Proceedings of the World Congress on Engineering and Computer Science 2016 Vol II
WCECS 2016, October 19-21, 2016, San Francisco, USA

Cooperative Explorations with Wirelessly Controlled Robots

G. Huang, R. Childers, J. Hilton, Z. Ye and Y. Sun

Abstract — **Robots have gained an ever increasing role in the lives of humans by allowing more efficient completion of tasks, ranging from healthcare to manufacturing. One area that robots have not been fully utilized in is multi-robots with autonomous exploration. In this research, we investigate how to use multiple robots collaboratively to explore and search the target more efficiently than a single robot is capable of. Given the task of finding a specific object from different areas, one robot will scout an area while the other robot will also scout a different area and pick up an object if it locates one. If the scout robot locates the object, then the scout will report the room that the object is located in by sending the room number between the robots. To achieve the tasks described above, robots should be at least able to communicate with each other, navigate different areas, detect objects, and grab the target. In this project, we have developed a communication method between robots, two navigation algorithms for two robots to avoid obstacles and navigate areas, object detection, and voice control. In addition, we developed some wireless phone control functions to provide the flexibility and convenience for users. Our experiments have demonstrated that above algorithms and methods can successfully make multi-robots work cooperatively to explore different areas and find the target.**

Index Terms — EZ-Robots, Autonomous, Exploration, Object Detection and Avoidance, Voice Control

I. INTRODUCTION

According to the International Federation of Robotics, the number of robot installations is estimated to increase by 12% on average per year from 2015 to 2017: about 6% in the Americas as well as in Europe, and about 16% in Asia/Australia. The trend towards automation continues to increase the volume of robot installations. Industry, linking the real-life factory with virtual reality, will play an increasingly important role in global manufacturing. The robotics industry is looking into a bright future [1].

Among the various types of robots that have been developed and used, rescue robot has been designed for the purpose of rescuing people from situations like mining accidents, urban disasters, hostage situations, and explosions, etc. Using rescue robots in these cases can minimize the risks imposed on the first respondents, reduce personnel requirements and fatigue, and allow access to otherwise unreachable areas. For example, rescue robots were used in the search for victims and survivors after the September 11 attacks in New York [2].

Manuscript received June 15, 2016; revised August 8, 2016. This work was partially supported by NASA EPSCoR Grant 2012 (No. NNX13AD32A).

Guofu Huang is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72034, USA.

Reese Childers is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72034, USA.

Joseph Hilton is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72034, USA.

Zixin Ye is with Central High School, 1500 Park Street, Little Rock, AR 72202, USA

Yu Sun is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72034, USA (Phone: 501-450-3385; fax: 501-450-5615; e-mail: yusun@ uca.edu).

Robot exploration is a hot research topic and a lot of research has been conducted in this area. The research in [3] addresses two tasks related to robotic extraterrestrial explorations: mapping and rover localization. But this approach only provides single robot exploration and it will need much more time than multiple robots to explore big area if this area is very big. The research in [4] formulates robotic exploration as a graph traversal problem. The robot is assumed to be able to autonomously traverse graph edges. Jennings et al propose a cooperative search and rescue method to allow a distributed team of mobile robots [5] to search for an object. When one robot finds the object, all robots will get gather around it to manipulate ("rescue") the object. The algorithm exploits parallelism, (with all robots searching concurrently) and is fully distributed (the robots communicate with each other without using a central server or supervisor). The strength of this approach is that it provides a very efficient way to let robots work cooperatively. The weakness is that it doesn't use a speech interface to better control the exploration process.

For the above work, only one involves multi robots and the other two only use one robot. None of them uses voice commands and wireless control functions.

In this work, we want to investigate the practical use of rescue robots to help people in common situations. While it is sufficient to have a single robot carry out tasks [3, 4] for relatively simple cases, the approach may be problematic a complicated situation. Taking a mining accident as an example, the space is too big for one robot to search. It may take the robot many hours to just cover a small area. However, in a serious accident, every minute matters. If the robot can search faster, it might increase the chance of saving lives. Although a team of robots may be controlled by manually by human operators, it is difficult to perform an efficient search with complete coverage in shortest time Computers can do this work much better than human operators as they can record every steps that the robots have gone and communication between robots can effectively avoid any exploration of an already explored area. In addition, computers can calculate much faster than human beings. Therefore, using a team of robots to search collaboratively will greatly increase the search efficiency.

Therefore, our motivation for this project is to develop a cooperative exploration system that involves multiple robots working cooperatively to search for an object. This research investigates algorithms and methods for independently exploring multiple rooms to search for an object, followed by retrieving an object by multiple robots working together. The objective is to speed up the process of search and retrieval. To this end, two robots are used to work together and communicate with each other. During the exploration process, users can also use voice commands to wirelessly control the robots.

The rest of this paper is organized as follows. Section II presents system overview & description. Section III discusses the overall exploration process and robot enhancement. Section VI and Section V present the proposed algorithms and the details of exploration using robot "Roli" and robot "SIX", respectively; Section VI provides experiments and demos; and Section VII concludes the paper.

II. System Overview and Description

In this research, we adopt two E-Robots: SIX (Fig. 1(a)) and Roli (Fig. 1(b)) [6]. Roli is a customizable planetary rover style robot kit; while SIX is a customizable robot with 6 legs and 12 degrees of freedom. Fig. 2 illustrates the system overview. Its working procedure is:

1) User starts the system using PC/Phone via arrow ***a***.
2) PC/Phone sends commands to SIX and Roli to start searching via arrow ***b***, arrow ***c*** and arrow ***d***.
3) SIX and Roli start searching in arrow ***e*** and arrow ***f***.
4) SIX helps Roli find the target, and when Roli locates the target, it sends back the video streams to PC/Phone asking whether or not to grab the target via arrow ***c***, arrow ***d*** and arrow ***b***.
5) User checks the video and then sends voice commands to PC/Phone to let Roli grab the target via arrow ***a***.
6) Roli grabs the object via arrow ***f***, and asks whether or not to return the target to the destination via arrow ***d*** and arrow ***b***.
7) User sends voice commands to PC/Phone to let Roli return the target.
8) Roli returns the target to the destination in arrow ***g***.

(a) SIX Hexapod (b) Roli

Figure 1: EZ Robots.

In order to let multi-robots work cooperatively to retrieve a specified object from different areas, the tasks have been divided between the two robots. SIX is a scout for Roli, and Roli is the robot to actually pick up the object. The robots then start on opposite ends of a hallway and start scanning the room number on the wall. The objectives of this research include: making multiple robots navigate a series of areas; enabling robots to transmit back the video signal to the system; providing voice control to users; developing algorithms for robots to retrieve specific objects from a random room; developing methods for communication between multiple robots; Wireless cell phone control. The system architecture, shown in Fig. 3, includes three modules: Communication, EZ-Builder Interface and Functions.

In the "Communication" module, all of devices will be connected either through WI-FI or Bluetooth or other communication techniques so that they can communicate with each other. Among these devices, there is a very important device "Router" which makes multiple connections between one controller and multiple robots possible.

"EZ Builder Interface" is a Program IDE for developers. Its functions are to collects data from robots and users so they can interact with each other.

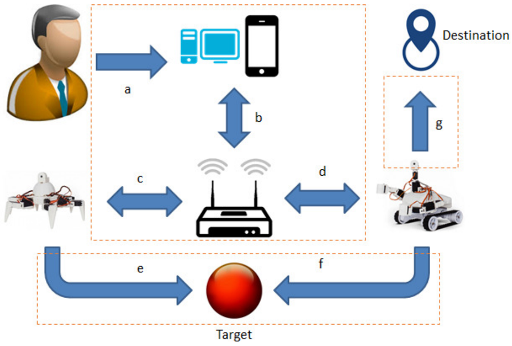


Figure 2: System Overview

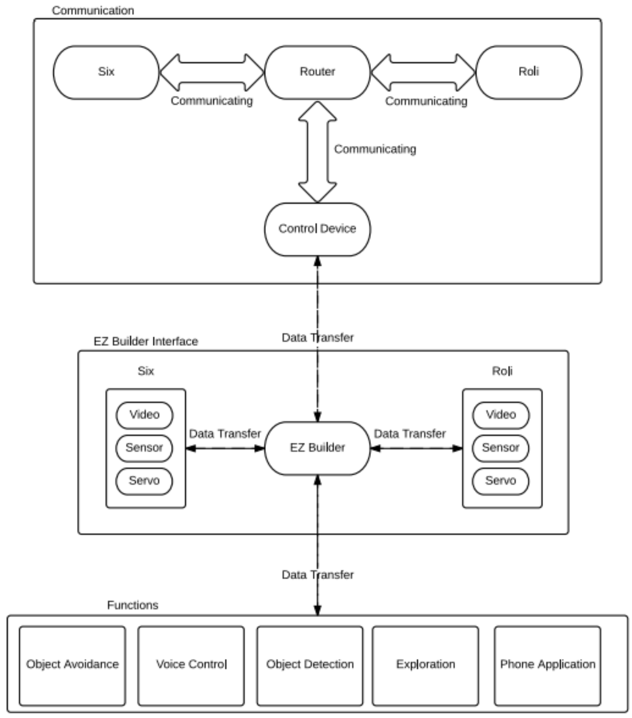


Figure 3: System Architecture

We have developed five functions in the "Functions" module, and each function is introduced below:

1) Obstacle Avoidance: The EZ-Robot does not come with this function. However, obstacle avoidance is essential for the robot to perform a search task. Therefore, we developed an efficient algorithm for obstacle detection and allow the robot to move forward, left, right or backward to avoid collision.
2) Voice Control: The EZ-Robot has a speech interface that can recognize several basic voice commands. We extend the interface by adding more voice commands such as "Robot Search", "Robot Grab", etc.
3) Object Detection: The camera of the EZ-Robot is used to detect an object. To achieve a high success rate of object detection in real time, we developed an algorithm that identifies the target object based on the object's color and size information.
4) Exploration: This is the most advanced function that combines object avoidance, object recognition, grabbing, signal process, communications, coordination, and other functions together. We have developed an exploration algorithm for each robot.
5) Phone Applications: we developed some phone functions to wirelessly control the robots, like moving forward, moving backward, etc.

ISBN: 978-988-14048-2-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Due to the limited space, we focus this paper on multiple robot cooperative explorations.

III. Exploration Process & Robot Enhancement

A. Overall Exploration Process

The whole process of exploration (Fig. 2) consists of three major steps. The 1st step is communication, including arrow a, b, c, d. This step includes human-robot communication and robot-robot communication through Wi-Fi.

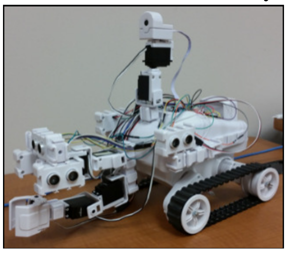
The 2nd step is searching and grasping (e and f). In this step, we proposed two navigation algorithms for the two robots in order to search for the object. In addition, the robots are able to detect the target and report the location to the system. Users can also use voice commands to control the robot to perform different actions. For example, the user can verify if the object detected by the robot is correct by examining the image from the camera. If it is, the user confirms by saying "Yes" and the robot will grasp the object. Otherwise, the user says "No" to instruct the robot to ignore the detected object and continue the search.

The 3rd step (arrow g) is to return the target object. After the object is grasped, the robot will ask the system whether the object should be returned. If "Yes", the robot will deliver the object to the destination. In this process, the robot will use a navigation algorithm to exit the room and detect the destination object return. If the user says "No", the robot will wait for 15 seconds and asks again to double check before giving up the object return task.

B. Robot Enhancement

In order for the robots to perform the object detection and grasping task better, we have made some modifications to the EZ-robots (See Fig. 4). For Roli, two of its three arms were removed to reduce the chance of blocking its view when exiting a room with the target object in its gripper. The servos originally used by the two arms were moved forward and used by the remaining arm to allow the robot to grasp the target object and move it out of the way of its front sensors to avoid occlusion. We placed the sensors away from the body of Roli to allow for navigation and to allow the navigation algorithm to predict where Roli needs to go next. Additional servos were added to the front of the robot to allow the robot to reposition its sensors (e.g. ultrasonic sensors) to avoid view occlusion. This makes it impossible for the robot to use the sensors for obstacle avoidance when exiting the room. In addition, two ultrasonic distance sensors have been added in two sides of the robot body. They are used to avoid collisions with obstacles.

For SIX, we have made some major modifications to its design in order to fit the needs of our project. The distance sensors on its left side were mounted in a way to keep them in parallel. The front sensor is installed in the same way. Another servo was used by the SIX to scan the sensors. .



(a) Roli

(b) SIX

Figure 4: Enhanced Roli and SIX

VI. Roli Exploration

Roli starts searching by reading a specific QR code attached to the room so as to locate and enter the correct room.

A. QR Code

Quick Response Code (QR code) is the trademark for a type of matrix barcode first designed for the automotive industry in Japan [7]. A barcode is a machine-readable optical label that contains information about the item to which it is attached. The reason why we use QR code is that the EZ Builder supports QR code reading from the signals transmitted from robot's video. By using the EZ scripts (similar to C language), we can read the data from a QR code.

B. Roli Search with Interactive Voice Control

The Roli's job in this project is object retrieval. We adjusted Roli's position so that it can read the QR code reliably. SIX is not able to make such fine movements, which leaves SIX in a more supportive role. Fig. 5 illustrates the start point at which the Roli reads the QR Code.

When starting a search task, the Roli first asks for permission from the user. After the user responds with "Yes" to the system, the Roli starts the searching task. If the user says "No", Roli will wait for 15 seconds and ask for permission again. Because the QR codes were installed according to the height of the SIX's camera, Roli will lower down its camera to read a QR code.

C. Roli—Camera Search

After reading a QR code, the robot enters the search mode as previously described. In this mode, it will periodically scan the room to search for the target object by first looking straight ahead (0 degree) and then scanning back and forth at a specific angle by using the neck servo. The scanning angle is incremented after each scan at an interval of 20 degrees until it reaches 180 degree. After this point, the scanning angle return to 0 degree and a new scan period as described above starts.

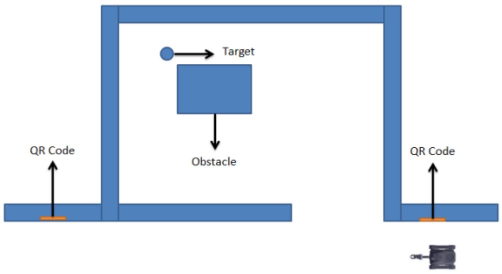


Figure 5: Start Point - Roli Read QR Code

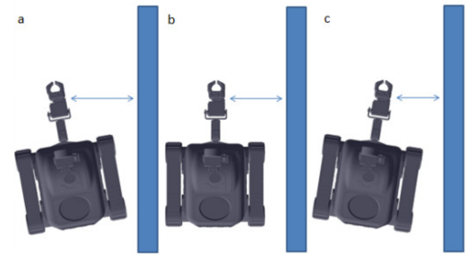
D. Proposed Roli Navigation Algorithm

If no target object is found after scanning the room thoroughly, the Roli robot will follow the perimeter of the room by using its ultrasonic sensors. The wall-following navigation method uses the walls as landmarks to maintain a known heading direction. Without using the landmarks, it is difficult for the robot to move along a straight line or turn a specific angle (e.g. 90 degrees). Over time, the robot will lose track of its heading and get loss.

The objecitve of this algorithm is to keep the distance between the robot and the wall within an acceptable range, e.g., 10-15 cm. The main sensors involved in this process are the three front sensors, whose readings are constantly updating to keep Roli from running into the wall. As Roli

ISBN: 978-988-14048-2-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

moves forward, the front right sensor will tell whether it is approaching the wall. This is done by setting some threshold values and making the robot move right forward if it is too close to the wall (Fig. 6a) or move left forward if it is leaving the wall (Fig. 6c).



a. Tilt to Left b. Normal c. Tilt to Right

Figure 6: Roli Detects Distances from the Wall.

The scheme to correct the heading deviation is as follows: (1) If the front right distance to the wall is bigger than the first threshold value 12 cm, the robot will move right forward for a short time (see Fig. 6a). (2) If the distance is even bigger than the second threshold value 15 cm, Roli will move right forward in for longer period of time. (3) If the front right distance to the wall is smaller than a threshold value 10, Roli will move left forward for a short time (Fig. 6c). (4) If the distance is even smaller than the second threshold value 6, meaning Roli is too close to the wall, the robot will reverse first and then move left forward for a longer period of time. (5) If the distance is in an acceptable range between 10 and 12 (Fig. 6b), Roli will move forward and update its sensors readings. (5) If the front distance to the wall is smaller than 30, Roli will turn left in a short time (and keep updating its front distance) until the front distance is greater than 30 cm.

To further ensure that a sufficient front clearance can be maintained we use another threshold value 20 cm and add another rule that whenever the front distance is smaller than 20 cm, Roli will reverse and turn left in a period of time. This rule enables Roli to handle different shapes of rooms. Roli stops moving periodically at the point when the total number of right-forward movement and left-forward movement reaches 20 and uses its camera to scan its surroundings to see if the object is in the room. If it spots the object, it will execute a script that sends Roli into object retrieval mode. The proposed exploration algorithm is summarized below:

1) Set the count to 0 to count the number of times that Roli updates its sensors' distances.
2) Roli updates distances.
3) System compares the Front distance with its threshold value. If the Front distance is greater than its threshold value, goes to the next step. Otherwise, turn left.
4) System compares the Right2 distance with its threshold value. If Right2 distance is greater than its threshold value, goes to the next step. Otherwise, turn left forward.
5) System compares the Right1 distance with its threshold value. If Front distance is greater than its threshold value, turn right forward, otherwise, turn left forward.
6) The count is incremented by 1 each time Roli updates its sensors. If the count is smaller than 20, go to step 2). Otherwise, Roli exits this algorithm and starts searching for the target object.

E. Roli—Process of Approaching Target

When Roli finds the target, it first asks the permission from the user to grasp the object. If the user says "Yes", then Roli will run "Grasp Target" script. Otherwise, the robot will ignore that object and continue searching.

Roli enters into the object retrieval mode with the confirmation from the user. In this mode, Roli attempts to keep the object in the center of the camera's image plane while moving towards it. Once the robot is close enough to the target, it attempts to grasp the object.

The robot detects the target object by processing the image from the incoming video stream. Reliable object detection is one of the most challenging aspects of this project. After some tests, we decided to use color information for object detection. The reason is that an object with a distinctive color can be reliably spotted from a large distance. The only issue. However, this approach is susceptible to background noise. To overcome this problem, a specific red color was chosen for the target object so that a specific threshold value could be used for object detection. To further narrow our margin of error improve the object detection success rate, we use objects with specific size as the targets.

After object is detected, Roli will attempt to pick up the object. To align the camera with the target object, the camera's image is divided into nine sections (Fig. 7), and the robot adjusts its position to ensure that the object stays in the Middle sections of the image. The sizes of each nine sections are set manually. Ideally, the Middle Middle section should account for 70% of the height and 30% of the width of the image. This is because in order to let the gripper better grab the object, keep the object as closed as possible is a key step to achieve it. If the cap is in the Middle Left/Right section, the robot will keep turning right/left to adjust the cap's position on the image plane until it is in the center. Once the cap is centered, the robot will move forward until it reaches the target. As the robot approaches the target, the cap moves downwards on the image. Therefore, the robot lowers its camera to keep the object in the center. At the same time, Roli's front distance sensor keeps updating the distance to the target. When the distance is below a certain threshold, Roli will make sure that the object is centered and then move to grasp the object.

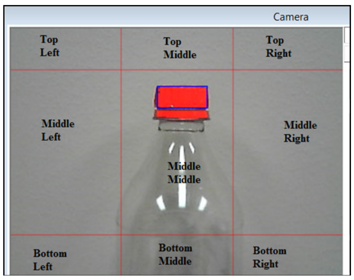


Figure 7: Camera Grid (Example)

F. Roli—Object Grasping and Release

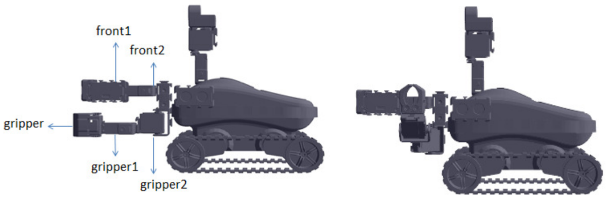
Roli detects that it is the time to grasp the target object If the front distance (measured by the front distance sensor) is less than 3 cm, In this case, the robot starts the object grasping procedure.

Once the target is grasped, Roli moves it out of its camera's view by using its front servos. This is because the object is so high that it may block the front ultrasonic distance sensor that must be used for exiting the room. Therefore, the robot moves the gripper together with the

ISBN: 978-988-14048-2-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

object to the left of the front distance sensor to avoid obstruction. The five servos used in this process are: front1, front2, gripper, gripper1 and gripper2 (Fig. 8). The grab process is summarized below:

1) Move front2 40 degrees up and wait for 500 milliseconds.
2) Move gripper2 77 degrees to the left.
3) Move neck2 to 90 degrees.
4) Move gripper2 to 20 degrees.
5) Move front2 to 90 degrees.
6) Move gripper1 to 80 degrees.



(a) Roli Before Grab (b) Roli After Grab

Figure 8: Roli Grab

Releasing the target object is an opposite process as follows: 1) Move gripper1 to 0 degrees; 2) Move front2 to 40 degrees; 3) Move front1 to 95 degrees; 4) Move gripper2 to 85 degrees; 5) Move front2 to 80 degrees; and 6) Move gripper to 30 degrees.

G. Roli—Return with Interactive Voice Control

After grasping the object, Roli will ask “Do you want me to return it?”. If Roli receives “No” voice command, it will wait for 15 seconds, then ask the permission to return the target again. If the user says “Yes”, then it responds with “No problem”. It will back up a few times and then move forward and right until it has found a wall. Once it has found the nearest wall, it starts exiting the room using our Navigation Algorithm. Meanwhile, the neck will head down to scan the QR code. Once Roli reads the QR code that is opposite to that it scanned for entering the room, Roli will head up to look for the blue destination marker.

The destination object was tracked similarly to the target by using a specific color and size to determine where the destination is. When it locates the destination marker, it will approach it similarly as it approached the target object. Once it reaches the destination marker, it releases and deposits the object and then shutdown.

H. Roli—Abnormal Retrieval with Voice Control

If the target is not in the room where Roli locates, Roli will have to go to that room to grab it. If SIX finds the target first, it writes a file with the room number for Roli to read. After reading the file, Roli will say: “SIX has found the object, do you want me to grab it?” When the user says “No”, Roli will ignore and delete that file and continues to search. Otherwise, Roli will enter the Scan QR Code mode and scan for the exit QR code. Once it exits the room, it will scan for the QR code that matches the room specified by SIX.

If it encounters a QR code that does not match the one specified, it will ignore it and continue searching. When Roli detects the specified QR code, it will put its head up and enter into search mode to search the target. When Roli finds the target, it will approach it and grab it. Then, Roli will head down to scan the exit QR code and follow its left wall to exit the room. Once Roli reads in the exit QR code, it put its head back to its normal position and search the blue destination marker. Compared to search for the object by a single robot, the collaborative search scheme reduces the time to find the object.

V. SIX Exploration

The way SIX navigates through a room is quite different from the way Roli does. This is due to that SIX’s unique walk cycle makes it difficult to produce accurate locomotion. Most importantly, SIX doesn’t have enough pins in its Ev4 controller to connect servos like Roli does. Accordingly, SIX is not required to pick up any objects. But scout and communicate the target objects’ location to Roli.

SIX begins on the opposite end of the hallway to Roli and will scan the wall for QR codes. As it walks down the hallway, it will be correcting its path continuously to walk along a straight line. When it has detected a QR code, it saves the room number into a buffer and then executes a script that allows it to enter the room.

A. Proposed SIX Navigation Algorithm

We also developed the navigation algorithm for SIX. Once SIX has entered the room it goes into a searching mode, in which SIX executes the Navigation Algorithm to follow the perimeter of the wall. The algorithm is designed to keep SIX’s trajectory in parallel with the wall. The basic idea is to compare the readings from the two ultrasonic distance sensors located in the same side of the robot.

First, if SIX detects a wall in its left. It will first analyze if SIX is too far from the wall. If it is, SIX will move left to get closer to the wall. Otherwise, it enters into a function called Parallel Movement. The rationale for adding this function is that it is impossible for SIX to make an exact 90º-turn. To address this issue, we wrote a script to allow Six to adjust its position and heading so that its movement is in parallel with the wall. This is done by using two sensors mounted on its left/right side. In Parallel Movement, SIX adjusts its position in order to be in paralleled with the wall with a certain distance. If it’s too close, SIX will move away from the wall. Then it compares two distances on its left/right. If Left1/Right1 is greater than Left2/Right2, then SIX will turn left/right. Otherwise, SIX will turn right/left.

In addition, SIX repositions its camera away from the wall in searching mode to allow detection of any possible objects in the room. It then checks the state of its sensors. During the Parallel Movement, SIX can detect if there is anything in the front. If there is nothing in the front, SIX is in state 1. It will then move forward and check its state. If SIX detects that there is an obstacle in the front, it is in state 2 and it will execute a script to turn 90º to the right and move forward. Finally, if SIX detects nothing to its left, it is in state 3. In this case, will make a left turn and move forward. At the end of each state, SIX update its sensor readings and determine the state for the next step.

In order for this method to work, certain conditions must be met. The robot must retain a certain distance from the wall that it is following. Whenever this condition is not met, SIX will either move toward or away from the wall. As SIX follows the perimeter of the room, it is constantly looking for the target object. The SIX Navigation Algorithm is summarized below:

1) SIX first tries to find the target. It exits this algorithm if the target is found
2) Otherwise, it updates its distance sensors and detects if there is a wall in its left.

ISBN: 978-988-14048-2-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

3) If there is a wall, SIX adjusts itself to be in parallel with the wall.
4) Then SIX detects if there is anything in front of it.
5) If there is nothing in front of it, SIX is in state 1 and moves forward.
6) If there is an obstacle in front of it, SIX is in state 2 and turns right by 90 degrees.
7) If there is no wall in its left, SIX is in state 3 and turn left by 90 degrees.
8) Go to step 1.

B. Locating the Target

Once SIX finds the target, SIX sends back the room number to the system which writes a file stored in a specific location for Roli to read. This informs Roli to come to grab the target. Then, SIX curl up into a ball and shutdown to save battery. If Roli encounters SIX, it will treat SIX as an obstacle and walk around SIX.

VI. Experiments and Demos

Our extensive experiments demonstrate that our proposed system and methods work successfully as a whole. The communication between the two robots and wireless phone control are also efficient. The demos of the exploration process can be viewed through our website: http://sun0.cs.uca.edu/~yusun/NASA%20Website/linkedWebsites/Gerald%20Website/video.html.

VII. Conclusion and Future Work

In this research, we have developed a robotic exploration system which can control multi-robots to work cooperatively to explore different areas and search for an target object. We have developed two navigation algorithms for two different robots, as well as methods for avoid obstacles, object detection, voice control, communication, and wireless phone control.

Regarding future work, there is a lot of potential to improve and expand our current work, such as improving search speed; improving network model to explore entire campus; and involving more robots in the exploration.

References

[1] www.ifr.org/industrial-robots/statistics/
[2] In the Aftermath of September 11 What Roboticists learned from the Search and Rescue Efforts AAAAI press release.
[3] Fengliang Xu. “Mapping and Localization for Extraterrestrial Robotic Explorations”, The Ohio State University 2004
[4] Dudek. G.; Jenkin M.; Milios E.; Wilkes D. “Robotic exploration as graph construction”, Robotics and Automation, IEEE Transactions on (Volume:7 , Issue: 6).
[5] Jennings J.S; Whelan G.; Evans W.F. “Cooperative search and rescue with a team of mobile robots”, Advanced Robotics, 1997. ICAR '97. Proceedings of 8th International Conference on.
[6] EZ-Robot. www.ez-robot.com
[7] http://en.wikipedia.org/wiki/QR_code

ISBN: 978-988-14048-2-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)