Multi-robot System Using Low-cost Infrared Sensors

Anubhav Kakkar, Shivam Chandra, Deepanshu Sood, Ritu Tiwari, Anupam Shukla
Soft Computing and Experts System Laboratory, Indian Institute of Information Technology and Management Gwalior

Article Info

Article history:

Received Jan 31, 2013
Revised May 28, 2013
Accepted Jun 25, 2013

Keyword:

Autonomous Robot
Infrared Sensors
Intelligent System
Object Detection
Obstacle Avoidance
Path Planning
T-Cycle

ABSTRACT

This paper proposes set of novel technique, methods, and algorithms for simultaneous path planning, area exploration and retrieval, obstacle avoidance, and object detection and retrieval by an autonomous multi-robot system considering the use of low cost infrared sensors. Ultimate function of our low cost system is to efficiently explore given unknown area and simultaneously identify desired objects by analyzing their physical characteristics. Explanation of scenario is done by two autonomous robots equipped with low-cost and low-range infrared sensors that perform assigned tasks by analyzing sudden changes in their environment. Along with identifying and retrieving desired objects, the proposed technique provides an inclusive analysis of area being explored. Novelties presented in this paper provide cost-effective solutions of area exploration and finding known objects in unknown environments without using high cost long range sensors and/or cameras. Additionally, our methodology is reasonably fast and uncomplicated in performing inter-related tasks such as avoiding obstacles, analyzing area as well as objects, and reconstructing that area using collected and interpreted information for unknown environments. Proposed methods and algorithms are simulated over a complex arena to show the defined operations and manually tested for physical environments which provided 78% correct results against several random parameters set.

Corresponding Author:

Anubhav Kakkar,
Soft Computing and Experts System Laboratory,
Indian Institute of Information Technology and Management, Gwalior, India
Email: anubhav.iiitm@gmail.com

1. INTRODUCTION

The exploration of unknown environments is one of the most fundamental problems being studied in the area of mobile robotics since past many years. The main considerations of this problem are to have an efficient path planning, autonomous exploration of area, analyzing the characteristics of environment, and building the maps. Added to this is the problem of finding some known object (or an object with some known characteristics) in the unknown environment. The developments going on in this area have given several of the successful instances of implementation on single robots. A more challenging task is to accomplish these complex jobs on multi-robot system. Certainly, multiple robots have the capacity of completing these tasks more efficiently but at the same time the path planning and area exploration technique should be proficient enough to generate desirable results. There are numerous applications of solutions to these problems in real life scenarios. The examples may include finding some interesting pieces of rock on unknown planet, searching an unknown species of plant with some desirable characteristics in unexplored forest, expediting an archeological area to find desirable objects from lost civilizations, etc [1]-[3].

In this paper, we have demonstrated an effective methodology for simultaneous path planning, area exploration, area retrieval, obstacle avoidance, and object detection by a multi-robot system using the low-range infrared sensors to minimize the cost. The main aim of our study is to suggest some innovative means of analyzing different characteristics of an unknown area, searching a ‘desirable’ while differentiating them against numerous undesirable obstacles, and successfully storing and retrieving all the necessary information to reconstruct the area after completion of its mission. The prime focus of this research is to accomplish the solutions for above defined problems in the most cost-effective manner along with assuring its practical applicability in future.

The paper is organized as follows. After a brief introduction of the research problem, aims, and purposes of the research, a detailed survey of literature is presented in the next section. This is followed by our proposed methodology with detailed account of various aspects related to it divided into three sections namely path planning, area exploration and retrieval, and obstacle avoidance and object detection. Next section presents the algorithm explained in the previous section after which results originated from experimental implementations are discussed. Finally, major findings and future prospects are discussed to conclude the paper.

An extensive review of literature available suggests that although a sufficient number of researches have been done in the area of path planning, area exploration, obstacle avoidance, and object detection yet to the best of our knowledge there is hardly any prominent work existing in the direction of accomplishing these complex tasks with the help of low cost infrared sensors. However, it would be highly significant to mention some of the significant works done by various researchers:

The evolution of different path planning and area exploration algorithms has been influenced by various purposes and requirements. N. Sariff and N. Buniyamin compiled and presented different techniques for path planning of autonomous navigation vehicles on the basis of readings of provided map and moving towards goal by avoiding the obstacles [4]. Behravesh and Farshchi gave a randomized search technique for path planning of mobile robots in the repetitive dynamic environment [5]. Nawi et al. used the backward chaining process for guiding the robot to plan its path autonomously [6]. Dongsung Kim and Ramakant Nevatia also proposed an efficient approach for indoor navigation by using the s-map representation of the environment [7]. Frederic Bougrault et al gave an innovative mechanism for maximizing the accuracy of map building by adaptively selecting the control actions and maximizing the localization accuracy by utilizing the Occupancy Grid with feature based simultaneous localizing and mapping (SLAM) algorithm [8]. Similarly, several other approaches for area exploration of unstructured indoor environment for the purposes of indoor cleaning that have been proposed such as random path planning coupled with local CCPP by Yu Liu et al [9]. Marija Dakulovic and Ivan Petrovic used the occupancy grid representation of area with every free cell in grid representing a particular node in the graph that is being searched to find complete path [10]. Cao proposed a hybrid vision system consisting of one omnidirectional vision and one stereo vision for providing the environmental information for 2D grid and 2.5D grid mapping respectively for planning path with the help of hybrid maps [11]. Shujun Lu and Jae H. Chung presented a weighted path planning approach based on the collision detection in which they used base and wrist force/torque detection sensors to estimate the collision position and collision force on manipulator [12]. Tonglin Liu et al presented an auto-adapted path planning methodology for shape shifting robots by reconfiguring the ability of AMOEBA-I robot to potential field method by which the ability of shape sifting robots assign through the narrow spaces is studied by including corner detection using modified potential field method [13]. Jianming Guo and Liang Liu improved the D* algorithm for mobile robot navigation in the dynamic and unknown environment using the WiRobotX80 mobile robot. The optimal path given by them is based on searching the neighbor grid in 16 directions [14]. The problem of path planning and remote robotic sweeping for humanitarian demining was addressed in three levels by ManjulaHemapala et al. by utilizing mobility enablers, demining outfits, and exploring the remote govern options [15]. Another path planning and area coverage methodology proposed by F.A. Pujol et al uses the image based system that requires real-time responses to reduce computation times for robotic path planning. They used mathematical morphology and digital signal processing to develop the path planning and area coverage [16]. Tiwari et al. modified the A* algorithm by taking into the account pruning of frontier cells to increase the optimal allocation of path plan [17]. Pang et al. gave a bionic self-learning algorithm based on Q-learning and A* for robot navigation in unknown environment [18]. Kala et al. presented the fusion of hierarchical multi-neuron heuristic search (MNHS) with probability based fitness to make an efficient coverage of area by planning a path while avoiding the obstacles [19]. Iker Aguinaga et al developed a disassembly planning techniques along with two path planning techniques based on single transactions and generation of a random search tree [20]. Richard Bloss used simultaneous localization and mapping (SLAM) with a series of stereo cameras to gather the image of 3D area and then analyzing the images to form a 3D map and monitor the location of robot in the area [21].

Multi-robot systems have also emerged out as a powerful domain of robotic path planning and area exploration. Karasher Singh and Kikuo Fujimurat also came up with a proficient technique of map making using the cooperative mobile robots in known areas [22]. Hui Wang et al worked to design a novel technique for exploring an area by multiple robots by integrating the results of simultaneous localizing and mapping attained by individual robots in the multi-robot system [23]. Wihua Sheng et al proposed the distributed bidding model for coordinating the movement of multiple robots to accomplish the area exploration given that communication range of robots is limited. They provide two distinctive approach to accommodate limited-range communication problem namely consideration of distance between robots and map synchronization mechanism [24]. Xiaolei Yu and Zhimin Zhao also proposed a novel method for navigation of mobile robots on the basis of chaotic exploration and thinning-based topological map self-construction in which they made the environmental recordings in the process of robot navigation and exploration [25]. Pal et al. proposed the use of modified A* based algorithm for planning the navigation of multiple robots along with the concept of leader-follower using wireless connection to cover a given area [26]. Also, in another work, they proposed the classification of given area into two layers namely “exploration layer” and “exploration and checkpoint visit layer” to deal with communication constraints in multi-robot exploration [27].

After reviewing several pieces of literature we can conclude that substantial amount of work has been done related to all the domains of our research problem. Researchers have given various algorithms for path planning and area exploration by autonomous robots, multi-robot coordination, exploration using multi-robot system, etc. But, most of the algorithms developed and methodologies formulated assume the availability of high range sensors. Thus, due to scarcity of work done to utilize the properties of short-range infrared sensors which are indeed very inexpensive as compared to other types of sensors, we could proceed with seeking solutions to our research problem using the IR sensors.

2. RESEARCH METHOD

2.1 Path Planning

The path to be followed by the two robots would be in the form of numerous horizontal and vertical expeditions termed as *decrementing t-cycles* (which are later on explained later in this section) in the opposite axes. To further proceed with our discussion, it should be assumed that our multi-robot system is exploring a rectangular arena even though methodology can be easily expanded to plan the path for areas of several other shapes. Initially, the two robots have to be placed at two diagonally opposite corners of the area that is to be explored considering that if the first robot is at the position (0,0) and other robot is at the position (X,Y), then the area covered by this methodology would roughly be a rectangle formed by the points (0,0), (X,0), (X,Y), and (0,Y).For the sake of simplicity, let us consider that the robot which is initially positioned at (0,0) (upper-left corner) is “bot1” and robot at positioned at (X,Y) (lower-right corner) is “bot2” which is indicated in the Figure 1.

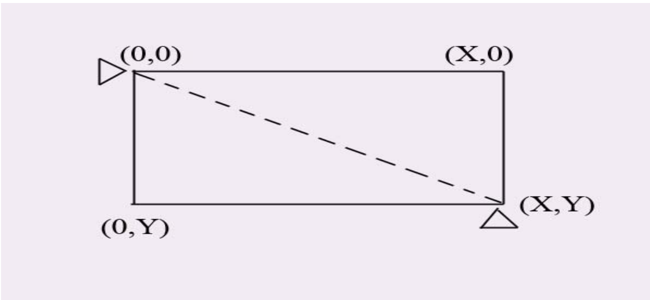


Figure 1. Area to be covered by the robots (absolute co-ordinate system)

Now, “bot1” should start its journey by aligning itself towards the positive side of x-axis and move forward in the same direction. Bot1 should keep on moving forward till one of its infrared sensor dedicated for detecting the end of area shows an infinite reading after which robot has to make a sharp right turn with clockwise movement of its right-sided wheels and anti-clockwise movement of its left-sided wheels and move forward for 0.1 seconds (for 200 rpm motors) (it should be noted that distance covered by the robot during this period when Bot1 moves in the y-axis should be extremely small as compared to the y-axis dimension of the area). After this, Bot1 should make another sharp right turn with the same mechanism and again start moving forward till its end-detecting sensors again detects the end following which the robot would be making a sharp left-turn by clockwise movement of its left-sided wheels and anti-clockwise movement of its right-sided wheels. We have termed this complete cycle as “t-cycle” [own reference] in our previous work with the first half part in which Bot1 move in the positive direction of x-axis termed as “positive half of t-cycle” and other half in which the Bot1 move in the negative direction of x-axis termed as “negative half of t-cycle”. The point at which robot switches from positive half to the negative half is termed as “transition phase of t-cycle”. Bot1 keeps on iterating this t-cycle several times either till the end-detecting sensor detects the end during the transition phase or desirable object is detected by either of the two robots.

Similarly, “bot2” starts its journey by aligning itself towards the negative side of y-axis and move forward in the same direction. It keeps on moving forward till its end-detecting sensor detects the end by sensing the infinite readings after which it makes a sharp left turn (similar to bot1) followed by a transitioning distance covered by the wheels for 0.1 seconds (for 200 rpm motors) (that should be extremely small as compared to the x-axis dimension of the area) and one more sharp left turn to begin the positive half of its t-cycle. Similar to that of bot1, bot2 makes numerous iterations of these t-cycles till the end-detecting sensor detect the end during the transition phase or object has been detected by either of the two robots. Furthermore, the responsibility of avoiding collision with Bot1 lies with Bot2 which is accomplished by checking if the next state of Bot2 is clashing up with the next state of Bot1. If the next state of Bot1 is clashing with the next state of Bot2, Bot2 stops for the time till Bot1 crosses this conflicting state and Bot2 is safe to move further to its next state.

However, in this algorithm the length of t-cycle is not constant but actually continually decrementing by the value equal to the distance travelled during the transitions by the other robot (note that distance traveled during the transition phase is same for every t-cycle for both the robots). For this, a variable “d” can be used to store initially store the length of first t-cycle calculated with the help of “quad encoders”, and then decremented every time a t-cycle is completed. This variable would be defining a sort of virtual wall for the robots encountering which they have to transit from *positive half of t-cycle to negative half of t-cycle in case of bot1* and *from negative half of t-cycle to positive half of t-cycle in case of bot2*. Therefore, with this path planning methodology, the two diagonal oppositely placed robots find their path using *“decrementing t-cycles”*. Again, it is to be understood that both the robots are making numerous iterations of the “path plan in one dimension” during any half of any t-cycle with a transition distance covered in the orthogonal axis which is extremely small as compared to the total dimensions of the area.

2.2 Area Exploration and Retrieval

The complete area is to be explored by two robots using the readings for their infrared sensors. We propose to use an array of five infrared transmitter-receiver pair installed on both the robots having infinitesimally very small distance relative to the dimensions of the area. As we know that transmitter of IR sender-receiver pair transmits some known intensity of infrared light and upon reflection from the surface of area underneath receiver receives the reflected light. According to the proportion of light received and transmitted infrared sensors generates some readings. We have already shown through experiments that a particular range of difference between the readings of IR sensors between the two instant of time represents a particular change in the color [28]. Considering the same work [28], we calibrated a table containing some fairly differentiable colors against their max and min values for infrared sensors’ readings. This table is to be referred by the robots to determine the colors of different parts of area.

Following is the explanation of how our robots would be exploring the area while recording the colors and objects/obstacles along with the position of the patches where they are occurring. Firstly, in order to determine the color of area as they are being passed by the robot, we would use the method of observing the change in readings of infrared sensors as they move from one state to the other. For this, we propose to use a variable “q” that records any change in the color in the area as robots progresses. To accomplish this, we need to use five variables say “p1, p2, p3, p4, and p5” on both the robots that read their respective values from five different infrared sensors installed in an array on both the robots. The readings can be taken at a delay of every 0.05 seconds and values stored in the respective variables of p-series. Then, the mean value calculated p-series is to be checked whether it lies in the range of a color (ranges of different colors can be seen in Appendix) that is different from the color where currently variable q lies. If it lies in a different range, the variable q is to be updated with that particular reading of p_{mean} and a new record containing the latest value of q along with the position is to be entered in the result file of respective robot.

The position of different patches with distinct colors within the area can be determined with the help of “quad encoders” installed on both the robots. The distances covered by the Bot1 and Bot2 in both the axes are to be recorded into the variables (x1, y1) and (x2, y2) respectively. For Bot1, the distance covered in x-axis is to be recorded by “quad encoders” while the distance covered in y-axis would automatically be known by keeping the count of number of t-cycles Bot1 has passed. However, for the Bot2, the distance covered in y-axis is to be recorded by quad encoders while the distance covered in x-axis would automatically be known by determining in which t-cycle currently the Bot2 is in the present state. In order to track of the current count and state of t-cycle, we need to use variables t1 and t2 (t1 for Bot1, and t2 for Bot2) which are to be updated by adding 0.5 each time they complete one half of the t-cycles. Now, if the floating part of the variables t1 and t2 is 5, then their respective robots are in the negative half of their t-cycles, while if their floating part is 0, then they are on the positive half. For Bot1, the value of variable x1 is exactly same as distance covered in the x-axis by Bot1 during positive half of its t-cycle while for the negative half it is "distance covered during the positive half of first t-cycle minus constant t1 multiplied by the distance covered by Bot2 during its transition phase minus distance covered by Bot1 in the negative half of the t-cycle". The value of variable y1 can be easily calculated by multiplying variable t1 by the distance covered by Bot1 in y-axis in its transition phase. While for Bot2, the value of variable y2 is exactly same as distance covered in the y-axis by Bot1 during negative half of its t-cycle while for its positive half it is "distance covered during the negative half of first t-cycle minus constant t2 multiplied by the distance covered by Bot1 during its transition phase minus distance covered by Bot2 in the positive half of the t-cycle". Contrary to Bot1, the value of variable x2 can be calculated by multiplying variable t2 with the distance covered Bot2 in x-axis during its transition phase. In this way, the Bot1 and Bot2 would be exploring the given area in collaboration with the patches explored by each one of them being in proportion with the orthogonal length of the axis to which they are aligned.

For retrieving this information after completion of journeys by both the robots, we would be using two result files stored separately on the two robots that could be combined to get the interpretation of complete area that has been initially unknown. As we are having five infrared sensors installed on both the robots dedicated for exploring the area, the result file for each robot would keep the track of five closely distant points. Now, as the robots move consecutively from positive half to negative half of the t-cycle, the result file would automatically have the information in the form of ten closely distant lines (five lines for one half and five lines for another half) to have records for complete t-cycles. Also, the average distance between any two lines is nearly 1.5cm (which is extremely small as compared to the dimensions of area) to make the breadth of a typical t-cycle nearly 15cm.

Finally, the area retrieval is to be performed by combining the information generated in the form of "meaningful records" by the two robots. We define records as entity containing the information about the color of a particular point, respective distances in the x-axis and y-axis, and if any object/obstacle is present at that particular point. However, robots would only be registering a new record either when change in the mean of readings by array of five infrared sensors is large enough to cross the range of color in which it was in the previous state or any object/obstacle is encountered. We would be assuming that characteristics of the area are not changing substantially for the area occurring between two meaningful records. Now, there can be different ways of recording the necessary information by the autonomous robots but we have adopted the following method. Both the robots would maintain a master file that would contain the records for every half of every t-cycle starting with "\\" followed by discrete data for different points where change of color is being detected or obstacle/object is being encountered along with their locations and finally terminated by "//". Both the robots repeat this pattern to reconstruct whole area. The master files collected from both the robots can further be manipulated for combining the map and redrawing the combined area using the pre-calibrated table. Area Exploration and retrieval is shown in Figure 2.

Figure 2. Area Exploration and retrieval

Figure 3. Object Detection, obstacle avoidance, and area exploration

2.3 Obstactle Avoidance and Object Detection

The obstacle avoidance and object detection are taken as parallel tasks in this methodology. Furthermore, this methodology assumes that "*every such object that is not desirable is an obstacle and thus needs to be avoided*". This assumption is considerable essential since for different missions the type of object required to be searched in an unknown environment may be different. Thus, an object which is "desirable" for a particular mission may not be desirable in other and thus becomes an obstacle for the robots. Also, the state of an object being desirable or undesirable may interchange during the same mission.

With this important assumption, we explain our methodology of detecting desirable object and avoiding undesirable obstacles using the infrared sensors. For this we recommend using two infrared sensors installed in the front of both the robots. Now, using these two infrared sensors we need to analyze the characteristics of the object in front of the robot and determine whether it is desirable or not. Hence, before sending the robot on the mission of finding the location of some desirable object in an unknown environment, we need to train our robots for some distinguishable characteristics of this object which they can use to differentiate the “desirable objects” from “undesirable objects”.

In our description, we have taken only two simple characteristics that can be analyzed with the help of low-cost and low-range infrared sensors. These characteristics are namely “color” and “shape” of the object. Therefore, before sending the robots for expediting the given unknown area, the robots have to be instructed about the desirable features which in our test case are color and shape of the object. The instruction can be in the form of a string like “Red Rectangle”, “Black Triangle”, etc however methodology can be expanded to cover complex objects having complicated shapes and multiple colors. Nevertheless, before the start of mission, both the robots have prior information about characteristics of desirable object and thus can simultaneously start the exploration of area and object detection. With the robots following the same path planning and area exploration methodology, a variable “f” continuously monitors the readings of infrared sensors dedicated for detecting obstacle/object (variable f1 for Bot1 and f2 for Bot2). Initially and for most of the time when there is no object in front of the object-detecting-sensors, the variable f has infinite reading allowing the robot to continue with its area exploration. However, as soon as the object-finding-sensors detect any object, the normal execution of t-cycle of the robot that has encountered the object halts for the time till these sensors analyze the characteristics of the object encountered. In order to analyze these characteristics in our test case, robot circumvent around the object. During the circumvent, the robot carries following two tasks: determining the color of the object using the same table used for exploring the area (see appendix) and tracing its own trajectory of path followed during circumvent to draw a rough shape for the same. After acquiring these two parameters robot matches them against the known characteristic parameters of desired object and if the match is found TRUE then it register this location as the location of desirable object or else ignores it by mapping a “hole” (proportion to the size of object) in its t-cycle. The circumvent path followed by the robots should be in such a direction so that it finally returns to its current t-cycle with the alignment of robot being in the same direction where it had previously been before the detection of object in front of it. This implies that Bot1 would circumvent around the object while keeping the track of distance covered in the y-axis which has traced back after circumvention is complete so as to get back to the same t-cycle from where it started. Similarly, Bot2 has to keep the track of distance covered in the x-axis so as to return to its original t-cycle.

Obstacle avoidance and object detection process being carried out by the robots along with area exploration is shown in the Figure 3.

3. SIMULATION MODEL

3.1. Path Planning

The following algorithm represents our path planning strategy as explained in the previous section. The path planning algorithm for both the robots is provided in two different functions running independently of each other.

```
Path_Planning_Bot1 ()
{
var t1=0; // to track the number of t-cycles. Bot1 starts from positive half of t-cycle
var end=false; // detect the end of area
align_x_axis (); // aligns the Bot1 parallel to x-axis
var stop=false; // flags the message if robot has to stop
varcount_states=0; /* count_states keeps the track of number of states in the first of first t-cycle and then
defines the maximum boundary for every t-cycle. It constantly decrease by 1 after completion of each t-cycle.
*/
while (!end)// first half of first t-cycle
{  move_fwd ();
   count_states++;
   end=read_end ();   }
t1++;
while (true) //further t-cycles
{
    if (t1%2==1) //end of positive half of t-cycle
    {   sharp_right_turn (); // making sharp right turn
         move_fwd (); // start forward move
```

```
        delay_Bot1 (200); /*moving forward for small distance that is infinitesimally very small as compared
        to y-axis dimension of area */
        stop= read_end (); // to stop the robot if end detecting sensor detects the end during transition phase
        if (stop)
        { System.exit (0); }
        sharp_right_turn (); // another sharp right turn to start new t-cycle
        while (!end) // negative half of t-cycle for Bot1
        {move_fwd ();
        end=read_end ();}
        t1++;   }
    if (t1%2==0)//end of negative half of t-cycle
    {   sharp_left_turn (); // making sharp left turn
        move_fwd (); // start moving forward
        delay_Bot1 (200); // allowing robot to move some distance forward
        stop=read_end (); // detecting the end during transition phase
        if (stop) // stop the robot if end is detected during transition phase
        { System.exit (0); }
        sharp_left_turn (); // starting the next positive half of t-cycle
        for (loop=1; loop<=count_states; loop++)  // moving the Bot1 till virtual wall (positive half of t-
        cycle for Bot1)
        { move_fwd (); }
        count_states--; // shifting boundary for next t-cycle
        t1++;   }
}}
Path_Planning_Bot2 ()
{
var t2=-1; // Bot2 starts from negative half of t-cycle
var end=false; // detect the end of area
align_y_axis ();// aligns the Bot2 parallel to y-axis
var stop=false; // flags the message if robot has to stop
varcount_state=0; // similar to that of Bot1 with the difference that it counts the boundary in y-axis
while (!end) //first half of first t-cycle
{  move_fwd ();
count_state++;
end=read_end ();   }
t2++;
while (true)//further t-cycles
{     if (t2%2==1) //end of positive half of t-cycle
    {     sharp_right_turn ();
          move_fwd ();
          delay_Bot2 (200);
          stop=read_end ();
          if (stop)
          { System.exit (0); }
          sharp_right_turn ();
          for (loop=1; loop<=count_state; loop++) // moving the robot till the virtual wall (negative half of t-
          cycle for Bot2)
          { move_fwd (); }
          t1++;   }
    if (t1%2==0) // end of negative half of t-cycle
    {      sharp_left_turn ();
           move_fwd ();
           delay_Bot2 (200);
           stop=read_end ();
           if (stop)
           { System.exit (0); }
           sharp_left_turn ();
           while (!end) // positive half of t-cycle for Bot2
           { move_fwd ();
```

```
        end=read_end (); }
        count_state--;
        t1++; }
}}
```

3.2 Area Exploration and Retrieval

The following algorithm represents our area exploration and area retrieval strategy as explained in the previous section. Our algorithm simultaneously takes the exploration and retrieval in one function which can be programmed identically in both the robots.

```
Area_explore_retrieval ()
{var q=null, p_mean;
varbot_id =1(or 2); //bot_id=1 if the algorithm is programmed on Bot1 else bot_id=2 if algorithm is
programmed over Bot2
var p1, p2, p3, p4, p5;
p1, p2, p3, p4, p5 ← read_irsensor1 (), read_irsensor2 (), read_irsensor3 (), read_irsensor4 (),
read_irsensor5 ();
q= p_mean= mean (p1, p2, p3, p4, p5);
store_new_record ((determine_color (q), absolute_pos (pos_x (), pos_y (), bot_id)); /* determine_color
(var x) returns the color according to the range in which value x is lying, pos_x () and pos_y () returns the
local positions with respect to the robot (either Bot1 or Bot2), absolute_pos (var x, var y) returns the absolute
positions with respect to common origin according to bot_id provided */
while (true)
{delay (50);//execution is delayed to take readings after every 0.05 milliseconds
    p1, p2, p3, p4, p5 ← read_irsensor1 (), read_irsensor2 (), read_irsensor3 (), read_irsensor4 (),
    read_irsensor5 ();
    p_mean= mean (p1, p2, p3, p4, p5);
    if (determine_color (p_mean)!=determine_color (q))//new record is stored only when the color is changed
    {
    q= p_mean;
    store_new_record ((determine_color (q), absolute_pos (pos_x (), pos_y (), bot_id));
    }
    if (ob_found ())
    { while (!ob_clear ()) //execution halts for the period till obstacle/object is cleared
        { delay (10); }
     }
    if (stop)
    { System.exit (0); }
}}
```

3.3 Obstacle Avoidance and Object Detection

The following algorithm represents our object detection and obstacle avoidance strategy as explained in the previous section. Object detection and obstacle avoidance is done simultaneously in this algorithm and the same function can be programmed identically in both the robots.

```
Object_detection_obstacle_avoidance (desired_shape, desired_color)
{
var f= null;
varbot_id = 1 (or 2); //bot_id=1 if the algorithm is programmed on Bot1 else bot_id=2 if algorithm is
programmed over Bot2
varob_clear = FALSE;
varob_found = FALSE;
While (true)
{
ob1, ob2 ← read_ob_sensor1 (), read_ob_sensor2 (); //for no object/obstacle ob1 and ob2 would remain
null
f = mean (ob1, ob2);
if (f!=null)
{
ob_found = TRUE;
```

initial_state = Bot_current_state ();
current_state= initial_state;
while (true)
{
Bot_next_state = find_next_state (max_f_reading (current_state)); // next state of robot is determined by finding that closest free position in which variable f has maximum value
current_state = execute_move (Bot_next_state);
temp_record_move (current_state);
record_ f_values(mean (ob1, ob2 ← read_ob_sensor1 (), read_ob_sensor2 ()));
if (is_aligned (initial_state, bot_id))
/ * function is_aligned () checks if the robot has been aligned parallel to its initial position when it started its circumvent along with the parameter of whether it is Bot1 or Bot2 */
{ob_clear= TRUE;
final_state=current_state;
break;
} }}}
shape_ob =determine_shape (access (temp_record_move ())); /* determine shape (file) access the temporary record of various states of the robot during circumvent and draw a schematic figure of the same in its temporary memory */
color_ob = determine_color (mean (access (record_ f_values ()))); // determine color () finds out the mean color of the object on the basis of f-values recorded by the robot during circumvent
if (shape_ob == desired_shape&&color_ob == desired_color)
{
record_pos_object (initial_state, final_state); // record the position of object as desirable object marking its initial and final positions if the shape and color of the object is found desirable
}
varob_clear = FALSE;
varob_found = FALSE; }

4. RESULTS AND ANALYSIS

The methodology and its subsequent algorithms proposed were implemented for a multi-robot system consisting of two identical robots having eight infrared sensors each over an arena having dimensions 250X200 cm. The surface of arena had total six colors distributed equally area-wise throughout the arena. Both the robots followed nearly the ideal planned path. As length-to-breadth ratio of arena was 5:4, the portion of area covered by Bot1 was approximately 45% in comparison to 55% by that of Bot2.

Table 1. Accuracy Results for Area Exploration & Retrieval

Color	Accuracy (Bot1)	Accuracy (Bot2)	Overall accuracy (Bot1+Bot2)
Black	100%	100%	100%
Brown	60%	70%	64.5%
Yellow	70%	80%	74.5%
Green	80%	90%	84.5%
Red	90%	80%	84.5%
White	100%	100%	100%

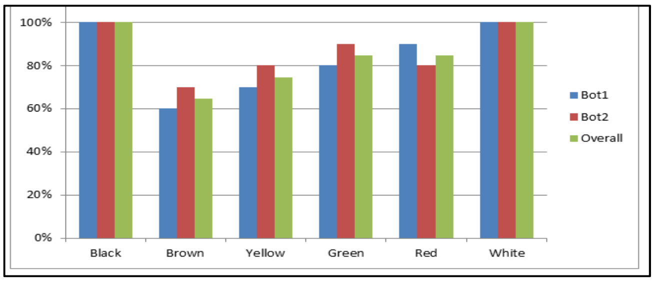


Figure 4. Accuracy results for area exploration & retrieval

After the exploration of area by both the robots, the master files from both the robots were obtained and combined to redraw the area by investigating the information contained in them. The accuracy of results for area exploration that is determining the correct color for different patches was overall 84.67% correct while the accuracy for object identification with desirable characteristics was nearly 90% correct. The Table 1 and figure shows the accuracy for detecting different colors by both the robots (all the colors were equally distributed throughout the arena).

Furthermore, the dimension of patch having a differentiated color from the neighboring patches was determined correctly for most of the time within the practical assumption that the information retrieved from the robots should reveal the basic schematic sketch of the actual shape of the patch. It was also found that if the color of the object is far from the color of its nearby ground in the Table 2, then color of that object is easier to detect as compared to those where their pair of color with that of the ground is close. Similarly, the change within the pair of colors that are more distant in the Table 2 is easier to detect for area as well.

The above obtained results show enough applicability of the proposed methodology and algorithms in the real-time scenarios for multi-robot systems. Furthermore, these results show several of practical considerations that need to be evaluated in order to use such low-cost and low-range infrared sensors. The implementation of this algorithm can further be extended to associate more number of characteristics (other than color and shape) with objects as well as the color of patches (for example green denoting grass, brown denoting soil, etc).

5. CONCLUSION

With the evolutions in the field of robotics, the tasks such as searching some defined object in an undefined area, expediting the unknown environments, etc are getting much interesting for modern day researches. In this work, we have shown an algorithmic approach for achieving these goals with the help of very short-range sensors. The intelligent system depicted in our work makes the use of a powerful methodology to overcome its hardware constrains of short sightedness while accomplish its task in a promisingly efficient way. Moreover, this work may be highly useful for on-going researches to construct low cost expedition devices that can go into the hostile environments to accomplish human desirable tasks.

Further researches in this direction may include testing the proposed methodology for more complex environments, exploring the undiscovered forests, tough mountain terrains, etc.

REFERENCES

[1] Chen J, Low KS, Dolan JM, Chien S, Thompson DR.*Decentralized Active Robotic Exploration and Mapping for babilistic Field Classification in Environmental Sensing*. 11th International Conference on Autonomous Agents and Multiagent Systems, Valencia. 2012: 44-52.

[2] Chan SK. *Distributed coverage with multi-robot system.* IEEE International Conference on Robotics and Automation, Orlando, FL. 2006: 2423-2429.

[3] Huang Y, Yuan J, Tao T, Sun F. *A Cooperative Approach for Multi-Robot Area Exploration*. IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei. 2010: 1390-1395.

[4] Buniyamin N, Sariff N. *An Overview of Autonomous Mobile Robot Path Planning Algorithms*.4th Student Conference on Research and Development, Selangor. 2006: 183-188.

[5] *Behravesh V, Farshchi MR.* A Novel Hybrid Algorithm for Robots Paths Planning.*International Journal of Robotics and Automation*. 2012: 1(4): 214-222.

[6] Nawi IM, Rahim MASA. *Autonomous Mobile Robot Path planning using backward chaining*. Advances in Electrical and Electronics Engineering - IAENG Special Edition of the World Congress on Engineering and Computer Science, San Francisco, CA. 2008: 182-191.

[7] Kim D, Nevatia R. *Representation and Computation of the Spatial Environment for Indoor Navigation*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA. 2004: 475-482.

[8] Bougrault F, Makarenko AA, Williams SB, Grocholsky B, Durrant-Whyte HF. *Information Based Adaptive Robotic Exploration*. Conference on Intelligent Robots and Systems, Lausanne, Swiherland. 2002: 540-545.

[9] Liu XLY. *Combined coverage path planning for autonomous cleaning robots in unstructured environments*. 7th World Congress on Intelligent Control and Automation, Chongqing. 2008: 8271 - 8276.

[10] Đakulovic M, Petrovic I. Complete coverage path planning of mobile robots for humanitarian demining. *Industrial Robot: An International Journal*, 2012: 39(5): 484 - 493.

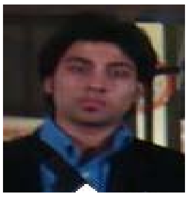
[11] Gu J, Cao Q. Path planning using hybrid grid representation on rough terrain. *Industrial Robot: An International Journal*.2009 :36(5): 497-502.

[12] Lu S, Chung JH. Weighted path planning based on collision detection. *Industrial Robot: An International Journal*. 2005: 32(6): 477-484.

[13] Liu T, Wu C, Li B, Ma S, Liu J. A novel auto-adapted path-planning method for a shape-shifting robot. *International Journal of Intelligent Computing and Cybernetics*. 2011: 4(1): 61-80.

[14] Guo J, Liu L. A study of improvement of D* algorithms for mobile robot path planning in partial unknown environments. *Kybernetes.* 2011: 39(6): 935-945.
[15] Hemapala M, Belotti V, Michelini R, Razzoli R. Humanitarian demining: path planning and remote robotic sweeping. *Industrial Robot: An International Journal*. 2009: 36(2): 146-156.
[16] Pujol FA, Chamizo JMG, Fuster A, Pujol M, Rizo R. *Use of mathematical morphology in real-time path planning*. *Kybernetes*. 2002: 31(1): 115-123.
[17] Pal A, Shukla A, Tiwari R. Multi Robot Exploration Using a Modified A* Algorithm. *Springer Intelligent Information and Database Systems*. Daegu, Korea. 2011: 506-516.
[18] Pang T, Ruan X, Wang E, Fan R. Based on A* and Q-Learning Search and Rescue Robot Navigation. *Telkomnika Indonesian Journal of Electrical Engineering*. 2012; 10(7): 1889-1896.
[19] Shukla A, Tiwari R, Kala R. Robotic path planning in static environment using hierarchical multi-neuron heuristic search and probability based fitness. *Journal of Neurocomputing*. 2011: 74(14-15): 2314-2335.
[20] Borro D, Aguinaga LMI. Path-planning techniques for the simulation of disassembly tasks. *Assembly Automation*. 2007: 27(3): 207-214.
[21] Bloss R. Simultaneous sensing of location and mapping for autonomous robots. *Sensor Review*. 2008; 28(2): 102-107.
[22] Singh K, Fujimurat K. *Map making by cooperating mobile robots*. IEEE International Conference on Robotics and Automation. Atlanta, GA. 2003: 254-259.
[23] Jenkin M, Dymond P, Wang H. Graph exploration with robot swarms. *International Journal of Intelligent Computing and Cybernetics*. 2009: 2(4): 818-845.
[24] Sheng W, Yangb Q, Tan J, Xi N. *Distributed multi-robot coordination in area exploration*. Robotics and Autonomous Systems. Amsterdam, The Netherlands. 2006: 945–955.
[25] Yu X, Zhao Z. A novel reactive navigation strategy for mobile robots based on chaotic exploration and TTM self-construction. *International Journal for Computation and Mathematics in Electrical and Electronic Engineering*. 2011: 30(2): 590-602.
[26] Tiwari R, Pal A, Shukla A. Multi-Robot Exploration in Wireless Environments. *Springer Cognitive Computation*. 2012.
[27] Tiwari R, Pal A, Shukla A.Communication constraints multi-agent territory exploration task. *Springer Applied Intelligence*. 2012: 1-27.
[28] Kakkar A, Saket, Punetha M, Shukla A. *A Novel Approach for Path Planning, Area Explore and Area Retrieval of An Autonomous Robot Using The Low-cost Infrared Sensors*. IEEE Conference on Industrial Electronics and Applications. Singapore. 2012.

BIOGRAPHIES OF AUTHORS



Anubhav Kakkar is currently a pre-final year student of Integrated Post Graduation program (B.Tech+M.Tech) in Indian Institute of Information Technology and Management Gwalior. His areas of interests include Robotics and Automations, Artificial Intelligence, and Computer Networks. He has published 02 papers in different international conferences/journals.

Shivam Chandra is currently a pre-final year student of Integrated Post Graduation program (B.Tech + M.Tech) in Indian Institute of Information Technology and Management Gwalior. His areas of interest include data communication, Robotics and Automations, and Robotic Visioning.

Deepanshu Sood is currently a pre-final year student of Integrated Post Graduation program (B.Tech + M.Tech) in Indian Institute of Information Technology and Management Gwalior. His areas of interest include Robotics and Automations, Robotic Visioning, and Soft Computing.

Dr. RituTiwari is currently the Assistant Professor in Information and Communication Technology department of Indian Institute of Information Technology and Management Gwalior. She has nearly eight years of teaching experience. Her areas of interest include Robotics, Soft Computing, Biometrics, and Artificial Neural Networks. She has published around 25 papers in various national and international conferences/journals.