Integrated Control Strategies Supporting Autonomous Functionalities in Mobile Robots

B. Sights[a], H.R. Everett[a], E. Biagtan Pacis[a], G. Kogut[a]
M. Thompson[b]
[a]SPAWAR Systems Center, San Diego
[b]The Naval Undersea Warfare Center

ABSTRACT

High-level intelligence allows a mobile robot to create and interpret complex world models, but without a precise control system, the accuracy of the world model and the robot's ability to interact with its surroundings are greatly diminished. This problem is amplified when the environment is hostile, such as in a battlefield situation where an error in movement or a slow response may lead to destruction of the robot. As the presence of robots on the battlefield continues to escalate and the trend toward relieving the human of the low-level control burden advances, the ability to combine the functionalities of several critical control systems on a single platform becomes imperative.

Keywords: robotics, control strategy, autonomy, localization, mapping, SLAM, communications.

1. BACKGROUND

From aerial and ground reconnaissance missions to explosive ordnance disposal, robots are quickly becoming an integral part of our military's arsenal. The battlefield assessment information provided by robots can help save lives, but the time and focus required to operate the robot may also make the soldier more vulnerable to attack. For this reason, soldiers prefer to use a robot only when the risk of serious injury is high, such as explosive ordnance disposal applications, or when they do not have to constantly monitor the robot, as is the case with many unmanned aerial vehicles.

The Technology Transfer Program at SPAWAR Systems Center (SSC), San Diego leverages work done at other government agencies, academia, and industry in order to facilitate more rapid advancement of the technologies required to produce an autonomous robot that can robustly perform in battlefield situations [1]. For each desired behavior, various approaches from several groups are evaluated, and then the best options are integrated onto test platforms to work in conjunction with the other behaviors. This is not a trivial task because each organization tends to have a particular preference with respect to programming language, hardware requirements, and operating systems. When blended into one cohesive behavior architecture, any number of individual caveats can prohibit quick integration. Once the behaviors have been successfully integrated on the test platform, the hardware must be scaled down for installation on man-portable robots, such as an iRobot *Packbot* or Foster Miller *Talon* [2].

Figure 1. *ROBART III*.

1.1 Test Platforms

The current test platforms used for the Technology Transfer Program at SSC San Diego are *ROBART III* and an iRobot *ATRV Senior*. From the standpoint of system optimization, *ROBART III* is the optimal platform for behavior integration and evaluation. This platform is currently equipped with a SICK scanning laser rangefinder, Sharp triangulation ranging sensors, passive-infrared (PIR) motion sensors, Polaroid ultrasonic rangefinders, a gyro-stabilized magnetic compass, and a fiber-optic rate gyro. There is also significant reserve capacity to host even more sensors and computational hardware. *ROBART III's* vision system includes a Visual Stone 360-degree omni-directional camera and a Canon pan-tilt-zoom (PTZ) camera. It also has an automated weapon payload system which includes a non-lethal Gatling-style gun and an appearance-based target recognition system that uses a laser sight for automated target acquisition. On the other hand, the *ATRV Senior* was chosen for worst-case-scenario localization testing because of its wide tires and skid steering, which leads to incredibly inaccurate dead-reckoning calculations. This platform is also equipped with a SICK scanning laser rangefinder, Polaroid ultrasonic rangefinders, and a Sony PTZ camera, but the real advantage to the *ATRV Senior* is its large payload capacity, which is currently being used to test human presence sensors and mobile manipulators.

2. CONTROL SYSTEMS

The four systems that host integrated behaviors resulting from the Technology Transfer Program are the drive control system, the obstacle detection and mapping sensor system, the vision system, and the weapon system. Efforts are currently underway to develop and integrate manipulator behaviors in FY ‘05.

2.1 Drive Control and Sensor Systems

Perhaps the most critical component of an autonomous mobile robot is the drive control system. Even a decidedly simple robot can perform useful tasks simply by feeding translational and rotational velocities to the drive system based on sensor data. The drive control system is expected to maneuver the robot in an environment while avoiding obstacles in its path. At the same time, it must continuously localize itself in order to allow higher level behaviors to accurately develop real world models of the environment which can be used to enhance the warfighter’s situational awareness (Figures 2) [2].

2.1.1 Drive Control

ROBART III’s drive computer is a *686 Core* from Compulab that is hosted by a daughterboard developed at SSC San Diego. This configuration gives us a processor running Linux at 266 MHz with seven serial ports, CAN, Ethernet , three USB ports, four DACs, 12 ADCs, and 50 DIO. This allows us to interface to the following sensors: SICK LMS 200 ladar, Microstrain 3DM-G compass, KVH E-Core fiber optic gyro, Polaroid sonar system, Sharp IR sensors, and tactile sensors, with room for further expansion.

The drive motors are controlled by the Solutions-Cubed *ICON PID Motor Controller*, which has been optimally tuned to produce fluid control of *ROBART III’s* motion. Commands are sent to the motor controller via a driver developed for the position interface of University of Southern California’s Player project [2]. The odometry and sensor data is used by the drive computer for collision-avoidance and dead-reckoning calculations, while a higher level computer uses the data to perform path planning and Simultaneous Localization and Mapping (SLAM).

2.1.2 Collision Avoidance

Idaho National Laboratory (INL), one of SSC San Diego’s strategic partners under the Technology Transfer Program, has developed collision-avoidance techniques specifically for use in dynamic unknown environments. The collision-avoidance algorithms take a behavior-based approach that emphasizes a tight coupling between sensing and action, with each of the sensors contributing to an array of robot-centric regions to which the robot responds, based on fuzzy-logic rules that control its translational and rotational velocities [1]. These rules not only apply to each individual region, but can be triggered by combinations and patterns found within the array of regions. In implementing this scheme, INL uses a subsumption architecture such as was employed on *ROBART I*, wherein atomistic behaviors such as collision avoidance run in parallel with, but can be subsumed by, other reactive behaviors, such as “maneuver-around” and “get unstuck”[4]. Collision avoidance is a bottom-layer behavior and although it underlies many different reactive and deliberative capabilities, it runs independently.

The algorithm also continuously calculates an *event horizon* representing the last possible moment for the collision avoidance behavior to successfully intervene upon goal-based behaviors at the current speed. By calculating this *event horizon* many times each second, the robot can smoothly scale down its velocity as a function of congestion without fully impeding motion. When a full stop is required, use of the *event horizon* ensures that the robot comes to a halt at the same distance from an obstacle regardless of its initial velocity [1].

2.1.3 Dead Reckoning

For dead reckoning calculations, the Technology Transfer Program uses the standard calculation methods using encoder counts, wheel radius, and the robot’s wheelbase, as described in *Sensors For Mobile Robots* [5]. On top of this calculation, a Kalman filtering algorithm combining sensor data from the Microstrain 3DM-G compass, the KVH E-Core fiber optic gyro, and the GPS receiver (for outdoor applications) stabilizes the calculations to cancel out errors associated with wheel slippage, unequal wheel diameters, and the actual effective wheelbase [6]. This method provides us with excellent dead reckoning calculations that facilitate much better SLAM capabilities.

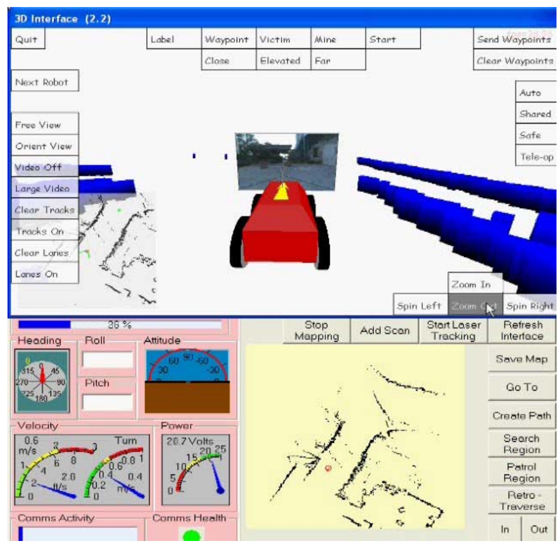

Figure 2. Augmented virtuality interface developed by integrating technologies from INL, SRI, and BYU [7].

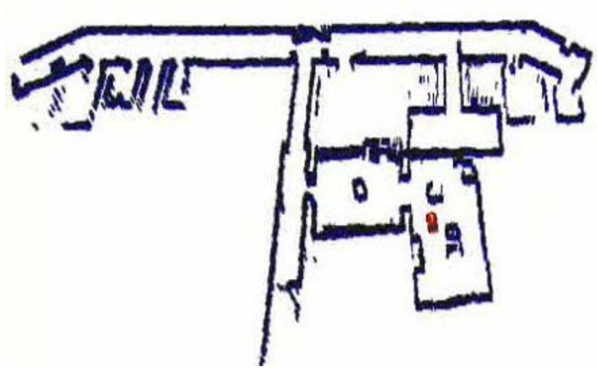

Figure 3. Map created by an iRobot *ATRV* exploring and mapping Battery Woodward, an underground WWII bunker at SSC San Diego.

2.1.4 Simultaneous Localization and Mapping (SLAM)

Now that the robot has an acceptable perception of its location, it can build a map as it traverses unknown terrain (Figure 3). The Consistent Pose Estimation (CPE) mapping technology was developed at Stanford Research Institute International (SRI). CPE efficiently incorporates new laser scan information into a growing map, and also addresses the challenging problem of loop closure, how to optimally register laser information when the robot returns to an area previously explored. CPE is also one method of performing Simultaneous Localization and Mapping (SLAM), based on original work by Lu and Milios [8], who showed that information from the robot's encoders and laser sensors could be represented as a network of probabilistic constraints linking the successive poses of the robot [1].

SRI has implemented and further developed localization algorithms using a representation of the robot's state space based on Monte Carlo sampling [9]. Introduced in 1970 [10], Monte Carlo Localization (MCL) methods have more recently been applied in the fields of target tracking, computer vision, and robot localization [9][11], with good results. The Monte Carlo technique inherits the benefits of previously introduced Markovian probability-grid approaches for position estimation [12], and provides an extremely efficient technique for mobile robot localization. One bottleneck in the MCL algorithm is the necessity for checking the posterior probability of each sample against the map, based on the current laser readings. SRI has developed an efficient method for performing this computation, using a correlation technique derived from computer vision algorithms [13].

2.2 Vision and Weapon Systems

Humans have a tendency to rely heavily on vision as their primary sensory input, and utilizing this capability can advance a robot's functionality immensely. Not only does it allow a robot to recognize objects and targets, but it also allows the human and the robot to understand the environment in a common way. Adding a weapon payload to work in conjunction with the vision system on the robot can amplify the ability of the robot to survive in a hostile environment or to engage enemies or weapons systems that may harm the soldier. To support the Warfighter's Associate Concept being developed at SSC San Diego[3], the vision system is expected to search for, recognize, and locate targets of interest. If the target is recognized as hostile, the vision computer can provide the weapon system with the proper information to prosecute the target.

2.2.1 Vision System

The vision processor on *ROBART III* consists of a Microspace *PC/104 MSM-P3 SEV* embedded computer running Linux at 700 MHz with 512 MB RAM, 512 MB Compact Flash, and using 2 Belkin VideoBus USB video digitizers. This currently allows for an error update rate to the weapon controller of about 5 Hz. The weapon controller consists of a Motorola *HC11* Microcontroller that does the fuzzy logic calculations, Hewlett Packard *HCTL-1100* PID controller chips to keep the motor output speeds at their respective set points, and National Semiconductor *LMD18200* H-bridges to drive the Pitman *GM8722G968* motors.

2.2.2 Target Recognition

ROBART III's target recognition system takes conventional digital images of potential targets as inputs and produces templates consisting of features calculated from the sample images. These features are matched against incoming images from either the 360-degree camera or rectilinear camera mounted on *ROBART III*'s head assembly. A probabilistic map of potential-matching targets is created in real-time for each image in the templates. Any match which exceeds a preset threshold is designated as a recognized target.

There are two primary algorithms which determine the likelihood of a match. The first is a conventional cross-correlation algorithm which correlates, pixel-by-pixel, the target template over each incoming image as a sliding window. Image hue is used in the correlation, providing two advantages. First, hue is independent of brightness, making the matching process independent of ambient lighting. Second, correlating on a single channel of data reduces the computational complexity of the process as compared to processing red, green, and blue color channels.

However, cross-correlation is perspective-dependent, in that the object being matched in the scene camera must have a similar scale and orientation as the image being used as the template. Therefore, a second algorithm is used to simultaneously match the seven Hu moments between the template image and the incoming image stream, since Hu moments are invariant to scale, rotation, and reflection. The combination of these two algorithms is a robust matching system with a very low probability of false-positives. The addition of a matching system using the Scale Invariant Feature Transform (SIFT) is also being investigated [14].

2.2.3 Target Acquisition and Tracking

The current mobility behaviors support target following using the object-recognition vector generated from the vision system to calculate translational and rotational velocities for the drive controller, which allows the robot to pursue the target while avoiding obstacles and running SLAM. The next step is to provide the ability for the weapon system to also track and prosecute a target while the robot is moving.

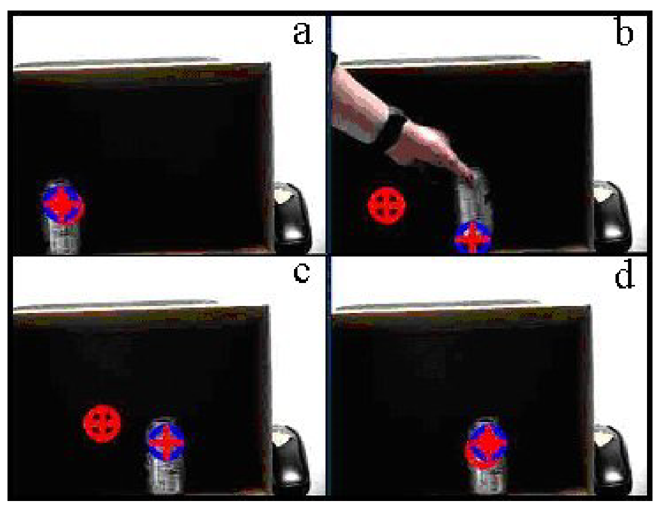

Figure 4. a) Vision System and targeting laser on detected vulnerability (soda can); b) Can is relocated and tracked in real-time c) Targeting laser servos to new location; d) Laser now relocated on new target position, ready to fire weapon.

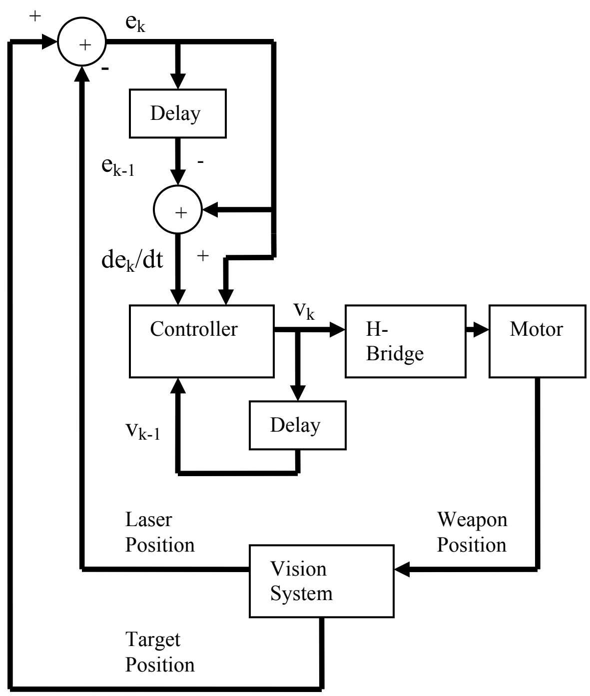

Figure 5. Block diagram of one axis of the combined vision and weapon control system.

ROBART III's current target acquisition strategy is simply to aim at and track the recognized target with the strongest match, but other acquisition strategies can easily be added to the system. The targeting process aims ROBART *III*'s arm-mounted weapon at the target in a two-stage process. The first stage involves panning the weapon to roughly the target location by measuring the current pan and tilt angles of the camera. The second stage employs fuzzy-logic and an active-laser-targeting system to precisely converge on the target.

The first stage uses a rough calibration between the weapon pan axis and either of two scene cameras that can provide the image coordinates of a target: the omni-directional visual sensor or the pan-tilt zoom camera. The omni-directional camera's central axis is parallel to the weapon's pan axis and at a known offset. This fixed geometry allows image coordinates from targets detected in the omni-directional image space to easily be converted to pan-axis coordinates in the weapon's pan-axis space. The image-space location of targets detected in imagery from the pan-tilt-zoom camera can similarly be easily converted into weapon pan-axis coordinates. The pan-axis of the pan-tilt-zoom camera is also parallel to the weapon's pan-axis, and at a known distance. The pan-tilt-zoom camera first uses a simple centering algorithm to center the target in its field-of-view, at which point the pan-coordinates can be read from the camera's pan-motor encoder, and used to pan the weapon to roughly the same orientation [14].

During the second stage, a bore-sighted laser aligned along the weapon's active barrel is turned on and off in synchronization with the vision systems frame grabber, and at one half the frequency of image capture. This allows simple image differencing to very accurately locate the laser dot in image

INPUTS		OUTPUT	
Image Error (e_k)		**Weapon Rotational Speed (v_k)**	
LN	Large Negative	FN	Fast Negative
MN	Medium Negative	MN	Medium Negative
SN	Small Negative	SN	Slow Negative
ZE	Zero	ZE	Zero
SP	Small Positive	SP	Slow Positive
MP	Medium Positive	MP	Medium Positive
LP	Large Positive	FP	Fast Positive
Previous Weapon Rotational Speed (v_{k-1})			
FN	Fast Negative		
MN	Medium Negative		
SN	Slow Negative		
ZE	Zero		
SP	Slow Positive		
MP	Medium Positive		
FP	FastPositive		
Change in Image Error (de_k/dt)			
LN	Large Negative		
MN	Medium Negative		
SN	Small Negative		
ZE	Zero		
SP	Small Positive		
MP	Medium Positive		
LP	Large Positive		

Figure 6. Input and output fuzzy sets for one axis of the weapon control system.

space (Figure 4). The inputs to the fuzzy-logic weapon controller are the difference between the laser position and the target position in the image space, the change in the difference between the laser and target position since the last measurement, and the weapon's previously set pan and tilt speed (Figure 5). The fuzzy rule sets for the pan and tilt axis are exactly the same, so we will only do the analysis for one axis. The fuzzy-set values of the fuzzy variables are shown in Figure 6, with a positive pan speed corresponding to a movement to the left (using unit-circle representation of the robot-space) and a positive tilt speed corresponding to a rotation of the orientation of the front of the weapon in the vertical direction [15].

After we calculate the degree of membership (DOM), μ, for each variable from the DOM functions, m(x), we can calculate the DOM of the output function by choosing the lowest DOM for each rule in our set of fuzzy rules (Figure 7). Finally, for each output fuzzy value, we take all the rules that apply to that set and choose the rule that produced the highest DOM as the output DOM for each output fuzzy set value[8]. Then we use

Previous Weapon Rotational Speed (v_{k-1})

Change in Image Error (de_k/dt)	LN	MN	SN	Z	SP	MP	LP
LN	LN	LN	LN	LN	MN	SN	ZE
MN	LN	LN	LN	MN	SN	ZE	SP
SN	LN	LN	MN	SN	ZE	SP	MP
ZE	LN	MN	SN	ZE	SP	MP	LP
SP	MN	SN	ZE	SP	MP	LP	LP
MP	SN	ZE	SP	MP	LP	LP	LP
LP	ZE	SP	MP	LP	LP	LP	LP

Image Error (e_k) = ZE

Figure 7. Cross section of the fuzzy control system's output rules for the case where e_k = ZE.

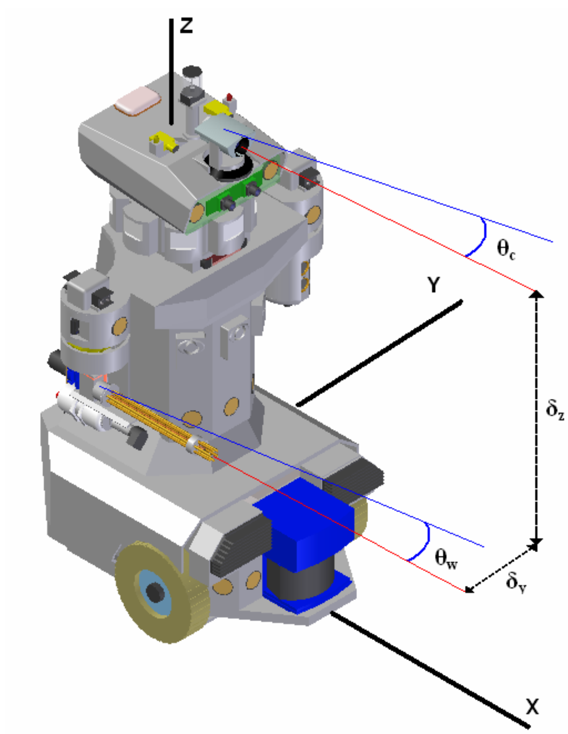

Figure 8. Diagram of pan axis angles and distances for the Canon PTZ camera and the weapon system on *ROBART III*.

the singleton method to calculate the output motor velocity for the pan and tilt axis in order to save computational time. For example, if we are calculating the weapon pan speed and the output degree of membership values are μ_{LN}, μ_{MN}, μ_{SN}, μ_{ZE}, μ_{SP}, μ_{MP}, μ_{LP}, and the singleton values corresponding to each output rule are s_{LN}, s_{MN}, s_{SN}, s_{ZE}, s_{SP}, s_{MP}, s_{LP}, then we can calculate the output crisp value as follows:

$$\text{Output} := \frac{\left(\mu_{LN}\cdot s_{LN} + \mu_{MN}\cdot s_{MN} + \mu_{SN}\cdot s_{SN} + \mu_{ZE}\cdot s_{ZE} + \mu_{SP}\cdot s_{SP} + \mu_{MP}\cdot s_{MP} + \mu_{LP}\cdot s_{LP}\right)}{\left(\mu_{LN} + \mu_{MN} + \mu_{SN} + \mu_{ZE} + \mu_{SP} + \mu_{MP} + \mu_{LP}\right)}$$

(1) [16].

This gives us a control system which is very robust, independent of the motion of the robot, independent of the robotic platform and the weapon used (the DOM functions can easily be adjusted), and which can be used to influence the motion of the robot when tracking a target.

2.2.4 Target Prosecution

Actual firing of *ROBART III*'s weapon is performed via teleoperation from a remote user interface. The user can also easily verify that the target has been hit. Voice feedback from *ROBART III* provides the user with real-time feedback of the recognition and acquisition process, with phrases such as, "Target acquired," etc.

2.2.5 Target Position Placement in Augmented Virtuality

Another benefit of the combined vision and weapon system is the ability to locate the position of the target with respect to the robot based on the pan and tilt angles of the camera and the weapon, as well as the known horizontal and vertical differences between the two devices, as follows:

Using simple triangulation-ranging geometry (see Figure 8), and calculating with respect to the pan axis we have:

$$\tan(\theta_c) := \left(\frac{\text{distance\_y}}{\text{distance\_x}}\right) \qquad (2)$$

$$\tan(\theta_w) := \frac{\left(\text{distance\_y} + \delta_y\right)}{\text{distance\_x}} \qquad (3)$$

By measuring the camera and weapon angles from encoder data, we can solve for the location of the target in the X-Y plane as follows:

$$\text{distance\_x} := \frac{\left(\delta_y\right)}{\left(\tan(\theta_c) - \tan(\theta_w)\right)} \qquad (4)$$

$$\text{distance\_y} := \text{distance\_x}\left(\tan(\theta_c)\right) \qquad (5)$$

When these distances are calculated, the position of the target can be mapped into the augmented virtuality model [3]. A benefit of using a point laser for image distance triangulation, as opposed to a typical sonar or ladar device, is the fact that the distance measured is the actual distance to the target, and not the distance to another object in between the robot and the target.

2.3 Manipulator Behaviors

The Technology Transfer Program is investigating efforts, such as those currently under development at the University of Texas at Austin, to develop autonomous manipulator behaviors that can expand the abilities of robots to perform search and rescue, explosive ordinance disposal, soldier assistance, and other missions that require the ability to manipulate objects (See Figure 9). Just as the weapons system requires the target to be either pre-taught or assigned by the soldier in real-time to the vision system, the manipulator will require the use of the vision system to activate behaviors. The addition of this ability to the

Technology Transfer Program will make it possible to field robots that are truly accepted by, and not a burden to, the warfighter [17].

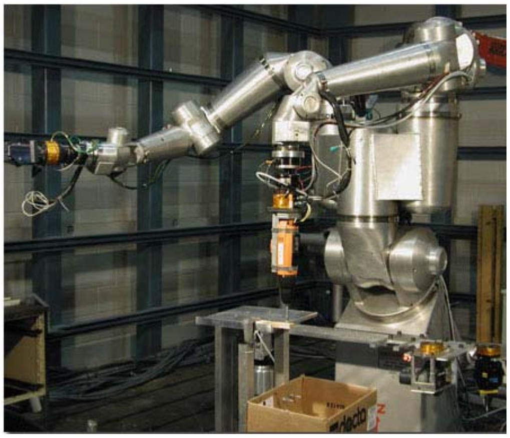

Figure 9. A 17 degree-of-freedom manipulator performing automated tasks.

3. THE FUTURE

Plans are currently in place to perform a comprehensive demonstration in the near term that utilizes all of the systems and capabilities discussed in this paper. Essentially, the robot would be required to enter an unknown bunker and autonomously create a map that contains images and information regarding targets of interest that are obtained by use of the vision system and other sensors (ie. chemical, radiological, etc). If a hostile target is located, the robot will engage and prosecute when given the command from the human operator. This would demonstrate the ability of a robot to perform reconnaissance sweeps on a building of opportunity without putting the warfighter in harm's way.

4. CONCLUSION

Military robotic capabilities are being rapidly expanded through the efforts of the Technology Transfer program at the Space and Naval Warfare Systems Center, San Diego. The individual components that make up the robot must be able to effectively control the motions of the various robot drive, vision, weapon, and manipulator systems to allow the combined effort to lead to the realization of more autonomous and useful robots. By standardizing the methods through which the different systems on each robot communicate, and focusing on the most promising behavioral techniques, more rapid advancement in autonomous functionalities can be facilitated. When these autonomous functionalities are combined with a more intuitive and informative user interface, the net benefit to the warfighter's situational awareness and safety are immeasurable.

REFRENCES

[1] E.B. Pacis, H.R. Everett., "Enhancing Functionality and Autonomy in Man-Portable Robots," Proceedings, SPIE Unmanned Ground Vehicle Technology VI, Defense and Security, Orlando, FL, 12-16 April, 2004.

[2] E.B. Pacis, H.R. Everett, et. al "Transitioning Unmanned Ground Vehicles Research Technologies," Proceedings, SPIE Unmanned Ground Vehicle Technology VII, Defense and Security, Orlando, FL, 28 March - 1 April, 2005.

[3] H.R. Everett, E.B. Pacis, G. Kogut, N. Farrington, and S. Khurana, "Towards a Warfighter's Associate: Eliminating the Operator Control Unit," SPIE Proc. 5609: Mobile Robots XVII, Philadelphia, PA, October 26 – 28, 2004.

[4] H.R. Everett, "A Microprocessor Controlled Autonomous Sentry Robot," Masters Thesis, Naval Postgraduate School, Monterey, CA, October, 1982.

[5] H.R. Everett, Sensors For Mobile Robots – Theory and Application, Massachusetts: A K Peters Ltd. Pub., 1995.

[6] J. Borenstein, "Measurement and Correction of Systematic Odometry Errors in Mobile Robots***",*** ***IEEE Transactions on Robotics and Automation,*** Vol. 12, No. 6, 1996, pp. 869-880.

[7] C.W. Nielsen, B. Ricks, M.A. Goodrich, D. Bruemmer, D. Few, and M. Walton. "Snapshots for Semantic Maps." Proceedings of the 2004 IEEE Conference on Systems, Man, and Cybernetics. October 10-13, 2004, The Hague, The Netherlands.

[11] Lu, F. and E.E. Milios, "Globally Consistent Range Scan Alignment for Environment Mapping," Autonomous Robots, 4(4), 1997.

[12] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte Carlo Localization: Efficient Position Estimation for Mobile Robots," Sixteenth National Conference on Artificial Intelligence (AAAI'99), July, 1999.

[13] J. Handschin. "Monte Carlo Techniques for prediction and filtering of non-linear stochastic processes." Automatica6, 1970.

[14] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo Localization for Mobile Robots," IEEE International Conference on Robotics and Automation (ICRA99), May, 1999.

[15]W. Burgard, D. Fox, D. Hennig, and T. Schmidt. "Estimating the absolute position of a mobile robot using position probability grids," Thirteenth National Conference on Artificial Intelligence, pp. 896-901, 1996.

[16] Kurt Konolige, Ken Chou: Markov Localization using Correlation, IJCAI 1999: 1154-1159.

[14] G. Kogut, **"**Target Recognition, Acquisition, and Prosecution from an Unmanned Ground Vehicle", Proceedings, SPIE Unmanned Ground Vehicle Technology VII, Defense and Security, Orlando, FL, 28 March - 1 April, 2005.

[15] B. Kosko, Neural Networks and Fuzzy Systems, New Jersey: Prentice-Hall Inc. Pub., 1992.

[16] S. Niku, Introduction to Robotics – Analysis, Systems, Applications, New Jersey: Prentice-Hall Inc. Pub., 2001

[17] C. Kapoor, et al, "Decision Making for Automation and Teleoperation of Mobile Manipulation Systems", DOD Proposal,2004.