Vision-Based Cooperative Localization for Small Networked Robot Teams

James Milligan[1], M. Ani Hsieh[1], and Luiz Chaimowicz[2]

[1]Scalable Autonomous Systems Lab,
Drexel University, Philadelphia, PA 19104, USA
{milligan.james,mhsieh1}@drexel.edu
http://drexelsaslab.appspot.com
[2]VeRLab, Computer Science Department
Federal University of Minas Gerais, Belo Horizonte, Minas Gerais, Brazil
chaimo@dcc.ufmg.br
http://homepages.dcc.ufmg.br/~chaimo/

Abstract. We describe the development of a vision-based cooperative localization and tracking framework for a team of small autonomous ground robots operating in an indoor environment. The objective is to enable a team of small mobile ground robots with limited sensing to explore, monitor, and search for objects of interest in regions in the workspace that may be inaccessible to larger mobile ground robots. In this work, we describe a vision-based cooperative localization and tracking framework for a team of small networked ground robots. We assume each robot is equipped with an LED-based identifier/marker, a color camera, wheel encoders, and wireless communication capabilities. Cooperative localization and tracking is achieved using the on-board cameras and local inter-agent communication. We describe our approach and present experimental and simulation results where a team of small ground vehicles cooperatively track other ground robots as they move around the workspace.

Keywords: networked robots, localization, applications development

1 Introduction

We are interested in the development of heterogeneous teams of air and ground vehicles for exploration and mapping of three dimensional (3-D) spaces where control and coordination strategies exploit the complementary actuation and sensing capabilities of aerial and ground vehicles. In particular, we envision the deployment of aerial vehicles cooperating with mobile ground robots equipped with range finders and cameras operating in an indoor environment and operating with smaller less capable robots. The aerial vehicles, with their ability to move in 3-D spaces, can provide information from vantage points inaccessible to the ground robots. The larger ground vehicles, with their larger payload capacity, can be equipped with high fidelity range sensors and cameras. Aerial and larger

ground robots can collaboratively build 3-D maps of the environment, help localize the team, and help deploy smaller ground robots. Once deployed, the smaller ground robots can be used to explore, search, and/or provide information from areas inaccessible to the larger robots.

We are interested in exploiting the capabilities of small ground robots to access regions in the workspace that are inaccessible to aerial vehicles and other larger ground robots. Examples include deploying a team of small ground robots to search for suspicious packages concealed under tables and benches, or provide information from vantage points inaccessible to the other robots in the team, *e.g.*, views through an air vent. Since these robots must be able to navigate in small confined spaces, their payload, and therefore their sensing capabilities, must be limited. We describe a cooperative localization and tracking strategy for teams of ground robots. The objective is to enable localization of a target located within the workspace that is not visible to the aerial vehicles and larger ground vehicles. We describe a strategy where a team of small ground robots cooperatively establish a local coordinate frame using their on-board cameras and local inter-agent communication. Cameras on the ground vehicles are used to observe identifying LED markers on each ground vehicle. Mutual observations define a visibility graph that is then used to triangulate the location of the ground vehicles in the local coordinate frame similar to [15]. Finally, assuming an aerial vehicle or larger ground robot can visually localize at least one element of each connected component of the small ground vehicle visibility graph, the local coordinate frame of the team of small ground robots can then be transformed into the global coordinate frame maintained by the aerial or larger ground vehicles.

In this work, smaller ground robots first establish a local coordinate frame using their on-board cameras and odometry. Information from an aerial vehicle is simulated using a static camera in the workspace and the estimate of the global state for one of the ground robots is communicated via the 802.11 wireless network. This information is then used to determine all the global poses for the team of ground robots within the workspace. Once localization has been achieved, one or a subset of the ground vehicles can then be tasked to drive to some region of interest to locate and localize a target of interest in the environment. The described approach can be scaled for teams of micro aerial vehicles (MAVs) cooperating with a team of larger unmanned ground vehicles (UGVs) and enables aerial vehicles to not only deploy teams of ground vehicles, but also to serve as localization anchors that can support a multitude of small teams of ground vehicles that can bring sensors to bear in regions inaccessible to MAVs and larger ground robots. The ability for the smaller UGVs to localize within a global coordinate frame enables more complex tasks such as autonomous exploration and searching and tracking of persons or items of interest hidden in the environment.

Existing work in cooperative localization [3, 6] has been shown using a wide selection of sensor combinations including range-only [17], bearing-only [14], and a combination of the two [16, 18]. Extensions of the general cooperative localization to that of a distributed system has also been addressed in [12]. In our

work, we rely on on-board cameras to extract relative orientation information and thus the localization problem considered constitutes a bearings-only localization problem. The majority of bearings-only vision-based localization in existing literature rely on known landmarks within the environment [1, 2, 7, 8, 10, 13]. Similar to [15], we rely on LED markers on each robot to enable a team of small ground vehicles to establish a local coordinate frame. State estimation and tracking is then achieved by fusing the information from the team using an Extended Kalman Filter (EKF) similar to [2, 4, 5, 7, 9, 11, 13].

The remainder of this paper is organized as follows: Section 2 briefly describes the scenario and the experimental testbed. Our Methodology is described in Section 3. We conclude with a brief summary and discussion of our experimental and simulation results in Section 4 and conclude in Section 5.

2 Experimental Testbed

As stated in the previous section, our objective is to develop a collaborative/cooperative localization and tracking strategy for a team of small, sensing limited team of ground robots for applications like exploration, target search, and providing situational awareness from vantage points inaccessible to aerial vehicles and larger ground robots. Since the objective is to use small ground robots, our robots are less capable than larger ground robots, *i.e.*, an iRobot Create platform. As such, we assume that our robots are only equipped with a single camera, odometry, and wireless communication capabilities.

Specifically, we consider a team of mSRV-1 robots (see Figure 1). The mSRV-1 is a differential-drive robot equipped with a 600 MHz Blackfin embedded processor, 802.11b wireless communication, a color camera, and wheel encoders. The robot has a footprint of approximately 10 $cm \times 9$ cm. The state of each vehicle is given by $X_i = [x_i\, y_i\, \theta]^T$ and summarized in Figure 2. Additionally, each robot on the team is equipped with an LED marker. The LED markers can be programmed with unique blinking patterns to enable unique identification of the individual robots similar to those described in [15].

3 Methodology

In this section we describe the development of our cooperative localization and tracking framework which consists of a state estimation step for estimating the state of the robots as they move within the workspace and an image processing step for identifying and localizing the robots within a local coordinate frame.

3.1 Extended Kalman Filtering

In our implementation, we use an Extended Kalman Filter with two independent update steps that incorporate information from the image processing step and from the robot odometry. As mentioned before the robot state is defined as

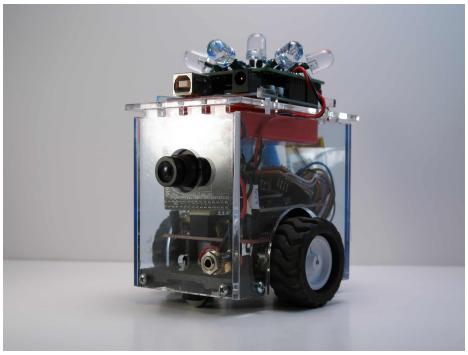


Fig. 1: mSRV-1 Robot

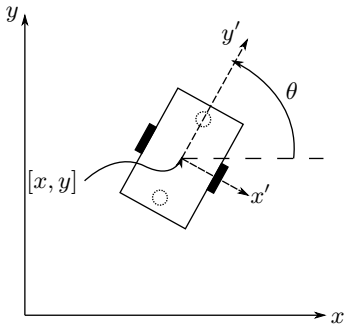


Fig. 2: mSRV-1 state

$[x, y, \theta]$. In the implementation of the EKF it is necessary to predict the system measurements in order to determine some residual value for filtering. In the image processing step, the primary output is a list of relative orientations to any LED markers in the field of view of any one of the robot cameras. Included with that are markers to denote which camera the information was taken from as well as which robot's LED markers are seen though this information is not directly used in the EKF.

Predicted outputs for the image processing step depend on both the field of view of the cameras as well as the state estimate for each of the robots. Since the state of a robot is defined with respect to a central point, there is some offset from this which defines the position of the camera. Figure 3 illustrates this idea. Here $[x', y', \phi']$ represent the position and relative orientation with respect to

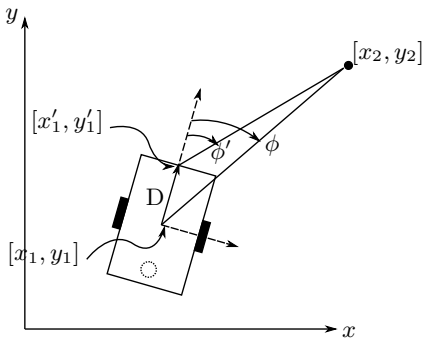


Fig. 3: Camera coordinate transformation

the camera. It is this ϕ' value that is being provided as a measurement. Since the EKF is based on the measurement residual between expected and measured, it is not necessary to bring this ϕ' value into the robot frame but rather the estimated measurement value could be carried out with respect to the camera position. The measurement prediction results in the following relationship:

$$h_\phi(\hat{x}_k) = \arctan\left(\frac{y_i - y_j - D\sin(\theta_j)}{x_i - x_j - D\cos(\theta_j)}\right) - \theta_j \tag{1}$$

where D represents camera offset distance from the robot center, the i components refer to the state information of the given LED and the j components, that of the camera.

Each robot is also equipped with wheel encoders and thus the output of the odometry is given as a column for the total distanced traveled at each time step.

3.2 Image Processing

At each time step, each robot is tasked with acquiring an image from its camera. In our system, we only consider the intensity of the robot LED markers in

the images rather than the marker colors, however, our approach can be easily extended to include color as another distinguishing feature. Given two images obtained by robot i at time steps k and $k+1$ denoted by $I_i(k)$ and $I_i(k+1)$, let $\Delta I_i = I_i(k+1) - I_i(k)$ denote the change in intensity in the image. As a robot moves through an image, the area where the robot was previously will have a decreased intensity while the intensity of the new location will increase as a result of the active LED markers. Thus, the region in ΔI_i, denoted as R_i with intensity values greater than some threshold $\delta_I > 0$ should give the new location of the LED marker in view and thus the robot.

Given that the robot dimensions are known, to improve the identification of the location of the LED marker in ΔI_i and filter out false positives, we take into account the distance of R_i from the ground plane and the size of R_i in relation to the known size of the robot. We then perform an opening operation followed by a closing operation to eliminate any smaller artifacts that may have shown up in ΔI_i. While both of these techniques improve the quality of the information extracted from the images, there is still no guarantee that a point of interest is in fact a robot. As such, the centroids of all R_i in ΔI_i whose intensities are greater than δ_I and whose areas are greater than $\delta_A > 0$ are then determined. Given the field of view of the camera, the centroid of each R_i in ΔI_i is then converted into a relative orientation of the point with respect to the camera as follows:

$$\phi_k = \alpha \left(\frac{w - 2p}{w} \right) \tag{2}$$

where ϕ_k represents the relative orientation of the LED marker with respect to the camera according to the right hand convention at time step k, α is the field of view of the camera, expressed as a $+/-$ value with respect to the camera normal, w is the width of the image, and p is the location within that image that is determined to be the centroid of the region of interest.

As mentioned before there are no guarantees that a region of interest in an image actually corresponds to a robot. Furthermore there is as of yet no information regarding which robot a given region of interest may correspond to. To address these issues, a vector, ρ, is created representing the robots that are in view for any given camera. Figure 4 illustrates one possible scenario for a robot.

Based on the estimated position of each of the robots at the previous time step, an estimated list of which robots are in view of Robot 0 can be created. In our scenario, each robot can potentially obtain an initial estimate of its position from an aerial vehicle or from another ground robot. In this instance, it is known that Robots 1 and 2 are within the field of view of Robot 0 denoted by the dotted lines, while Robot 3 is outside of Robot 0's field of view. The point F in Figure 4 represents a false positive in the image for which an orientation was provided though it does not correspond to a known robot.

This information is combined with the relative orientation computed using each of the estimated robot poses which we denote by $\hat{\phi}_j(k)$ to create a vector ψ_i, such that ψ_i for the example given in Figure 4 is given as: $\psi_0 = [Null, \hat{\phi}_1, \hat{\phi}_2, Null]^T$ where both the orientation to the robot itself and any ori-

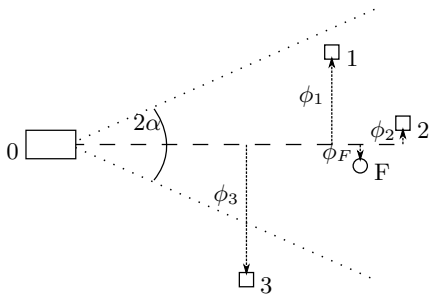


Fig. 4: Example case for Robot 0

entations to robots outside the field of view are given by $Null$. The $Null$ values insure that no information from these values is propagated through the system.

At this point, each ϕ given by Equation 2 is compared with each of the entries in ψ_i and matched with the entry that provides the smallest difference in orientations. This is essentially computing the residual values of the relative orientation measured and the ones estimated between all possible robots in view and provides an initial approximation for the image processing output. For any given robot there may be multiple orientations provided. For instance, in the example shown in Figure 4, ϕ_F would be attributed to Robot 2 as well as ϕ_2 since that is the closest match. Each robot is checked to ensure only one match is found in a given image. If there are multiple matches, the match with the least distance to the predicted orientation, based on the estimate pose, is taken to be the correct match. Since the size of the intensity blobs in the difference image is roughly the size of the robot itself, small errors in the position estimates at this point are not likely to drastically change the values returned by the imaging system, i.e., attributing an incorrect match to a robot.

With one match assigned per robot, another check is performed to make sure that all matches are within a given threshold. For the example shown in Figure 4, if the LED marker for Robot 2 did not appear in the image and yet the F did, ϕ_F would be returned as the relative orientation to Robot 2 since it was the best and only match for that robot. However, by ignoring differences in the relative orientation greater than that some appropriately chosen threshold, our system returns no information for Robot 2 instead of incorrect information at point F.

The vector ψ now represents a column vector in which each row corresponds to a robot in the team and the entries represent relative orientations as reported by the cameras. Since the EKF requires both the orientation as well as which robot camera the information came from, ψ is expanded to the matrix O for which each row entry reflects the following information: $[\phi, camID, ledID]$. In our EKF implementation, $camID$ and $ledID$ values are used to determine the locations for the Jacobian blocks of the observation matrix. For the scenario

provided the values for ψ, O, and H would be as follows:

$$\psi_0 = \begin{bmatrix} Null \\ \phi_1 \\ \phi_2 \\ Null \end{bmatrix} \tag{3}$$

$$O = \begin{bmatrix} \phi_1 & 0 & 1 \\ \phi_2 & 0 & 2 \end{bmatrix} \tag{4}$$

$$H = \begin{bmatrix} H_C & H_L & 0 & 0 \\ H_C & 0 & H_L & 0 \end{bmatrix} \tag{5}$$

where H_C represents the derivative of Equation 1, as presented in Section 3.1, with respect to the camera states, and H_L the derivative with respect to the LED states. Each entry shown in 5 consists of a 1×3 block so that each row of H corresponds with an image system output and each column a state of the system.

In the full system, values for O as provided by each camera are concatenated and a single H matrix is created to include all entries provided by each camera. For demonstration purposes the above values are assuming that Robot 0 is the only available camera in the team.

In summary, consider the case where the team is deployed in an environment where one or a subset of the robots is tasked to drive from their initial locations to some goal positions in the workspace. Our collaborative localization and tracking strategy for each robot can then be summarized as follows:

```
if the robot has not reached the goal location
    compute desired velocities (linear and angular)
    drive the robot
    predict current robot state based on control inputs (EKF predict)
    collect new measurements (images & odometry)
    extract relative orientation information
    update the state and covariance estimates (EKF update)
    check if the robot has arrived
end
```

4 Results

In our experiments, we consider a team of 5 mSRV-1 robots where one robot was tasked to drive from an initial position to a final goal position using the estimated state provided by the team. Since our robots are quite resource constrained, all image processing, state estimation, and computation was implemented on a centralized computer to ensure the experiments can be completed in a reasonable amount of time and robots can move at a reasonable speed. The general implementation of the calculations and image processing, while done on a central

computer, was carried out as if it were on individual processors, simulating a decentralized implementation. With demonstrated success in this manner experimentally, we believe our approach can be easily decentralized once the image processing has been successfully implemented on each robot's embedded processor. Furthermore, an initial state estimation for at least two of the robots is provided by an overhead localization system. The overhead localization system provides information that can be provided by an aerial vehicle or larger ground robot with view of at least one of the ground robots.

In order to validate the results of our collaborative localization and state estimation framework, several experiments were conducted where the estimated states are compared to ground truth measurements provided by our overhead localization system. The accuracy of our overhead localization system is approximately ± 20 mm. Figure 5 shows three trial runs and the predicted error ellipses.

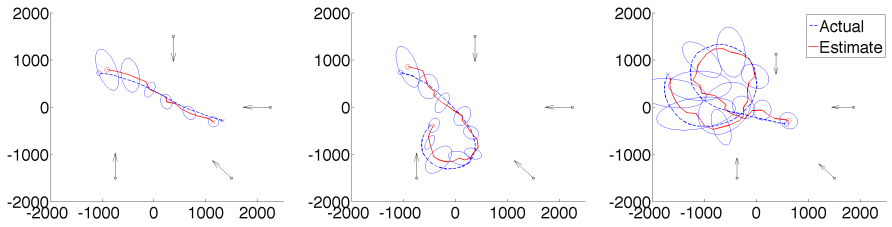
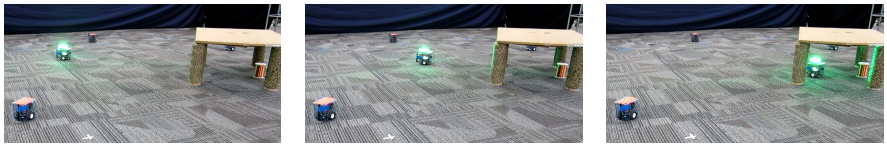


Fig. 5: Three example trials. Start and end positions are shown as $\circ$ and $\times$ respectively and cameras are denoted with squares, oriented as shown. Axes shown in mm.

The error at each time step was calculated as the 2-norm between the ground truth and the estimated position. The mean error over the full run as well as the standard deviation of the error is provided for each trial. A total of 12 trials were completed with varying start and end positions comprising a total of 278 pose estimates. The average error in position over all 278 estimates was 132 mm with a standard deviation of 67 mm. Despite the lack of high fidelity odometry sensors on our robots, the accuracy of our strategy is about a robot's bodylength. Based on the covariance matrices of the EKF it is also possible to see the assumed error in the measurements from the robot's perspective. The ellipses shown on Figure 5 represent the location of the robot with 80% confidence based on noise in the measurements. As can be seen the number of robots that can see the traveling robot as well as the direction from which the robot is seen has a large affect on the error in the system.

In addition to evaluating the position error of the system, we considered an example scenario where the smaller ground vehicle cannot be easily localized by an aerial vehicle or larger ground robot. This is shown in Figure 6 where a robot was tasked with capturing an image of a suspicious object underneath a

low table. The view of the object is obstructed from the point of view of any aerial vehicles. Furthermore, the low clearance means the space is inaccessible to larger ground vehicles. Using our cooperative localization and tracking strategy the mSRV-1 is able to infiltrate the space to localize the target.



(a) (b) (c)

Fig. 6: Surveillance of suspicious object under table.

Finally, the show the scalability of the proposed strategy, we simulated a scenario where a team of five robots must simultaneously track three moving robots. In our simulation, the error on relative orientation measurements was assume to follow a Gaussian distribution based on errors obtained in experiments. Figure 7 shows one such trial in which three robots were tasked to drive 10m towards three different goal locations. Using the methodology presented, the two static robots were successful in providing the information necessary to localize and track the three others.

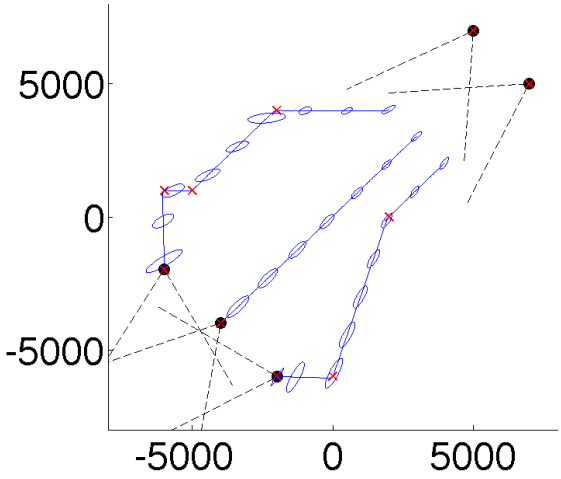


Fig. 7: Multiple driver simulation. Dashed lines represent field of view of the robots, solid lines show the paths traveled, interim goals are shown as $\times$, axes are in mm.

5 Conclusions

In this work, we presented a cooperative localization and tracking framework specifically designed to enable small ground vehicles to operate in regions of the workspace that is inaccessible to aerial vehicles and/or larger ground robots. Specifically, we address the localization and navigation of small ground vehicles in such confined spaces. An EKF-based position update scheme was presented in which bearing information received from on-board cameras and odometry information results in pose estimates for the team that are accurate to approximately one robot body length. One direction for future work is full distributed implementation of the strategy on a team of robots capable of on-board image processing. A second direction for future work includes the integration of the approach with a team of aerial vehicles capable of providing the initial global pose estimates for the team.

For all of the scenarios presented, the location of the static robots responsible for tracking the moving robots were predefined. The positioning of these tracking robots has a large effect on the accuracy of the system. As illustrated in Figure 8, a system consisting of two robots with parallel views is much more susceptible to large errors due to measurement noise than a system of two robots with perpendicular views.

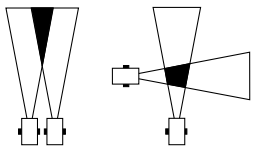


Fig. 8: Error envelopes for different camera poses.

In Figure 8, lines are shown to represent an error envelope on the relative orientations as provided by the cameras. This error is then extended into the $[x, y]$ position measurements as represented by the shaded region. The position error in the parallel view scenario is unbounded and highly undesirable. This idea of *best sensing locations* is examined in [16], in which observers are directed to move in order to reach the optimal imaging location given a target's position. We are interested in extending our approach to include such formation control strategies and eliminate the need for predefining the positions of the tracking robots.

References

1. Bishop, A.N., Anderson, B.D.O., Fidan, B., Pathirana, P.N., Mao, G.: Bearing-Only Localization using Geometrically Constrained Optimization. IEEE Transactions on Aerospace and Electronic Systems 45(1), 308–320 (Jan 2009)

2. Bonnifait, P., Garcia, G.: A multisensor localization algorithm for mobile robots and its real-time experimental validation. IEEE International Conference on Robotics and Automation (April), 1395–1400 (Apr 1996)
3. Howard, A., Matarić, M.J., Sukhatme, G.: Putting the 'I' in 'team': an ego-centric approach to cooperative localization (2003)
4. Huang, G.P., Trawny, N., Mourikis, A.I., Roumeliotis, S.I.: On the Consistency of Multi-robot Cooperative Localization. Proceedings of Robotics: Science and Systems (Jun 2009)
5. Jetto, L., Longhi, S., Venturini, G.: Development and experimental validation of an adaptive extended Kalman filter for the localization of mobile robots. IEEE Transactions on Robotics and Automation 15(2), 219–229 (Apr 1999)
6. Kurazume, R., Nagata, S., Hirose, S.: Cooperative positioning with multiple robots (1994)
7. Leonard, J., Durrant-Whyte, H.: Mobile robot localization by tracking geometric beacons. IEEE Transactions on Robotics and Automation 7(3), 376–382 (Jun 1991)
8. Loizou, S.G., Kumar, V.: Biologically inspired bearing-only navigation and tracking. 2007 46th IEEE Conference on Decision and Control pp. 1386–1391 (2007)
9. Mourikis, A., Roumeliotis, S.: Performance Analysis of Multirobot Cooperative Localization. Robotics, IEEE Transactions on 22(4), 666–681 (Aug 2006)
10. Reza, D.S.H., Mutijarsa, K., Adiprawita, W.: Mobile robot localization using augmented reality landmark and fuzzy inference system. In: Electrical Engineering and Informatics (ICEEI), 2011 International Conference on. pp. 1–6 (2011)
11. Roumeliotis, S.I., Bekey, G.A.: Collective localization: a distributed Kalman filter approach to localization of groups of mobile robots (2000)
12. Roumeliotis, S., Bekey, G.: Distributed Multi-Robot Localization. IEEE Transactions on Robotics and Automation 15(5), 781–795 (Oct 2002)
13. Se, S., Lowe, D., Little, J.: Local and global localization for mobile robots using visual landmarks. In: Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on. vol. 1, pp. 414–420 vol.1 (2001)
14. Sharma, R., Quebe, S., Beard, R., Taylor, C.: Bearing-only Cooperative Localization. Journal of Intelligent & Robotic Systems pp. 1–12 (2013), http://dx.doi.org/10.1007/s10846-012-9809-z
15. Shirmohammadi, B., Taylor, C.J.: Self Localizing Smart Camera Networks. ACM Transactions on Sensor Networks 8(2), 11:1–11:24 (Mar 2012)
16. Zhou, K., Roumeliotis, S.I.: Multi-robot Active Target Tracking with Combinations of Relative Observations. IEEE Transactions on Robotics 27(4), 678–695 (Aug 2011)
17. Zhou, X., Roumeliotis, S.: Robot-to-Robot Relative Pose Estimation From Range Measurements. IEEE Transactions on Robotics 24(6), 1379–1393 (Dec 2008)
18. Zhou, X.S., Roumeliotis, S.I.: Determining the robot-to-robot 3D relative pose using combinations of range and bearing measurements (Part II). 2011 IEEE International Conference on Robotics and Automation (Part II), 4736–4743 (May 2011)