Relative localization with Computer Vision and UWB Range for Flying Robot Formation Control

B. Ramirez[1,†], H. Chung[3,†], H. Derhamy[2], J. Eliasson[2,†] and J.C. Barca[1,†]

Abstract—**Relative localization is a core problem for swarm robotics since each swarm node must determine where neighboring robots are located to accomplish cooperative tasks such as formation control. In this paper we present a system that performs relative localization between a stationary marker and a flying robot in real-time. Relative distances are obtained using ultra-wide band radio, while a low-cost webcam provide angle measurements. To achieve the latter, we employed the Camshift algorithm and a Kalman filter. We tested our system outdoors during daylight using centimeter-accuracy GPS measurements as ground truth. Three data sets have been collected from a series of experiments and it shows that errors in estimated relative positions are between *+/-0.190 m* on the x-East axis and *+/-0.291 m* on the z-North axis at 95% confidence level.**

Keywords-Flying robot; Relative Localization; UWB Range; Computer Vision; Camshift.

I. Introduction

A swarm of flying robots will soon play a major role in a wide range of applications, such as search and rescue, target tracking, localization, and mapping. A swarm uses relatively simple agents or nodes that can cooperatively use their resources to perform complex tasks. In these tasks, it is important to measure or estimate relative positions with respect to nearby nodes [4,18].

Some of the main applications of relative localization are formation forming and keeping, which are considered fundamental capabilities required for the autonomous operation of cooperative tasks [17,13]. This is why, robot-to-robot relative localization remains one of the core problems of swarm robotics.

For a single flying robot self-localization can be tackled by simultaneous localization and mapping approaches [17]; however, external and internal localization systems are still widely used for multi-robot relative localization. These systems include; GPS, Wi-Fi, ultrasound, sonar, infrared markers, ultra-wide band radios (UWB), or (multi-) camera systems just to name a few. Such systems are used for closed-loop control, formation control or ground truth position measurements [15].

For example, the Differential Global Positioning System (DGPS) is able to acquire Real Time Kinematic (RTK) position and elevation measurements; however, it will not operate under trees, next to buildings or indoors. This sensor also requires a ground station to produce accurate position measurements, limiting the operation range of the system [14].

Wi-Fi can be used for coarse grained localization inside buildings via signal strength measurements, or by determining which access point a device is connected to. Wi-Fi provides a rough position estimate, usually in the range of 5-15 meters in accuracy, in many cases this is sufficient to maintain formations for unmanned ground vehicles or slow moving robots [1]. A drawback with this approach is that the need for access points makes it difficult to implement these systems in large environments.

To perform relative localization in any environment, it is highly desirable to have good precision and high-frequency of sensor measurements. With these readings one can accurately measure the relative range and angle to neighboring nodes. Their relative pose can also be determined [5]. To produce accurate relative position estimates, robots make use of these measurements and fuse them with their own sensor data [9].

Sonar is one of the most popular sensors used in localization methods, as it is simple and inexpensive and provides distance information. However, in general sonar has difficulty in measuring the accurate direction of objects. Another inexpensive localization system is the use of onboard cameras to localize neighboring robots [5]. H.-D. Kang [8] used wide angle cameras to create a panoramic image with 360 horizontal field of view. Where neighboring robots could be tracked in order to compute their relative 3D positons.

One issue when using cameras, is to distinguish and track objects in outdoor and indoor environments. Active and passive markers have been used to address the visual identification and tracking problem and it has been shown that this can lead to low computational cost tracking systems [2,3].

A. Breitenmoser [2] developed a computationally lightweight 3D localization system equipped with target

[1] Brian Ramirez Espinosa and Jan Carlo Barca are with the Faculty of Information Technology, Monash University, Clayton VIC 3800, Australia brian.espinosa@monash.edu, jan.barca@monash.edu.

[2] Hasan Derhamy and Jens Eliasson are with the Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, 971 87 Luleå, Sweden, hasan.derhamy@ltu.se, jens.eliasson@ltu.se.

[3] Hoam Chung is with the Department of Mechanical and Aerospace Engineering, Monash University, Clayton VIC 3800, Australia hoam.chung@monash.edu.

[†] The authors are with the Monash Swarm Robotics Laboratory, Monash University, Clayton VIC 3800, Australia.

978-1-5090-3549-6/16/$31.00 ©2016 IEEE

modules, which included four active LED emitters and a monocular camera. However, in outdoor environments, the light emitted from the LEDs could be washed out by sunlight and the predefined shape could be occluded by obstacles.

Alternatively, passive markers with distinctive colors can be tracked, even when occlusions are present using robust color tracking algorithms and a Kalman filter. This filter can estimate the next position of the target, with the knowledge of previous observations [3]. These algorithms also require additional distance measurements to complete three-dimensional relative position estimation.

UWB radios for range measurements [11] have been a hot topic in wireless networks and real-time localization systems, recently. UWB offers very interesting properties and capabilities such as high bandwidth, low sensitivity for multi-path interference and relatively accurate ranging. The use of UWB in e.g. mining applications has been studied by Daixian [6].

Once the relative positions are estimated, formation control algorithms such as behavior based [8,19,20,21], virtual structures [16,19], hierarchical formations [12,19,21] and morphogenesis [7,19,20], can be used. These methods typically assume full knowledge of the formation state expressed as the relative distances and bearings among the adjacent vehicles [9].

In this paper, we propose a 3D relative positioning system suitable for fast moving flying robots. The system is designed to operate with low computational power, since flying robots have harsh constraints on processing power and payload weight. The system estimates its own 3D position in relation to a colored marker. Two UWB radios are used to determine the distance between the two nodes (flying robot, marker). A computer vision system mounted on the flying robot calculates the relative angle between the nodes.

This paper is arranged as follows: Section II describes the hardware and functionality of our proposed relative localization system. Section III proceeds by detailing the performance of our system as measured by a set of experiments. Conclusions and future directions of research are provided in Section IV.

II. System Design

The proposed technology measures the distance between two nodes (flying robot, marker) by means of two UWB distance measuring sensors, one on each node. Angle measurements are obtained separately with a webcam mounted onto the flying robot. To process sensor data a central processing unit (ODORID-XU4) and an open source autopilot (Pixhawk), where mounted onto the flying robot see Figure 1 and Figure 2.

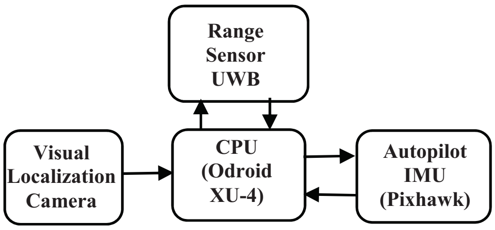

Figure 1. Flying robot's payload.

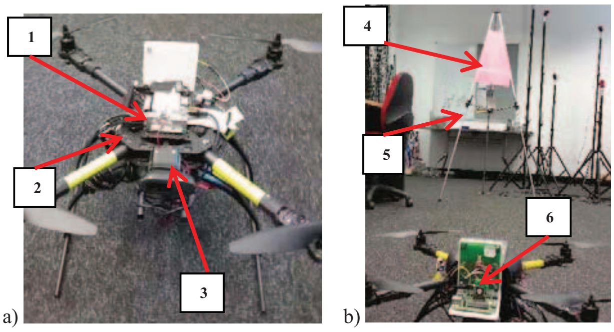

Figure 2. a) Flying robot setup 1: Odroid XU4, 2: Pixhawk, 3: Camera; b) Relative localization system setup 4: Colored-marker, 5: UWB-marker, 6: UWB-flying robot

A. Distance Measurement

As the first step to determine the relative 3D pose of the flying robot in relation to the marker. We have to measure the distance between the nodes. To achieve this, we selected the Decawave DW1000 chip. Given that this UWB radio can perform ranging with an accuracy of +/- 10cm at up to 200m away within Line-of-Sight (LoS). This chip also supports two forms of distance calculation. It supports using time difference of arrival, and time of flight (ToF) distance calculation.

A choice was made to focus on ToF between two UWB nodes, in order to allow a minimum of two nodes to perform distance measurements. Because the absolute times of transmission and reception of each node in the pair, are used only relative to the same clock domain, the ToF measurements does not need clock synchronization between two pairs. This is very beneficial when high update rates are required.

The nodes are setup in two modes, one as an Anchor and the other one as a Tag. There are a total of six messages between the two nodes used in a ranging exchange: *1-blink, 2-init, 3-poll, 4-response, 5-final, and 6-measurement report.* The ranging process is as follows: The Tag is a sleepy node which broadcast *blink* messages, searching for the Anchor within range. This is the discovery message. Anchor nodes are actively listening/waiting for *blink* messages from any Tag.

Once a *blink* is received the Tag will begin the ranging handshake by sending an *init* message. The Tag responds with a *poll* message at time T_{sp} and is received by the Anchor at time T_{rp}. The Anchor then sends a *response* at time T_{sr} message which is received and responded at T_{rr} and T_{sf} respectively with a *final* message to the Tag and is received at T_{rf}. The *final* message is the last message in the ranging handshake. The sixth message is an optional *measurement report* message sent from the Anchor to the Tag. This message holds the ToF calculated from the four timed messages from T_{sp} to T_{rf}. This interaction can be seen in Figure 3.

For low-power operation on a quadcopter, it is assumed that both nodes are battery powered. The nodes periodically switch between the roles of Anchor and Tag, in order to actively initiate ranging with any given node.

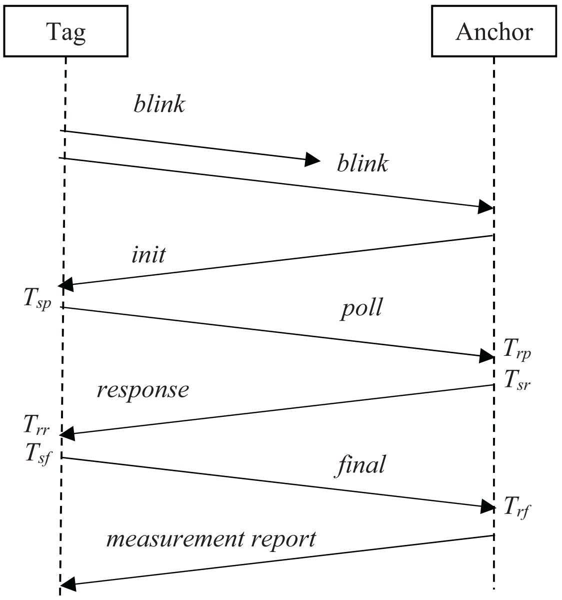

Figure 3. UWB ranging procedure

B. Angle Measurements

With the distance between nodes determined by the UWB sensor. The next step was to determine the 3D relative localization of the flying robot, by calculating the angle between the nodes. This was achieved, through visually tracking the colored marker from the mounted camera on the flying robot and connected to the onboard Odroid XU-4. To distinguish the marker from the rest of the environment, it was decided to use a colored passive marker. A color that is not predominant in an environment, like bright pink can be easily tracked, which is the one we used.

To track this distinctive color, we made use of OpenCV, which includes the Camshift algorithm for color tracking. This algorithm is an adaptation of the Meanshift algorithm for object tracking that is a robust, non-parametric technique [22]. The algorithm climbs the gradient of a probability distribution function, to find the local maxima of a data set. The main difference between both algorithms, is that Camshift uses continuously adaptive probability distributions, which means the target`s probability distribution may be recomputed in every frame. This lets the target`s size, shape and appearance change in every frame [23].

To obtain robust results against different lighting conditions, we used the HSV histogram (Hue Saturation Value) color space in our algorithm. We only chose hue and saturation color channels, because these represent the marker`s main color features. The histogram values are then to be within the discrete pixel range of the 2D probability distribution image using (1) [22,23].

$$p_u = min\left({}^{255}/_{max(q)}\, q_u, 255\right),\ \ u = 1,2 \tag{1}$$

Given that two histograms are used, we define the n image pixel locations as $x_{i=1\ldots n}$, $y_{i=1\ldots n}$, the histograms as $q_{u=1,2}$ and the PDF as $p_{u=1,2}$. The histogram values are rescaled from [0, *max(q)*] to the new range [0, 255], where pixels with the highest probability of being in the sample histogram will appear as visible intensities in the 2D histogram back-projection [22,23].

The mean location (centroid x_c, y_c) within the search window of the discrete probability image is found using the zeroth and first moments ($M_{00,}$ $M_{10,}$ M_{01}). Given that *I(x,y)* is the intensity of the discrete probability image within the search window, as we show in (2) [22,23].

$$M_{00} = \sum_x \sum_y I(x,y)$$

$$M_{10} = \sum_x \sum_y xI(x,y)$$

$$M_{01} = \sum_x \sum_y yI(x,y)$$

$$x_c = \frac{M_{10}}{M_{00}};\ y_c = \frac{M_{01}}{M_{00}} \tag{2}$$

In order to solve the occlusion problem, saturation and value components are adjusted and in combination with a Kalman filter, has proven to increase the accuracy of the tracking algorithm. In order to use the Kalman filter implemented in OpenCV to estimate the state of position of the marker given noisy observations from the camera mounted on the flying robot. This means specifying: *F(k)*, the state-transition model; *H(k)*, the measurement model, for each time-step *k,* see (3), *Q(k)*, the process noise covariance; *R(k)*, the measurement noise covariance, were defined through tests [10].

$$F(k) = \begin{bmatrix} x_{ck-1} & 0 & Vx_{ck-1} & 0 \\ 0 & y_{ck-1} & 0 & Vy_{ck-1} \\ 0 & 0 & Vx_{ck-1} & 0 \\ 0 & 0 & 0 & Vy_{ck-1} \end{bmatrix}$$

$$H(k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{3}$$

Once the position of the marker has been determined on a 640x480 pixel image, we converted the (x_{ck},y_{ck}) pixel coordinates into distance in meters from the center of the whole image. In order to calculate the real distance each pixel represents, determined by the distance from the camera to the target at any given time. We made use of the known area of the colored marker and the area in pixels that the marker represents on the image to characterize the sensor as Figure 4 shows.

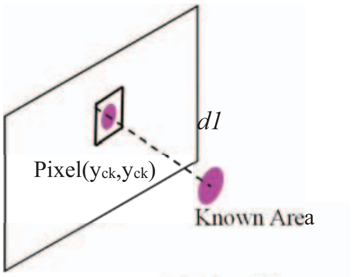

Figure 4. Representation of the characterization of the camera

III. Experiment design and Results

When dealing with a formation of flying robots, inaccurate relative position estimates could jeopardize the whole system. These inaccuracies can make the formation unstable and eventually break apart or in the worst case members can collide with each other. With these issues in mind and to prove that our system is viable option as relative localization system, we tested the accuracy of our relative position estimates. We made use of a SwiftNav DGPS as ground truth, which has a reported standard deviation of +/- 2cm when in the differential RTK mode.

The experiments occurred outdoors. The colored marker was placed in an open space, without any obstructions to the quadcopter (see Figure 5). Before any test the GPS position from the marker surveyed for an hour at 5Hz, in order to calculate the mean of its position. Three tests were performed, where the flying robot moved with constant altitude, while measuring the relative d and $x = x_{ck}$ position of the marker.

$$\theta = \cos^{-1}(x/d); \quad z = \sin(\theta)/d \tag{4}$$

With these readings the onboard computer calculated the Θ angle and z position of the flying robot in relation to the marker using (4). In our case x denotes East z North, and y Down, respectively. We left the height of the flying robot at a constant *alt =3 m*, in order to maintain constant visual LoS with the marker. A ground station PC running Ubuntu 14.04 recorded the RTK positions in meters (East, North), which the DPGS provides. Where the *(0,0)* coordinates represent the marker's position, see Figure 7. Every time the UWBs reported range measurements, the onboard computer recorded the distance measurement, the position of the marker in the image and the RTK position, at the same time to sync the data.

When a test began, the first step was to select a region of interest (RoI) for the Camshift algorithm and detect the color of the marker (Figure 6). When the flying robot moved, we kept visual LoS with the marker, while maintaining the same heading at all times. This was done to avoid moving the center of the image, every time the flying robot faced the marker at another angle.

By having the UWB sensor we avoid computing the distance measurement with the camera. This allows for a computationally lightweight system, although while testing our system we noticed that the UWB had blind spots when ranging. We conclude that this was due to the UWB antenna not being large enough to transmit in a spherical radio wave.

To determine the accuracy of our system, the distributions of the errors on the x East axis and z North axis are shown in Figure 8. Both distributions show that the 95% of the error data lies between +/- two standard deviations as also shown in Table 1, which means that that most of the errors spread between *+/-0.196 m* on the x East axis and *+/-0.32 m* on the z North axis. The results show larger errors in the *z*-axis, given that the *z* position was calculated with *d* and *x*; which inherently contain errors.

Furthermore, the average error measurement of the three test and the averaged error measurements are *x = +/-0.073 m, z= +/-0.105 m* for the three tests (Table 2). These results verify that our relative localization system is capable of providing accurate relative position estimation.

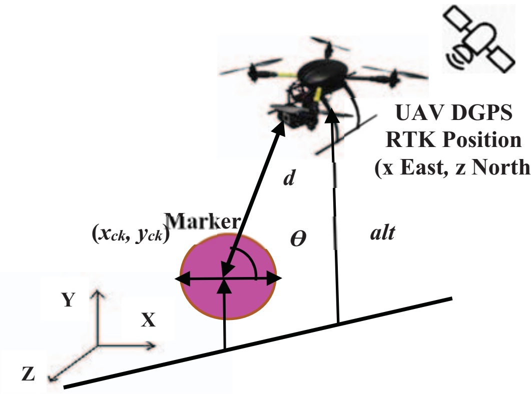

Figure 5. Representation of the formation control setup

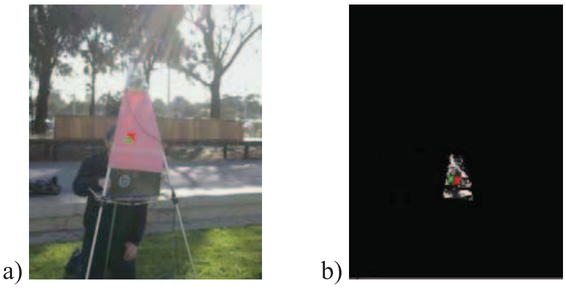

Figure 6. a) Tracking of the colored marker b) Black and white image of the RoI. Red "x": Position of the marker without the Kalman filter. Green "x": Position of the marker with the filter.

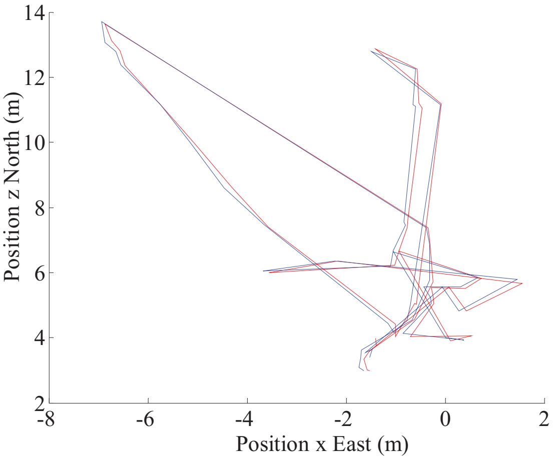

Figure 7. Third test: Flying robot relative position with respect to the marker (0,0), measured with our relative localization system and the DGPS RTK position. Blue line: flying robot's position measured with the UWB and the camera. Red line: flying robot's RTK position measured with the DGPS.

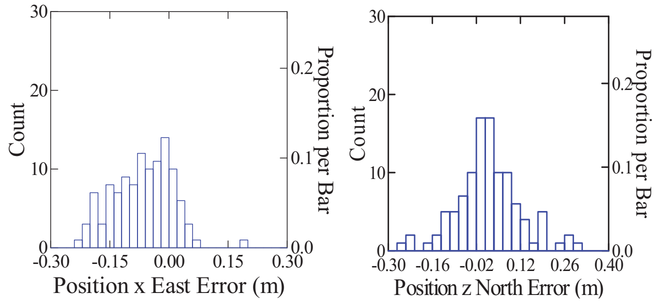

Figure 8. Normal distribution of the error on the x EAST coordinate and z NORTH coordinate.

	POSITION x EAST ERROR (m)	*POSITION z NORTH ERROR (m)*
NUMBER OF DATA POINTS	*116*	*116*
MINIMUM	*-0.190*	*-0.602*
MAXIMUM	*0.130*	*0.291*
DISTRIBUTION MEAN	*-0.054*	*-0.017*
DISTRIBUTION STANDARD DEVIATION	*0.057*	*0.160*

Table1. Statistics of the normal distribution of the error on the x East coordinate and z North coordinate.

	TEST 1		***TEST 2***		***TEST 3***	
	x EAST (m)	*z NORTH (m)*	*x EAST (m)*	*z NORTH (m)*	*x EAST (m)*	*z NORTH (m)*
AVERAGE ERROR	*+/- 0.017*	*+/- 0.068*	*+/- 0.105*	*+/- 0.177*	*+/- 0.080*	*+/- 0.070*

	x EAST (m)	***z NORTH (m)***
OVERALL AVERAGE ERROR	*+/-0.067*	*+/-0.105*

Table2. Average error of *x* and *z* axis over three tests and the overall average error in meters

IV. Conclusion

In this paper, we present a low-cost relative position estimation system for flying robots. Our experimental results show that the combination of UWB radios and a simple computer vision system is a viable option for computationally cheap relative position estimation between two nodes. During the experiments, we can estimate relative positions with errors between *+/-0.190 m* on the x East axis and *+/-0.291 m* on the z North axis at 95% confidence level. During experiments we noticed that the UWB had blind spots when ranging, which will be improved in the next version of our hardware implementation. The proposed system will be applied to autonomous formation flight of flying robots for mapping GPS-denied environments.

References

[1-23]

[1] J. Biswas and M. M. Veloso, "Wifi localization and navigation for autonomous indoor mobile robots," 2010.

[2] A. Breitenmoser, L. Kneip, and R. Siegwart, "A monocular vision-based system for 6D relative robot localization," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, 2011, pp. 79-85.

[3] V. Caglioti, A. Citterio, and A. Fossati, "Cooperative, distributed localization in multi-robot systems: a minimum-entropy approach," in *Distributed Intelligent Systems: Collective Intelligence and Its Applications, 2006. DIS 2006. IEEE Workshop on*, 2006, pp. 25-30.

[4] H. Cheng, J. Page, and J. Olsen, "Cooperative control of UAV swarm via information measures," *International Journal of Intelligent Unmanned Systems,* vol. 1, pp. 256-275, 2013.

[5] I.-S. Choi, J.-S. Choi, and W.-J. Chung, "Sonar sensor based relative localization for leader-following formation control," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, 2011, pp. 202-205.

[6] Z. Daixian and Y. Kechu, "Particle Filter Localization in Underground Mines Using UWB Ranging," in *Intelligent Computation Technology and Automation (ICICTA), 2011 International Conference on*, 2011, pp. 645-648.

[7] K. Hosokawa, T. Tsujimori, T. Fujii, H. Kaetsu, H. Asama, Y. Kuroda*, et al.*, "Self-organizing collective robots with morphogenesis in a vertical plane," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, 1998, pp. 2858-2863.

[8] H.-D. Kang, J.-H. Kawk, C.-H. Kim, and K.-H. Jo, "Multiple robot formation keeping and cooperative localization by panoramic view," in *ICCAS-SICE, 2009*, 2009, pp. 3509-3513.

[9] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil*, et al.*, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems,* vol. 76, pp. 539-562, 2014.

[10] C. B. Liu, C. C. Chen, and X. Li, "Object Tracking System in Dynamic Scene Based on Improved Camshift Algorithm and Kalman Filter," in *Applied Mechanics and Materials*, 2014, pp. 2061-2064.

[11] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2015, pp. 1730-1736.

[12] S. R. Musse and D. Thalmann, "Hierarchical model for real time simulation of virtual human crowds," *Visualization and Computer Graphics, IEEE Transactions on,* vol. 7, pp. 152-164, 2001.

[13] L. E. Parker and F. Tang, "Building multirobot coalitions through automated task solution synthesis," *Proceedings of the IEEE,* vol. 94, pp. 1289-1305, 2006.

[14] Y. Qu, Y. Zhang, and Q. Zhou, "Cooperative localization of UAV based on information synchronization," in *Mechatronics and Automation (ICMA), 2010 International Conference on*, 2010, pp. 225-230.

[15] I. M. Rekleitis, G. Dudek, and E. E. Milios, "On the positional uncertainty of multi-robot cooperative localization," in *Multi-Robot Systems: From Swarms to Intelligent Automata*, ed: Springer, 2002, pp. 3-10.

[16] W. Ren and R. Beard, "Decentralized scheme for spacecraft formation flying via the virtual structure approach," *Journal of Guidance, Control, and Dynamics,* vol. 27, pp. 73-82, 2004.

[17] J. Stipes, R. Hawthorne, D. Scheidt, and D. Pacifico, "Cooperative localization and mapping," in *Networking, Sensing and Control, 2006. ICNSC'06. Proceedings of the 2006 IEEE International Conference on*, 2006, pp. 596-601.

[18] X. S. Zhou and S. I. Roumeliotis, "Determining the robot-to-robot 3D relative pose using combinations of range and bearing measurements: 14 minimal problems and closed-form solutions to three of them," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, 2010, pp. 2983-2990.

[19] J. C. Barca, A. Sekercioglu, and A. Ford, "Controlling formations of robots with graph theory," in *Intelligent Autonomous Systems 12*, ed: Springer, 2013, pp. 563-574.

[20] W. L. Seng, J. C. Barca, and Y. A. Sekercioglu, "Distributed formation control in cluttered environments," in *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*, 2013, pp. 1387-1392.

[21] S. Yu and J. C. Barca, "Autonomous formation selection for ground moving multi-robot systems," in *Advanced Intelligent Mechatronics (AIM), 2015 IEEE International Conference on*, 2015, pp. 54-59.

[22] E. Emami and M. Fathy, "Object tracking using improved CAMShift algorithm combined with motion segmentation," in *Machine Vision and Image Processing (MVIP), 2011 7th Iranian*, 2011, pp. 1-4.

[23] J. G. Allen, R. Y. Xu, and J. S. Jin, "Object tracking using camshift algorithm and multiple quantized feature spaces," in *Proceedings of the Pan-Sydney area workshop on Visual information processing*, 2004, pp. 3-7.