

A Portable, Autonomous, Urban Reconnaissance Robot

L. Matthies, Y. Xiong, R. Hogg, D. Zhu, A. Rankin, B. Kennedy
Jet Propulsion Laboratory, Pasadena CA, 91109

M. Hebert, R. Maclachlan
Carnegie Mellon University, Pittsburgh PA, 15213

C. Won, T. Frost
IS Robotics, Somerville MA, 02143

G. Sukhatme, M. McHenry, S. Goldberg
University of Southern California, Los Angeles CA, 90007

ABSTRACT

Portable mobile robots, in the size class of 20 kg or less, could be extremely valuable as autonomous reconnaissance platforms in urban hostage situations and disaster relief. We have developed a prototype urban robot on a novel chassis with articulated tracks that enable stair climbing and scrambling over rubble. Autonomous navigation capabilities of the robot include stereo vision-based obstacle avoidance, visual servoing to user-designated goals, and autonomous vision-guided stair climbing. The system was demonstrated in an urban reconnaissance mission scenario at Fort Sam Houston in October 1999. A two-axis scanning laser rangefinder has been developed and will be integrated in the coming year for indoor mapping and position estimation. This paper describes the robot, its performance in field trials, and some of the technical challenges that remain to enable fieldable urban reconnaissance robots.

1. INTRODUCTION

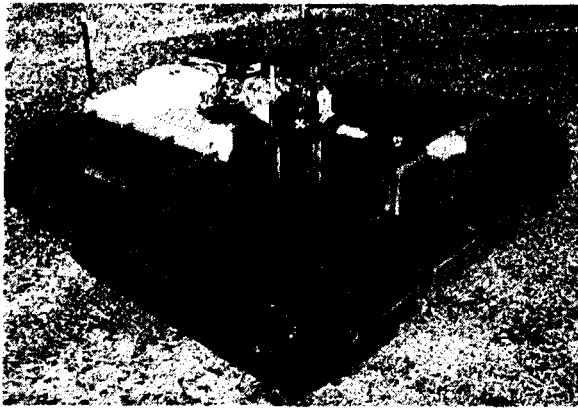
Urban hostage situations, disaster relief, and urban conflicts are extremely dangerous, particularly when entering buildings where no prior intelligence information may be available. Unmanned reconnaissance robots may reduce the danger by providing imagery and maps of outdoor and indoor areas before human personnel move in. Such platforms will be most practical if they are small enough to be carried and deployed by one person. This paper describes a 20 kilogram prototype tactical mobile robot we have developed for urban reconnaissance and reviews the autonomous navigation capabilities it has achieved. These capabilities include stereo vision-based obstacle avoidance at up to 80 cm/sec, visual servoing to goals, and vision-guided ascent of multiple flights of stairs.

Through 1999, the objectives of our project were to develop portable robots that could approach, enter, and map the inside of a building in daylight. In 2000 and beyond, these objectives will extend to doing such operations in the dark. Overall weight and dimensions of the robot are constrained by the requirement of portability to be on the order of 20 kg or less and 65 cm long or less. The mobility platform must be able to negotiate obstacles typical of urban areas, including curbs, stairs, and rubble. Autonomous navigation capabilities are required to minimize the burden on both the operator and the communication system. Autonomous perception has always been a limiting factor for robots; therefore, enabling high-speed autonomous navigation and mapping in such a small vehicle is a key challenge for this program.

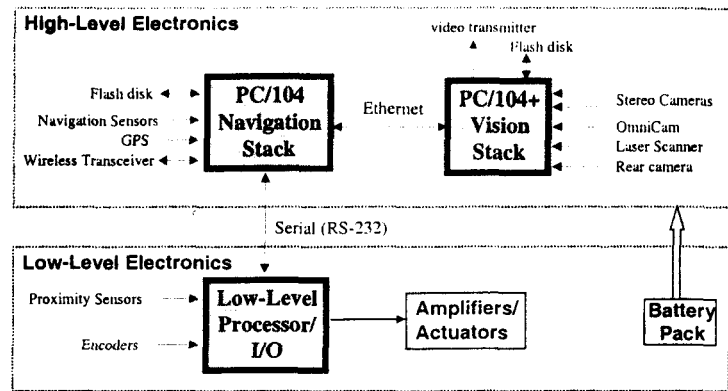
While a great deal of prior mobile robot research exists, the vast majority has either used much larger vehicles or worked primarily indoors. Some highlights of prior work include autonomous cross country navigation systems developed on HMMWV's¹ [1], Mars rover research prototypes [2], and indoor mapping systems developed on wheeled indoor robots [3]. Contributions of our effort include a small mobility chassis suitable for mixed outdoor and indoor terrain, design of a small, two-axis scanning laser rangefinder for indoor mapping and position estimation, packaging of a large number of sensors and significant computing power in a 20 kg robot, and algorithms for new autonomous navigation capabilities in the areas of obstacle avoidance, visual servoing, and stair climbing.

Section 2 describes the system architecture of our robot, including the mobility chassis, processors, sensors, and communication subsystems. Section 3 summarizes algorithms we have developed for obstacle avoidance, visual servoing, and stair climbing. Performance of the system in extensive trials is assessed in section 4. Section

¹ High Mobility Multi-purpose Wheeled Vehicle



(a)



(b)

Figure 1: Urban reconnaissance robot. (a) Entire vehicle, showing stereo cameras on the front and Omnicam behind the stereo cameras. (b) Electronics architecture.

5 summarizes the contributions of this work, the current limitations of the robot, and key areas for future work.

2. SYSTEM ARCHITECTURE

2.1 Chassis and Electronics

The mobility chassis is the "Urban II" tracked platform developed at IS Robotics (figure 1a). Tracked articulations in the front of the robot can do continuous 360 degree rotation and enable crossing curbs, climbing stairs, and scrambling over rubble. A 3 kg NiCad battery pack provides about 100Wh of energy; peak driving speed is currently 80 cm/sec on flat ground. In the Urban II, motor control is performed by a Motorola 68332 processor, which also reads 14 infrared and 7 sonar proximity sensors that are distributed around the waist of the vehicle. The 68332 communicates over a serial line to a PC104+ based Pentium II processor in the "high level electronics" subsystem, which was developed at the Jet Propulsion Lab (JPL). This subsystem does all of the vision, mapping, and most of the navigation functions.

The high level electronics contain two PC104+ stacks that communicate by ethernet, with a 166 MHz Pentium II on the "navigation" (or "nav") stack and a 233 MHz Pentium II on the "vision" stack (figure 1b). Image processing functions are isolated from other interrupt traffic on the vision stack, which interfaces to a forward-looking stereo camera pair, an omnidirectional camera, and the scanning laser range finder. The nav stack interfaces to a 900 MHz, 115 kb/s radio modem (from Freewave), 3-axis gyros (Systron Donner) and accelerometers (Etran), a compass and inclinometer package (Precision Navigation), and a GPS receiver with up to 2 cm precision in carrier phase differential mode (Novatel). Overall weight as of October 1999 was approximately 22 kg. Power dissipation standing still was 76W; power required for driving will be discussed in section 4. A 2.4 GHz analog video

transmitter (EyeQ) can transmit either the Omnicam or one of the stereo cameras. The high level electronics are packaged in a 10x10x5 inch payload compartment known as the "e-box", which is cooled with two fans on the back of the robot.

2.2 Sensors

Selection of the sensor suite was driven by needs for daylight obstacle detection, mapping, position estimation, and goal designation and tracking. Three-axis gyros and accels, summarized above, are essential for position estimation and for heading estimation during driving and stair climbing. Obstacle detection is enabled by forward-looking stereo cameras, infrared (IR) and sonar proximity sensors looking in various directions, and the scanning laser rangefinder. To provide adequate coverage, the stereo cameras have a field of view of 97x74 degrees. The stereo imagery is processed into 80x60 pixel disparity maps on the vision stack using algorithms developed previously at JPL [4]. To provide adequate dynamic range and control of image exposure for operation from bright sunlight to dim indoor lighting, we required the cameras to have software-controllable exposure time; this limited the camera selection to one vendor of CCD board cameras (Videology). The implementation of the IR and sonar sensors had noise problems which have yet to be resolved; hence, they were not used in the experiments described in section 4.

The laser rangefinder is designed to support obstacle detection, indoor mapping, and indoor position estimation; it consists of rangefinding electronics developed by Acuity Research and a two-axis scanning mechanism developed at JPL. The electronics are a modified version of the Accurange 4000 sold commercially by Acuity; the modifications reduce the receiver aperture from a 3 inch diameter to 2x1.5 inches and change the electronics from one 3x6 inch board to two 3x3 inch boards, which mount

in an "L" configuration (figure 2). These changes enable low-profile integration into a small robot. The scanner is designed to allow continuous 360 degree panning at 600 RPM with tilt variable from -10 to +15 degrees; planned revision of the pan motor selection will increase these numbers to over 3000 RPM and -15 to +30 degrees. Using a PC104+ version of the Acuity "High Speed Interface" board, the maximum sample rate is 50,000 samples/sec; we typically acquire 1000 samples/revolution. The laser diode can be operated at 3 mW or 20 mW output power. Integration and testing are still in process; however, initial tests at 20 mW so far indicate a range precision of 1 mm (1 sigma) out to 10 m, against a cardboard target at normal incidence, and ability to measure range up to an incidence angle of 70 degrees against the same target at 3 m.

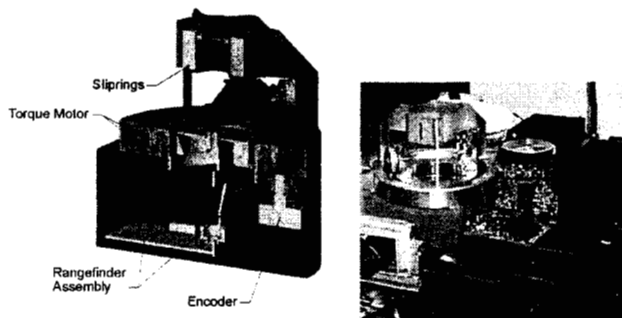


Figure 2: Laser rangefinder CAD model and prototype

Goal designation and tracking is done with either the stereo cameras or with an "Omniscam" panoramic camera, developed by Cyclovision from an optical design pioneered at Columbia University [5]. The Omniscam uses catadioptric optics with two mirrors to provide a 360 horizontal field of view and 70 degree vertical field of view. A small CCD board camera with fully electronic exposure control is integrated into the base of the Omniscam optics. Omniscam imagery can be captured digitally onboard or multiplexed to the analog video transmitter.

3. NAVIGATION MODES

Navigation can be controlled at three levels of autonomy: pure teleoperation, safeguarded teleoperation, and autonomous navigation. In teleoperation mode, imagery can be transmitted to the user over either the analog video transmitter or the radio modem. In safeguarded teleoperation mode, an onboard obstacle avoidance behavior modifies the user's teleoperation commands via an arbitration scheme discussed below. In autonomous navigation mode, obstacle avoidance is combined with onboard goal-seeking behaviors (visual servoing or waypoint following) that take the robot toward a user-designated goal. Finally, the autonomous navigation mode can also do vision-guided stair climbing up a user-specified number of

flights of stairs. The rest of this section outlines the operation of the autonomous navigation modes.

3.1 Obstacle Avoidance

At present, obstacle avoidance (OA) relies exclusively on stereo vision; noise problems in the implementation of the IR and sonar proximity sensors currently prevent their use. The OA algorithm is a JPL adaptation for this vehicle of the "Morphin" system developed at CMU [6]. A "positive" obstacle detection algorithm [7] is applied directly to the disparity map with different thresholds to discriminate three cases: no obstacle, traversable obstacle, or nontraversable obstacle. These labels are applied to pixels in the disparity map based on obstacle height at each pixel, then projected down onto the ground plane to create a local occupancy grid map (figure 3). Thresholds for the three cases correspond to the ability of the chassis to cross step discontinuities with the articulations stowed or deployed at a 45 degree angle of attack; thus, "no obstacle" is defined as a step of less than 9 cm, "traversable obstacle" is a step between 9 and 20 cm, and "nontraversable obstacle" is a step greater than 20 cm.

For the resolution and field of view of the stereo vision system, the occupancy grid map is 2.5 m wide and extends 2.5 m ahead of the vehicle, with 10 cm per cell. Cells in the occupancy grid are either unknown, empty, or filled with a traversable or a nontraversable obstacle. The obstacle regions are grown by a fraction of the vehicle width, then a local path search evaluates a predetermined number of steering arcs to produce a "goodness" value and a maximum velocity for each arc. Arc goodness is zero if the arc runs into a nontraversable obstacle within 1 m of the robot; otherwise, a penalty function assigns a goodness value between zero and the maximum value (255) based on whether the arc encounters a nontraversable obstacle beyond 1 m or passes through cells with traversable obstacles. The maximum velocity for an arc is a heuristic linear function of its goodness. Ultimately, our intent is to use the map to determine appropriate angles for the arms; for now, however, arm angles for each mission segment are set by the user. The vector of votes for all arcs is output to the arbiter, which combines them with votes from the currently active goal-seeking behavior as described below. For the experiments described in section 4, the obstacle map was created from scratch for each new stereo image pair (ie. there is no map merging over time). A version with map merging has been tested off-line, but not yet tested on the vehicle. The entire OA system runs at about 4 Hz.

3.2 Visual Servoing and Waypoint Following

The visual servoing (VS) and waypoint following (WP) capabilities were developed at CMU to provide goal-seeking behaviors to complement OA. In visual ser-

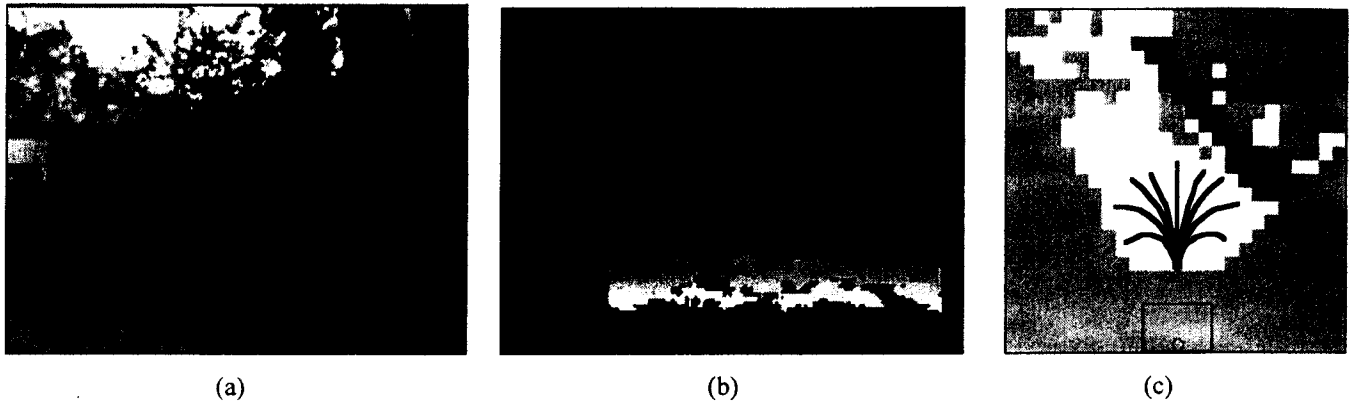


Figure 3: Obstacle avoidance behavior. (a) Left image from stereo pair, viewing a sidewalk with a short concrete wall. (b) Disparity map for this scene (bright is close, dark is far); the sidewalk and wall both are sensed well. (c) Occupancy grid created from the disparity map and positive obstacle detection algorithm, showing example steering arcs used in path evaluation. The rectangle at the bottom is the robot; gray areas are unknown, white are empty, black are nontraversable obstacles. The “traversable obstacle” class does not appear in this scene.

voing, an image is sent to the operator from the Omnicam or one of the stereo cameras, the operator designates a rectangular region in the image to serve as the goal, then the robot tracks the goal as a template as it approaches the goal. Only template-based tracking methods are used to allow goals to be arbitrary objects in the scene.

A number of techniques are used to provide robust tracking and to cope with large scale changes as the robot nears the goal [8]. For the Omnicam, target designation and tracking is performed on a “virtual image” that is a perspective projection of a 90 degree field of view from the Omnicam onto a virtual image plane in the direction of the target. Templates begin with a typical size of about 40x40 pixels. Tracking begins with a 2-D correlation search over +/- 16 pixels in the virtual image to determine the image plane translation of the template; then an iterative, linearized affine match procedure determines image plane rotation, scale, and skew changes of the target. As the robot moves, the original template from the first image is matched against each new frame and the aggregate affine transformation between the first and current frame is computed until the target size has changed enough to warrant reinitializing the template. Rapid robot motions or occlusions can cause loss of track; in this event, an attempt is made to reacquire the target by using the correlation search over a double-size window of the image (+/- 32 pixels). For template sizes of 40 to 50 pixels, the tracker runs at over 15 Hz on the 233 MHz Pentium II in the vision stack. Two tracking examples are shown in figure 4.

There are two types of waypoints for waypoint following: ground plane waypoints and direction waypoints. For ground plane waypoints, the operator designates a series of pixels in the image, which are converted into points on the ground by intersecting the direction vector through each pixel with the ground plane. The ground

plane estimate is updated at every frame, using the current robot attitude estimate, so as to update the waypoint coordinate estimates; this is referred to as an “incremental flat earth” assumption for the waypoint coordinates [8]. The robot then attempts to drive through the sequence of waypoints. For direction waypoints, the operator designates one pixel in the image; that direction vector is projected onto a line in the ground plane, which the robot tries to follow. If obstacles force the robot to deviate from the line, the direction waypoint module later directs the robot back toward the designated virtual line on the ground. Thus, this mode does more than just maintain a fixed steering direction, since it actually tracks the originally designated line.

In both VS and WP modes, a vector of steering and velocity votes that aim at the goal is generated and passed to the arbiter for combination with votes from OA. The arbitration algorithm uses vetoes (zero votes) from the OA module to eliminate nontraversable steering directions, performs a linear combination of the OA and goal-seeking votes for the remaining directions, and outputs the “best” steering vote to the lower level controller for execution. The minimum of the velocities for this direction from OA and the goal-seeking behavior is passed as the velocity command to the controller.

3.3 Stair Climbing

The operator initiates stair climbing by aiming the robot at a stairwell using teleoperation, inputting whether the stairwell is clockwise or counter-clockwise, giving the number of flights to ascend, and entering “go”. The robot uses edge detection in one of the forward-looking cameras to see the stairs and begin the ascent. Different algorithms apply when the robot is on the stairs and on the landings between each flight of stairs.

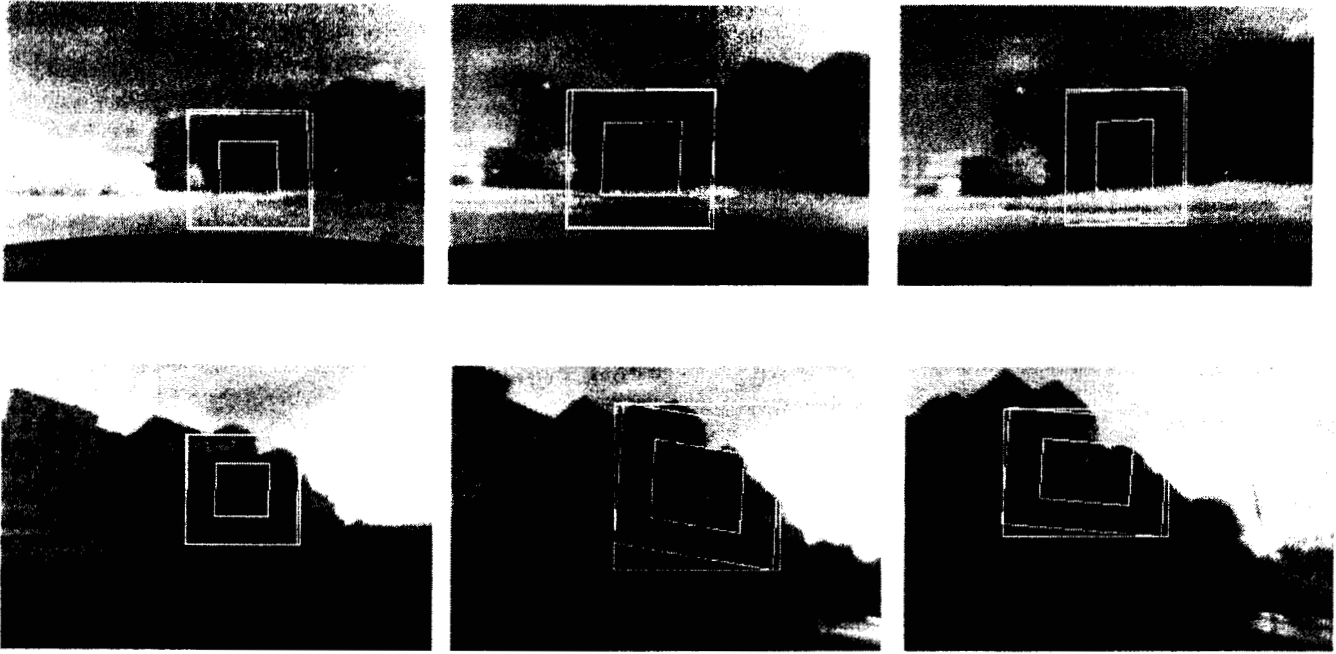


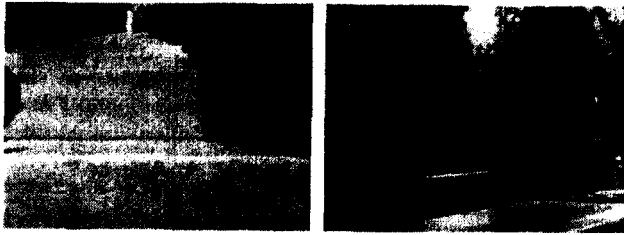
Figure 4: Two examples (top and bottom) of tracking targets on a building at a test site at Fort Sam Houston in San Antonio, Texas. The inner box shows the template; the outer box shows the search window in the virtual image.

The robot is required to handle indoor and outdoor stairwells, stairwells bounded by walls, stairwells bounded only by railings, and a variety of lighting conditions, including strong shadows and looking directly into the sun. Since bounding walls cannot be guaranteed, range sensors that rely on sensing walls are precluded. Therefore, we based our approach to ascending each flight of stairs on using one of the forward-looking stereo cameras to detect the horizontal edges of each stair step and the endpoints of each edge [9]. The orientations of the stair edges are used to guide the robot in the uphill direction; the endpoints are used to steer away from the walls.

To cope with strong shadows and significant appearance variations in steps (figure 5a), the edge detection algorithm takes a “least commitment” approach to finding the near-parallel straight lines that are the edges of the steps. First, a Canny edge detector is run with a low gradient magnitude threshold to find even weak edgels (figure 5b). Edgels are then linked into straight line segments. The dominant orientation of all of the edges is found by histogramming the edgels in all of the edges and choosing the greatest peak within ± 45 degrees of horizontal; all edges with orientations further than some threshold from the dominant orientation are then discarded (figure 5c). Since some steps may be detected as multiple short line segments, the remaining edges are filtered to merge nearly collinear segments. Finally, any edge that is still less than $1/4$ of the image width in length after merging is discarded (figure 5d). Some distracting edges with inconsistent orientations can still remain at this stage.

It is possible to derive a simple equation that relates the slope of the line segments in the image, assuming they are the edges of steps, to the angle of rotation θ between the robot heading and the centerline of the stairwell [9]. One can also derive a simple equation that relates the endpoints of the line segments in the image to the ratio q of the distances from the robot centerline to the left and right endpoints of each stair in 3-D. Since there can still be some outliers in the detected stair edges, we compute θ and q for each candidate stair edge, reject those for which the left and right endpoints are on the same side of the vehicle, find the median θ , and reject those edges whose θ value is far from the median. This produces a final set of filtered stair edges as seen in figure 5e. Since the sensitivity of estimating θ is poor near the horizon line, we compute a final estimate of θ as a weighted average of the estimates from each remaining edge, weighted by the inverse distance of the edge from the horizon line. The final θ estimate is used to recompute q for every edge, and the median q is used together with the final θ in the steering controller.

Steering of the robot on stairs is determined by the need to align parallel to the centerline of the stairwell and to avoid colliding with the walls or rails; that is, to keep θ small and q close to 1. Our work to date has focused on the image analysis above, so we currently use a fairly simple control algorithm that uses a proportional mapping from θ and q to steering corrections. The alignment criterion is considered more important than the centering criterion, so when θ is large we steer purely by that criterion



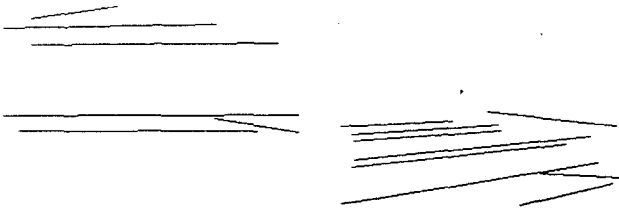
(a) Two examples of stairwells



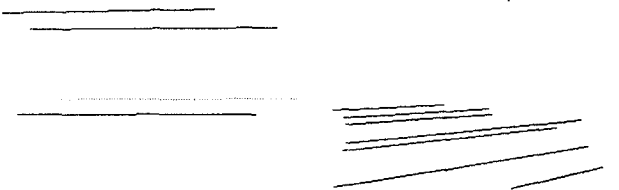
(b) Raw edge detector results



(c) Line segments after dominant orientation filter



(d) After merging segments and deleting short ones



(e) Final stair edges after median filter

Figure 5: Stair edge detection at several stages of the algorithm for two outdoor examples with shadows.

until the alignment is corrected. If edge detection fails to find any edges in a given frame, the system relies on the

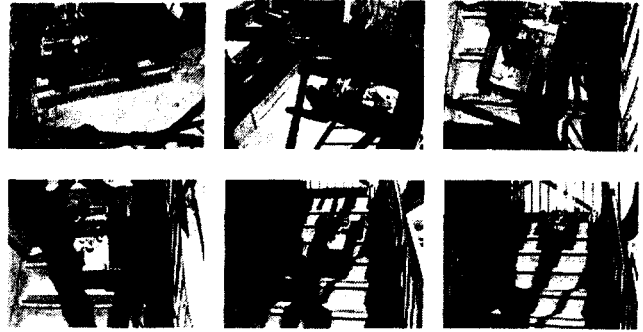


Figure 6: Climbing several flights in strong shadows.

heading gyro to estimate θ for the current frame and steers accordingly.

These algorithms are able to cope with strong shadows, high levels of image noise, and even the presence of the sun in the image (figure 6). Visual stair detection runs at about 3 Hz on the vision stack CPU.

Inclinometers in the compass module are used to determine when the robot reaches a landing. Two methods have been implemented to turn corners on landings: (1) a vision-based method that essentially uses the OA capability to do "wall following" around the edge of the landing until another stairwell comes into view, and (2) a purely deadreckoned method that uses the gyros and track encoders to execute two 90 degree turns separated by a driving segment of appropriate length. The deadreckoned method works very well if the dimensions of the landing are known. The vision-based method works on stairwells with adequate visual texture for stereo vision to perceive either the floor or the wall.

4. PERFORMANCE ASSESSMENT

An urban robotics fiscal year-end demonstration was conducted at an abandoned hospital, the former Brooks Army Medical Center (BAMC), on the grounds of Fort Sam Houston in San Antonio, Texas, in October 1999. An urban reconnaissance scenario was shown in which the robot was commanded to approach the building from behind a grove of trees about 150 meters away. Obstacle avoidance, visual servoing, direction waypoints, and teleoperation were used to reach the base of an external stairwell, then vision-guided stair climbing was invoked to autonomously ascend four flights of stairs. This entire sequence was performed flawlessly numerous times. In future years, this scenario will be extended to include several robots, one or more of which will carry tools for breaching doorways to allow access to the interior of the building; once inside, the mission will continue with indoor reconnaissance and mapping using the cameras, laser rangefinder, and other sensors on the robots.

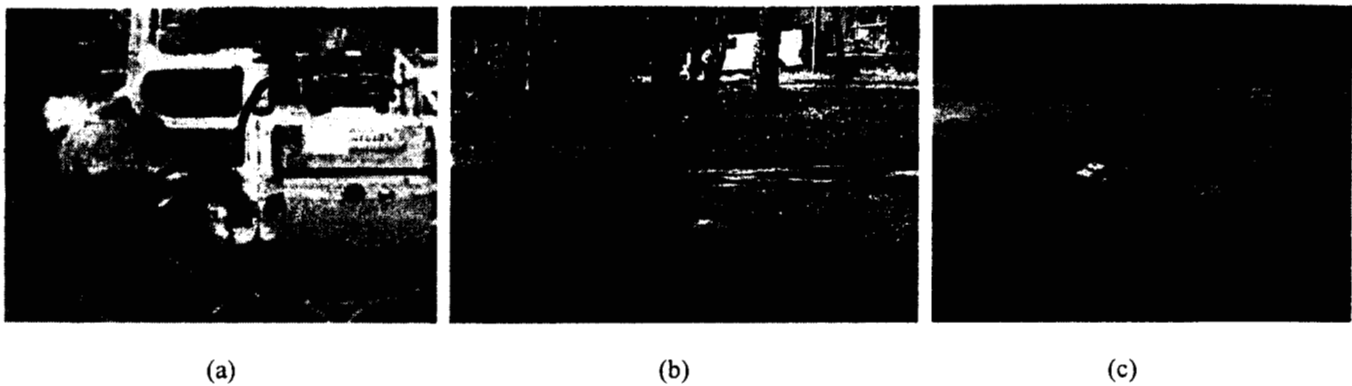


Figure 7a: Field testing at BAMC, Fort Sam Houston. (a) Aerial view of hospital grounds. Black line shows path of robot from deployment point in a grove of trees up to an external stairwell on the building. (b) Robot at deployment point. Autonomous obstacle avoidance in direction waypoint mode took the robot through the trees. (c) A performance limitation: an instance where the robot got high-centered on the edge of a sidewalk. Teleoperation is presently necessary to recover in this situation.

Figure 7a shows an aerial view of the grounds behind the hospital; the red line drawn on the photo shows the approximate path of the robot from behind the grove of trees to the base of the stairs. Figure 7b shows the robot in position behind the trees; waypoint following with obstacle avoidance was used to pass through the trees at up to 80 cm/sec. Visual servoing was used to cross the parking lot by tracking a HMMWV parked at the far side of the lot (not shown in the photo). After teleoperating under the HMMWV, waypoint following was used to reach the base of the stairs, with teleoperation used again to adjust position at the base of the stairs. Autonomous stair climbing enabled the robot to ascend all four flights of stairs with one command from the operator.

As mentioned earlier, power dissipation standing still was 76 W; on straight driving segments at 80 cm/sec, it was about 145 W, and on stairs it peaked about 250 W. Typical time and energy usage for this entire scenario was under 15 minutes, including all operator actions, and about 25 Wh. The robot occasionally suffered hangup failures when the belly pan between the tracks hung on a low, narrow obstacle; for example, this occurred in figure 7c on the edge of a sidewalk. Enhancements to the obstacle avoidance/obstacle negotiation subsystem will be required to deal with such situations. The tilt-axis gyro was found to saturate occasionally on the stairs. Communication system performance was adequate in this scenario, although the analog video transmitter produced noisy video in some circumstances. In general, obstacle avoidance, visual servoing, and stair climbing were quite reliable, even under relatively difficult lighting conditions.

5. CONCLUSIONS AND FUTURE WORK

Mobile robot technology has reached a level of maturity at which it should soon be possible to field rugged,

portable mobile robots for urban reconnaissance missions that include indoor and outdoor traverses and onboard mapping. In just one year, we have developed a prototype of such a vehicle and demonstrated key autonomous navigation capabilities that are necessary for such missions, including obstacle avoidance, visual servoing to user designated goals, and autonomous vision-guided stair climbing. Visual servoing and vision-guided stair climbing are new achievements for mobile robots; in addition, we believe this system represents a new milestone in the level of integration in such a small robot.

The technical challenges are now as much in system integration issues as they are in research problems for the component technologies. In particular, power and thermal management are significant issues, as is maintaining communication in the urban environment. In the area of power, we are comparing the locomotion power requirements of tracks versus the power requirements of wheels, and examining hybrid tracked/wheeled locomotion systems, in order to optimize driving speed versus power requirements over a mixture of terrain types. We are also examining alternatives to NiCad batteries as onboard energy sources, including the development of high-power lithium ion rechargeable batteries to extend the energy availability by roughly a factor of three.

In more recent communication testing at the San Antonio site, we used a 17 inch yagi antenna at the operator control station and a 17 inch elevated feed point antenna on the robot for the 900 MHz radio modem link. This combination of antennas allowed us to maintain communication with the robot from about 500 m away from the building, even when the robot was deep inside the building. We are also optimizing performance of transmission of compressed video over the 115 kb/s radio modem to enable teleoperation over that link; currently, we can teleoperate with 160x120 pixel imagery transmit-

ted with JPEG compression at a rate of 6 frames per second and a latency of under 250 ms. This will allow us to remove the analog video transmitter in future versions of the system. We are also exploring alternate radio frequency bands to enable better propagation in urban and wooded areas.

Miniaturization and packaging are also significant problems, particularly as we move to adding sensors for night operation. More research also needs to be done in such areas as outdoor autonomous position estimation, mapping, and obstacle avoidance in more difficult terrain.

6. REFERENCES

1. M. Hebert, B. Bolles, B. Gothard, L. Matthies, M. Rosenbloom, "Mobility for the Unmanned Ground Vehicle", in *Reconnaissance, Surveillance and Target Acquisition (RSTA) for the Unmanned Ground Vehicle*, O. Firschein (ed.), PRC Inc., 1996.
2. R. Volpe, J. Balaram, T. Ohm, R. Ivlev, "The Rocky 7 Mars Rover Prototype", *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Osaka, Japan, Nov 1996.
3. S. Thrun et al., "MINERVA: A second generation mobile tour-guide robot", *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Detroit MI, 1999.
4. Y. Xiong and L. Matthies. "Error analysis of a real-time stereo system", *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Puerto Rico, June 1997.
5. S. Nayar, "Catadioptric omnidirectional camera", *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Puerto Rico, June 1997.
6. R. Simmons, L. Henriksen, L. Chrisman, G. Whelan, "Obstacle avoidance and safeguarding for a lunar rover", *Proc. AIAA Forum on Advanced Developments in Space Robotics*, Madison WI, August 1998.
7. L. Matthies, A. Kelly, T. Litwin, G. Tharp, "Obstacle detection for unmanned ground vehicles: a progress report", *Robotics Research: The Seventh International Symposium*, G. Giralt and G. Hirzinger (eds), Springer-Verlag, 1996.
8. M. Hebert, R. MacLachlan, P. Chang, "Experiments with driving modes for urban robots", *Proc. SPIE Conference on Mobile Robots*, Boston MA, September 1999.
9. Y. Xiong and L. Matthies. "Vision-guided autonomous stair climbing", *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, April 2000.

7. ACKNOWLEDGEMENTS

This work was supported by the Defense Advanced Research Projects Agency under NASA task order 15089.

- 02 00 - 1/06