A Team of Robotic Agents for Surveillance

Paul E. Rybski,Sascha A. Stoeter, Michael D. Erickson,
Maria Gini, Dean F. Hougen, Nikolaos Papanikolopoulos*
Center for Distributed Robotics
Department of Computer Science and Engineering
University of Minnesota Minneapolis, MN 55455
{rybski,stoeter,merickso,gini,hougen,npapas}@cs.umn.edu

ABSTRACT

This paper presents the hardware and software components of a robotic team designed for security and surveillance applications. The team consists of two types of robotic agents. The first type is a larger, heavy-duty robotic platform, called the "ranger." Rangers are used to transport, deploy, and supervise a number of small, mobile sensor platforms called "scouts," the second type of robotic agent. In an example scenario, the scouts are deployed into an office/lab environment, navigate towards dark areas, and position themselves to detect moving objects using their cameras. A ranger communicates with each of the scouts and determines whether there are objects of potential interest within the observed area. The paper also includes experimental results for individual scout and ranger-scout activities.

1. INTRODUCTION

For security and surveillance applications, an area is typically observed either by (1) multiple remote sensing devices that report to a coordination agent or (2) a mobile agent that patrols the required area. In both cases, the problem of adequate sensor coverage exists. In case 1, the problem is spatial: Are there enough sensors in the right locations? In case 2, the problem is temporal: Will the mobile agent be in the right place at the right time?

One possible solution is to combine the two approaches into one. A mobile agent that is capable of long distance travel can cover a large area and deploy smaller, less mobile agents in various locations. The smaller agents can be given the responsibility of sensing a small area and can have the flexibility to change their vantage points to make sure that all of their local area is observed. A coordination agent can then communicate with the sensing agents, query them for information, and move them remotely to increase the area viewed by them.

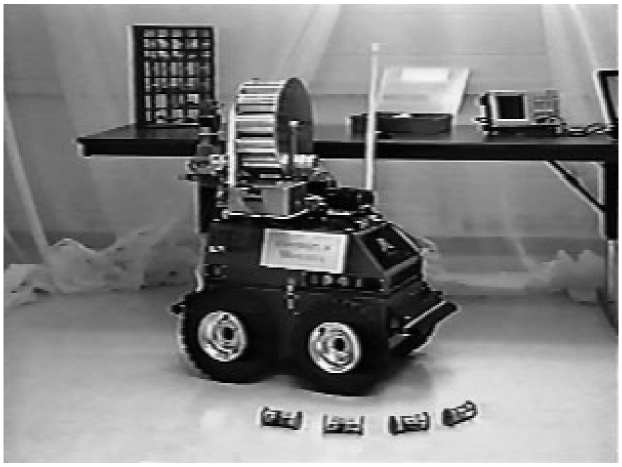


Figure 1: A ranger with four scouts in the foreground.

This is the solution that this paper suggests. The robots that are used are customized RWI ATRV-Jr^{TM}-based robots called "rangers" and a group of extremely small custom mobile sensor platforms called "scouts" (see Figure 1). Rangers are capable of navigating long distances without needing to recharge their batteries and are capable of navigating off-road terrain. Due to their size, however, not all areas may be accessible to them. The small size of the scouts makes them much easier to operate in these areas but presents a different set of problems including decreased range, battery lifetime, computational power, and sensing ability. By putting both kinds of robots into a team, the benefits of both can be achieved.

In this team, rangers are used as the primary navigational and computational resources. Their responsibilities include traversing the environment, selecting appropriate locations that are to be observed, and deploying the scouts into those areas. Once the scouts reach their designated locations, a ranger contacts each of the scouts in turn to analyze the area.

The paper is organized as follows: Section 2 presents related work, Section 3 describes the hardware components of the team, and Section 4 describes the software aspects. Experimental results are presented in Section 5 and analyzed in Section 6.

---

*Corresponding Author

2. RELATED WORK

Automatic security and surveillance systems using cameras and other sensors are becoming more common. These typically use sensors in fixed locations, either connected ad hoc or, increasingly, through the shared communication lines of "intelligent buildings" [16]. These may be portable to allow for rapid deployment [17] but still require human intervention to reposition when necessary. This shortcoming is exacerbated in cases in which the surveillance team does not have full control of the area to be investigated, as happens in many law-enforcement scenarios. Static sensors have another disadvantage. They do not provide adaptability to changes in the environment or in the task. In case of poor data quality, for instance, we might want the agent to move closer to its target in order to sense it better.

Mobile robotics can overcome these problems by giving the sensor wheels and autonomy. This research has traditionally focused on single, large, independent robots designed to replace a single human security guard as he makes his rounds [9]. Such systems are now available commercially and are in place in, for example, factory, warehouse, and hospital settings [10, 12, 15], and research continues along these lines [4, 13, 18]. However, the single mobile agent is unable to be many in places at once—one of the reasons why security systems were initially developed. Further, large mobile robots are unable to conceal themselves, which they may need to do in hostile or covert operations. They may also be too large to explore tight areas.

Multiple robots often can do tasks that a single robot would not be able to do or do them faster, as described in the extensive survey by Cao et al. [3]. The tasks traditionally studied with multiple robots are foraging [11], which involves searching and retrieving items from a given area; formation marching [1], which involves moving while maintaining a fixed pattern; map making [2]; and janitorial service [14], where robots have to clean a room in an unfamiliar building by emptying the garbage, dusting the furniture, and cleaning the floor.

Multiple mobile robots for security have recently been investigated [6]. In this case, the robots were meant to augment human security guards and fixed sensor systems in a known and semi-tailored environment.

3. HARDWARE COMPONENTS

The two hardware component types in the system correspond to the two robotic agent types, the rangers and the scouts.

3.1 Rangers

Rangers are based on the ATRV-Jr.^{TM} mobile robotic platform from RWI, a division of IS Robotics. The rangers are equipped with a novel launching mechanism (Figure 2), designed and built at the University of Minnesota, which is used to deploy scouts into the environment as needed. The launcher can hold up to 10 scouts in its magazine and is capable of deploying them up to a range of 30 m. Rangers can travel distances of up to 20 km, giving the scouts a much greater range than they would have if they needed to transport themselves. Once the ranger reaches its desired location, it can rapidly deploy the scouts into difficult to access places.

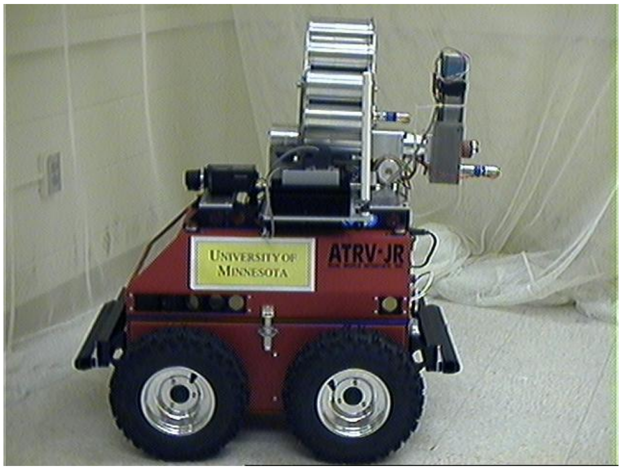


Figure 2: A ranger robot with scout launcher.

Each ranger is equipped with a 233 MHz Pentium-based PC running Red-Hat Linux which is linked to the robot's sensors and actuators with RWI's rFLEX^{TM} control interface. The rangers are controlled using RWI's Mobility^{TM} API (an object-oriented, CORBA-based modular robotics control architecture).

3.2 Scouts

Scouts are custom cylindrical robots 40 mm in diameter and 110 mm in length (see Figure 3) possessing a unique combination of locomotion modes. A scout can roll using its wheels (one on each end of its body) and a leaf spring "foot" mounted underneath for stabilization. It is also able to hop by winching its spring foot around its body and releasing it in a sudden motion.

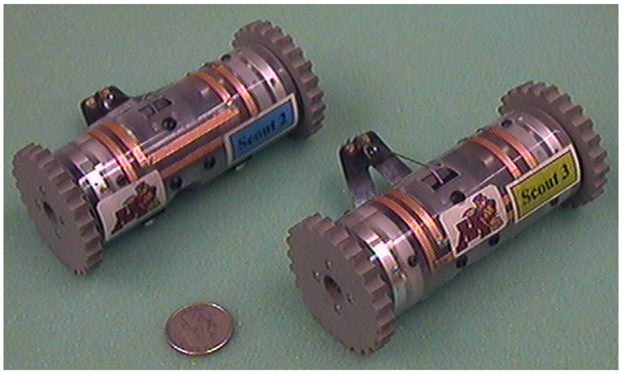
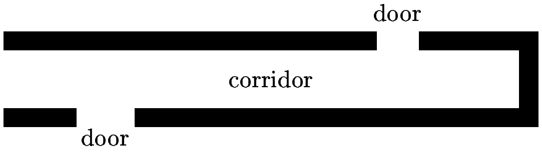


Figure 3: Two scout robots (shown next to a quarter for scale).

For the scenario presented in this paper, each scout possesses a miniature video camera and a wireless video transmitter. The camera consists of a monochrome single chip CMOS video sensor and a miniature pinhole lens. Video data is broadcast back to a receiver via a 900 MHz analog video transmitter. Each scout also possesses a miniature RF data



(a) Real-world layout.

(b) Robot perception.

Figure 4: The environment.

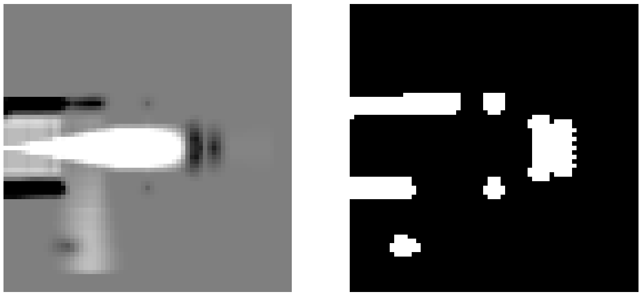
transceiver for receiving commands from rangers and transmitting status information back. Scouts are discussed more fully in Hougen et al. [7, 8].

4. SOFTWARE COMPONENTS

In order for the rangers and the scouts to coordinate their efforts and work together properly, a proxy-processing system has been developed which allows for the scout's control programs to run on a computer separate from the scout hardware. This is important because the small size of the scout's design severely restricts the speed and computation power of its on-board computer. The scout's limited on-board computational resources can only handle the most basic low-level control routines, such as setting the pulse width modulation frequency of the motor controllers, handling the sensor payload, and decoding information received on the inter-robot RF data link. High-level scout control is achieved by executing the scout motion control algorithms on a different computer. In the experiments described here, all of the scout programs run as separate processes on the ranger's on-board computers. These scout control programs send commands to the scout hardware through an RF data link.

While this proxy-processing scheme means that each physical scout *robot* is dependent on a ranger robot, this does not mean that the corresponding scout *agent* is not autonomous. Rather, the scout agent can be seen as distributed, with its sensors and effectors located in the scout robot and its behavioral controller located on board a ranger's computer. This takes advantage of the nature of artificial intelligent agents—they are not limited to a single physical location.

We have developed behaviors for a scenario in which rangers will find interesting areas to explore and deploy scouts into them. In our scenario a ranger is placed in a building to traverse the corridor and launch scouts into rooms that it finds along its path. A second ranger is used as a communication agent to coordinate the actions of the deployed scouts. The scouts must find dark areas in which to conceal themselves and watch for moving entities (such as people).



(a) Evidence grid. (b) After thresholding.

(c) Closest obstacles along path. (d) Area of right door.

(e) Hough space of area of right door.

Figure 5: Processing of an evidence grid.

4.1 Ranger Behaviors

Door detection and motion control are solely based on sonar input. Concurrent to the ranger's motion, sonar readings from two side-sonars and one front-sonar are integrated into an evidence grid [5]. Evidence grids partition the world into a number of discrete cells. Each cell carries a probability value describing the likelihood of that part of the world being free space. A sensor model expresses the effect of a reading from that sensor on the evidence grid. This allows for readings from different sensor sources to be combined into a unified model of the world. Here, the evidence grid covers an area of 4 m × 4 m centered around the robot where each cell is 6.25 cm on a side. The environment in Figure 4(a) is perceived by the ranger as depicted in Figure 4(b). White areas are considered free whereas black spots are likely to contain obstacles. Gray regions indicate insufficient knowledge to assume either occlusion state.

To identify doors or any other opening in a wall the evidence grid surrounding the ranger (Figure 5(a)) is treated as a grayscale image. Note that the ranger moves from left to right in the image. First, a threshold is applied to retain occluded regions resulting in Figure 5(b). Figure 5(c) shows the cells containing obstacles closest to the axis of motion. The remaining pixels to the left and right of the ranger are split into two sub-images, one such image shown in Figure 5(d), and then projected into Hough space to find lines in the image that correspond to the walls. Figure 5(e) shows the Hough space of the right side of the ranger. The darkest pixel in this image corresponds to the location of the wall with respect to the ranger. Lastly, openings are searched for along these lines within a dynamically chosen strip. If the opening is wide enough, i.e. about 1 m, then it is classified as a door.

When a door is detected, the ranger moves back to center itself in the door frame, turns to face the door and launches a scout. After successful deployment, it continues to seek out further rooms until all scouts have been exhausted from the magazine.

4.2 Scout Behaviors

Several simple behaviors have been implemented for the scouts. The only environmental sensor available to the scout is its video camera, the use of which presents several problems. First, the scout's proximity to the floor severely restricts the area it can view at a time. Secondly, since the video is broadcast over an RF link to the ranger for processing, the quality of the received video often degrades due to of range limitations, proximity of objects that interfere with transmission, and poor orientation of the antennas. Figure 6 shows an example image received from the scout's camera. The RF noise degrading this image increases the difficulty of distinguishing the objects from the background.

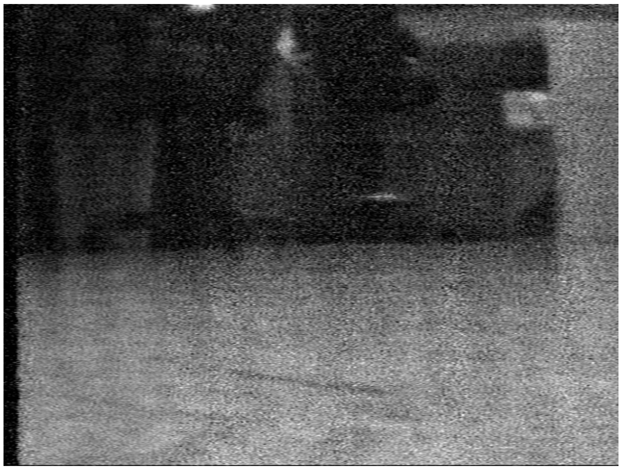


Figure 6: The world from the scout's point of view. Here the scout is viewing a lab bench and two chairs at a range of 2m.

The scout behaviors are:

Locate Goal: Determining the location of the darkest (or lightest) area of the room is accomplished by spinning the scout in a circle and checking the mean value of the pixels in the image. Since the scout has no encoders on its wheels to determine how far (or even if) it has moved, frame differencing is used to determine whether motion took place. The circular scan is accomplished in a number of discrete movements. Before each move, the scout must determine the quality of the video and set a threshold to filter out RF noise. This is accomplished by doing image differencing on a stream of video and increasing a difference threshold until RF noise is filtered out. Once the threshold is set, the robot takes an image, rotates for half a second, takes a new image, and subtracts the new image from the old one. A large difference indicates movement. There are several instances where this approach can fail, however. First, if the transmitted image quality is so low that motion in the image cannot be distinguished from noise. Second, if the robot is operating in an area of very low light or very uniform color, there may not be enough detail in the images to generate significant differences.

Drive Towards Goal: Identifying a dark area to move towards is a simple matter of scanning across the image at a fixed level on or about the level of the horizon and determining the horizontal position of the darkest area in the image. The mean pixel values in a set of overlapping windows in the image are determined. The scout selects the darkest window and drives in that direction. The scout knows that it should stop when its camera is either pressed up against a dark object, or the scout is in shadows. Scout motion in this behavior is continuous and the scout does not check its movements by frame differencing (unlike the discrete movements of the previous behavior). This is because the scout is unable to move very quickly. The difference between subsequent frames captured during forward motion is minimal, making it very difficult for the robot to detect forward motion.

Detect Motion: Detecting moving objects is accomplished using frame differencing. Once the scout has been placed in a single location, it sets its frame differencing noise threshold in the same way as described in the Locate Goal behavior. The scout then subtracts sequential images in the video stream and determines whether the scene changes at all (caused by movement in the image.)

Handle Collisions: If the scout drives into an obstacle, all motion in the image frame will stop. If no motion is detected after the scout attempts to move, it will invoke this behavior and start moving in random directions in an attempt to free itself. In addition to freeing the scout from an object that it has driven into, this random motion has an additional benefit. If the scout is in a position where the video reception quality is extremely bad, the static in the image will prevent the scout from detecting any motion (regardless of whether it is hung up on an object). Moving the scout changes the orientation of the antenna which may help improve reception.

5. EXPERIMENTS AND RESULTS

Three different experiments were devised to test the scout and ranger's ability to function in an environment, report back useful data and operate successfully as a team. The

Figure 7: Top view of experiment 1.

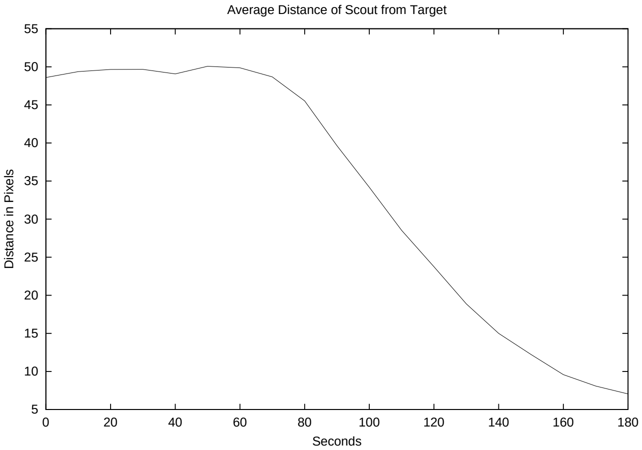


Figure 8: Experiment 1: Average distance (nine runs) of the robot from the target. Distance is in pixels, determined in Figure 7. 1 pixel is approximately 3cm.

first two experiments tested the ability of the scout to locate useful goals in various environments and move towards them. The third experiment tested the ability of the rangers and scouts to work together to achieve a useful goal.

5.1 Experiment 1

The first experiment was to determine, in a controlled environment, how well the scout could locate and move towards an appropriately dark area. This experiment was designed to examine the scout's behaviors in an analytical fashion.

For the first experiment, a controlled environment was constructed. This environment consisted of a 2.5 m × 3 m enclosed rectangle with uniformly-colored walls. A 1 m × 0.5 m black rectangle set up on one side of the environment as the target for the scout. The scout was started 1.5 m away from the center of the target.

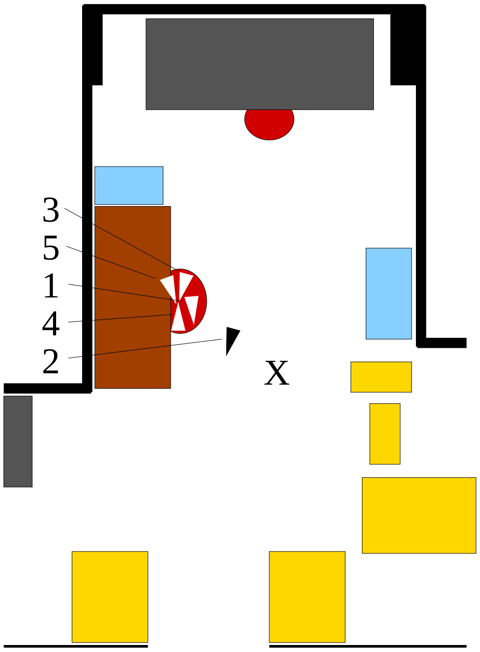


Figure 9: Experiment 2: Lab environment, showing locations of the scout for all five runs. X marks the starting position used in all runs and numbered arrows correspond to final position and orientation for individual runs. Ovals represent chairs under which scouts may hide. Chairs are positioned at a table and a lab bench, both of which also provide hiding opportunities. Other objects are impassable.

Nine experiments were run to see how long it would take the scout to locate the black target object and move itself towards it. A camera was mounted on the ceiling of the room and was used to view the progress of the robot from above. A simple tracking algorithm was used to automatically chart the progress of the scout as it moved towards the target. Figure 7 shows the view from this camera as well as a superimposed plot of the path that the scout took to reach its objective during one of its nine runs. In each case, the scout successfully located the target and moved towards it.

Figure 8 shows a plot of average distance the scout was away from the target vs. time for all of these runs. In the first 70-80 seconds, the scout uses its Locate Goal behavior to find the dark spot. Once it identifies it, the scout starts its Drive Towards Goal behavior until it comes in contact with the goal, somewhere between 150 and 160 seconds.

5.2 Experiment 2

The second experiment was set up to determine how well the scout could position itself in a more "real world" environment—a somewhat cluttered office or lab space. For these experiments, the scout's ability to locate a dark area was combined with the ability to turn towards the lighter areas and search for moving objects.

Two environments were used for this experiment. One was a lab environment with chairs, a table, lab benches, cabinets,

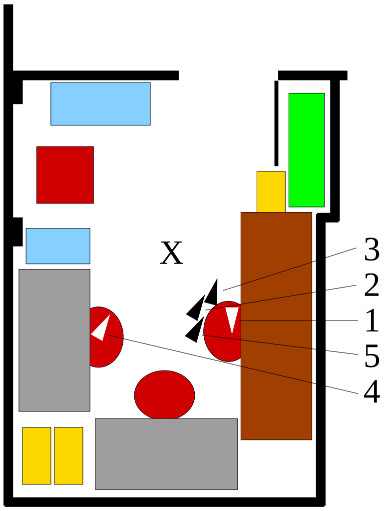


Figure 10: Experiment 2: Office environment, showing locations of the scout for all five runs. X marks the starting position used in all runs and numbered arrows correspond to final position and orientation for individual runs. Ovals represent chairs under which scouts may hide. Chairs are positioned at a table and at two desks, all of which also provide hiding opportunities. Other objects are impassable.

boxes, and miscellaneous other materials (see Figure 9). The other was an office environment with chairs, a table, desks, cabinets, and boxes (see Figure 10). The floor of the lab is a shiny, light tile surface of relatively uniform color whereas the floor of the office is a carpet of medium and dark piles providing a high localized contrast. This difference in surface brightness and contrast were accounted for in the scouts vision behaviors which were effectively self-calibrating. Five runs were conducted in each environment, using a fixed starting point for the scout in each room (shown as an X in Figures 9 and 10).

In four of the five runs in the lab environment, the scout chose the same chair under which to hide (locations 1 & 3-5 in Figure 9). On run number 2, however, the scout wound up roughly 0.5m out from under the chair in a relatively exposed position (location 2 in Figure 9). In all five runs the scout ended up facing towards a relatively bright area of the room. However, in run 4 this happened to be towards the rear of the room. Time required for these runs are given in column 2 of Table 1.

Similarly, in four of the five runs in the office environment, the scout chose the same chair as its destination (locations 1-3 & 5 in Figure 10). On run 4 the scout chose the other nearby chair (location 4 in Figure 10). In four of the five runs the scout wound up facing brightly lit areas roughly towards the door of the office. On run 1, though, the scout became physically stuck under the chair, forcing it to face the somewhat darker area towards the back of the room. Time required for these runs are given in column 3 of Table 1.

While collecting data, several experiments had to be aborted and restarted due to problems with radio communication. In these experiments, the scout found itself surrounded by objects that interfered with RF transmissions. In other experiments, the scout's batteries ran low and had to be replaced before data collection could continue.

5.3 Experiment 3

The third experiment was designed to determine if the combined scout/ranger team could carry out an entire surveillance mission. This mission combines all behaviors described above. The scouts are initially manually loaded into the launcher, mounted on Ranger 1. Rangers 1 and 2 are positioned as shown in Figure 11. From there on the actions of the team are autonomous. Ranger 1 moves down the hall, finds doors, and launches the scouts through doorways. Each scout, through proxy processing with Ranger 2, finds the darkest area visible from its landing site, drives to the dark area, turns around to face the more brightly-lit room, and begins watching for motion. In all of the experiments, the scouts were able to detect the motion of a person walking through the areas, either the lab or the office space. The final positions of the scouts are also shown in Figure 11. Time required for these runs are given in column 4 of Table 1.

Run	Lab Environment	Office Environment	Coordinated Actions
1	3	4	11
2	4	4	8
3	3	4	19
4	2	6	11
5	5	4	14

Table 1: Duration of Experiments 2 (Lab and Office Environments) and 3 (Coordinated Actions). Time in minutes.

6. ANALYSIS

As can be seen from the experimental results, the scout robots take a fair amount of time to reach their destinations. The scouts currently have a maximum speed upwards of 17 cm/s (10.2 m/min.) The majority of this reported run time is taken up by the execution of the Locate Goal behavior, during which the scout is only rotating in place. When executing this behavior, the scouts must spin in a circle and locate the darkest (or lightest) area in the environment. As their cameras have a very narrow field of view, the scouts must turn in a full circle in order to view the entire area. Because the scout does not have any encoders on its wheels, there is no way of knowing exactly whether or not it actually completed its rotation. Thus, the scout is programmed to spin in place approximately twice to guarantee a full rotation. In practice, due to the occasional command packet that is lost due to RF noise, the scout will spin anywhere between 1.5 and 1.75 times when executing this behavior. Once the scout has located a patch of environment which is "better" than the others, it must spin until it is oriented towards it again. Because the scout has no encoders, it cannot blindly turn back to the proper direction and start driving forward. The scout must actively search for the previously selected visual area. This two-step spinning can sometimes take up to two minutes to execute.

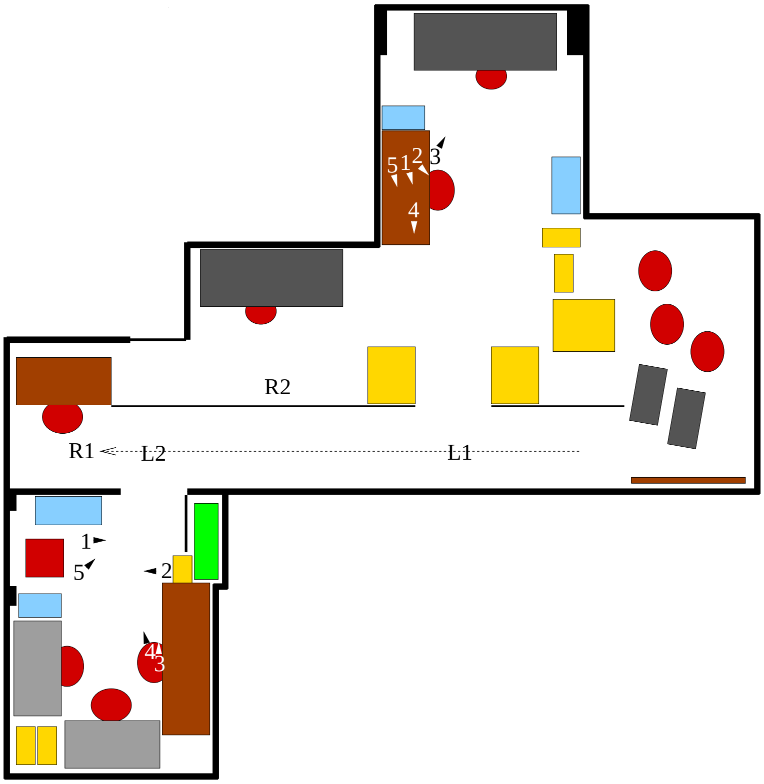


Figure 11: Experiment 3: Coordinated agent behaviors, showing path taken by Ranger 1, launching sites, and final location of scouts on all five runs (numbered arrows showing orientation and position). Ovals represent chairs under which scouts may hide. Chairs are positioned at tables, desks, and lab benches, all of which also provide hiding opportunities. Other objects are impassable.

When choosing a hiding place, the scout robot picks a direction which has the darkest average pixel value and moves towards it. In these experiments, the scout usually drove beneath the chairs or the desks. When approaching such an object, the robot would occasionally drive right into one of the legs of the piece of furniture, impeding its motion. The robot would believe that its goal had been reached (its camera appeared completely black) and would try to turn towards the lighter areas of the room (away from the dark object that it was pressed against). While rotating, the robot would often collide with the supporting leg again, causing the Handle Collision behavior to become active. The Handle Collision behavior would cause the robot to make small random motions until it moved freely again. Executing this behavior could take anywhere between 30-60 seconds, increasing the overall completion time of the task.

The Handle Collisions behavior was observed to activate in two other cases as well. The first case occurred when the scout was facing a relatively featureless wall. While spinning, the scout would detect no difference between the before and after views of the wall. This would activate the Handle Collisions behavior, usually moving the scout to a location that had more visual information to analyze. The second case occurred when the robot found itself in a place with a great deal of RF interference. After setting the threshold to remove the noise, the threshold would be too high to allow recognition of any robot motion. Believing that the scout was stuck, the Handle Collisions behavior would activate and would usually move the robot out of the RF "dead zone" and back into an area where it could transmit a cleaner signal.

7. SUMMARY & FUTURE WORK

The system as presented in this paper handles a task where cooperation increases performance by increasing reliability. By having its sensors spread throughout the environment with several agents, rather than concentrated on a single agent, there is less chance of an observation being missed. Further, because some of the agents are small and more easily hidden, even persons attempting to avoid detection by the system are more likely to be detected than in the case of a single, large robotic security guard. The controlling agent architecture is distributed in nature, allowing the controlling algorithms for the smaller "computationally challenged" robots to run on a computer separate from the physical body of the robot. This provides for greater flexibility and power for interpreting the environmental information provided to the robot.

Future work will be to improve our system to handle security and surveillance tasks that *require* cooperation. For example, in some cases we may want members of our team to work in locations beyond the range of their RF links to one another. That is, while increasing the power and, therefore, the range of the ranger radios would be highly beneficial, there will always be cases where we would need our agents to work at distances where direct communication between supervisors and their subordinates is not possible. We plan to use some team members as mobile communication relays (as well as mobile sensors) to achieve success in such domains. Finally, the problem of retrieving the scouts will be explored. Actuators could be added to the rangers which would allow them to retrieve the scout robots once the mission was completed. In the described experimental scenarios, there is no plan for recovery of the scout robots once they are deployed. In a field application, the scout robots could very well be disposable. Retrieving scouts is a challenging problem which is compounded by the fact that at the end of a mission, the scout robots may have drained their batteries completely. In this case, the scouts would not be able to rendezvous with the ranger for a pickup, requiring that the ranger actively search for them.

8. ACKNOWLEDGEMENTS

This material is based upon work supported by the Defense Advanced Research Projects Agency, Microsystems Technology Office (Distributed Robotics), ARPA Order No. G155, Program Code No. 8H20, issued by DARPA/CMD under Contract #MDA972-98-C-0008.

9. REFERENCES

[1] T. R. Balch and R. C. Arkin. Behavior-based formation control for multiagent robot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, Dec. 1998.

[2] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proc. of the IEEE Int'l Conference on Robotics and Automation*, 2000. To appear.

[3] Y. Cao, A. Fukunaga, and A. Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–27, 1997.

[4] R. Dillmann, M. Kaiser, F. Wallner, and P. Weckesser. PRIAMOS: An advanced mobile system for service, inspection, and surveillance tasks. In *Modelling and Planning for Sensor Based Intelligent Robot Systems*, volume 21 of *Series in Machine Perception and Artificial Intelligence*. World Scientific, Singapore, 1995.

[5] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer*, 22(6):46–57, 1989.

[6] H. R. Everett and D. W. Gage. From laboratory to warehouse: Security robots meet the real world. *Int'l Journal of Robotics Research*, 18(7):760–768, July 1999.

[7] D. Hougen, S. Benjaafar, J. Bonney, J. Budenske, M. Dvorak, M. Gini, D. Krantz, P. Y. Li, F. Malver, B. Nelson, N. Papanikolopoulos, P. Rybski, S. Stoeter, R. Voyles, and K. Yesin. A miniature robotic system for reconnaissance and surveillance. In *Proc. of the IEEE Int'l Conference on Robotics and Automation*, San Francisco, CA, Apr. 2000. To appear.

[8] D. Hougen, J. Bonney, J. Budenske, M. Dvorak, M. Gini, D. Krantz, F. Malver, B. Nelson, N. Papanikolopoulos, P. Rybski, S. Stoeter, R. Voyles, and K. Yesin. Reconfigurable robots for distributed robotics. In *Government Microcircuit Applications Conference*, Anaheim, CA, Mar. 2000. To appear.

[9] T. Kajiwara, J. Yamaguchi, Y. Kanayama, S. Yuta, and J. Iijima. Development of a mobile robot for security guard. In *Proceedings of the 15th Int'l Symposium on Industrial Robots*, volume 1, pages 271–278, Tokyo, Japan, 1985.

[10] A. Kochan. HelpMate to ease hospital delivery and collection tasks, and assist with security. *Industrial Robot*, 24(3):226–228, 1997.

[11] M. Matarić. Behavior-based control: Examples from navigation, learning, and group behavior. *Journal of Experimental and Theoretical Artificial Intelligence*, 9(2–3):323–336, 1997.

[12] T. Orwig. Cybermotion's roving robots. *Industrial Robot*, 20(3):27–29, 1993.

[13] A. Osipov, V. Kemurdjian, and B. Safonov. Mobile robots for security. In *Proceedings of the 1996 2nd Specialty Conference on Robotics for Challenging Environments, RCE-II*, pages 290–295, Albuquerque, NM, 1996.

[14] L. E. Parker. On the design of behavior-based multi-robot teams. *Journal of Advanced Robotics*, 10(6):547–578, 1996.

[15] C. Pellerin. Twenty-first century sentries. *Industrial Robot*, 20(2):15–17, 1993.

[16] J. Porteous. Intelligent buildings and their effect on the security industry. In L. D. Sanson, editor, *IEEE Int'l Carnahan Conference on Security Technology*, pages 186–188, Sanderstead, Surrey, England, Oct. 1995.

[17] D. Pritchard, R. White, D. Adams, E. Krause, E. Fox, M. Ladd, R. Heintzleman, P. Sprauer, and J. MacEachin. Test and evaluation of panoramic imaging security sensor for force protection and facility security. In *IEEE Int'l Carnahan Conference on Security Technology*, pages 190–195, Alexandria, VA, Oct. 1998. Larry D. Sanson, ed.

[18] M. Saitoh, Y. Takahashi, A. Sankaranarayanan, H. Ohmachi, and K. Marukawa. Mobile robot testbed with manipulator for security guard application. In *Proc. of the IEEE Int'l Conference on Robotics and Automation*, volume 3, pages 2518–2523, Nagoya, Japan, 1995.