Camera Placement and Multi-Camera Fusion for Remote Robot Operation

Brenden Keyes, Robert Casey, Holly A. Yanco
Computer Science Department
University of Massachusetts, Lowell
One University Avenue, Olsen Hall
Lowell, MA 01854
{bkeyes,rcasey,holly}@cs.uml.edu

Bruce A. Maxwell, Yavor Georgiev
Swarthmore College
500 College Ave.
Swarthmore, PA 19081
{maxwell, ygeorgi1}@swarthmore.edu

***Abstract*— This paper studies the impact of camera location and multi-camera fusion with real robots in an urban search and rescue task through two sets of experiments. In the first, we compared a camera with an overhead view to a traditional forward looking camera. In the second, we compared the use of a single forward looking camera to the use of two cameras, one on the front of the robot and one on the rear. Our experiments show that an overhead view that includes the robot chassis significantly increases the situation awareness of the operator as measured by the number of collisions during a maze traversal. We also found that having two cameras, one forward-facing and one rear-facing, results in improved situation awareness. The addition of the rear-facing camera also eliminates many of the collisions that typically occur in the back of the robot when using a single camera.**

I. Introduction

Robots used for urban search and rescue [USAR] are deployed in dangerous areas and controlled by a remote operator. The USAR task is particularly challenging for developers of (semi-)autonomous robot systems. A typical disaster site presents many challenges for robot navigation [3], [4], [10]. To further research in the area, USAR is entering its sixth year as a competitive research challenge for roboticists, with teams worldwide developing hardware and software platforms for the task. Most of the successful teams have used some form of teleoperation, with video as the primary means of giving the operator a sense of awareness of the robot's environment.

Situation awareness, the operator's knowledge of the robot's immediate and larger-scale environment, is one of the most critical factors in success at teloperation tasks [12]. To achieve better situation awareness (SA), teams have tried a variety of cameras, lens types, and camera locations with different degrees of success. Prior studies of USAR systems in competition have shown that particular combinations of factors provide an operator with more or less SA, but the complexity and variety of systems at a competition, combined with the limited number of runs, make it difficult to evaluate any single factor with experimental significance [18], [11]. A team with an optimal camera location, for example, may have poor navigational control or inadequate hardware for the terrain, rendering the excellent camera position useless with respect to completing the task.

One idea that appears to be successful across multiple teams is the concept of having an overhead camera that is capable of seeing the robot chassis [7]. The argument is that the ability to see the robot and its local environment gives the operator a better sense of the robot's location with respect to obstacles, victims, or other potential difficulties. Several USAR teams with overhead cameras have been quite successful in the competitions. However, the importance of the overhead viewpoint relative to other factors is not clear given the complexity of the USAR robot systems.

Wang et al. have undertaken experiments on camera orientation in a simulated USAR environment [16]. They compared the difference between gravity referenced cameras that maintain a single orientation relative to gravity and egocentric cameras that are fixed on the robot. In their study, the gravity referenced camera, which allowed users to perform the task better, also included a view of the front of the robot chassis to give the user a sense of the robot's orientation relative to gravity. In their discussion, they mention that they were not able to separate the effects of the camera orientation from the fact that the robot's chassis was visible in the overhead view, since they did not test the case of a gravity-referenced camera without the robot's chassis in view.

Hughes et al. [6] have undertaken similar studies of coupling and decoupling the robot and camera motion, including a case with two cameras and integrated views. Like Wang et al., they used the USAR simulator described in [15].

Another experiment using the USAR simulator tried to solve the "soda straw" view of the world normally provided by a single camera by piecing together images from five forward looking cameras into a cross [14]. When moving to real robots, sending video from five cameras will require a great deal of bandwidth. However, the study does show that better views of a remote environment assist in more effective navigation.

In this paper, we present the first studies of the impact of camera location and multi-camera fusion on actual robots in a USAR task. We devised two sets of experiments. In the first, we compared a camera with an overhead view to a traditional forward looking camera. In the second, we compared the use of a single forward looking camera to the use of two cameras, one on the front of the robot and one on the rear.

Our experiments show that an overhead view that includes the robot chassis significantly increases the situation awareness of the operator as measured by the number of collisions during a maze traversal. We also found that having two cameras, one forward-facing and one rear-facing, results in improved SA as measured by a reduction in the number of collisions from the one camera case to the two camera interface cases.

II. Hardware & Software Overview

We performed two sets of experiments on two robotic platforms, described below. The first system was developed at Swarthmore College for testing the use of overhead and forward looking cameras. The second system was developed at UMass Lowell for testing the use of multiple cameras.

A. Overhead and Forward Camera System

(a) (b)

Fig. 1. (a) iRobot Magellan Pro with forward and overhead cameras. (b) Operator station.

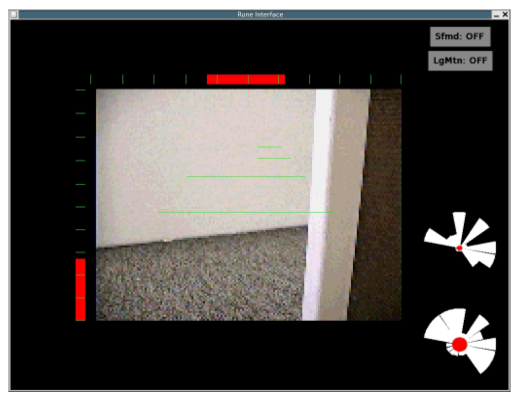
The platform used for the overhead vs. forward experiment is an iRobot Magellan Pro. The robot has three rings of sensors around the base: 16 bump sensors, 16 short range infrared sensors, and 16 sonars. In addition, the robot has an on-board 900MHz Pentium III, a wireless 802.11b bridge, and a framegrabber with three NTSC video inputs. The two cameras, shown in Figure 1(a), are Canon VC-C4 cameras with pan-tilt-zoom capability, controlled via serial port connections.

The operator station, shown in Figure 1(b) is a dual-processor Pentium IV system running Linux (Debian Sarge distribution) with a 19in LCD monitor and a Wingman Extreme USB joystick controller. All of the controls for the system that the user is permitted to access during the trials are available through the joystick and its associated buttons.

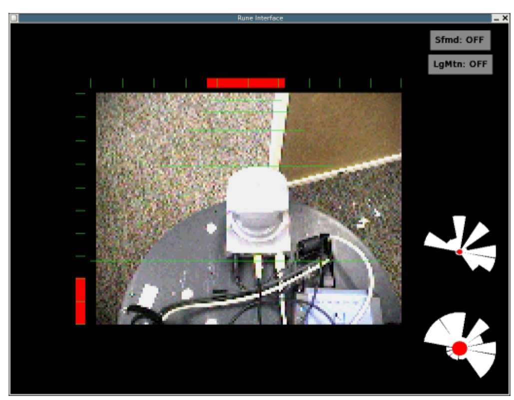
The software system for the Magellan consists of three modules running concurrently on the robot and the interface running on the operator station. All communication between the modules on the robot and the interface takes place through IPC, developed by Simmons [13].

The three modules on the robot are a Navigation module [Nav], a vision module [SVM], and a monitor program [Robomon] that watches the other two, starting, halting, or restarting them as necessary. For more detail on the navigation module see Maxwell et al. [9], and for the vision module see Maxwell et al. [8]. The Navigation module is capable of responding to velocity commands and executing them with a safe mode on or off. In safe mode, the robot will slow down as it approaches an object it detects using sonar and IR, and it will stop if it senses an object within 6-10cm. The robot ignores its sensors when safe mode is turned off.

The relevant capability of the vision module for this task is its ability to send compressed images with a variety of options through the IPC communication protocol. For the user trials, we used only the smallest and most highly compressed images. However, the user could choose between color or greyscale images on the fly using the slider on the Wingman joystick: the greyscale images are slightly faster to update, but some objects are difficult to detect without color. In addition, the vision module controls the PTZ cameras and can take directions for pan, tilt, and zoom actions from the operator interface via IPC messages. The operator controls the pan, tilt, and zoom features of the active camera via the buttons on the top of the main joystick handle. Finally, the vision module controls which camera is active. When switching between cameras is allowed, the operator can use a button on the front of the joystick to send a message to the vision module to switch cameras. The camera switch is completely executed in software, and the delay is very short (< 1s). Figure 2 shows views of the same scene from the forward (figure 2a) and overhead cameras (figure 2b).



(a)

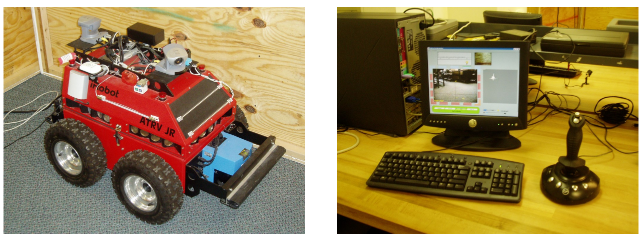


(b)

Fig. 2. (a) Forward camera and (b) overhead camera view of the same scene.

The operator interface, also shown in Figure 2, consists of a main panel showing the view from the currently active camera. Green bars overlaid on the video show 0.5m distances projected onto the ground plane and always stay pointing in the forward direction relative to the robot, even when the camera is panned to the side. In addition, the red bars above as well as to the left of the video window show the user the current pan-tilt position of the cameras. The width of the red bars indicates the zoom setting: a wider bar means a higher zoom setting. In the upper right corner, there are two text boxes. The upper box indicates whether safe mode is on or off. The lower box is a timer showing how long the user has taken so far on the current task. In the lower right of the screen are two displays showing the current readings of the sonar (upper) and the infrared sensors (lower). The white triangles show distance to detected obstacles (free space).

B. Front and Rear Camera System



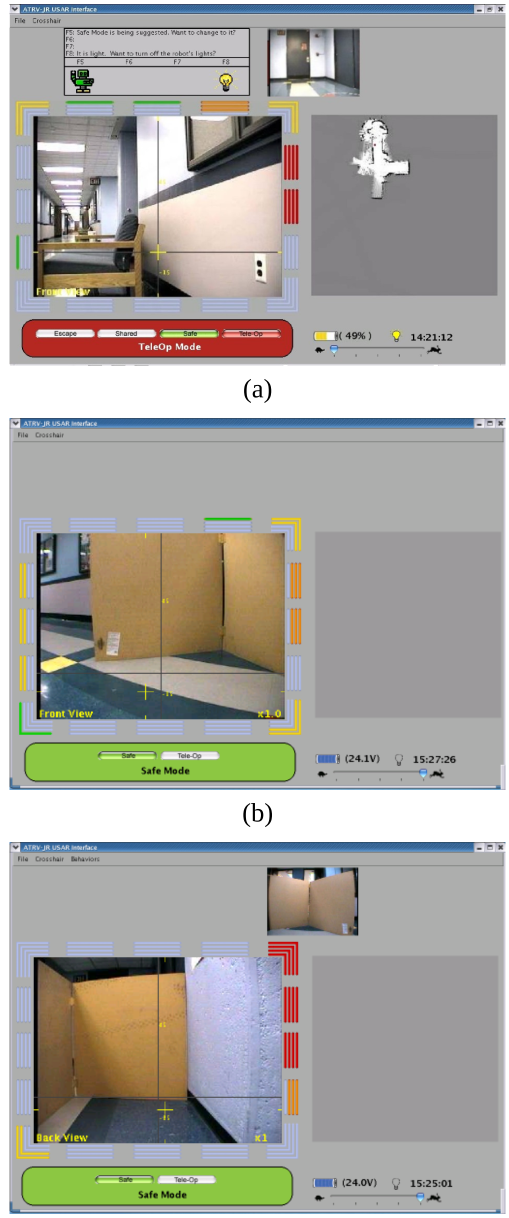
(a) (b)

Fig. 3. (a) iRobot ATRV-JR with front and rear cameras. (b) Operator station.

The robot platform used for the testing of the number of cameras is an iRobot ATRV-Jr robot (see , figure 3a). The robot has a 2.8GHz Pentium IV processor, a SICK laser rangefinder, a ring of 26 sonars, a lighting system, a four stream frame grabber, and two Canon VC-C4 pan-tilt-zoom cameras. The cameras are equipped with wide angle lenses, resulting in an 80 degree field of view when the camera is fully zoomed out (opposed to 47.5 degrees without the wide angle lens). One camera is mounted on the front of the robot and the other on the back.

The robot has four autonomy modes: teleoperation, safe, shared, and autonomous, based upon [2]. In the teleoperation mode, the operator makes all decisions regarding the robot's movement. In safe mode, the operator still directs the robot, but the robot will not allow the operator to drive it into obstacles. These two modes are similar to the autonomy modes on the system described above for the camera view tests; only these two modes were used in the experiments so the systems would have similar capabilities.

Using the rear camera, we have created an Automatic Direction Reversal (ADR) mode. We made it possible to reverse the robot's travel direction in a way that makes the front and rear of the robot virtually identical from the user's perspective. When the user switches to the rear (or front) camera view, the interface automatically remaps the joystick drive commands and the display of range information accordingly. This means that the user can drive the robot into narrow confines without having to back out; the user can simply select the opposite camera view and drive out as if driving forward. This is safer and more efficient than backing out or physically turning the robot around.



(a)

(b)

(c)

Fig. 4. (a) The full interface designed for the USAR system. (b) The simplified interface with a single camera view. The interface looked like this for both the single camera and switchable two camera experiments. (c) The simplified nterface with two camera views. The camera displayed in the larger window can be switched with the camera displayed in the smaller window.

The interface has been designed with the experience of three years of studying existing interfaces designed by other institutions for USAR [12], [17], [5], [18]. A major influence on our design philosophy comes from the observation that users of USAR interfaces become so absorbed in the video display that they ignore all other information on the interface. We exploit this behavior in our design by placing important information, such as ranging data, on and around the main video display to make it difficult for users to overlook. Our full interface is showing in figure 4a. Around the video window are indicators showing the current readings of the sonars (with red being the closest to gray the farthest). On the video window, crosshairs are drawn to indicate the pan and tilt positions of the camera. To the right of the large video window is a display of the map being built as the robot moves around the environment. Below the video window is the selection area for modes and below the map area is the speed control and battery indicator. Above the large video window is a display for the robot's suggestion system. (A full description of the robot system and interface can be found in [1].)

For these experiments, we modified the interface to make it more analogous to the interface used for the camera placement experiments. The two interface conditions used for the experiments are shown in figure 4b and c. In both, the large video window on the left shows the current camera view. Figure 4b shows a single video window; this interface was used for testing the single camera condition as well as the switched two camera display. Figure 4c shows two video windows; this interface was used for the two camera, two display condition.

In the two window interface, the primary camera (related to the current forward direction of travel) is shown the larger window. The smaller window displays a "rear view mirror" view of the world by inverting the video stream from the camera on the current "back" of the robot. The camera views can be switched using a button on the joystick. Switching the camera views invokes the ADR mode described above. The interface controls are shown in figure 3b.

III. Experimental Design

For both sets of experiments, we used the same experimental design. Each experiment had three conditions. The overhead and forward camera experiments were run on 19 subjects ranging in age from 18 to 45, with 9 men and 10 women. The forward- and rear-looking camera experiments were run on 19 subjects ranging in age from 18 to 50, with 11 men and 8 women.

1) A member of the research team would first explain the interface to the operator and demonstrate the use of the joystick and the features of the robot system. During the training phase, the robot was in the same room as the operator, which contained a small training maze. The same member of the research team trained all of the operators. Training generally took 10-15 minutes.
2) The operator would then navigate the robot through the training maze until he or she felt comfortable navigating.
3) The operator was then given a map of the test maze, which was in a separate room, and the task of finding three objects identified on the map. The map and test arena for the camera placement tests are shown in figure 5; the single vs. dual camera tests used a similar map and arena, with wider corridors to accommodate the larger ATRV-JR robot.
4) The operator would then undertake three runs. Each run used a different camera setup. For the overhead vs. forward tests, the three camera setups were forward, overhead, and switchable. For the one vs. two camera tests, the three camera setups were forward, a single window with a switchable view of the front and rear cameras, and two windows displaying the front and rear cameras. To ensure that a learning effect did not account for the variation in the data, we randomized the order in which the operators used the different camera configurations.
5) The task for each run was to go to each of three objects specified on the map. All operators completed the same three layouts which had unique start positions and object locations. To eliminate any learning effect, the ordering of the layouts was changed from operator to operator.
6) Observers mapped the path of the robot through the maze by hand, noting collisions and the time of each collision relative to the start of the run. In addition, we tracked time to each object, total time for each run, logged all commands send from the interface to the robot, and videotaped the operator interface.

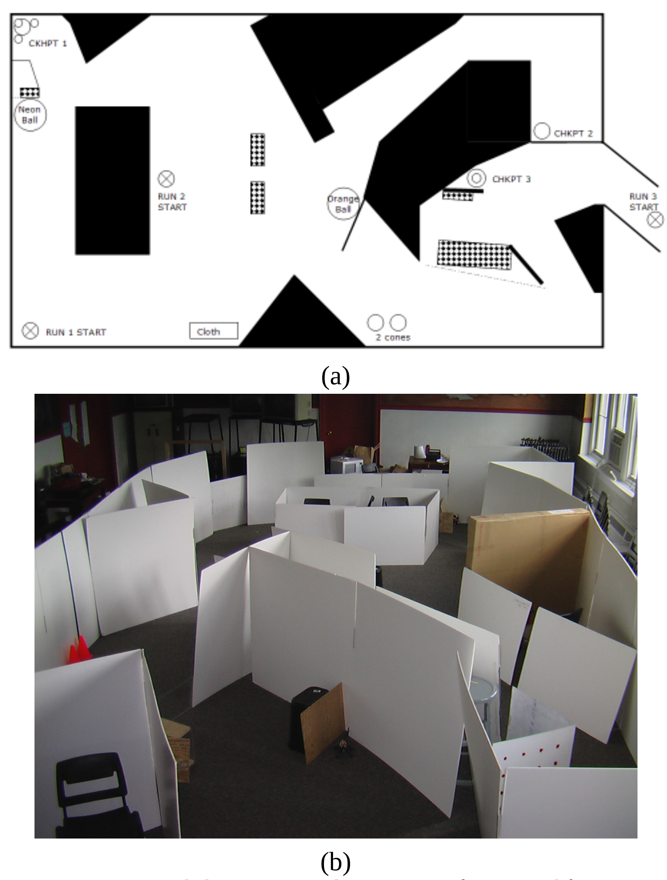


Fig. 5. (a) Map provided to operators. (b) Overview of maze used for testing.

IV. Results & Discussion

A. Overhead v. Forward Camera

For all subjects we compared total time on task and the number of collisions for the three camera setups. The means, standard deviations and standard errors are shown in table I.

Case	Time (s)	σ_t	$\frac{\sigma_t}{\sqrt{N}}$	Collisions	σ_c	$\frac{\sigma_c}{\sqrt{N}}$
Forward	435	168	39	4.6	2.0	0.47
Overhead	407	181	42	1.3	1.5	0.34
Switching	436	196	45	1.0	1.3	0.29

TABLE I
Comparison of Time and Collisions Results by Camera Configuration

While the overhead camera produced a mean time that was about 10% faster than the other two camera cases, this difference is not significant ($p = .42$ for overhead vs. forward with a paired t-test, $df = 18$; $p = .50$ for overhead vs. switching). For the number of collisions per run, however, the overhead and switching cases are significantly different from the forward camera case ($p < .001$ for both cases, while there is no significant difference between the overhead and swtiching cases ($p = .48$)). In fact, having access to the overhead camera virtually eliminated collisions for this task for almost half the subjects. Eight operators had zero collisions for the overhead case; eight operators had zero collisions for the switchable case, and six operators had no collisions in either case. Of the six people with zero collisions in both the overhead and switchable cases, their average number of collisions using the forward camera was 3.3, with a minimum of 2 and maximum of 5. All subjects had more collisions using the forward camera than in the overhead or switchable cases.

We also analyzed the data based on the run ordering. Table II shows the results based on the run order. Time on task drops by less than a standard deviation between the runs, but does show a consistent trend as people learn the interface and the task. Note, however, that the number of collisions drops only slightly from run to run, and the differences are not statistically significant. Therefore, we can conclude that the reduction in collisions is primarily due to the camera configuration and not due to operators learning the system.

Case	Time (s)	σ_t	$\frac{\sigma_t}{\sqrt{N}}$	Collisions	σ_c	$\frac{\sigma_c}{\sqrt{N}}$
Run 1	510	210	48	2.7	2.5	0.58
Run 2	396	140	32	2.2	2.1	0.48
Run 3	372	157	36	2.0	2.3	0.54

TABLE II
Comparison of Time and Collision Results by Run Order

The final analysis looked at the case where operators had the choice of using either camera. Table III shows the average amount of time spent using each camera and the average number of switches. On average, operators used the overhead camera almost three times more than the forward camera. The difference is statistically significant ($p < .001$), demonstrating that, overall, the operators used the overhead camera more than the forward camera when they had a choice. In fact, the eight operators who had no collisions during this run used the overhead camera on average five times more than they used the forward camera (365s versus 76s), an even stronger preference than that demonstrated by the group as a whole.

Variable	Mean	σ
Time on Forward Camera	120s	138s
Time on Overhead Camera	318s	204s
Number of Switches	11.7	13.7

TABLE III
Analysis of switching configuration

The aggregate numbers, however, do not well represent the population, which had multiple modes. For example, of the 19 operators, nine spent less than 15% of their time using the forward camera, and only one of those switched cameras more than 5 times the entire task, giving an average of less than 4 switches per run for people using the overhead camera at least 85% of the time. The other ten operators spent at least 20% of their time using the forward camera, and switched cameras on average 16 times. Only three of 19 operators used the forward camera a majority of the time, and the operator who used the forward camera the most still used the overhead camera 27% of the time. In contrast, two operators used the overhead camera for the entire run without switching at all.

These findings all support the conclusion that an overhead view that potentially includes the body of the robot significantly decreases the number of collisions operators make with the environment.

B. Number of Cameras

Both of the two camera cases resulted in improved situation awareness, measured by the number of collisions with the environment; results are shown in table IV. The most significant difference ($p < .02$ for a two-tailed paired t-test, $df = 18$) is between the single camera configuration and the two camera display. There is also a significant difference between the single camera configuration and the switched camera display ($p < .04$). The number of collisions in the two camera configurations are not significantly different ($p = .67$).

Case	Time (s)	σ_t	$\frac{\sigma_t}{\sqrt{N}}$	Collisions	σ_c	$\frac{\sigma_c}{\sqrt{N}}$
1 Cam	401	132	31	5.4	3.2	0.73
2 Cams, Switched	433	145	34	3.9	2.7	0.63
2 Cams, 2 Disps	375	154	36	3.6	2.7	0.62

TABLE IV
Comparison of Time and Collisions Results for One vs. Two Cameras

There is no significant difference between camera configurations for the time needed to complete a run (see table IV). We appear to see a learning effect with the time needed to

Case	Time (s)	σ_t	$\frac{\sigma_t}{\sqrt{N}}$	Collisions	σ_c	$\frac{\sigma_c}{\sqrt{N}}$
Run 1	442	137	32	4.3	3.1	0.70
Run 2	414	153	36	4.8	2.9	0.66
Run 3	353	133	31	3.8	3.0	0.68

TABLE V

COMPARISON OF TIME AND COLLISION RESULTS BY RUN ORDER FOR ONE VS. TWO CAMERAS

complete runs based upon the averages (see table V). However, these differences are not significant. Additionally, we do not see a learning effect with the number of collisions; there is no significant difference in the number of collisions based upon the order of the runs.

Case	Back Hits (mean)	σ
1 Cam	0.58	0.84
2 Cams, Switched	0.21	0.42
2 Cams, 2 Disps	0.5	1

TABLE VI

NUMBER OF COLLISIONS OCCURRING IN THE BACK OF THE ROBOT

Our results show that situation awareness in the back of the robot is improved for the two camera conditions, whether or not the rear camera is currently being displayed. Having the ability to see behind the robot reduces the number of collisions, as shown in table VI. However, the difference between the back hits for the single camera case is significantly different for only the switched two camera interface ($p < .02$). When both cameras are displayed, the hits are not significantly different than the single camera case ($p = .17$). We believe that the difference is due to the fact that although both windows are displayed, the user still focuses on the primary video window, missing things in the rear camera view. Additionally, when the user switches the camera that is displayed in the primary window, the ADR mode is turned on, making the direction of travel change with respect to the camera being used. In the two camera display, we found that subjects would back up using the smaller "rear view" video window, resulting in additional collisions.

Variable	Mean	σ
Time on Forward Camera	317s	177s
Time on Backward Camera	128s	99s
Number of Switches	7.7	7.2

TABLE VII

ANALYSIS OF SWITCHING CONFIGURATION: SINGLE VIDEO WINDOW, TWO CAMERAS.

We analyzed the number of switches made between the two cameras in the single and double window cases (see tables VII and VIII). In both cases, the subjects spent more time using the forward camera than the rear camera. This result is not unexpected, as most operators will spend more time moving the robot forward.

Variable	Mean	σ
Time on Forward Camera	259s	149s
Time on Backward Camera	113s	131s
Number of Switches	4.9	5.3

TABLE VIII

ANALYSIS OF SWITCHING CONFIGURATION: TWO VIDEO WINDOWS.

One of the subjects did not switch to the rear camera in either of the two camera interfaces. The other eighteen subjects made at least one camera switch in the single display case. However, we found that three of these eighteen subjects did not switch cameras at all in the two display case. One of these subjects noted that he didn't need to toggle the cameras because he had both views on the interface.

The subjects were asked which camera view they preferred. Ten of the nineteen preferred the front camera, seven of the nineteen didn't have a preference, and one subject preferred the back camera, saying that it was "better for some reason."

Five of the ten people who preferred the front camera said that they liked it because they could see the front bumper when they tilted the camera down all the way. (The laser rangefinder in the front of the robot makes the front bumper stick out more than the back bumper; the rear camera can not be tilted down enough to see the back bumper.) When the front camera is tilted down, the operator can see the whole bumper and 2-3 inches on either side of the bumper. We found that the five users who had the strategy of looking at the bumper to localize the robot in the environment had fewer collisions (mean: 8.0 collisions, standard deviation: 4.1) than the other fourteen subjects (mean: 14.7 collisions, standard deviation: 6.6). This finding correlates with the results obtained while testing with the overhead camera.

We found that most of the hits occurred not on the robot chassis, but on the tires; 75% of the times that the robot hit an obstacle with its front, it hit with the tires. These tires lay just outside the visible area and widen the robot by about five inches on each side. Despite warnings by the instructor, users continually went on the assumption that the boundaries of the video reflected the boundaries of the robot. Also of interest, we found that 71% of all hits occurred on the tires.

Fifteen of the nineteen subjects, or 79%, preferred the interface with two camera displays. Three of the subjects preferred the interface with two cameras that could be switched in a single video window. Two of these subjects had little computer experience, leading us to believe that they might have been overwhelmed by the two video windows. The final subject expressed no preference between the two interfaces with two cameras, but did prefer these two to the single camera case. No subject preferred the single camera case.

When asked to identify the best feature(s) of the interface, the subjects mentioned pan-tilt-zoom (8 of 19 subjects), switching between cameras (6), the sonar display (4), safe mode (2), teleoperation mode (2) and the crosshairs showing the pan and tilt of the camera (1). When asked to identify the least favorite features of the interface, the subjects stated the control for the pan-tilt-zoom (5), having no view of the tires (3), needing to hold down the joystick trigger to move the robot (3), and safe mode stopping their progress (3). Four could not identify a least favorite feature.

V. Conclusions

Our results with the overhead camera nicely complement the prior study by Wang et al. on camera placement that found that a gravity-referenced camera whose view included the robot chassis provided greater situation awareness than a robot-referenced camera that did not include the robot within its view [16]. While Wang's study had the confounding factor of the frame-of-reference, their results are consistent with ours in terms of an overhead versus a forward view, and they support the conclusion that including the chassis of the robot in the camera view has a significant impact on situation awareness in remote robot operation.

We also found that subjects in our front and rear camera study developed the strategy of tilting the camera all the way down to see the front bumper and that these subjects had fewer collisions with the environment, further emphasizing the importance of being able to view one's own body in reference to the environment. Therefore, we conclude that, if you can only have one camera on a robot during remote operation, position the camera such that the user can view at least a portion of the robot within the environment.

If, on the other hand, you can have two cameras, use the second camera to give the user quick access to rear view rather than an alternative forward view, since access to a rear-facing camera significantly improves situation awareness compared to only a forward facing camera. In our overhead study, operators simply did not use the forward camera as much when they had a choice between forward mounted and overhead mounted cameras.

There are still open questions with regards to camera placement on a robot. For example, we only tested the overhead camera at one height relative to the robot. The improvement in situation awareness may have come from being able to see the robot–as was anecdotally the case when watching operators drive through tight spaces–but it may have also come from the higher perspective of the scene being closer to the operator's own perspective than the forward camera, which was only 30cm above the ground. There are also variations on the placement of a rear camera that merit exploration, especially with an interface design that includes both views.

Our current results, however, should have an immediate impact on the design of robots for remote operation, especially within the robotic USAR community. Both anecdotally and quantitatively, the operator's situation awareness is improved by both permitting the operator the option of viewing the robot within its environment and, especially for asymmetric robots, giving the operator quick access to a rear view.

VI. Acknowledgements

This work was supported in part by NSF IIS-0308186, NSF IIS-0415224 and NIST 70NANB3H1116. Thanks to Mike Baker, Andrew Chanler and Munjal Desai for assisting with the user tests at UMass Lowell.

References

[1] M. Baker, R. Casey, B. Keyes, and H. A. Yanco. Improved interfaces for human-robot interaction in urban search and rescue. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, The Hague, Netherlands, October 2004.

[2] D. Bruemmer, D. Dudenhoeffer, and J. Marble. Dynamic autonomy for urban search and rescue. In *Proceedings of the AAAI Mobile Robot Workshop*, Edmonton, Canada, August 2002.

[3] J. Casper. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. Master's thesis, University of South Florida, Department of Computer Science and Engineering, 2002.

[4] J. Casper and R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 33:367 – 385, June 2003.

[5] J. L. Drury, D. Hestand, H. A. Yanco, and J. Scholtz. Design guidelines for improved human-robot interaction. In *Proceedings of CHI2004*, Vienna, April 2004. Poster Presentation.

[6] S. Hughes, J. Manojlovich, L. M., and J. Gennari. Camera control and decoupled mtion for teleoperation. In *Proc. of IEEE Int'l Conf. on Systems, Man and Cybernetics*, volume 2, pages 1339–1344, October 2003.

[7] P. Lima, L. Custodio, L. Akin, A. Jacoff, G. Kraezschmar, B. K. Ng, O. Obst, T. Rofer, Y. Takahashi, and C. Zhou. Robocup 2004 competitions and symposium: A small kick for robots, a giant score for science. *AI Magazine*, 2005.

[8] B. A. Maxwell, N. Fairfield, N. Johnson, P. Malla, P. Dickson, S. Kim, S. Wojtkowski, and T. Stepleton. A real-time vision module for interactive perceptual agents. *Machine Vision and Applications*, 14:72–82, 2003.

[9] B. A. Maxwell, N. Ward, and F. Heckel. A configurable interface and architecture for robot rescue. In *AAAI Mobile Robotics Workshop*, San Jose, CA, July 2004. AAAI.

[10] M. Micire. Analysis of the robotic-assisted search and rescue response to the world trade center disaster. Master's thesis, University of South Florida, July 2002. Also available as CRASAR-TR2002-6.

[11] J. Scholtz, B. Antonishek, and J. Young. A field study of two techniques for situation awareness for robot navigation in urban search and rescue. In *Proceedings of the 14th IEEE International Workshop on Robot and Human Interactive Communication*, Nashville, TN, August 2005.

[12] J. Scholtz, Y. J., J. L. Drury, and H. A. Yanco. Evaluation of human-robot interaction awareness in search and rescue. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, LA, April 2004.

[13] R. Simmons and D. James. Inter process communication: A reference manual. December 2001.

[14] M. Voshell, D. D. Woods, and F. Phillips. Overcoming the keyhole in human-robot coordination: Simulation and evaluation. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting*, Orlando, September 2005.

[15] J. Wang, M. Lewis, and J. Gennari. A game engine based simulation of the nist urban search and rescue arenas. In D. F. S. Chick, P. J. Sanchez and D. J. Morrice, editors, *Proc. of the 2003 Winter Simulation Conference*, pages 1039–1045, 2003.

[16] J. Wang, M. Lewis, and S. Hughes. Gravity-referenced attitude display for teleoperation of mobile robots. In *Proc. of the Human Factors and Ergonomics Society 48th Annual Meeting*, pages 2662–2666, 2004.

[17] H. A. Yanco and J. Drury. 'Where Am I?' Acquiring situation awareness using a remote robot platform. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, The Hague, Netherlands, October 2004.

[18] H. A. Yanco, J. L. Drury, and J. Scholtz. Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition. *Journal of Human-Computer Interaction*, 19(1 and 2):117–149, 2004.