Alonzo Kelly
Anthony Stentz
Omead Amidi
Mike Bode
David Bradley
Antonio Diaz-Calderon
Mike Happold
Herman Herman
The Robotics Institute
Carnegie Mellon University
alonzo@ri.cmu.edu

Robert Mandelbaum
Sarnoff Corporation
Princeton, NJ
rmandelbaum@sarnoff.com

Tom Pilarski
Pete Rander
Scott Thayer
Nick Vallidis
Randy Warner
The Robotics Institute
Carnegie Mellon University

Toward Reliable Off Road Autonomous Vehicles Operating in Challenging Environments

Abstract

The DARPA PerceptOR program has implemented a rigorous evaluative test program which fosters the development of field relevant outdoor mobile robots. Autonomous ground vehicles were deployed on diverse test courses throughout the USA and quantitatively evaluated on such factors as autonomy level, waypoint acquisition, failure rate, speed, and communications bandwidth. Our efforts over the three year program have produced new approaches in planning, perception, localization, and control which have been driven by the quest for reliable operation in challenging environments. This paper focuses on some of the most unique aspects of the systems developed by the CMU PerceptOR team, the lessons learned during the effort, and the most immediate challenges that remain to be addressed.

KEY WORDS—autonomous mobility, mobile robot, unmanned ground vehicle, UGV, UAV, obstacle avoidance, motion planning, ladar, stereo

The International Journal of Robotics Research
Vol. 25, No. 5–6, May–June 2006, pp. 449-483
DOI: 10.1177/0278364906065543

1. Introduction

The DARPA PerceptOR (Perception for Off Road Navigation) program was, to our knowledge, the first of several recent DARPA robotics programs to have emphasized independent evaluative testing. As of this writing, it was followed by the LAGR (Learning Applied to Ground Robots) program. This program is characterized by extensive data collection, performed by an independent evaluation team, and objective measures of performance. Hence, the test results were not, in this case, generated by the authors, and they will not be reported here in great detail. Fortunately, the test team has also published the test results in Krotkov et al. (2006). This article focuses on the architecture of the system and the design issues, elements and solutions which have consumed most of our attention in the quest to improve our performance in field tests.

The PerceptOR program approaches outdoor autonomous mobility research from the perspective that, while everything needs improvement, perception performance is the factor most limiting the performance of contemporary systems. Nonetheless, relevant experimental work can often only be conducted in the context of an operational fielded system containing all essential components, so the performance evaluation is conducted on such a system.

Over a three year period, the objectives and our experiences in field tests have driven us to implement new approaches at all levels of the traditional autonomy software hierarchy—from gross motion planning to reactive low level control. The article discusses the motivation for, and design, and the implementation of some of the most significant new elements.

Early designs and intermediate results have been reported elsewhere (Stentz et al. 2002, 2003). The article outlines the final design of the system produced by the CMU PerceptOR team, results achieved, and some of the most immediate challenges that remain to be addressed.

This article describes a software system consisting of perhaps 500 000 lines of source code. Its integration and tuning on the PerceptOR program alone required perhaps 30 man years of effort. Some of the integrated algorithms themselves, including local and global planning, helicopter control and mapping, and infrastructure and support elements, have a 15 year development legacy. The fielded system consisted of all of the core hardware and software components than have existed on any similar previous system at CMU.

1.1. Related Work

The scope of this effort makes it impossible to provide anything other than an abridged treatment of all of the prior work on every relevant application area and subspecialty of robotics The following summary attempts to identify a few seminal, summary, or representative papers, or influential systems from the history of the field while also pointing out some of the new trends.

Although this work has used an unmanned air vehicle (UAV), it will not be described in detail. UAVs have become a popular area of research in recent years. Recent trends include, for example, custom miniaturized vehicles (Bouabdallah, Murrieri, and Siegwart 2005). The UAV was used largely as a pre-existing system in this work, but two papers have been produced to disclose our UGV-UAV integration concept (Stentz et al. 2002, 2003). Detailed descriptions of the helicopter UAV system and references to the field in general can be found in: Amidi, Kanade, and Miller (1998); Miller and Amidi (1998); Miller et al. (1999).

Related work for unmanned ground vehicles can be organized in terms of applications. The potential applications of off road robotic ground vehicles have long been recognized. Much of the work to date has been motivated by military (Baten et al. 1998; Krotkov et al. 2006) and space (Andrade et al. 1998; Schenker et al. 2003) applications although agriculture (Hagras et al. 2002), mining (Roberts, Corke, and Winstanley 1999), and forestry have received more recent attention. On this latter point, there is an extensive literature on the topic of field robots (Thorpe and Durrant-Whyte 2001) which relates to the uses of mobile robots in diverse applications. Such work relates to substantially stationary vehicles, semi-structured environments, multi-robot coordination, and the position estimation problem in the absence of obstacles, etc. This review of prior work has avoided addressing related work of such broad application scope in order to concentrate particularly on work which relates to the goal of driving long distances in challenging environments with minimal human intervention. This is the core problem addressed by PerceptOR.

Related work can also be organized in terms of environments, whether terrestrial or not, the degree of structure assumed, and the degree of support from humans assumed. One of the earliest autonomous mobile robots was the Stanford Cart which, in 1978, spent more time thinking than moving (Moravec 1980) in a very structured environment. An early JPL robot (Thompson 1977) also constructed polygonal terrain models from visual input. A decade later, robots appeared which were able to achieve continuous, very low speed motion (Olin and Tseng 1991). Three major bodies of work, distinguished by environments, began to emerge: unmanned ground vehicles (UGVs), road followers, and planetary rovers.

Early work in unstructured, terrestrial, off-road terrain tried to achieve a balance between speed and terrain complexity (Chang, Qui, and Nitao 1986; Dunlay and Morgenthaler 1986a; Daily et al. 1988; Keirsey, Payton, and Rosenblatt 1988; Olin and Tseng 1991). The number of elements of these early systems which can still be clearly identified in today's off-road systems is a striking testament to the quality of this early work. Speeds continued to increase with improved approaches and advances in the underlying technology (Feng, Singh, and Krogh 1990; Mettala 1993; Langer, Rosenblatt, and Hebert 1994; Sreenivasan and Wilcox 1994; Goldberg, Maimone, and Matthies 2002).

Other work emphasized the valid assumptions of road operations, and vehicle speeds were increased as a result (Dunlay and Morgenthaler 1986b; Dickmanns and Zapp 1987; Baten et al. 1998). Only twenty years after the Stanford Cart, these systems were able to operate at highway speeds (Pomerleau and Jochem 1996). On occasion, road following principles have been applied to semi-structured environments such as underground mines (Duff and Roberts 2003). The leader–follower scenario, where a possibly manned lead vehicle is followed by one or more unmanned followers has also been called *convoying* and *platooning*. It has been an important strategy for early deployment in military scenarios (Gage and Pletta 1987; Kehternavaz, Griswold, and Lee 1991; Broggi et al. 2000; Shoemaker and Bornstein 1998) and it is often proposed for either off-road or on-road scenarios. Another important strategy has been teleoperation (Aviles et al. 1990; Lescoe, Lavery, and Bedard 1991; Mettala 1993). This subfield generated some of the earliest experiences with the many difficulties of human-machine interfaces including interface delays, situational awareness and the challenges of effective collaboration over low bandwidth communications.

A third body of work emphasized rougher terrains and improved environmental models for the purpose of planetary exploration (Giralt and Boissier 1992; Simeon and Dacre-Wright 1993; Iagnemma et al. 2004). Of course, the successful deployment of the Mars Exploration Rovers is a landmark in the history of robotics. As of this writing, the rovers known as Spirit and Opportunity have logged more than 6 km of autonomous traverse on another planet (Biesiadecki, Leger, and Maimone 2005).

The quest for relevance to power-limited planetary rovers fostered specialized efforts in passive sensing (Nishihara 1984; Faugeras et al. 1993; Langer, Rosenblatt, and Hebert 1994; Kanade et al. 1996) which were also relevant to the desire for passive sensing in UGVs. In recent years, stereo vision has become an off-the-shelf commercially available modality (Konolige 1997). While many present day robots use single scan line ladar ranging devices, only a few efforts have been undertaken to develop a two dimensional ladar designed for outdoor robots (Bergh et al. 2000) and no solution designed for this application is commercially available today.

Work desiring to characterize and understand the terramechanical issues of ground vehicle mobility arose first in the context of planetary rovers (Sreenivasan and Wilcox 1994). Later, planning algorithms began to consider deformable terrain (Cherif 1999). More recent work has concentrated on the prospects for learning terrain parameters on-line (Iagnemma et al. 2004; Ojeda et al. 2006; Manduchi et al. 2005).

A subspecialty in computer vision has addressed the need for both environmental modeling and obstacle detection algorithms. Starting in the late 80s this work had emphasized the use of laser rangefinder data (Hebert, Kanade, and Kweon 1988; Nashashibi et al. 1994; Gowdy et al. 1990; Talukder et al. 2002) to produce local elevation maps and somewhat later, stereo vision (Matthies and Grandjean 1994) was used. The value of merging color and range information for terrain classification was recognized as early as Langer, Rosenblatt, and Hebert (1994) and Belluta et al. (2000). Recent work, which has inspired some of our efforts, includes the application of PCA to range data in Vandapel et al. (2004) and the more contemporary quest to distinguish compressible vegetation from incompressible obstacles (Macedo, Manduchi, and Matthies 2000; Lacaze, Murphy, and DelGiorno 2002).

Simultaneously, the remote sensing community has addressed similar problems from an aerial perspective that are now being addressed by ground robots. The absorption properties of chlorophyll (Willstatter and Stoll 1913; Clark et al. 2003), for example, have been exploited elsewhere for a long time.

The value in route planning of expansive digital elevation maps, obtained from aerial and space platforms, has also been recognized relatively early (Hotz, Zhang, and Fua 1993) and it continues to be used today (Lacaze, Murphy, and DelGiorno, 2002). Also, the cruise missile TERCOM system has been an important existence proof of the value of terrain elevation information in guidance (Golden 1980).

Unlike the case of scanning laser radars, commercial solutions for vehicle position estimation have been available since before the dawn of modern robotics. Commercial integrated INS-GPS solutions are available today across the price-performance spectrum. Nonetheless, desires to reduce costs, integrate perception, or control the filtering used have led researchers in robotics to develop their own custom inertial navigation from component sensing (Barshan and Durrant-Whyte 1995; Roumeliotis, Johnson, and Montgomery 2002; Sukkariel et al. 2000).

Robotics and computer vision have pursued two related challenging problems for several decades. The visual odometry solution implemented here (Nistér, Naroditsky, and Bergen 2004), is closely related to the problem and the solution originally posed as structure from motion (Huang 1994). However, in robotics, the more commonly encountered formulation of the problem is simultaneous localization and mapping (Dissanayake et al. 2001). In recent years, solutions have appeared which are tailored to expansive outdoor environments (Guivant and Nebot 2003).

By analogy to perception, a class of motion planning algorithms specifically designed for outdoor mobile robots has also evolved over time (Olin and Tseng 1991; Chatila and Lacroix 1995). The planner that drove the Mars Exploration Rovers for several autonomous kilometers is described in Goldberg, Maimone, and Matthies (2002). Among the more enduring local planners used for unmanned ground vehicles is that reported in Lacaze et al. (1998).

1.2. Problem Statement

Off-road autonomy is among the most ambitious of our aspirations for mobile robot technology. Such robots must operate effectively under forest canopy while positioning satellite signals are occluded and the trees themselves present natural mazes to challenge motion planning. In both fields and forests, ground covering vegetation obscures the shape of the underlying solid surface (if any), while it also hides any underlying mobility hazards. In alpine areas, terrain slopes require safe operation subject to the persistent threat of rollover, while precipitous ledges threaten to end the mission in an instant.

While active sensing sometimes performs better in the absence of background solar illumination, effective passive sensing under low light and darkness conditions remains an unsolved problem. Precipitation, dust, smoke, and fog easily defeat most optical sensing modalities while relatively long wavelength sensing like sonar and radar is effective only at the cost of much reduced, and frankly inadequate, angular resolution.

The set of situations to avoid is much more diverse than for indoor robots. On the assumption that atmospheric and ground cover conditions permit effective sensing, the robot must assess and correctly avoid, negotiate, or engage such hazards as:

- Any impact with objects, terrain, or vegetation which would exert damaging forces on any part of the robot including wheel and undercarriage collisions with rocks (positive obstacles) or the near or far edge of holes or ledges (negative obstacles), forward, reverse, or side collisions with trees, and perception sensor collisions with overhanging branches.

- Any loss of mobility caused by lost traction due to terrain mechanical properties (mud, ice) terrain geometric properties (steps beyond vehicle capabilities, slopes), or lost reaction forces (deep water, ballistic motion, rollover).

- Any loss of mobility caused by retarding shear forces (high centering) or compressive or tensile forces leading to entrapment.

While physics provides the means to predict vehicle motion (given gravity and the material properties of the environment), such properties must often be inferred from the very indirect information available from visual and geometric sensing. The robot will often have path options for how to achieve its goals. However, the conservative approach of avoiding anything that might be a problem can lead, in challenging terrain, to an overly timid robot incapacitated by an inflated perception of the risks it is facing.

While the challenges are daunting, the potential for autonomous field robots to increase our productivity while isolating humans from danger is high. Nonetheless, well-intentioned and carefully constructed demonstrations intended to illustrate the potential of field robots are not (and are not intended to be) rigorous evaluations of capability in the real world. One of the objectives of this work has been to better quantify contemporary performance and reliability levels in order to set an agenda for the future. This desire leads to a need to perform experiments in a more controlled fashion, to record significant amounts of data, and to analyze it with a view toward generating an objective assessment. Our task has been to construct the most capable off-road mobile robot we could, and to subject it to challenges as well-quantified as we could, while recording as much useful data, in the most objective fashion possible. The most complete description of the test results appears in Krotkov et al. (2006) under the authorship of the test team. This article concentrates on documenting the system architecture and concept design while documenting its relationship to the problems encountered and lessons learned during the program execution.

1.3. Outline

The article is organized into 12 sections as follows:

Section 1. Introduction—this introduction

Section 2. Experimental Approach—presents the organization of the test team into multiple groups and the test philosophy.

Section 3. Platform Design—outlines the UAV and UGV hardware designs.

Section 4. Software Design—describes the UGV software at an architectural level.

Section 5. State Estimation—describes unique aspects of our approach to position estimation during intermittent GPS availability.

Section 6. Reactive Planning and Control—describes a low level reactive subsystem empowered to respond to exceptions often due to failures in higher level autonomy layers.

Section 7. Perceptive Autonomy—describes unique aspects of our approach to perception in forested and vegetated terrain.

Section 8. Mapping—describes unique aspects of our approach to mapping and data fusion of aerial and ground derived data.

Section 9. Map Processing—describes salient aspects of our algorithms to process terrain data generated by aircraft and satellites.

Section 10. Motion Planning—describes the integration of two levels of planning to preserve the capacity to think strategically while reacting quickly to perceptual information.

Section 11. Project Results—presents a chronology of the six main field experiments and how they have influenced the design presented earlier.

Section 12. Conclusions and Outlook—describes the major lessons learned, and major unresolved issues that seem worthy of further investigation.

2. Experimental Approach

DARPA structured the PerceptOR program for an unprecedented level of controlled experimentation.

2.1. System Under Test

The goals of the program are to assess the readiness of technologies for autonomous ground mobile robots. Nonetheless, the system under test is considered to be a man-machine team consisting, in our case, of one operator, one ground robot, and sometimes, one air robot. The air robot is conceptually under exclusive control of the ground robot. In some tests, the operator is permitted to monitor the ground robot and intervene at will (at the cost of incurring communications bandwidth). In others, the operator may view data and assist the robot only when it asks for help or when test administration is interested in having the operator attempt to correct a robot failure that caused an emergency stop (ESTOP).

2.2. Performance Metrics

The desire to measure and improve performance leads naturally to a quest to define metrics that quantify performance. Among the data collected in each test is the number of times field safety personnel intervened in order to prevent vehicle damage, the communications bandwidth used to and from the operator, system ability to reach desired waypoints, instantaneous and average speed, distance, and time. System developers also record all data into and out of the robot in order to facilitate their own analysis, aimed at improving performance in subsequent tests.

The overall goal has been to simultaneously maximize autonomy, reliability, and speed. An attempt to maximize overall test scores raises the issue of how different aspects of performance should be traded against each other to produce a single scalar utility metric for comparison purposes. For example, increased speeds come at the cost of increasing demands on computation, resolution, and system response times, so a given system design would be expected to become less reliable at higher speeds. The test philosophy has been to avoid the temptation to define such an ultimate metric in order, among other things, to test the technology rather than the creativity of the development team in tuning the system trade-offs to the immediate environment.

2.3. Test Approach and Execution

The PerceptOR program emphasizes independently administered evaluative testing as the primary mechanism to drive progress. Specifically, the sequel will distinguish regularly the development team, which constructed and maintains the robot hardware and software, from test administration, whose role is to independently define tests and assess performance.

2.3.1. Blind Evaluative Testing

Tests are conducted on an unrehearsed basis, meaning the development team has no detailed knowledge of specific terrain before the tests commence. Software maintenance during a period of continuous testing (lasting perhaps one week) is permitted as necessary but the intent and desire is to prevent extensive tuning of system behavior to the terrain before the testing starts. While the development team may eventually see the test courses during the conduct of the tests, the specific individuals who operate the system are denied any undue information until all tests are complete. The operator is “firewalled”—deliberately placed in a windowless remote location, able to assess the situation only via video and data telemetry, and isolated from all information sources that would not be available in a realistic setting. At times, operators may receive a prior briefing of the relevant aspects of the test site based on what analysts can determine from aerial and satellite imagery. The operator is also often denied significant time to study any prior information that may be provided. The overall intent is to simulate the conditions of actual deployment of UGVs where no one would or could know the terrain in detail. A secondary intent is to allow assessment of how operator experience with the terrain affects performance.

2.3.2. Test Operations

For each test, the operator is initially provided with an ordered series of goals to achieve in the form of GPS waypoints. These are chosen by test administration and provided, possibly augmented by aerial ladar and color imagery, only immediately before test execution commences. The execution of tests follows a structured process involving several teams. The test administration and development teams are reorganized into three teams during test execution as outlined in Figure 1.

In a more realistic setting there might be an operations team consisting of several human operators (for several robots) but our interest in reducing human supervision levels, and in evaluating operator workload, led to a reduction of this team early in the program to only one operator on duty at one time.

The field team (Field) follows the robot in the field. This team is responsible for positioning the robot at the start waypoint, and repositioning it in the event the robot immobilizes itself etc. Field is also responsible for keeping the robot safe (via wireless emergency stop, called ESTOP) and recording with measurements, imagery, and video, such performance indicators as excursion time, waypoint achievement, and apparent assessment of hazards it encounters. ESTOPs are permitted to prevent robot damage or human injury, but they sometimes indicate that the robot would otherwise have made a mission-ending decision. These events are therefore tracked carefully and considered to negatively affect performance evaluation. The ESTOP operator must walk a difficult line between placing the robot at undue risk and unnecessarily preventing it from encountering a difficult area that it would have correctly assessed. A second field team is responsible for following the air vehicle and is similarly empowered to intervene for safety or test administration reasons.

The field teams are complemented by the analysis team (Analysis). This team is housed in a remote “situation room” incorporating multiple projected displays and it listens to all radio, video, and data traffic. It has, in principle, all information available to it. This team receives data feeds from the operator workstation (to see what the operator is doing), data and sensor video feeds from the robot (including GPS position to allow tracking motion over geo-registered aerial imagery

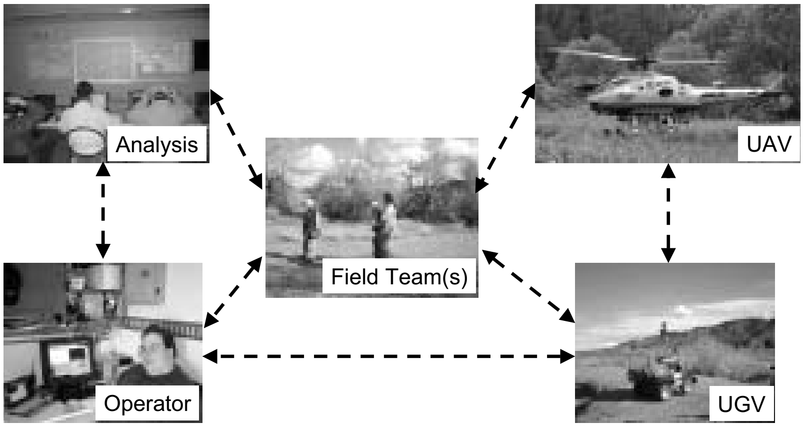

Fig. 1. Operations, field, and analysis teams. Several remotely situated and partially firewalled groups are coordinated in each test.

and topographical maps), and video and audio feeds from the field teams. Analysis is uniquely able to assess and correlate what the robot thinks is happening, what the operator thinks is happening, and what is really happening. This team may interrogate the operator or ask for clarification of what the field video is showing but multi-channel radio communication is routed and controlled to keep the operator uninformed.

In principle, test administration operates independently from the development team which constructed and maintains the robot system. However, test execution was found to be more effective when members of the development team supported both the Field and Analysis teams by being physically present, and engaged in the conduct and analysis of tests.

3. Platform Design

Intuitively, for global planning purposes, given a choice between the view of a ground vehicle or an air vehicle, the aerial view would be preferred due to its broader viewing range and its relative immunity to terrain and (small) vegetation occlusion. A second advantage of the aerial view is its unique capacity to see inside negative obstacles which are so difficult for ground vehicles to detect.

Our top level approach to improving unmanned ground vehicle (UGV) system performance was to partially free the ground vehicle from the limitations of a ground based perspective by employing an organic unmanned air vehicle (UAV) under the direction of the UGV. Other aspects of the approach included fusion of multiple perception sensing modalities in order to better discriminate the multiple hazards listed earlier.

3.1. Retrofitted Ground Vehicle

The ground vehicle is based on the Honda Rubicon All Terrain Vehicle (ATV). Elements for human-aboard driving were removed entirely and replaced with a custom autonomy retrofit. It was considered important to have two operational vehicles at all times in order to continue testing on one while the other was being serviced. Two design iterations produced two generations of UGVs. The first two vehicles were designed to support carrying and launching the UAV from a platform on each UGV. A few performance issues including weight, high center of gravity, and reliability were addressed in a second generation design based on the same base ATV platform.

These second generation vehicles added encoders and suspension instrumentation for improved 3D odometry, and a modified transmission for more reliable gear shifting. The newer design also included an auxiliary generator to power additional sensors and computing. Details of the second design are provided below.

3.1.1. Sensors

The UGV is equipped with several perception sensing modalities (Figure 2) that can be grouped into several categories:

Pose sensors. Odometry sensors (engine encoder and steering angle potentiometer), Novatel RTK 2cm DGPS, ground speed radar, suspension travel linear potentiometer, and a Smiths land inertial navigation unit.

Mapping and obstacle detection sensors. Custom-made 2-D scanning ladars based on the SICK LMS sensor, 2 stereo camera pairs with hardware stereo engine based on the Sarnoff Acadia chip, several color and FLIR cameras to help with terrain classification, and a pressure sensitive bumper. There are also three different variations of the scanning SICK ladar. The front bottom unit is a horizontally scanned ladar (the fast axis is vertical), and the front middle unit is a vertically scanned ladar. The top and the rear/mast units are both horizontally

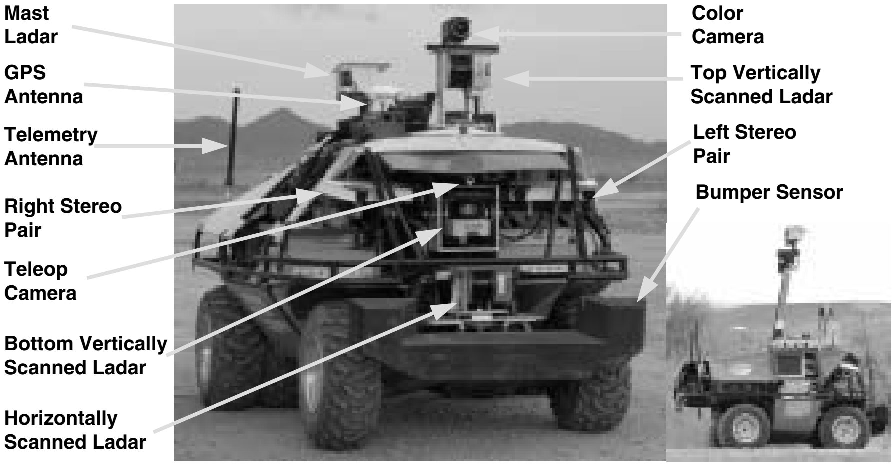

Fig. 2. Retrofitted UGVs. (Left) Frontal view of UGV showing most external sensors. (Inset) Side view of UGV showing mast extended. The mast raises the scanning ladar to a height of 2.5 m—significantly increasing the detection distance for negative obstacles.

scanned ladar, but each is also equipped with a slip ring, so it is capable of spinning 360 degrees continuously, capturing data from all around the vehicle.

Teleoperation sensors. Several cameras are mounted around the body for exclusive use by the operator. The operator also has access to all other sensor data.

3.1.2. Computing

Major computing needs include range image perception, local and global planning, stereo ranging, video compression, data logging and low level control. Each UGV is equipped with six GHz speed computers and a custom stereo processor:

Sensor interface computer. This computer, a 2.4 GHz Pentium IV, interfaces to most of the sensors with the notable exception of the stereo and teleoperation cameras. It also serves as the wireless ethernet router for the other computers.

Planning and stereo computer. This computer, a Dual 1.8 GHz Athlon, houses the Sarnoff Acadia Stereo engine card, which processes the stereo images to produce depth maps at high update rate. It also serves as the computing platform for both the local and global planners.

Video logging computer. This computer, a 1.8 GHz Pentium IV, digitizes compresses and logs the images from four on-board teleoperation cameras. These images are then made available to the remote operator on-demand. Several “TIVO” like capabilities are made available to the operator, allowing real-time, time shifted, or distance shifted video to be displayed at any time.

Auxiliary interface computer. This computer, a 2.4 GHz Pentium IV, digitizes and fuses the data from the top ladar and camera, allowing the remote operator to view fused range and color imagery.

Low level controller. This computer, a 1 GHz VIA C3, controls all the actuators, and interfaces to all the low-level sensors, such as the engine encoder, steering and mast angles.

All computers are housed in three separate air-cooled enclosures. Gigabit ethernet is used to communicate data between computers. All computers are provided with a common pulse per second (PPS) signal to achieve real time clock synchronization across all processors to within 50 microseconds. Such precise synchronization of the different computers is critical to ensure accurate registration of sensor data routed through processors with otherwise independent clocks. Among other things, this capability permits perception data to be pose tagged very precisely and it permits the generation of multimodal imagery discussed in the sequel. The PPS signal generates an interrupt which is used to determine the offset between the local and master clocks.

3.2. Retrofitted Air Vehicle

The UAV, a pre-existing CMU helicopter test bed (Figure 3), collects and transmits world-registered 3D laser and stereo-matched range and obstacle data during flight. It is built over

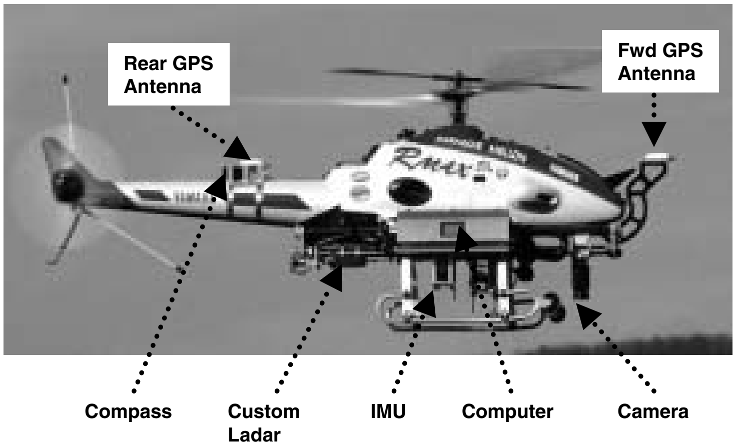

Fig. 3. Autonomous Air Vehicle. Yamaha Rmax Helicopter was retrofitted for autonomy.

a Yamaha Rmax platform. The airframe is 14 ft long, weighs 210 lb, contains a 250cc 2 cycle engine, and has a 77 lb payload capacity. The system supports fully autonomous takeoff, landing, and way point following.

3.2.1. Sensors

The high resolution custom-designed ladar scanner is based on a Riegl spot laser rangefinder. It provides 28 KHz measurement frequency and an accuracy of 2–5 cm. The single axis spinning mirror scanner (push-broom configuration) is strapped down to a calibrated chassis. A pair of Sony digital 1M pixel color cameras provides stereo data over a firewire interface. State estimation is based on a Litton tactical grade inertial measurement unit (IMU) and two carrier-phase L1/L2 GPS Novatel OEM-4 WAAS receivers. A custom 15th order Kalman filter implements a latitude-longitude inertial navigation system mechanization, providing state estimates at 100 Hz.

3.2.2. Computing

1 GHz Pentium-based processors running vxWorks are used for ladar and stereo sensing and they are augmented by custom interface boards for the Navigation computer.

3.3. Integration

Our pre-existing UAV was a capable, completely autonomous robot in its own right. Raw range data (whether ladar or stereo) was transmitted down to the UGV over wireless and the uplink to the UAV consisted of sparse waypoints to follow, given its capacity to close its own motion and attitude control loops based on its own native guidance system.

For reliability, communications between the UAV and UGV were asynchronous and the UGV did not depend either on the availability of UAV data or on the UAV receiving and executing its commands. Both vehicles were self–reliant and independently competent.

4. Software Design

The UGV software system described in this paper has evolved from local (Kelly and Stentz 1998) and global (Saridis 1983), planning systems that we developed for the Demo II program. Although they were developed some time ago, they were integrated closely for the first time in the work described here.

4.1. Multi-Layer Architecture

The overall architecture, like very many contemporary systems, consists of several nested layers of intelligent behavior. One common multi-layer architecture (Gat 1998) distinguishes stateless (control), memory-based (sequencer) and predictive (deliberator) layers. Another distinguishes organization, coordination, and execution layers (Saridis 1983). Like Albus (1992), our architecture distinguishes reasoning on different time and space scales. This structure arises fundamentally from the need to be both intelligent and responsive at the same time. This architecture is presented more from the perspective of an after-the-fact description of the result of multiple design iterations rather than a before-the-fact design that guided our development efforts.

As shown in Figure 4, three layers known as reactive autonomy, perceptive autonomy, and deliberative autonomy are distinguished. Each layer has the capacity to control the vehicle based on information which is more abstract, more spatially expansive, and of less spatial and temporal precision

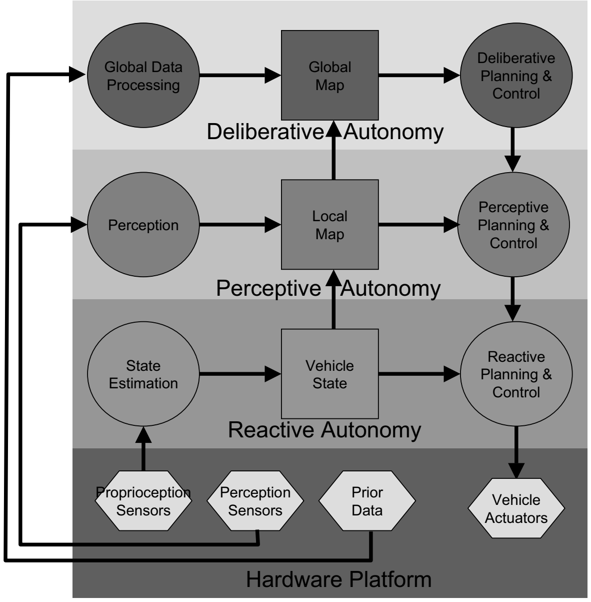

Fig. 4. Software Architecture. Three nested layers of autonomy are shown. Higher levels use more abstract, lower resolution, spatially expansive models, while incorporating more memory, longer prediction horizons, and consequently slower response times.

than lower layers. Higher layers are characterized by both increased memory and increased predictive horizon, and this comes at the cost of reduced response times.

4.1.1. Reactive Autonomy

This layer performs functions requiring fast response times and this responsibility tends to result in the capacity to implement only limited or even no memory or look ahead.

4.1.2. Perceptive Autonomy

This layer is defined by heavy reliance on the processing of radiative, non-contact sensing. It uses a commensurate amount of memory and prediction of the future in order to react intelligently to the proximal and/or immediately visible environment.

4.1.3. Deliberative Autonomy

This layer uses expansive models of the environment, and considerable amounts of memory and look ahead, to guide the vehicle on a strategic basis.

4.2. Data Flow

Each layer involves a sensory/data interpretation component, a world model, and a planning/control component. Vehicle state is considered a world model. Reactive autonomy, despite its name, can engage in limited look ahead. In addition to the population of world models by sensor and external data, world model information flows upward in the model hierarchy and it is abstracted and/or reduced in precision as a result. Conversely, command information flows downward and is elaborated and increased in precision as a result.

Whereas each layer has its own dedicated source of information which is unknown below it, the actuation side is characterized by increasingly abstracted models of the same underlying vehicle as the hierarchy is ascended. Each layer influences the behavior of a single contended resource—the vehicle—through such mechanisms as:

- hierarchical control: lower levels elaborate and follow higher level commands;
- arbitration: deliberative elements must resolve disagreements arising from conflicting simultaneous goals such as obstacle avoidance and waypoint seeking;
- exception handling: faster elements deal with exceptions quickly and propagate them upward.

When higher layers are present, they provide commands which are elaborated and executed by lower layers. However, each layer has the capacity to control the vehicle independent of the layers above it to generate respectively the behaviors of (dense) waypoint following, obstacle avoidance, and global path planning. Each layer has the authority to ignore the commands passed to it in order to respond quickly to an exception or emergency.

We have found it to be reasonable to trust the lowest level decision maker to bring the vehicle to a stop or avoid the obstacle immediately in view. Lower levels may have shorter sensor horizons and limited contextual information but they are more competent in detecting exceptions and in implementing default responses which remove them. Generally, the context for such events is that higher levels of the hierarchy have already failed, so lower levels are granted the authority to temporarily suspend the quest to achieve higher level goals in order to bring the vehicle to a safe state and then ask for higher level guidance.

The local map is a 2-1/2D environment representation whose main purpose is to record and organize perceptually-derived information, at fairly high resolution, in a region surrounding the vehicle whose size is commensurate with the planning horizon of local planning. The global map, by contrast, accumulates data over a much wider area at reduced level of detail. Data in the local map is positioned based on the local pose estimate, and data in the global map is positioned based on the global pose estimate. These estimates are described immediately below.

5. State Estimation

Although position estimation was not intended to be the focus of our work, the extreme dependence of perception, mapping, and planning performance on the performance of position estimation resulted in significant effort to achieve required robustness levels in natural environments.

5.1. Dual Pose Estimates

For reasons elaborated later in Section 8.1, two different state estimates are provided by the state estimation system. Designated "global" and "local" these estimates are distinguished by their management of the trade-off between absolute and relative accuracy that arises when a high quality position fix arises after a period of pure dead reckoning. From the perspective of absolute accuracy, it is appropriate to incorporate a high quality fix and to allow the estimate to jump to the improved estimate. From the perspective of relative accuracy (roughly, smoothness), it is appropriate to ignore the improved fix so that environment models based on data which spans the jump do not themselves incorporate a discontinuity.

The global pose estimate is computed from all available sensing several times a second. This estimate is used to control motion with respect to globally specified waypoints, and to fuse onboard and externally provided geo-referenced data.

The local pose estimate is computed based only on sensing which does not project onto position states. It does not process GPS readings, nor would it process terrain aids, such as absolute landmarks, if they were available. This estimate drifts with respect to the global estimate but it is completely immune to characteristic jumps in GPS position. Such jumps occur before loss of lock on satellite(s), during constellation changes, because of multi-path effects, and at the re-acquisition of satellites whose signal was temporarily occluded. This estimate is used to provide feedback to motion control and to estimate motion relative to obstacles for obstacle avoidance purposes. Neither of these uses requires an absolute sense of position on the earth. Note however, that the perceived slope of a patch of imaged terrain depends on its shape as perceived by vehicle-fixed sensing and the attitude estimate used to place the individual point data in a terrain map. Hence, it was important that the pitch and roll estimates of local pose have bounded absolute error.

5.2. Local Pose—3D Odometry

Position estimates satisfying the stringent requirements of ladar data registration are not available yet from commercial sources, so it was necessary to construct our own estimate. 3D odometry using wheel encoders and the attitude and heading from the Smiths land inertial unit were the basis of our local pose estimate. This navigation solution amounts to integration of the linear and angular velocity vectors while accounting for their directions in 3D. In principle, the problem is straightforward under certain ideal assumptions. In practice, several nonidealities come into play, as discussed below.

5.2.1. Wheel Slip

Wheel slip is a common occurrence but it is not completely observable from measurements of wheel rotation rates. These errors are corrected in the global pose by the GPS measurements, but currently they remain a problem for local pose. Radar-based speed sensors were quickly eliminated from consideration as viable terrain relative speed sensors for this application because of their dead band near zero speed and unidirectional sensitivity. We also investigated visual odometry, as discussed below.

5.2.2. Suspension Deflection

Given the weight distribution of the vehicle, the deflections of the suspension system depend on terrain pitch and vehicle acceleration. Since the vehicle velocity vector is generally parallel to the terrain tangent plane, these deflections imply a misalignment with respect to the forward axis. Attempts to compensate based on suspension measurements were not as accurate as desired due to additional unmeasured deflections generated by tire compression. Our final solution to this problem was to fit a linear model that predicts chassis attitude from body attitude.

5.3. Visual Odometry

Sarnoff Corporation developed a visual odometry system for use on the UGVs (see Nistér, Naroditsky, and Bergen (2004) for a description of the system). Although it was never integrated into the full system, the final field test showed some promising results. Performance of visual odometry was measured on two vehicle runs. The first was a waypoint-following run during which the vehicle was denied GPS measurements. The second was driven manually by an experienced operator to simulate the performance of the planning and perception systems (i.e., not a straight path from one waypoint to the next). Both of these runs showed improvement over the older version of the 3D odometry system (without the Smiths inertial system). Table 1 summarizes the results.

5.4. Global Pose

The global pose estimate is determined from a nonlinear Kalman Filter which uses all of the information available for local pose as well as GPS. The nonlinear nature of the Kalman filter embodies the assumption that the vehicle is pointing in the direction in which it is moving. Consequently, GPS fixes

Table 1. Visual Odometry Results

Run	Mechanical Odometry Error	Visual Odometry Error
Autonomous (96 m total)	0.711 m	0.678 m
Manual (163 m total)	5.52 m	3.65 m

are used to update vehicle attitude and heading as well as position. While this can be valuable, jumps in GPS can cause jumps in attitude and heading as a result.

The initial PerceptOR system used a heading gyro along with an attitude measurement system (vertical gyro/reference) as input to 3D odometry. The poor performance of the heading gyro made it difficult to distinguish heading drift due to gyro errors from heading drift due to GPS jumps. Various outlier filtering schemes were tried originally with limited success. Eventually we were able to use the relatively high quality attitude and heading from the Smiths inertial unit to reject filter innovations due to unreasonable jumps. This technique resulted in long term accuracy due to GPS and short term accuracy due to the inertial sensing.

6. Reactive Planning and Control

This component is analogous to the autonomous nervous system in animals. Its job is to monitor system health and react instantly to external stimuli indicating danger. In the absence of such situations, this component closes coordinated control loops which depend on knowledge of the vehicle state in order to execute the commands coming from the layer above.

6.1. Trajectory Following

In principle this layer is responsible for following continuously specified paths or dense waypoints, for articulating sensor heads based on body attitude, and even for visual servoing if bandwidth requirements justify this. The capacity to do much of this was not used on PerceptOR due to the sparse waypoint description of the task and the high update rates of the perceptive autonomy layer.

6.2. Timeout Exceptions

As is typical in many real-time systems, a watchdog timer was implemented to cause an exception to be raised, and an emergency stop to be issued, if the low level controls did not receive a command at least every 500 ms. Operator intervention would normally be required to resolve this situation.

6.3. Collision Exceptions

Whereas perception has the primary responsibility to detect obstacles, several reactive measures are also implemented here in order to detect and respond to failure of higher layers.

Three classes of “bumpers”, virtual, inertial, and physical, were created over the course of the program to address such situations as software bugs, obstacles occluded by vegetation, and occasional legitimate needs to drive blindly when sensors are occluded by tall grass or trees.

Virtual bumpers were regions immediately in front of the vehicle which were rapidly and repeatedly tested for intersections with a subset of all ladar data. Virtual bumpers had to be shut off in tall grass. Inertial bumpers were based on detecting rapid changes in the accelerometer readings coming from the IMU. Such changes indicate a collision of some kind. Finally, a true mechanical bumper was mounted to the front of the vehicle and equipped with sensors to indicate pressure. A threshold on these indications could indicate either a collision or a situation where vehicle motion was being resisted successfully.

Once a bumper was activated, the system was safely brought to a halt and, depending on configuration settings it would either attempt to autonomously avoid the obstacle or ask the operator for help.

6.4. Control Alarms

Three classes of system health alarms are related to motion measurements. A mobility alarm is raised under conditions of an excessive following error. Such error occurs either with motion that was not commanded (e.g., due to a broken encoder wire) or with lack of motion (e.g., due to an obstruction) in response to an issued command. A mobility alarm would typically require operator intervention to resolve.

Attitude alarms detect proximity to static tip over and are tripped when the vehicle attitude exceeds predefined safe limits that incorporate some margin. Again, operator intervention would often be required since autonomy had failed to understand the situation already.

Stability alarms check the specific forces from the strapdown IMU against the support polygon in order to assess proximity to dynamic tip over. The design for such alarms was available (Diaz-Calderon and Kelly) but they were not implemented. Depending on the degree of proximity and the amount of dynamic contribution to the problem (versus slope), slowing down or reducing curvature may be a valid autonomous recovery mechanism.

6.5. Body Perception

Several useful field-validated techniques fall conveniently into a category defined by inferring properties of the environment based on how the body reacts to them. When a bumper alarm was raised for example, it was necessary to record an obstacle (after all, there can be no more direct evidence of one) at the position in front of the vehicle. Otherwise, the system would repeat the same mistake when the exception was cleared. Once the obstacle was there, perception was disallowed to clear it and the rest of the obstacle avoidance system reacted appropriately.

Other examples of body perception include clearing obstacles which might occur underneath the vehicle if there was direct evidence from motion sensing that there clearly was no obstacle. Finally, as discussed in the perception section, wheel elevation is a valuable check on the predicted elevation of the supporting surface when it can otherwise be predicted based only on inference of vegetation density.

7. Perceptive Autonomy

While many earlier UGV programs concentrated on barren and rolling terrain, our approach to UGV perception was forced to evolve in order to address the challenges of competent operations in vegetated and forested terrain.

7.1. Perception Architecture

The overall data flow through the perception system is represented in Figure 5. Notable aspects of the system include partially computational / partially optical generation of multi spectral, multimodal imagery, and a commitment to full 3D representations. Data fusion is distributed in the system, occurring at the state estimation and perceptual (image) levels, the mapping (map) level and the control (command) level. Obstacle/hazard detection is also distributed, with perception responsible for image interpretation and mapping/planning responsible for interpreting neighborhoods in the various maps.

7.2. Image Plane Registration/Interpolation

The output of this process is either a synthetic image formed from a single origin encoding range, color, and IR at every pixel (rangified-color) or a set of ladar range pixels with an associated color and IR value (colorized-range). When stereo vision was used, the registration of range and appearance data is intrinsic. While Near IR and color data was fused optically, these data streams were fused computationally with range data from ladar (Figure 6).

The availability of dense range data renders the problem of correlating it with projective appearance data a relatively straightforward process. It requires a calibrated relative pose between the sensor centers of projection but no search for correspondences. Colorizing range data is somewhat easier than “rangifying” color data due to vehicle motion and sampling issues. Appearance data was mapped onto sparser range data by subsampling for obstacle avoidance purposes and range data was mapped onto denser appearance data for human interface purposes.

7.3. 3D Volumetric Density Mapping

While, in many earlier works, systems have been able to function using an overhead planar projection of the robot and the environment, and while we continue to do so at the global level, the overhead projection of the local map is derived from a more detailed 3D representation (Figure 7). In this representation, the range and appearance information in the multimodal image is converted into colorized (x,y,z) points which are arranged into a coherent 3D array of point lists to enable rapid lookup based on position.

Like the planar local map discussed later, this data structure is implemented as a circular queue which wraps around in 3 dimensions in order to avoid the need to move the data in memory as the vehicle moves over distances much larger than the size of the map. Both the 2D and 3D local maps have coordinates aligned with and orthogonal to gravity to facilitate attitude predictions. Resolution in both cases is 20 cm on a side for both area and volumetric cells. Maps are typically extended out to about 20 meters (the point where perception data was of much less value) and are 20 meters wide.

In this data structure, the entire volume swept by the vehicle during candidate motions can be tested for collisions. While the height of the column of data intersected by the vehicle does not vary, its vertical position in the 3D map does vary. Nonetheless, given the vertical position of the column, it is possible to identify the relevant data and collapse its net assessment into a 2D cell in a planar representation used in subsequent planning calculations.

The basic attribute extracted in each 3D cell is the “density” computed as the ratio of the number of hits to hits plus misses similar in principle to that used in Lacaze et al. (2002) and Bilarski and Lacaze (2000). A cell gets a hit if the ladar beam terminates inside it and a miss if it was passed through. This calculation requires that each cell in the line from the sensor to the returning cell be identified in a comparatively expensive ray tracing process.

7.4. Geometry Based Terrain Classification

Among the more pervasive and demanding requirements for operations in vegetation is the discrimination of terrain from vegetation—of rocks from bushes. Experiments in the Sonoran desert, for example, indicated a pressing need to distinguish rocks from sage brush and similar growths. Failure to make the distinction leads to frustrating behaviors including unnecessary detours (of a timid system) around benign vegetation or collisions (of an aggressive system) with rocks misclassified as vegetation.

Quite often, pure geometric sensing using stereo or ladar provides scant basis for making this distinction but there is value in performing principle components analysis (PCA) on the data in each voxel (Vandapel and Herbert 2004). A multiresolution analysis of the ladar data was applied. Clutter-like objects found at coarser scales were analyzed at finer scales by

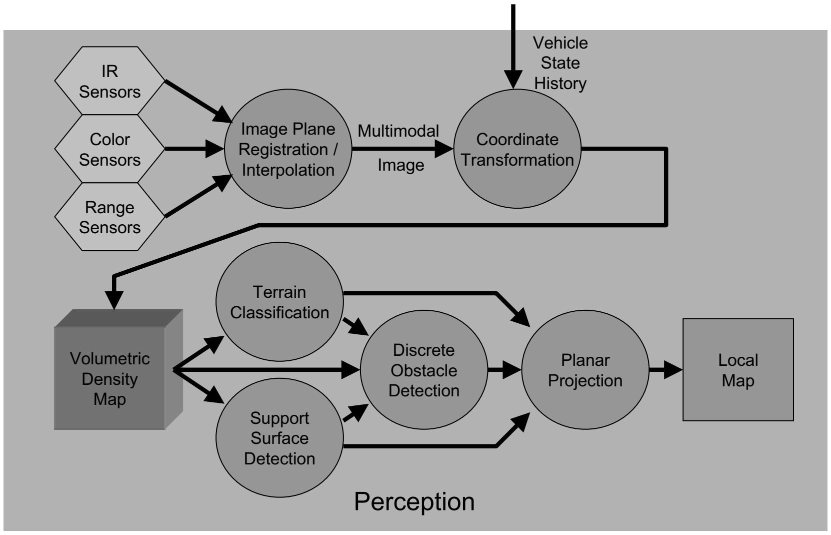

Fig. 5. Perception Software Architecture. Notable elements include partially synthetic generation of multispectral, multimodal imagery, and a commitment to full 3D representations.

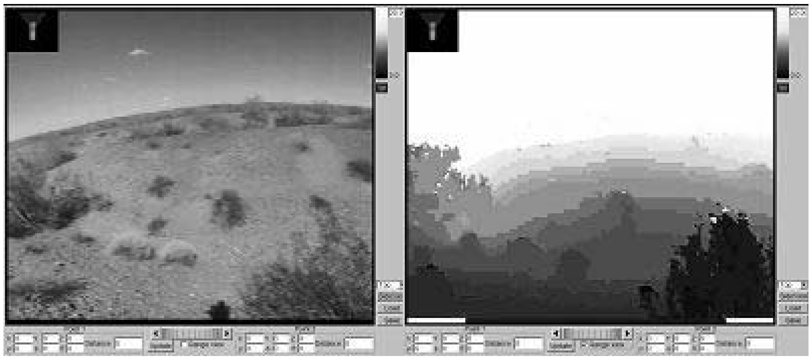

Fig. 6. Computational Image Plane Registration. Dense range data is relatively easy to correspond with appearance data given calibrated inter-sensor poses and the ability to correctly associate historical vehicle poses with range pixels.

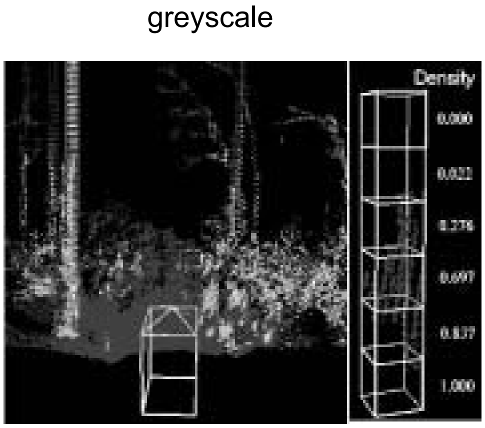

Fig. 7. 3D Perception. Left: The point cloud to the left indicates the typical 3D structure of a forested scene on a trail. Right: Density scores in a vertical column through the point cloud.

means of the PCA of their sub regions. Rocks and sagebrush are both clutter-like at coarser scales, but the former often reveal planar surfaces at finer scales, allowing for discrimination. The usefulness of this method was borne out in the first long distance run carried out at Yuma Proving Grounds, when the vehicle was able to traverse considerable vegetation without colliding with any rocks.

7.5. Multi-Spectral Terrain Classification

The unique spectral absorption properties of chlorophyll provide an additional valuable cue. Kauth and Thomas (1976) noticed that plotting Near-Infrared (NIR) reflectance against Red reflectance for satellite images produced a scatter diagram that they called the "tasseled cap-agram" because of its resemblance to a tasseled cap. Figure 8 shows this scatter plot created from one of our images. The scatter plot contains a blue sky line (bottom right) as well as the soil line (middle) and the vegetation point (marked in top left). Pixels containing vegetation and blue sky are thereby well separated from everything else.

A basic multi spectral sensor was created by aligning the optical axis of a color and a NIR camera with a cold mirror (a beam-splitter which reflects visible light while transmitting NIR light). Vegetation pixels were then detected by thresholding based on their distance in the red–NIR scatter plot from the soil line. Richardson and Wiegand (1977) pursued this approach with their Perpendicular Vegetation Index (PVI).

This method produced robust, real-time vegetation detection during extensive testing in outdoor environments (Bradley et al. 2004). Although the system performed very well in natural environments, it was observed to fail on certain synthetic materials such as polyester clothing and vehicles painted with pigments that were reflective in NIR. Fortunately these synthetic materials generally present dense surfaces that are easily classified as non-vegetation by the geometry-based classifier described above.

7.6. Support Surface Detection

The fundamental need of planning to predict vehicle motion in the future generates a need to know the shape of the support surface—the surface which will support the vehicle weight—because the mapping of motion commands onto future states of motion depends on the terrain shape.

Of course, the problem of identifying the load bearing surface is not straightforward in environments with vegetative ground cover. In meadows and under forest canopy, we have found it necessary to estimate the supporting surface based on the content of the 3D density map. The ray tracing algorithm maintains a record of the lowest hit or laser pass-through in each vertical column of the map. The lowest pass-through indicates an upper bound on the elevation of occluded terrain (in the range shadow of a rise, for example). The lowest hit is nominally on the ground but specular laser returns (from standing water, for example) must be filtered to avoid corrupting this calculation and a limited evidence accumulation period is used in order to limit the impact of outliers generated by long range measurements.

7.7. Learning to Estimate the Support Surface

The algorithm described so far is based on the assumption that the vegetation is sparse enough to be penetrated by ladar. When vegetation is not sparse, a certain amount of inference is necessary. Based on the learning principle originally developed for agricultural robots (Wellington and Stentz 2003), a neural network was developed which learned how to adjust a preliminary estimate of the load bearing surface elevation based on observing the ground truth data generated when the vehicle drives over terrain in the learning phase. Figure 9 shows the operation of the neural net in a typical scene.

Despite all efforts applied to this problem, cases of dense vegetation remain where no reliable support surface can be detected. As a result, and for reasons of terrain self-occlusions, it was necessary for planning computations to be robust to missing terrain elevation data.

7.8. Discrete Obstacle Detection

As Figure 5 suggests, obstacle detection algorithms operate on terrain type information, density information, and support surface information. Operations both within a voxel and in a voxel neighborhood are used. The basic principle involved is to search for the signature of a particular class of obstacle in all the available data in a region.

When a supporting surface estimate is available, the number of hits above the surface in a column, weighted by measured and inferred (from terrain type) density, provides an indication of "positive" obstacles. Negative obstacles are inferred from the lack of data in a column—which is known because the support surface elevations are upper bounds rather than real measurements.

Regions where the vehicle would sustain excessive roll or pitch are also detected here by computing the attitude that the vehicle would assume at each position given the terrain elevations expected under its wheels.

Certain environments exhibit a high prior probability that a specific signature is meaningful. Fallen trees, for example, are very common in certain forests, and they present a characteristic semi-circular profile in single vertical ladar scans (Figure 10). When operating in such environments, a circular template was matched to points in each such scan. A few heuristics were used to prune false positives that require concavity, density, a large enough radius, and a center near the support surface. Those matches that pass these tests were collected in an overhead map, and a Hough transform was then used to find lines.

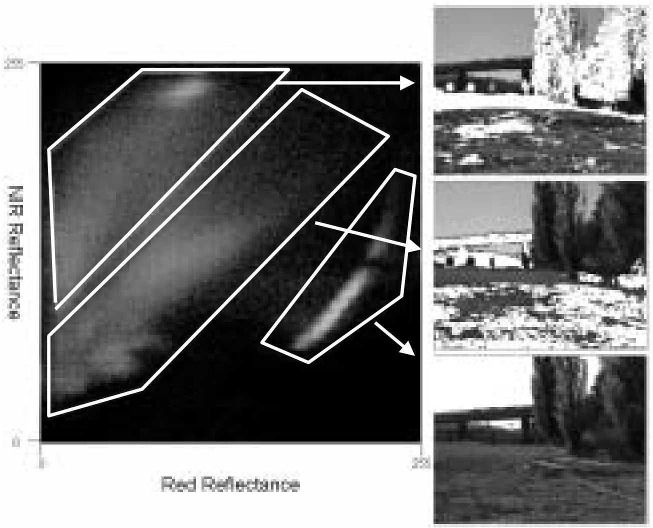

Fig. 8. Classifying Image Pixels by Red and Near-Infrared Reflectance. The different regions in the scatter-plot (left) of Near-Infrared (900–1000 nm) vs. red (600–670 nm) correspond to distinct regions of the image used to generate the scatter-plot. A discriminative feature of vegetation is that it reflects much more strongly in the NIR band than it does in the red band, and pixels in top left polygon correspond well to regions of vegetation in the image. Clear sky also has a distinct response which is much stronger in the red band than in the NIR band (bottom right polygon).

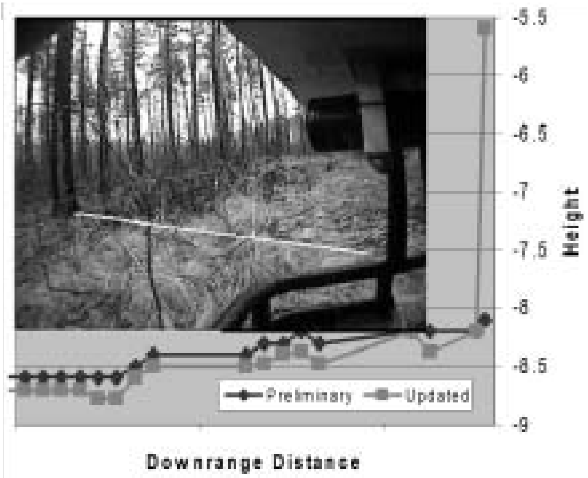

Fig. 9. Computing the Load Bearing Surface Under Ground Cover. The line in the inset image symbolically represents the slice through the world model density scores which is being processed. The neural net lowers the grass and raises the tree which is the correct adjustment of the load bearing surface.

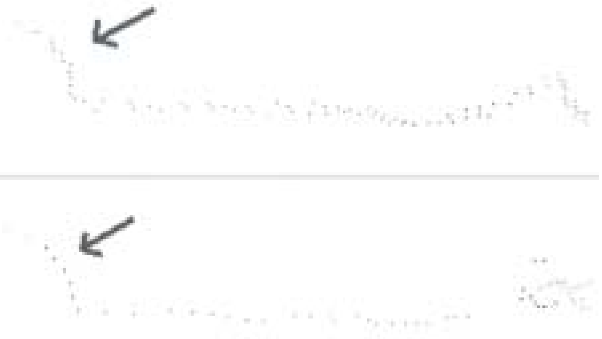

Fig. 10. Profiles of Logs in Ladar Scans. The image shows two consecutive scans of a log by a vertically scanning ladar. The arrows point to the log's profile in the scans. It appears as a smoothly curved surface connected to the ground plane. The cluster found at the right hand side of the scans corresponds to a small bush. The shape of this cluster differs significantly from scan to scan.

8. Mapping

In the present context, mapping refers to the process of recording, organizing, transforming, registering, resampling, and fusing multiple sources of information. The need to register data before it can be fused, the data association problem as it occurs in continuous mapping contexts, has proven to be the dominant design driver. Various disappointments in field experiments have driven us to redesign the mapping approach in order to respond to the challenges of generating high fidelity perception on a moving platform and provide a framework for the fusion of information from diverse sources. Our approach to mapping inherits ideas from our earlier work and adds a few new ones.

8.1. Accumulation—Distortion Trade-off

This principle drives many aspects of our approach to mapping. Nonideal pose estimates cause distortions in environmental models that are created with them. Pose error accumulation rate often increases with motion, motion is more difficult to measure on rough terrain, and sensitivity to these errors increases as the desired fidelity of perception increases.

There may at times be a fundamental requirement to accumulate data in a region; for example, in order to compute a region property the size of a wheel. Nevertheless, the value of excess data accumulation (beyond the fundamental requirements) is at odds with the cost of the cumulative effects of pose error. In other words, the benefits of longer term data fusion eventually become drawbacks due to the effects of blur. Ironically, despite good intentions, too much oversampling of the environment eventually incorporates enough distortion to make it impractical to reliably resolve the features of interest at the scale of interest. Several design principles have emerged to manage this trade-off. The impact of distortion can be minimized by:

- Minimum Accumulation: accumulating no more data than necessary. Hence, when obstacle signatures can be computed from one scan line of ladar data, they should be. When a few lines are necessary, then only a few lines should be used.

- Exploit Signal Properties: exploiting the best properties of pose estimate. It is better to compute region properties within ladar scan lines than across them. If two sets of consecutive lines must be accumulated, then the computation is organized to compute high fidelity local results in each set first, and then merge them in a manner consistent with the larger error accumulated between the sets.

- Engineer the Distortion Signal: providing the best possible pose signal(s). Ladar(s) can be oriented to extract preferred information from the faster of the two rotation axes. Multiple custom-designed pose estimates can easily be generated for multiple purposes.

These principles drive our approach to position estimation and mapping. A hierarchical arrangement of data structures provides decreasing periods of data accumulation and increasing levels of detail as the hierarchy is descended. An associated hierarchy of position estimates trades absolute accuracy for relative accuracy as that hierarchy is descended. Data is accumulated with one pose estimate but localized with the next highest estimate and, if necessary, re-localized to track the growing misregistration between the two.

8.2. Mapping Architecture

Three maps encoding relevant properties of the environment are notable; the perception (volumetric) cube, the local map, and the global map (Figure 11). As data flows from the cube to the local and then to the global map, it is abstracted and averaged over time and space. Two types of information flow. Terrain elevation data (including a flag indicating if it was measured or is an inferred upper bound) is used by the local planner to predict the motion of the vehicle through 3D space. This information is not passed to the global map. Cost information is the common semantic language of all information sources participating in global data fusion. The “cost” in a cell is intended to represent the risk level associated with driving the vehicle over it. A distinguished cost of “lethal” designates an obstacle to be avoided entirely.

Obstacle detection algorithms that required high resolution were performed on the point cloud data in the perception cube. These operations therefore used all the resolving power of the sensors before the results were quantized for the local map. Although resolution would normally have to be reduced downstream as well, we were able to maintain a 20 cm cell size at all three levels, while reducing dramatically the amount of information in each cell. Among other advantages, this allowed us to minimize the number of (expensive) resampling steps involved.

The global map was structured as an array of maps, where each individual map was known as a “channel”. Individual channels allow a perpetual distinction of cost data based on its source, and this allows fusion algorithms to adjust based on the potentially time varying quality of individual sources. Local map cost data comes from ground based perception and is likely to be the best assessment available unless the global pose estimate is drifting due to lost GPS. The UAV is better at detecting holes when operating at low altitudes. DTED (Digital Terrain Elevation) data available from various remote sensing sources is well localized but often of reduced resolution. Most of the time, our highest performance fusion algorithm was a conservative one—computing the maximum cost of all channels at a given location.

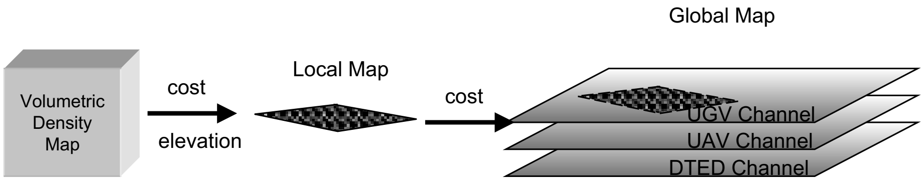

Fig. 11. Mapping Architecture. Three maps are maintained which span the spectrum of reduced extent/high detail to increased extent/low detail.

A second advantage of the multi-channel global map is the capacity to perform lazy registration as discussed below.

8.3. Lazy Registration

Figure 12 illustrates the operation of the two highest levels of mapping during a GPS jump. The local map is re-registered to the global map at high rates so that while the relative position (local-level shape) of perception-derived data remains stable, its global position is adjusted continuously.

8.4. Minimum Accumulation in Wrappable Maps

In contrast to the temporal error dynamics of pure inertial guidance, pose error accumulates in pure odometry (lost GPS) with total distance traveled. Accordingly, it becomes useful to record "age" tag information in map cells with the distance traveled since system startup. Such tags make it possible to perform two important functions to prevent misregistration:

- During writing (fusion): when the same terrain is re-imaged after an intervening gap in motion, data incoming to a cell is significantly younger than the data in the cell. The two sets of data are not likely to be correctly registered and the data in the cell is likely to be or have been regenerated in a nearby cell, so the data in the cell is cleared before the new data is added.

- During readout: when terrain data has a tag which is significantly younger than the present vehicle distance tag, the data in the cell is likely not in the correct position relative to the vehicle so the cell is considered to be empty, though is not explicitly cleared until the above case occurs.

These mechanisms are performed at the cellular level because the terrain self occlusion makes it unlikely that all cells in any region nominally in view will, in fact, be visited by a range reading. Over the lifetime of a single cell, these mechanisms create a typical sequence of states where data is first added, then interrogated for a while, then its data is hidden for some time until it is cleared and the cycle repeats.

Given the desire to remember the maximum amount of information subject to the above constraints, it is natural to allocate an amount of memory sufficient to hold the largest region of space that can be remembered. The recycling of cells that occurs creates an opportunity to avoid shifting of data in memory to reflect vehicle motion.

The perception cube and local map are respectively 3D and 2D arrays of attribute cells accessed with ring-buffer semantics. The mapping of spatial positions onto array indices is performed modulo the relevant array dimension, creating a wraparound. These maps are designated "wrappable" maps because space conceptually wraps around at their edges. The global map is not presently wrappable but it could be if necessary.

8.5. UAV Mapping

By contrast to UGV mapping, the UAV could assume a mostly unobstructed view of the sky and the GPS satellites. It relied on high performance position and attitude estimation in order to produce overhead ladar maps of outstanding quality as shown in Figure 13. Unlike nearly all aerial mapping practiced today, the UAV produces such maps on-line with no postprocessing. This data can even be broadcast live during flight.

9. Map Processing

An overhead projection like an elevation map provides a unique opportunity to recognize terrain features that may not be so easily found in a single image or ladar scan. While many things are possible, our efforts in this regard were restricted to a few specific issues that became critical at some point during the program.

9.1. Trail Extension

A single ladar scan provides a limited amount of information but sequences of scans or the map produced from them provides evidence of expansive and relevant scene structures. We processed the global terrain map to find the signatures of trails in otherwise natural terrain. The approach is reminiscent of many other lane detection approaches (Dickmanns and Zapp

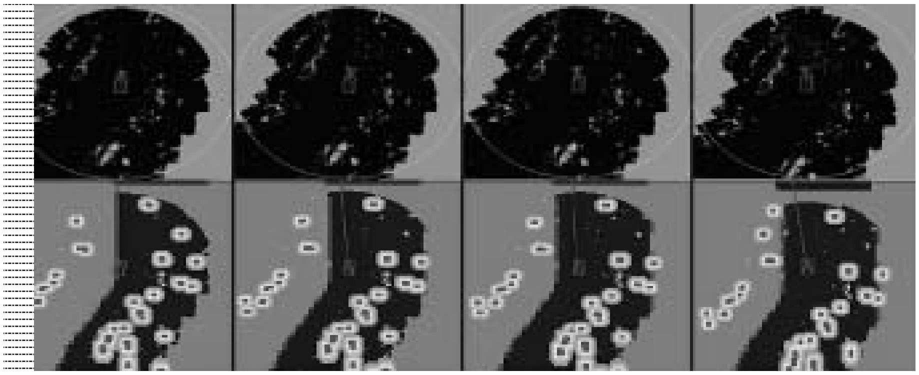

Fig. 12. Lazy Registration. Overhead renderings of the local map (top) and corresponding global map (bottom) are shown at four closely spaced periods of time. Over this time, the global estimate moves the vehicle to the right by one vehicle width due to GPS drift. In the global map, the dark roadway is geo-registered and does not move. The obstacles, however, were located by perception and placed in the local map so they shift right with the vehicle. In this way, obstacle avoidance becomes immune to GPS jumps.

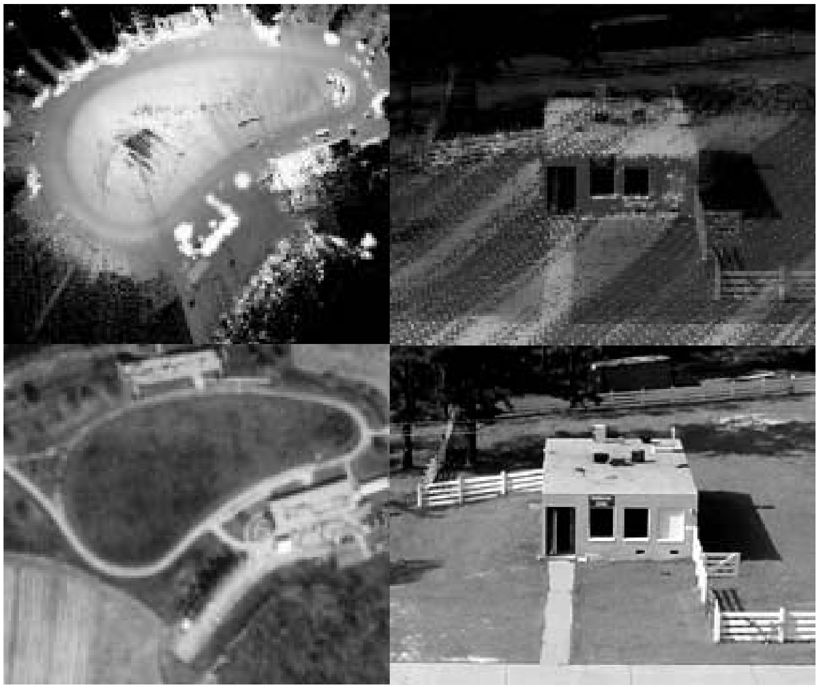

Fig. 13. UAV Mapping Performance At Global and Local Scales. (Top Left) Digital Elevation Map produced by the UAV. (Bottom Left) Orthophoto from another source. (Top Right) Fused ladar and color data of a building. (Bottom right) Photograph from same perspective.

1987; Pomerleau and Jochem 1996) in that a template encoding expectations about the scene is correlated to find the degree and position of a match.

Our approach was to coax the planner to follow trails by extending regions of low cost along their detected directions. A set of 1D kernels encoding a characteristic cost profile in the transverse direction was convolved with the map in front of the vehicle. If a sufficient number of sufficiently strong responses were detected that fit a line well, costs were lowered along this line. Though simple in design and unsuited to winding paths, the concept of trail extension has demonstrable merit on long, divergent paths, as illustrated in Figure 14.

9.2. Global Data Processing

Although the original intent was to extract benefit from the UAV only in detecting holes, ledges and other negative obstacles, our use of it eventually migrated toward having it generate global data on demand in order to supplement any DTED data that was available for the site. As it turned out, much of our DTED data was also generated from aircraft which flew at higher altitudes than the nominal 50 meter altitude of our UAV. UAV and DTED data were therefore processed similarly.

Available DTED data provided 10–20 range returns per square meter on average, and UAV data was roughly 3 times as dense. Data quality was such that simple local algorithms were quite effective. The data was first sorted into 1 meter 2D cells. This can be viewed as a degenerate form of the perception cube used for UGV perception—where columns are only one cell high and cells are much taller in the vertical direction. Once the data was sorted, data processing was divided into the stages outlined below.

9.2.1. Canopy Rejection

Air vehicles operate, of course, above occluding trees. The degree to which this canopy occludes ladar signals varies with location, time of year, species, etc. but it is often possible to filter out the canopy to leave the ladar hits which penetrated to the ground. By keeping only the lowest hits in each cell, it was possible to reject the canopy, when it occurred at all, very reliably.

9.2.2. Vegetation Rejection

At various times, planes were fit to the data both within and across cells depending on the context. Significant scatter in the residuals from the plane indicated ground vegetation which meant that the next step (hazard detection) would be unreliable, so the cell was simply classified as vegetation.

9.2.3. Hazard Detection

Thereafter, high deviation of the center from the plane indicated a negative obstacle (hole) while the slope of the plane itself was a reliable measure of local slope. The former signature would result in a lethal cost being passed to the planner whereas the latter would generate a continuous cost proportional to the slope.

10. Motion Planning

The motion planning system is also arranged hierarchically in order to address the need to be both smart and fast. It is composed of the D* global planner (Stentz 1995) and the Ranger local planner (Kelly and Stentz 1998). These elements were integrated for the first time on PerceptOR.

10.1. Motion Planning Architecture

The basic design principle used is a division of responsibility based on the extent of the planning horizon (Figure 15).

In the local region, relatively high fidelity models of vehicle shape and maneuverability are needed for reliable obstacle avoidance. Farther away, such details are unlikely to have a significant effect on the decisions which must be made now. Global planning can therefore use less detailed models of both the vehicle and the environment in order to look farther ahead in time and space.

The local planner performs a convolution of a rectangular vehicle model for a number of continuous paths computed online. These convolution paths are derived from a terrain map encoding elevation as a continuous variable. By contrast, the global planner performs heuristic search in an 8-connected grid where the convolution is approximated and pre-computed into a 2D configuration space.

The estimated cost of the local portion of each alternative is added to the estimated cost of the global portion, and a penalty is added for any misalignment of two portions at the point of joining. This form of planning decision fusion evolved over time in answer to the shortcomings of more straightforward alternatives.

10.2. Local Planning

The local planner was originally designed to support high vehicle speeds over natural, barren, terrain. To address the high degree of constraint imposed by high speed motions and Ackerman steering, the system uses forward modeling to predict response trajectories from a pre-stored set of steering and speed command signals which are intended to span the space of feasible motions more or less uniformly. As far forward in space as the stopping distance, the response trajectory can depend significantly on terrain shape, so 3D motion prediction is performed based on the terrain elevations produced from the perception cube.

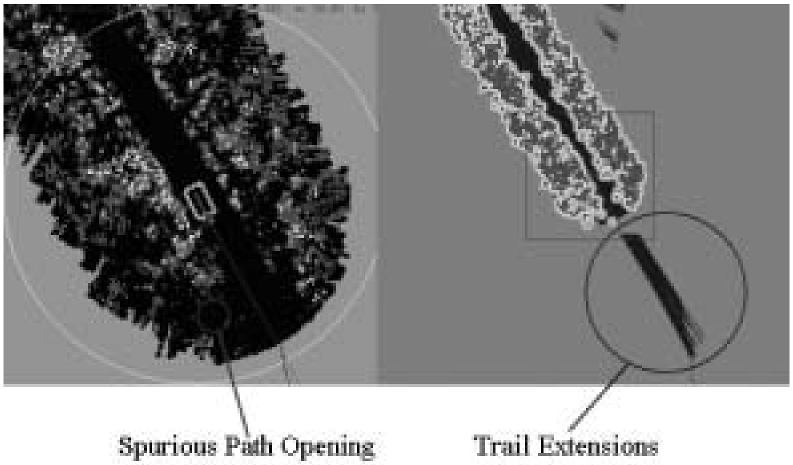

Fig. 14. Trail Extensions and Spurious Path Avoidance. The local map (left) shows an opening in the forest that would attract the planner in the absence of trail cost enhancement. The corresponding global map (right) demonstrates how low cost extensions to the trail direct the path planner along the putative direction of the trail. At the end of the trail extensions, the planned path veers to the vehicle's right directly toward the goal.

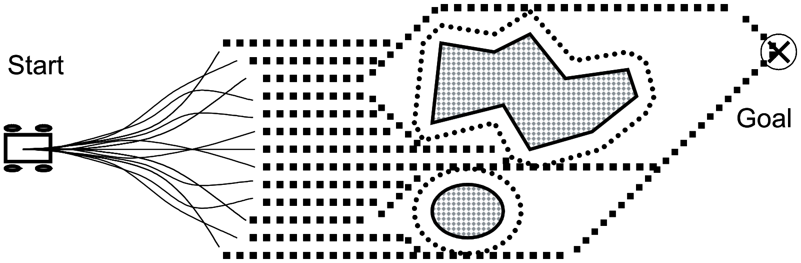

Fig. 15. Hierarchical Motion Planner. Candidate trajectories are produced by forward simulation of the vehicle dynamics out a distance beyond the stopping distance. Beyond that, the optimal remaining path in an 8-connected grid with expanded obstacles is used. An elevation grid is used in the high fidelity portion of the motion simulation. The total line integral over a cost field of both portions of each path is used to select the option to execute.

10.2.1. Dual Regimes of Steering Look Ahead

The responsibility to command a stop, when necessary to avoid a collision, originates at this level in the system. During motion look ahead, the two functions of motion prediction and hazard convolution are conceptually distinct. Once motion prediction has computed how the vehicle will move over the terrain, only then does it become possible to determine if any obstacles are intersected by the vehicle volume.

Local planning look ahead often extends well beyond the stopping distance in order to enable avoiding problems well before they occur. Doing so permits the use of high resolution perception data if it exists at such ranges.

The look ahead region is clearly divided into two regions and it is searched at roughly 10 Hz. The region inside and slightly beyond the stopping distance is designated the "reactive" zone. If all motion options at any time are deemed unsafe before the trajectory leaves this region, the vehicle is commanded to stop. The region beyond the reactive zone is designated the "deliberative" zone. Obstacles appearing in this region significantly raise the perceived cost of a trajectory in order to discourage its selection but the trajectory it is not eliminated outright.

10.2.2. Speed Generation and Negative Obstacles

The desire to continue moving also originates at this level of the system. Outside the context of a panic stop, the speed command is generated based on the overall situation as perceived by local planning. If all trajectories are deemed unsafe in the deliberative zone, the speed is reduced. Otherwise, it may be increased if it is lower than the prevailing speed goal.

Speed generation is particularly important in the context of negative obstacles. A system which can safely assume an absence of negative obstacles is free to operate at much higher speeds than one which cannot. Negative obstacles are particularly problematic for several related reasons. These obstacles can only be detected based on an absence of evidence to the contrary. The missing terrain in a hole does not generate any range data. Unfortunately, there are many other situations, such as terrain self-occlusion, where there is an absence of evidence of anything. The fundamental distinction to be made is whether a particular absence of data is due to a negative obstacle. The capacity to correctly perform this discrimination increases as range to the obstacle decreases. Unfortunately, the capacity to avoid the obstacle also decreases with proximity to it.

When the terrain is tilted toward the sensor, or when considering holes with two sides, higher sensor angular resolution can help. However, cases remain (Figure 16) where improved angular resolution will not help.

Facing these realities, it becomes clear that perception alone cannot solve the problem. Our ultimately successful approach involved several parallel lines of attack including the sensor mast mentioned earlier. From the perspective of the computational approaches, negative obstacle avoidance is a concrete example of the need to plan in spite of uncertainty.

Gaps in the terrain map are possibilities of negative obstacles. Short gaps can be safely ignored but longer ones are assessed at increasingly higher costs as they approach the reactive zone. This leads indirectly to a reduction in speed. If they enter the reactive zone, the associated path is stricken. If the gap is wide enough, it will also cause all trajectories to be stricken and a stop command to be issued. The emergent behavior of these measures is that the system will avoid ambiguous situations if there is an option to do so and it will react forcefully only when more graceful approaches have not provided sufficient information in a sufficiently timely manner.

10.3. Global Planning

The global planner is required to estimate the remaining path cost from the ends of the local path planner's candidate paths to the next goal. The planner represents the area of operations using a two-dimensional grid, called the navigation map. If aerial map data is available, the global planner loads the cost data from the global data processing system into the grid cells. If aerial data is not available, the navigation map is initialized to an intermediate cost value.

The global planner ran typically at a rate of 10 Hz, so it was able, in principle, to respond rapidly to the arrival of new knowledge. For example, perception might inform it that an assumed corridor was, in fact, closed off. A grid data structure of fixed resolution was used here because distances between waypoints were limited to roughly 1 km and it was permissible to stop and re-plan at each waypoint. Multi-resolution versions of the planner have been implemented in other contexts as well.

The costs used represent (roughly) mobility risk. The tuning of these costs/risks emerged as a major design issue during the execution of the work. Correct system behavior was only obtained by squarely addressing the question of what amount of extra excursion is justifiable to avoid a given level of risk. Nonetheless, we can report few principled methods beyond understanding that the tradeoff exists and tuning the costs in order to get correct behavior under various circumstances.

It is fundamental that every element of risk at a point must be associated with some distance through safe terrain that the system is willing to traverse in order to avoid the risk. One notable success, mentioned in Section 11.7 was a strategy of transforming human derived cost assessments exponentially to increase sensitivity to slight variations near the high end of the cost spectrum.

During route traversal, the global planner receives mobility cost updates from the vehicle's onboard sensors, and also from the UAV, if it is mapping concurrently. The global planner re-plans from the vehicle's current location to the goal, taking into account all new information written into the navigation

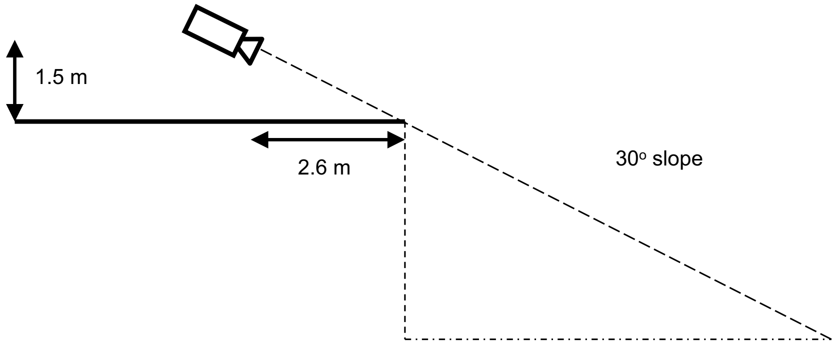

Fig. 16. Discriminating Negative Obstacles. Regardless of sensor angular resolution, a benign 30 degree slope cannot be distinguished from a lethal ledge until a sensor 1.5 meters high is 2.6 meters from the ledge.

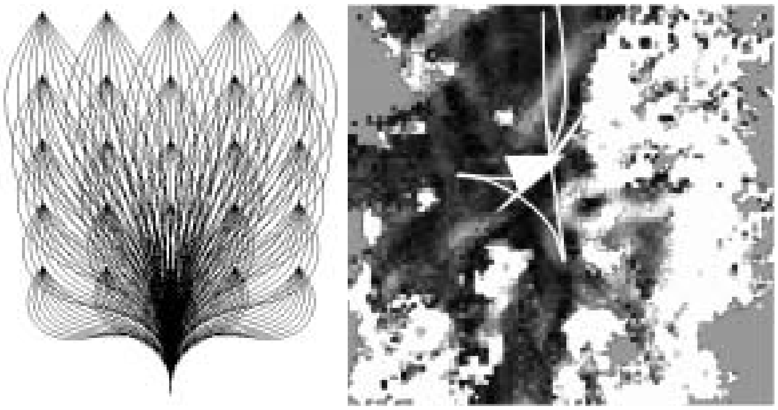

Fig. 17. Nonholonomic Motion Planner. Left: In this zero heading slice (restricted to forward motions for display) steering functions to a regular array of neighboring poses are encoded. Right: A custom 5 point turn is generated to turn after detecting a natural cul-de-sac.

map. Thus, the vehicle does not follow a single, static global path; instead, the global path changes every time new sensor information arrives. The advantage of this approach is that the vehicle is able to escape from deep cul-de-sacs that are larger than the current sensor field of view. The global planner keeps an accurate picture of terrain mobility across the length of the entire traverse. The planner can direct the vehicle to turn around and drive out of problem areas.

The global planner uses the D* algorithm to plan paths. D*, like A*, finds the optimal (lowest-cost) path between the vehicle's current location and the goal. However, D* is very efficient at re-planning when mobility costs in the navigation map change, especially those near the vehicle's current location. For typical maps, D* is able to re-plan hundreds of times faster than re-planning from scratch with A*. Therefore, we are able to re-plan in real-time. We used the D* algorithm early in the program, then we switched to Field D* (Ferguson and Stentz 2005) in the second half. Like most grid-based planners, D* produces "jerky" paths that consist of straight and diagonal line segments through the grid. In fact, the true optimal path cannot typically be represented in the grid, since all paths are required to pass through grid vertices. The resultant paths deviate by as much as 8% in cost and 22 degrees in heading from the true optimal path. Field D* does not require paths to pass through the grid vertices. It uses interpolation-based smoothing during the planning process to select paths that are much closer to the true optimal.

10.4. Behaviors

The above fusion of local and global path planners is used when the system is moving at reasonable speed over the terrain. In cases of any errors in the global map, where the global planner's simplifications matter, or when there is no prior map, it is inevitable that the system will find it necessary to stop suddenly. Once this occurs, a series of prioritized behaviors may take control in order to deal with this exceptional situation. The above planners are used by the Navigate behavior. When it releases control, a set of behaviors are used to verify the assumption of no safe forward path and to implement various recovery mechanisms.

First, the Lookaround behavior performs a slow speed ladar scan to regenerate the area in view under the ideal circumstance of the vehicle remaining stationary. This scan will generally remove any false positives in the obstacle map that were created due to motion blur or insufficient data. If so, the system switches back to Navigate and proceeds.

If this mechanism does not generate a way forward, the way is blocked and global planning will already have computed a new solution which guides the vehicle backward and ultimately around the forward impasse. The system next executes a very effective reactive backup and turn maneuver, called Goaround, which attempts to align the vehicle with the direction preferred by global planning. The maneuver is attempted multiple times with increasing backup distances. During most of the program, rearward perception was not present, so the system drove backward based only on terrain memory. If at any time during the backup the vehicle aligns with the direction preferred by the global planner, the backup maneuver is halted and normal forward operation is resumed.

If after a fixed number of backup attempts, the Goaround behavior is unable to align itself to the path or is unable to proceed for any other reason, the system transitions to the next behavior, a nonholonomic motion planner embodied in the Planaround behavior. If this behavior fails, the operator is asked for assistance.

The nonholonomic motion planner is based on a generalization of a grid which we call a pose lattice (Figure 17). The space of robot poses (x,y,heading) is discretized into a dense 3D grid and the trajectory generator described in Kelly and Nagy (2003) is used to create an implicitly repeated control set which connects any node to every reachable neighboring node out to some practical radius. The pose lattice structure can then be interpreted as a graph for purposes of implementing heuristic search (A* is implemented at the moment). The result is a regular sampling of state space which encodes, to finite resolution, all possible motions between neighboring states using only feasible motions. Complex motions respecting the constraints of local obstacles are then generated by optimal network search.

11. Project Results

Test results on the PerceptOR program are tallied at the system level in order to focus effort on overall performance. Subsystem performance was closely monitored by the development team, but was secondary from the perspective of the independent test team because of their focus on overall performance in realistic operational scenarios. This section uses this same strategy of emphasizing system level performance, though it also contains and explains the evolution of components.

11.1. Test Courses

Table 2 enumerates the 6 different tests conducted, specifying location, general terrain type, and date and season of the test.

Additional information on the test courses is provided by Krotkov et.al. (2006) in a detailed report on the tests themselves.

11.2. Typical Run

One of the tests of the fifth exercise is illustrated in detail in Figure 18, which contains an overhead view of a composite cost map over the full experiment, with insets containing close-up views of the cost map and/or view from an onboard video camera. In this test, the robot was positioned at the northern (topmost) portion of the course, and given five (5) waypoints to reach the southern (bottommost) position on the

Table 2. Test Site Locations, Terrain Types, and Date Visited

Location	Terrain Type	Date (Season)
Fort A.P. Hill, Virginia (near the Drop Zone)	Virginia woodlands	February, 2002 (Winter)
United States Army Yuma Proving Grounds, Arizona (near the Vinegaroon Wash)	Arizona desert	April/May, 2002 (Spring)
United States Marine Corps Mountain Warfare Training Center, California	California mountains	July/August, 2002 (Summer)
Fort Polk, Louisiana	Louisiana woodlands	October, 2002 (Autumn)
Fort A.P. Hill, Virginia (near Training Area 1B)	Virginia woodlands	December, 2003 (Winter)
United States Army Yuma Proving Grounds, Arizona (near the Vinegaroon Wash)	Arizona desert	February, 2004 (Winter)

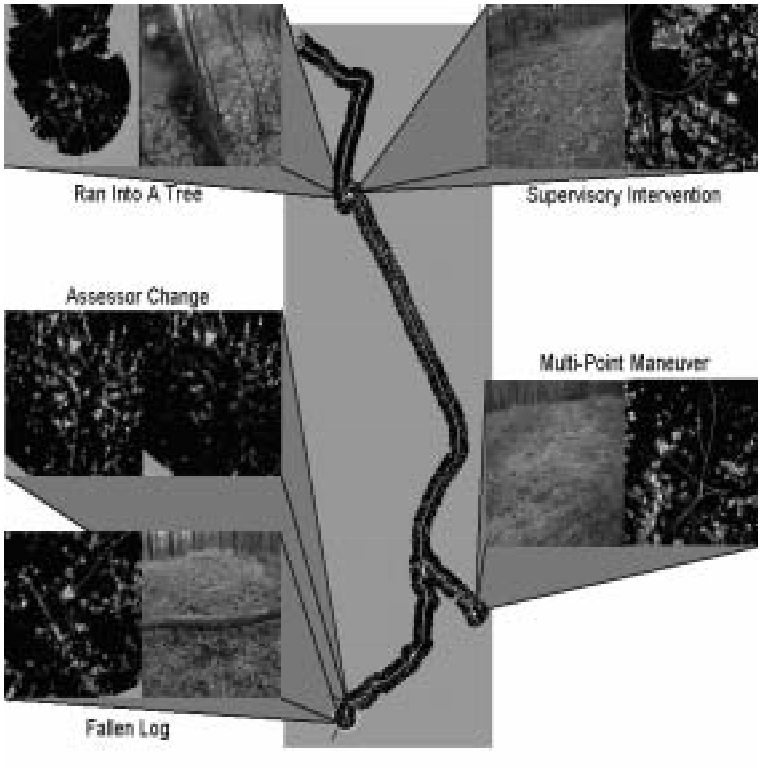

Fig. 18. Test Run with Forest, Meadow, and Trail. After driving 50 meters through a forest, the system collides harmlessly with a small tree. The remote operator takes control to clear the error and move the vehicle away from the hazard. The system acquires a trail and follows it autonomously for a long distance before determining that the trail is no longer heading toward the goal. It then executes a multi point turn maneuver, retraces its path, and exits the trail at an appropriate place. The operator reconfigures the software to tolerate tall grass (assessor change) near the end of the run and it terminates at a fallen log visible as the bright streak in the lower left cost image.

map (near the "Fallen log" inset). The straight-line path between waypoints was about 750 meters. The robot had no explicit bounds assigned (or even suggested) by the test team. The autonomy software constructed a map approximately of the scale shown as an autonomous response. The robot had no prior data; so much of the map was unobserved—represented as the textureless background. The trail of black, with a central white jagged line, is the path taken by the robot. (Black = safest, increasing white represents less safe.)

11.3. Experiment 1: Virginia Woodlands

The first experiment contained three test courses traversing Virginia woodlands. We borrow the course descriptions from Krotkov et al. (2006).

> The first course was a trail through the woods, encountering forks, standing and fallen trees in the path, ruts, and washouts. The second course was a woods course through sparsely wooded area encountering holes, logs, brambles, bushes, and closely spaced trees. The third course was a meadow course passing through a variety of grasses, an infantry trench, ponds, trees, and operation on side-slopes.

Though official tests were conducted, this experiment largely proved to be a calibration of expectations and a lesson for the next experiment. We were unable, in the 10-month window from project kickoff to testing, to retrofit the unproven base vehicle, design a system to meet capability requirements, and implement and pre-test to the reliability levels required for a solid week of unrehearsed field experiments. Still, the exercise proved enlightening to both the development and test teams to better understand what was expected and needed, in both directions.

At the component level, several technical issues were identified quickly. Most notably, we recognized a need for far greater reliability in the drive-by-wire vehicle—electrical power generation and distribution, mobility control, and similar "pure engineering" issues. What worked just fine for occasional demos and non-time-critical testing was simply insufficient for a solid week of on-schedule testing. Though it may appear obvious, it also points out the difficulty in moving from a demo research mentality to a more rigorous scientific evaluation mentality. We devoted significant resources to rigorous engineering issues: stronger error handling in software, component and integrated system test planning, and more formal execution of test plans.

Another notable component change, responding to the experience of this trip, was the move to maintaining two pose estimates, as explained in Section 5.1. Dealing with forests, which caused long periods of low quality or absent GPS coverage, while experiencing increased wheel slip, motivated the desire for the two different types of estimates.

It also became clear that our experimental mechanism for externally rotating a single-axis ladar scanner was an extremely powerful tool for building 3D models without vehicle motion. The basic design was also relatively low cost and it appeared highly reliable. By the next test, this "poor man's two-axis ladar" was the backbone of the autonomous perception system.

From this point forward we adopted a mentality to develop core capabilities and expand out from there, understanding that new features would arrive at each test with differing levels of technical readiness. We used three categories of readiness: high (meaning that it is integrated and well tested), medium (meaning that it usually has been integrated into the system but is not yet tested), and low (meaning that it is still in a component development stage). This classification strategy helped explain to the test team what was done and what was still in the pipeline.

11.4. Experiment 2: Arizona Desert

The second experiment contained four courses traversing Arizona desert at Yuma Proving Grounds. (Six courses were actually defined, but mobility limitations of our UGVs prevented safe testing on two courses focusing on extreme slopes and side slopes.) Krotkov et al. (2006) describe the courses as follows:

> The Wash and North courses featured open, rocky terrain characteristic of a desert wash. The Gully course climbed into and through a sinuous gully. The Ledge course traversed slopes with significant side slopes and drop-offs. Vegetation on the courses ranged from small, easily traversable plants to moderately-sized bushes, saguaro cactus, and dead logs.

Going into these tests, our system was far simpler than the description laid out in earlier sections of the article, but it was far more reliable than in the first test. This reliability allowed extensive testing. We completed nearly 10 000 meters of travel during the experiment, more than 90% of that distance under autonomous control.

Five findings stood out most prominently from this trip. First, relatively simple processing of a priori overflight ladar data proved powerful in supporting the planning of routes. Overflight ladar data density measured approximately 10 hits per square meter. Such density allowed fine-scale processing more like onboard processing than traditional prior data processing. In one experiment, the vehicle navigated blindly, using only the a priori cost map derived from this high resolution overflight data. The run was 100% autonomous, completing the 300-meter Wash course in approximately 8 minutes while covering approximately 360 meters. By comparison, running with onboard perception but without prior data at the time often took 30 minutes to complete with multiple operator interventions and occasional ESTOPs, while covering a greater distance (typically 400–500 meters). Though this one result was insufficient to draw conclusive judgments, it was sufficient to suggest that high resolution prior data was a powerful tool, even with straightforward processing.

The second major finding was that establishing operator context, or situational awareness, was extremely challenging, and that more live video was often insufficient to address this problem. This historically well-known issue began to receive more of our attention. By the next test, we had deployed our first version of an on-vehicle video server that could provide live or logged video to the operator, with logs accessible based on time or position of (or distance traveled by) the vehicle. This feature, described briefly in Section 3.1.2, gave the operator the ability to disable video transmission, but then recall the video later as desired. Given the vegetative clutter in most of the tests conducted, this ability was often useful for finding alternative means to “progress” in the test by moving back and taking a different branch of a path.

The third major finding was that the system often detected obstacles, but was unable to respond adequately before encountering the obstacles. While a continued effort to drive down latency was considered—and acted upon—the problem appears fundamental to high-level processing and control. Nature yielded inspiration: develop a “central nervous system” for the vehicle. By the next test, the first version of the reactive planning and control system (discussed in Section 6) had been implemented, able to rapidly stop forward mobility based on events such as detection of collisions on the front bumper.

The fourth major “finding”, in retrospect, might actually have been a mistake. Given the high quality of our prior data, and the relative immaturity of our onboard perception systems, we concluded that overwriting prior aerial data with live perception data was often a poor idea. This led us to develop methods of fusing data based solely on cost values and the source of the data. Cost fusion ultimately proved far more subtle and complex than we had hoped at this point, and in fact it remains on the list of important issues for further research.

The final finding was that neither the human nor the autonomous system was equipped to handle negative hazards, especially with natural ledges in which the terrain on either side is traversable. As shown in Figure 19, even out-of-body photos do little to ease the difficulty of detecting and estimating the danger posed by a drop-off from a ledge down into a wash. Onboard sensors were even less revealing to the operator, misleading him on several occasions to try to drive off lethal ledges.

The autonomous system fared only marginally better. It was configured to respond only to *positive* confirmation of trouble. A negative hazard, though, is negative in more ways than one: it not only drops down (negative change in height), but its signature is an absence of information (negative, or lack of, information). Figure 20, for example, shows a cloud of ladar points with a large gap from the near range to the far range. This gap is caused by occlusion. Closer inspection reveals that the occlusion is not from a positive obstacle, but rather from drivable terrain that falls off in elevation with increasing range from the vehicle. During Experiment 2, we were not using this signature information. By Experiment 6, when we returned to Yuma, we had directly exploited this property in a coupled perception-planning approach, as noted in Section 10.2.2.

Programmatic reevaluation led to a fundamental change in the Flying Eye (FE) system following this test. Up to this point, the FE had been based on a small Bergen helicopter frame, which proved to be expensive, risky, and severely limiting the science of exploring air-ground coupling. Weight minimization was paramount, forcing engineering compromises across the system: pose sensor quality, computing hardware, battery weight, and perception sensor options. Following this test, the decision was made to move to using a much more powerful, already automated aircraft based on the Yamaha R50, and support a transition to the eventual Yamaha RMAX (a newer model of a similar scale), as noted in Section 3.2.

11.5. Experiment 3: California Mountains

The third experiment contained four test courses traversing California mountain terrain at the United States Marine Corps Mountain Warfare Training Center (MWTC). Krotkov et al. (2006) describe the test courses as follows:

> The courses traveled through mountainous, alpine terrain in the high Sierras. The Alpine course (segmented into Alpha, Bravo, and Charlie) traveled approximately 500 m through pines large and small, standing and fallen, scrub vegetation, rocks large and small, and a grassy meadow, and crosses a stream on a “hasty” bridge on the Charlie segment. The Trail course began in a large exposed clearing and climbed on an unimproved road, crossed into the treeline, and departed from the road and traveled cross-country through alpine terrain.

Unlike the first two experiments, the test site was located far from the base camp, which proved highly complicating and greatly reduced available test time. At this test, the test team introduced the analysis center, as noted in Section 2.3.2. Until then, the test direction had insufficient real-time understanding of the overall test, leading to lower quality decision making. The analysis center represented a major step forward, not only for the test team but also for the development team, since few observers could watch from the field, and even they did not have the capability to monitor the internals of the autonomous system the way the analysis center eventually could.

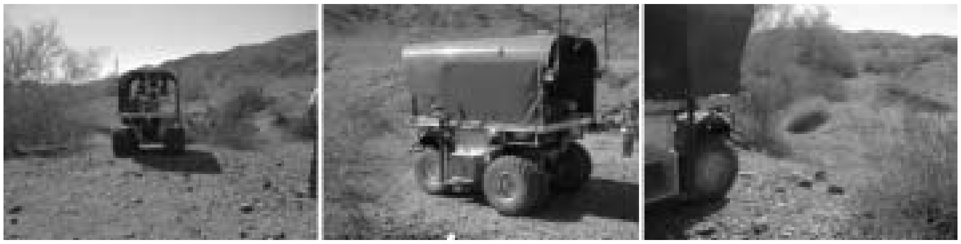

Fig. 19. Photos of the UGV at the top of a lethal ledge, attempting to enter the gully. Even with three photos, the ledge is not easily observed. The operators, with less information than is shown in this figure, chose to move forward, leading to an ESTOP during teleoperation.

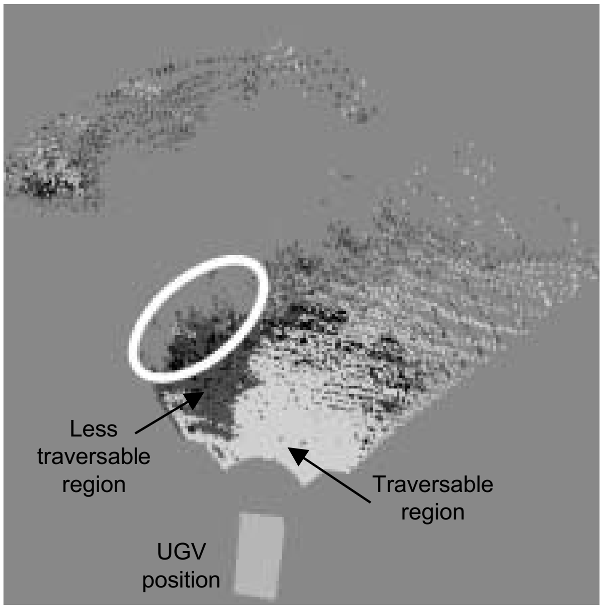

Fig. 20. Detection of ledge. The figure shows a nearly overhead 3D view of ladar returns color coded based on cost evaluation. The data shown here shows the first detection of a ledge (circled) inside the obstacle detector. The large gap beyond these detected points is the most "real" signature of a ledge from a distance, however. This signature was not fully exploited until our return trip to Yuma, in Experiment 6.

One major finding from this trip was that our global pose estimator was severely flawed. Desert conditions had masked the issue, which was triggered by low quality GPS or reacquisition after long GPS outage. This motivated continued work on the core filter itself, while also reinforcing a commonly held belief that visual odometry could be a helpful addition.

Another finding was that, with the addition of a blind go-to-point feature available from the operator control station (OCS), we nearly completely eliminated the need for detailed steering based teleoperation when stop-and-go behavior is acceptable. The operator could easily insert nearby waypoints toward which the UGV would drive, while ignoring perceived cost maps. Yuma had suggested that this was possible, but left some question. This test cemented the finding for us. In future tests, we rarely even connected the joystick for standard teleoperation controls.

A major surprise in the perception area was that dust was more problematic than at Yuma. In the desert, we had expected serious troubles, but had only light to moderate issues—they were detected, but not often enough to cause frequent replanning. At MWTC, however, the dust qualitatively appeared finer, and persisted in the air longer, causing more frequent troubles. Based on dust, vegetation, and overhangs, we finally concluded that we had gone about as far as possible with simple scan line processing of ladar data, and moved on to 3D volumetric analysis, as described in Section 7.3.

The recurring theme of sensor fusion grew into a related theme of map registration/alignment, which we referred to as consistency. Problems included converting between local and global frames, dealing with multiple resolutions, and software design limitations that prevented proper communication of certain types of hazards between the local and global planners. This led to ongoing work, as we began to understand the breadth and complexity of these related issues.

11.6. Experiment 4: Louisiana Woodlands

The fourth experiment contained four test courses traversing Louisiana woodlands at Fort Polk. Krotkov et al. (2006) describe the test courses as follows:

> The courses traversed classical southern woods with loblolly pine, long-leaf pine, slash pine, scrub oak, red oak, sparse meter-tall grasses, palmetto, and occasional yucca. Wetter areas featured 3-meter-tall grasses, bamboo-like scrawny vegetation, and thickets of brambles. The Whiskey course started on a sandy wash, crossed a creek, and climbed a rutted, grassy trail

> along the treeline. The Xray course started on a well-established dirt road, traveled through pine trees, and ended inside a concertina wire barrier. The Yankee course started in loblolly pines at the treeline, descended through a field with knee-high grasses and occasional small pines, crossed a well-established dirt road, and entered a fortification surrounded by concertina wire and a 3-meter-tall soil earthworks berm. The Zulu course traveled through sparse woods (tall, thin loblolly pines, spaced approximately 2–6 m apart, with no low branches and many needles on the ground) and dense woods (dense concentrations of pine trees with occasional pockets of meter tall grasses and thick areas of broad-leaf plants about 1 m high that could hide fallen logs, brambles, or holes). One of the courses traveled through dense, knee-high ferns. A person has an impression of "wading" through the ferns. Underneath the ferns were objects such as fallen logs in various stages of decay, and stumps 10–20 cm in diameter and 30 cm tall.

Tests here revealed common failures in ladar and stereo perception, with an inability to properly handle cross slopes and trenches. Both issues would be addressed by the next test. We also concluded that autonomous perception—and the operator—lacked sufficient visibility around the vehicle. We had added sensors to assist, but had not taken full advantage of them. It was believed that constructing panoramas would aid the operator, and that making those panoramas include 3D information would further aid the operator in understanding the context. Both actions were taken in preparation for the next experiment, leading to the ability for the operator to see views like those in Figure 6.

Experiments at a MOUT (Military Operations on Urban Terrain) site also confirmed that the autonomous system was now capable of complex emergent behaviors and reliable safe navigation. With the exception of wires, cables, and thin posts, a system designed to drive through dense grass was easily able to recognize and navigate the largely geometric world of urban terrain. (Note that these experiments focus on static terrain—moving objects were not present.) An example of such complex planning is shown in Figure 21, an overhead view of a planner cost map (brighter pixels are hazards) overlaid with the path of the robotic vehicle. The vehicle began at the bottom of the figure.

Another finding was that, with all the attempts made to tolerate dust, vegetation, and other compressible objects, the system now was rarely able to detect wire, including militarily important concertina wire. Ladar clearly detects concertina wire, but processing techniques usually considered it vegetation and ignored it. This issue remains a critical one for real-world off-road operations. (Note that for on-road operations, detection may be easier, since the confusion element of vegetation is removed.)

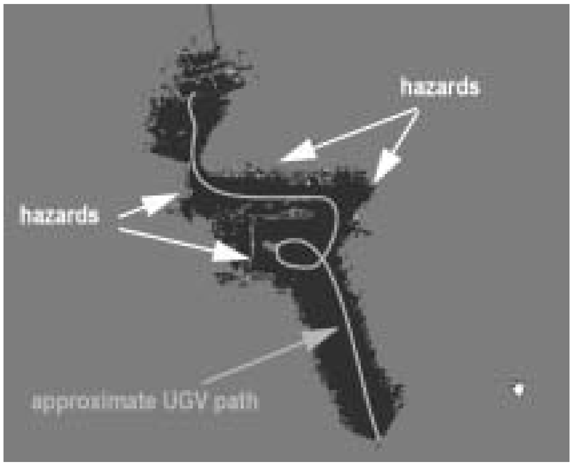

Fig. 21. Maps from a run at the Fort Polk MOUT site. Traveling from the bottom of the figure, the UGV approached a fence, explored along the fence (to the left in the figure), encountered another fence, looped around to the right of the original opening, made a sharp left turn to squeeze between the fence and a building, and eventually squeezed through an opening to reach the goal.

Two other features were common, given the torrential rains encountered prior to and during test execution: mud and water. Given the sensors available, neither substance produces an obvious signature. Detection of these substances using our sensor suite remains an open issue as well.

Finally, even with the success of the planning system, it was still relatively easy to construct situations in which the local and global planners would fail to find a useful solution forward. The situation usually involved needing to achieve a particular heading at a particular point in order to continue moving forward, but being constrained by objects around the point, as shown in Figure 22. The vehicle needed to swing wide to make the turn, but lacked the reasoning power to make this determination. This clear, simple example drove us toward more sophisticated planning capabilities as described in Figure 17.

11.7. Experiment 5: Virginia Woodlands

The fifth experiment contained three courses traversing Virginia woodlands at Ft. A. P. Hill—the same type of terrain as in Experiment 1, but in a different test area within the base. Krotkov et al. (2006) describe the test courses as follows:

> The courses traversed densely forested terrain. The OnOff course ran through a meadow with tall grass, down an unimproved dirt road through woods, across a second grassy meadow, and

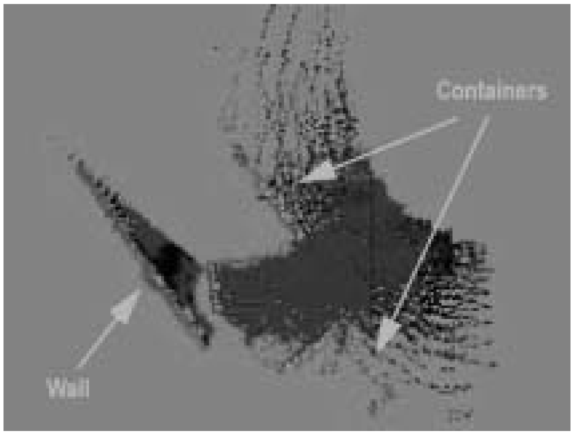

Fig. 22. Planning system failure in tight situations. The UGV detects two containers between which it successfully navigates. Then, the UGV detects a wall and decides to turn right. Due to mechanical turning limits, the vehicle can not complete the turn. In response, it backs up, but without understanding that it must "swing wide" in order to make this turn. The planner ended in a limit cycle that the operator eventually chose to break, and to manually complete the turn.

> along a rough track through the woods obstructed by many fallen trees. The Grass course followed a rough path under tall trees, encountering large logs, some in plain sight, and others hidden in tall grass and vegetation. The ZigZag course ran through fairly dense woods, and crossed an untraversable ditch resembling an infantry trench. Throughout, there were many fallen logs and patches of thick underbrush. Together, these courses constituted the most cluttered, densest, and most confined of all the PerceptOR tests. This terrain, encountered under snow, freezing rain, and rain conditions, posed significant challenges to ground mobility.

Numerous incremental improvements going into this phase yielded improved performance, despite the most cluttered terrain of all experiments conducted. Stereo perception proved powerful, equaling ladar performance in the most complex terrain. Interestingly, stereo tended to miss certain fine-grained features entirely, which allowed it to move on despite the complexities. Ladar could see tall tufts of grass, for example, and had inconsistent results attempting to evaluate the tradeoffs of going through it vs. around it. In slightly more open terrain, ladar-based operations proved faster and more reliable overall, but stereo still proved useful.

Another major success was the performance of pose estimation. With a variety of canopy conditions, GPS coverage was wildly inconsistent. In perhaps the longest stretch of zero-GPS operations, the UGV drove approximately 400 meters on the OnOff course with an accumulated error of only 2% of distance traveled in these non-ideal circumstances.

Mapping issues were finally being addressed. We reached our final strategy of brute-force resampling of the entire local map into the global map—a strategy that avoided pitfalls of numerous clever "tricks" that had been attempted previously. Coming out of this trip, we made our final discovery about the importance of cost ratios, rather than absolute cost values. Intuition had misled us: comparing costs of 10 to 20 and 100 to 110 seemed intuitively similar: both differ by a delta of 10 cost levels. Yet, from the planner perspective, the comparison was 10/10 to 20/10, and 100/100 to 110/100—1:2 and 1:1.1—which obviously leads to radically different planner behavior. In practice, the system had historically been surprisingly sensitive to subtle variations in smaller cost values, while being almost blind to much larger changes (deltas) in higher cost values. With this new understanding, we implemented a conversion function that forced the planner internally to represent a constant change in the costs it was given as cost in its planning space. This final configuration proved far more reliable and stable.

During this experiment, we also conclusively demonstrated that a tightly coupled FE can in fact give advance warning to its UGV companion in avoiding hazards. In one experiment, a coffin-sized hole was dug in a field of waist-high grass, creating a lethal hazard that the UGV could not detect until it was nearly into the hole. The FE flew ahead, detected the hazard, and published the information to the ground vehicle. The UGV received the data, integrated it into its maps, planned around it, and successfully avoided contact with this hole although it was never observed directly by the UGV and was not known to either vehicle before the test started.

While these UGV-FE operations were successful, it was also clear that tightly coupling the FE hampered the quality of FE data collection. This effect was compounded by the push-broom ladar scanning strategy employed, because any change in FE heading required a difficult compromise between speed of execution and density of scan coverage on the ground. The FE sensing system was not designed to collect dense data during tight turns. The decision was made here to focus more on loosely coupled UGV-FE operations for the next test, in which the FE would be flown in several different mapping exercises, including wide-area mapping as well as in the mapping of specific areas of interest identified in the prior data. This approach appears more viable from a scalability standpoint, allowing collections of UGVs and FEs to work together in a task-driven approach.

11.8. Experiment 6: Arizona Desert

The sixth experiment focused on three new courses in the same general area as Experiment 2, in the Arizona desert at Yuma Proving Grounds. In addition, experiments were conducted on several of the courses used in Experiment 2, including the

Wash and Ledge courses. Krotkov et al. (2006) describe the three new courses as follows.

> The three courses were longer than in previous tests. The Fish Course, with a straight-line distance of 700 m between waypoints, traveled through a wash featuring characteristic vegetation (easily traversable scrub, moderate-size bushes, fallen logs, non-traversable trees, and occasional cactus) and rocks ranging in size from easily traversable pebbles to non-traversable boulders. The Dorsey Course, with a 600 m straight-line distance between waypoints, tested the ability to drive "cross corridor" across approximately a half-dozen washes. The Bernie Course, with a 1200 m straight-line distance between waypoints, traversed a gravel road and off-road paths.

During Experiment 2, the Wash course was a reasonably challenging test. This course was approximately 300 meters in length, and in Experiment 2 our autonomous system often took 30 minutes and drove 400–500 meters to complete the course, with several operator interventions and several ESTOPS. In this second visit, the final system completed the test fully autonomously without ESTOPs in several of the repetitions, both for stereo-based operations and ladar operations. The median distance traveled over 8 experiments was down to 335 meters, in median time of approximately 6 minutes.

One key finding in this test was conclusive evidence that sensor height greatly aids in the perception of hazards. The UGV was updated with a sensor mast that elevated an additional ladar approximately 1 meter higher than the other sensors. With the mast deployed, the system was able to see further, to see over most nearby objects, and it therefore had more information with which to plan.

Another key finding was that the exponential adjustments to the cost spectrum finally made the system respond more rationally to variations in the cost map. High cost areas were now almost always avoided, when previously they would frequently give only mild respond to high cost but nonlethal objects.

Another conclusive finding was that the coupled handling of potential negative hazards (by planning and perception) worked extremely well. The perception system recognized gaps in data as potential areas of concern, but did not have to designate them as necessarily lethal to the vehicle. The planner would add cost to any path that would go through such areas, thereby discouraging but not preventing such action. If it chose to proceed, it would also slow the vehicle down, to the point that it would stop just short of entering such a gap. By that point, the perception system was able to gather enough information from close proximity to determine if the gap was a real hazard or just a benign shadow. This coupled approach amounts to an effective active perception strategy. It was the only effective mechanism we found in addition to raising the sensors.

The separation of reactive local planning from deliberative local planning was also critical. By introducing two explicit ranges (Section 10.2.1), the planner could consider both stopping maneuvers with short motions and turning maneuvers with long motions. The myopic behavior of earlier versions was eliminated by extracting all the useful information we could from distant perception information.

11.9. Summary of Autonomy Results

Approximate cumulative data for all six exercises are summarized below in Table 3. A rough sense of improvement over time can be obtained by comparing the cumulative scores for all 6 exercises with those of the final one. The table includes metrics regularly tracked by the independent test team:

- ***Number of Runs***: The number of runs for the official record.
- ***Distance***: The total distance traveled during the official runs, as reported by the robot navigation system.
- ***Duration***: The total time during the official runs.
- ***Speed***: The average speed over all distance traveled.
- ***Uplink Distance Density***: The total uplink volume (total number of bits sent from the OCS to the vehicle) normalized by the total distance traveled.
- ***Downlink Distance Density***: The total downlink volume (total number of bits sent from the vehicle to the OCS) normalized by the total distance traveled.

These metrics were chosen to demonstrate the magnitude of the testing conducted and because of their relevance to tradeoffs and requirements in realistic operational scenarios.

12. Conclusions and Outlook

This section attempts to formulate some of the main lessons learned and some key issues for further investigation.

12.1. Lessons Learned

Our experiments have confirmed that visibility is critical to both human and autonomous control. Logged video playback capabilities in the operator interface, sensors placed high on masts for negative obstacle detection, and the occasional rearward looking ladar were all very effective strategies.

The basic premise of PerceptOR—that perception is the most limiting aspect of contemporary systems—seems to have been borne out in our experiments. Perception for low-speed

Table 3. Cumulative and Final Data for PerceptOR Program Exercises (CMU Team Only)

Item	Unit	Cumulative	Last Test
Number of Runs		183	36
Distance Traveled	m	81,094	29,376
Test Duration	s	230,606	49,172
Average Speed	m/s	0.35	0.6
Uplink Distance Density	Kb/m	34.5	4.8
Downlink Distance Density	Kb/m	705	327

operations in essentially geometric rigid worlds—even in unmapped terrain and outdoors with dust, rain, etc—seems to be essentially a solved problem from the scientific perspective. Our experiments at Yuma on two occasions were highly successful for this reason. Adding the possibility of vegetation, water, mud, loose gravel, etc. immediately and substantially reduces performance. False alarms and missed detections substantially increase. While our system exhibited limited competence on slopes it is not clear whether this is fundamental or an aspect of our limited attention to the matter.

Our perception systems were also increasingly myopic as speeds increased. Our experiences substantiate the prevailing opinion that achieving higher speeds in such complex terrain is a difficult challenge.

Among the more daunting sensor fusion challenges that we faced was not the need to register data, or to compute optimal estimates based on noise models, but the need to simply render perception results in consistent cost units. A significant amount of system tuning related to the problem of trading off additional excursion for the risk encoded in the costs passed to planning. A second issue was achieving consistency in the costs reported by aerial data and ground data derived processing. Ongoing efforts are addressing the second issue with machine learning.

Prior data was observed to have a very positive effect only in situations where the environment is complicated enough. It takes a very long time to solve a maze by exploration, and natural environments can be mazelike. On the other hand, the use of maps expressed in ground fixed coordinates requires a position estimate of sufficient accuracy to extract value from them. Barring an ability to detect forest trails in perception, for example, a two meter pose error would lead the system consistently into the trees beside a trail because the map and the position estimate disagreed with the data in plain view from a human's perspective.

Thus, position estimation is an Achilles heel of autonomous systems which must share data with any other agents. GPS performs poorly under canopy and the units we used could not be trusted to determine the quality of their own estimates.

In terms of self-awareness, our planners lacked higher level contextual information that caused them to be ignorant of which decisions are important and unimportant. Sometimes there was but one narrow channel connecting A to B and the system consistently failed to make the turn into the channel. A more self aware system would slow down and make sure this critical matter was executed correctly. Conversely, when the costs of two different alternatives are almost identical, the system can react to the noise in the perception system by continuing to change its mind about how to proceed, even to the point of exhausting all available margin to avoid an obstacle.

The advantages of real time replanning therefore come at the cost of a system which can change the global plan too radically and too often. Planning may not be complete in the classical sense in the face of continuously changing information.

A matter which seems similar to uncertainty awareness is risk awareness. It would perhaps be useful to incorporate sufficient understanding for the system to know that, although a path between the trees will allow it to "cut the corner" to the next waypoint, following the trail is probably more prudent so long as it goes in roughly the correct direction.

The disagreement between local and global planners on the mobility characteristics of the vehicle continues to be a largely unresolved issue. At the point where the planning horizons join, the discontinuous changes in heading that may occur lead to the predictable effects of planning infeasible motions. An exception occurs which may or may not be resolvable by autonomy.

12.2. Future Work

The final fielded system incorporated a special subsystem of parameter files which contained the major "knobs" used to influence system behavior. We made little headway in determining a canonical and minimal set of knobs upon which all others should depend. For example, various system tuning parameters might have been changed at times in order to ultimately convey to the system that the prior probability of fallen trees is presently particularly high or that tall grass in the present area is or is not likely to hide anything underneath it. The human operator became very skilled at changing parameters on the fly when the system became confused.

It seems likely that an appropriate set of knobs relates at least in part to the local environmental context. One of the more relevant aspects of common sense knowledge that could benefit an outdoor mobile robot is the awareness that it is presently in a field, a forest, a dry river bed, a road, or a forest trail. Such contextual information could be used to condition or alter its prior assumptions about the prevalence and magnitude of certain hazards.

Among the more promising avenues of investigation is unsupervised learning based on proprioceptive information, gathered when the vehicle drives over terrain that was evaluated by the planner. The compressibility of vegetation can be learned, for example, from experience.

12.3. Outlook

The PerceptOR program has been unusually rigorous in its approach to evaluative testing of unmanned ground vehicles. The data presented here and in companion publications constitutes an unusually thorough, structured, and detailed evaluation of UGV technology. While the overall objective was simply to quantify the performance level of contemporary systems, our performance has clearly improved over time in response to test results, and many opportunities for more performance improvements obviously remain. Indeed, any field accelerates with the advent of objective methods to define and measure progress.

Some of the grand challenges of outdoor autonomy are well understood. A few are detecting small obstacles at high speeds, detecting negative ones at any speed, passive operations at night, and perceiving the groundplane and hazards hidden beneath obscuring vegetation. In addition to some progress on some of these fronts, the main contribution of the PerceptOR program has been the most rigorous attempt to date to quantify the present state of the art.

References

Albus, J. 1992. A Reference Model Architecture for Intelligent Systens Design. In: *An Introduction to Intelligent and Autonomous Control*, pp. 57–64, Kluwer.

Amidi, O., Kanade, T., and Miller, J. R. 1998. Autonomous Helicopter Research at Carnegie Mellon Robotics Institute. Proceedings of Heli Japan ‘98, April.

Andrade, G., Amara, F., Bidaud P., and Chatila, R. 1998. Modelling of Robot-Soil Interaction for Planetary Rover Motion Control. *Proceedings of IEEE/RSJ ICIRS*, Victoria, Canada.

Aviles, W. A., Hughes, T. W., Everett, H. R., Umeda, A. Y., Martin, S. W., Koyamatsu, A. H., Solorzano, M. R., Laird, R. T., and McArthur, S. P. 1990. Issues in Mobile Robotics: The Unmanned Ground Vehicle Program Teleoperated Vehicle (TOV). *Proceedings of Mobile Robots V*; SPIE, Boston, MA.

Balarski, S. and Lacaze, A. 2000. World Modelling and Behavior Generation for Autonomous Robot Ground Vehicle. *Proceedings of IEEE ICRA*, San Francisco, USA.

Barshan, B. and Durrant-Whyte, H. F. 1995. Inertial navigation Systems for Mobile Robots. *IEEE Trans. on Robotics and Automation* 11(3).

Baten, S., Mandelbaum, R., Luetzeler, M., Burt, P., and Dickmanns, E. 1998. Techniques for autonomous, off-road navigation. *IEEE Intelligent Systems Magazine*, 57–65.

Belluta, P., Manduchi, R., Matthies, L., Owens, K., and Rankin, A. 2000. Terrain Perception for Demo III. *Proceedings of IEEE Intelligent Vehicles Symposium*, pp. 326–331.

Bergh, C., Kennedy, B., Matthies, L., and Johnson, A. 2000. A compact and low power two-axis scanning laser rangefinder for mobile robots. Seventh Mechatronics Forum International Conference, Atlanta, Georgia.

Biesiadecki, J., Leger, C., and Maimone, M. 2005. Tradeoffs Between Directed and Autonomous on the Mars Exploration Rovers. In *Proceedings of International Symposium of Robotics Research*, San Francisco.

Bouabdallah, S., Murrieri, P., and Siegwart, R. 2005. Towards Autonomous Indoor Micro VTOL. *Auton. Robots* 18(2):171–183.

Bradley, D., Thayer, S. M., Stentz, A., and Rander, P. 2004. Vegetation detection for mobile robot navigation. CMU Technical Report 04–12.

Broggi, A., Bertozzi, M., Fascioli, A., Guarino LoBianco, C., and Piazzi, A. 2000. Visual perception of obstacles and vehicles for platooning. *IEEE Trans. Intell. Transport. Sys.* 1(3).

Chang, T. S., Qui, K., and Nitao, J. J. 1986. An Obstacle Avoidance Algorithm for an Autonomous Land Vehicle. *Proceedings of the SPIE Conference on Mobile Robots*, pp. 117–123.

Chatila, R. and Lacroix, S. 1995. Adaptive Navigation for Autonomous Mobile Robots. 7th International Symposium on Robotics Research, Munich, Germany.

Cherif, M. 1999. Motion Planning for All-Terrain Vehicles: A Physical Modeling Approach for Coping with Dynamic and Contact Interaction Constraints. *IEEE Transactions on Robotics and Automation* 15(2):202–218.

Clark, R. N., Swayze, G. A., Livo, K. E., Kokaly, R. F., Sutley, S. J., Dalton, J. B., McDougal, R. R., and Gent, C. A. 2003. Imaging spectroscopy: Earth and planetary remote sensing with the USGS Tetracorder and expert systems. *J. Geophys. Res.* 108(E12), 5131, doi:10.1029/2002JE001847, pp. 5–1 to 5–44, http://speclab.cr.usgs.gov/PAPERS/tetracorder.

Daily, M., Harris, J., Kiersey, D., Olin, D., Payton, D., Reiser, K., Rosenblatt, J., Tseng, D., and Wong. V. 1988. Autonomous Cross Country Navigation with the ALV. *Proceedings of the IEEE International Conference on Robotics and Automation*, Philadelphia, Pa, pp. 718–726.

Diaz-Calderon, A. and Kelly, A. On-line Stability Margin and Attitude Estimation for Dynamic Articulating Mobile Robots. *International Journal of Robotics Research* 24(10):845–866.

Dickmanns, E. D. and Zapp. A. 1987. Automonous high speed road vehicle guidance by computer vision. In: Automatic Control—World Congress, 1987: Selected Papers from the *10th Triennial World Congress of the International Federation of Automatic Control*, (ed R. Isermann) pp. 221–226, Munich, Germany, Pergamon.

Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., and Csorba, M. 2001. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on Robotics and Automation* 17(3):229–241.

Duff, E. and Roberts, J. 2003. Wall-Following with Constrained Active Contours. *Proceedings of the 4th International Conference on Field and Service Robotics*.

Dunlay, R. T. and Morgenthaler, D. G. 1986a. Obstacle Detection and Avoidance from Range Data. *Proceedings of SPIE Conference on Mobile Robots*, Cambridge, MA.

Dunlay, R. T. and Morgenthaler, D. G. 1986b. Obstacle Avoidance on Roadways Using Range Data. *Proceedings of SPIE Conference on Mobile Robots*, Cambridge, MA.

Faugeras O. et al. 1993. Real-time correlation-based stereo: algorithm, implementation and applications. INRIA Technical Report n. 2013.

Feng, D., Singh, S., and Krogh, B. 1990. Implementation of Dynamic Obstacle Avoidance on the CMU Navlab. *Proceedings of IEEE Conference on Systems Engineering*.

Ferguson, D. and Stentz, A. 2005. Field D*: An Interpolation-based Path Planner and Replanner. Carnegie Mellon Robotics Institute Technical Report CMU-RI-TR-05-19.

Gage, D. and Pletta, B. 1987. Ground vehicle convoying. *SPIE Mobile Robots II* 852:319–328.

Gat, E. 1998. Three-layer architectures. In: Artificial intelligence and mobile robots: case studies of successful robot systems, pp. 195–210 , MIT Press.

Giralt, G. and Boissier, L. 1992. The French Planetary Rover VAP: Concept and Current Developments. *Proceedings of IEEE Int'l Workshop on Intelligent Robots and Systems*, IEEE, Piscataway, N.J., pp. 1391–1398.

Goldberg, S., Maimone, M., and Matthies, L. 2002. Stereo vision and rover navigation software for planetary exploration. *Proceedings of the IEEE Aerospace Conference*, pp. 2025–2036.

Golden, J. P. 1980. Terrain Contour Matching (TERCOM): A Cruise Missile Guidance Aid. *Proceedings of International Soc. for Optical Eng. (SPIE) Image Processing for Missile Guidance* 238:10–18.

Gowdy, J., Stentz, A., and Hebert, M. 1990. Hierarchical Terrain Representation for Off-Road Navigation. *Proceedings of SPIE Mobile Robots*.

Guivant, J. and Nebot, E. 2003. Implementation of simultaneous navigation and mapping in large outdoor environments. Robotics Research: The Tenth International Symposium, Springer.

Hagras, H., Colley, M., Callaghan, V., and Carr-West, M. 2002. Online Learning and Adaptation of Autonomous Mobile Robots for Sustainable Agriculture. *Auton. Robots* 13(1): 37–52.

Hebert, M., Kanade, T., and Kweon, I. 1988. 3-D Vision Techniques for Autonomous Vehicles. The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Technical Report CMU-RI-TR-88-12.

Hotz, B., Zhang, Z., and Fua, P. 1993. Incremental Construction of Local DEM for an Autonomous Planetary Rover. Workshop on Computer Vision for Space Applications, Antibes.

Huang, T. S. 1994. Motion and Structure from Feature Correspondences: A Review. *Proceedings IEEE* 82:252–268.

Iagnemma, K., Kang, S., Shibly, H., and Dubowsky, S. 2004. On-line Terrain Parameter Estimation for Planetary Rovers. *IEEE Transactions on Robotics* 20(2):921-927.

Iwata, T. and Nakatani, I. 1992. Overviews of the Japanese Activities on Planetary Rovers. *Proceedings of 43rd Congress of the International Astronautical Federation*.

Kanade, T. et al. 1996. A stereo video rate dense depth mapping applications. *Proceedings of CVPR96*.

Kauth, R. J. and Thomas, G. S. 1976. The Tasseled Cap—A Graphic Description of the Spectral-Temporal Development of Agricultural Crops as Seen by Landsat. LARS: Proceedings of the Symposium on Machine Processing of Remotely Sensed Data, West Lafayette, IN: Purdue University, pp. 4B-41-4B-51.

Kehternavaz, N., Griswold, N., and Lee, S. 1991. Visual control of an autonomous vehicle (BART)—the vehicle following problem. *Transactions on Vehicular Technology* 40(3).

Keirsey, D., Payton, D., and Rosenblatt, J. 1988. Autonomous Navigation in Cross-Country Terrain. *Proceedings of Image Understanding Workshop*.

Kelly, A. and Nagy, B. 2003. Reactive Nonholonomic Trajectory Generation via Parametric Optimal Control. *International Journal of Robotics Research* 22(7–8):583–601.

Kelly, A. and Stentz, A. 1998. Rough Terrain Autonomous Mobility—Part 2: An Active Vision, Predictive Control Approach. *Autonomous Robots* 5:163–198.

Konolige, K. 1997. Small Vision Systems: Hardware and Implementation. Eighth International Symposium on Robotics Research, Hayama, Japan.

Krotkov, E., Fish, S., Jackel, L., McBride, W., Perschbacher, M., and Pippine, J. 2006. The DAPRA PerceptOR Evaluation Experiments. To appear, Autonomous Robots.

Lacaze, A., Murphy, K., and DelGiorno, M. 2002. Autonomous mobility for the DEMO III Experimental Unmanned Vehicle. Association for Unmanned Vehicle Systems—Unmanned Vehicle.

Lacaze, A., et al. 1998. Path planning for autonomous vehicles driving over rough terrain. Proceedings of the IEE

ISIC/CIRA/ISAS Joint Conf., pp. 50–55.

Langer, D., Rosenblatt, J. K., and Hebert, M. 1994. A Behavior-Based System for Off-Road Navigation. *IEEE Trans. Robotics and Automation* 10(6):776–782.

Lescoe, P., Lavery, D., and Bedard, R. 1991. Navigation of Military and Space Unmanned Ground Vehicles in Unstructured Terrains. Proceedings of the Third Conference on Military Robotic Applications.

Macedo, J., Manduchi, R., and Matthies, L. 2000. Ladar-based discrimination of grass from obstacles for autonomous navigation. *Proceedings of International Symposium on Experimental Robotics*, pp. 111–120.

Manduchi, R., Castano, A., Talukder, A., and Matthies, L. 2005. Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous Robots* 18:81–102.

Matthies, L. H. 1992. Stereo Vision for Planetary Rovers: Stochastic Modeling to Near Real-Time Implementation. *International J. Computer Vision* 8(1):71–91.

Matthies, L. and Grandjean, P. 1994. Stochastic Performance Modeling and Evaluation of Obstacle Detectability with Imaging Range Sensors. *IEEE Transactions on Robotics and Automation*, Special Issue on Perception-based Real World Navigation 10(6).

McGovern, D. E. 1990. Experiences and Results in Teleoperation of Land Vehicles. Sandia Report SAND90-0299, Sandia National Laboratories, Albuquerque, NM.

Mettala, E. 1993. Reconnaissance, Surveillance and Target Acquisition Research for the Unmanned Ground Vehicle Program. *Proceedings of Image Understanding Workshop*, Morgan Kaufmann, San Francisco, pp. 275–279.

Miller, J. R. and Amidi, O. 1998. 3-D Site Mapping with the CMU Autonomous Helicopter. Proceedings of the 5th International Conference on Intelligent Autonomous Systems (IAS-5).

Miller, J. R., Amidi, O., Thorpe, C., and Kanade, T. 1999. Precision 3-D Modeling for Autonomous Helicopter Flight. *Proceedings of International Symposium of Robotics Research (ISRR)*.

Moravec, H. P. 1980. Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover. PhD Thesis, Stanford University, Stanford, CA.

Nashashibi, F., Fillatreau, P., Dacre-Wright, B., and Simeon, T. 1994. 3-d autonomous navigation in a natural environment. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 94)*, 1:433–439.

Nishihara, H. K. 1984. PRISM: A Practical Real-Time Imaging Stereo Matcher, Massachusetts Institute of Technology, Cambridge, MA.

Nistér, D., Naroditsky, O., and Bergen, J. 2004. Visual odometry. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, 1:652–659.

Ojeda, L., Borenstein, J., Witus, G., and Karlsen, R. Terrain Characterization and Classification with a Mobile Robot, to appear. *Journal of Field Robotics*.

Olin, K. and Tseng, D. Y. 1991. Autonomous Cross-Country Navigation. *IEEE Expert* 6(4):16–20.

Pomerleau, D. and Jochem, T. 1996. Rapidly Adapting Machine Vision for Automated Vehicle Steering. *IEEE Expert: Special Issue on Intelligent System and their Applications* 11(2):19-27.

Richardson, A. J. and Wiegand, C. L. 1977. Distinguishing vegetation from soil background information. *Photogrammetric Engineering and Remote Sensing* 43:1541–1552.

Roberts, J., Corke, P., and Winstanley, G. 1999. Development of a 3,500 tonne field robot. *The International Journal of Robotics Research* 18(7):739–752.

Roumeliotis, S., Johnson, A., and Montgomery, J. 2002. Augmenting Inertial Navigation with Image-Based Motion Estimation. Proceedings of 2002 IEEE International Conference on Robotics and Automation, Washington D.C., May 11–15, pp. 4326–4333.

Saridis, G., 1983. Intelligent Robot Control. *IEEE Trans on Automatic Control* AC-28(5):547–556.

Schenker, P., Hunstberger, T., Pirjanian, P., Dubowski, S., Iagnemma, K., and Sujan, V. 2003. Rovers for Agile Intelligent Traverse of Challenging Terrain. Proceedings of the 7th International Symposium on Artificial Intelligence, Robotics and Automation in Space, i-SAIRAS, Nara, Japan.

Shoemaker, C. M. and Bornstein, J. A. 1998. The Demo III UGV program: A testbed for autonomous navigation Research. IEEE International Symposium on Intelligent Control, Gaithersburg, MD.

Simeon, T. and Dacre-Wright, B. 1993. A practical motion planner for all-terrain mobile robots. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 93)*, 2:1357–1363. IEEE.

Sreenivasan, S. V. and Wilcox, B. H. 1994. Stability and traction control of an actively actuated micro-rover. *Journal of Robotics Systems* 11(6):487–502.

Stentz, A. 1994. Optimal and Efficient Path Planning For Partially-Known Environments. *Proceedings of IEEE ICRA*.

Stentz, A. 1995. The Focussed D* Algorithm for Real-Time Replanning. *Proceedings of IJCAI-95*, August.

Stentz, A., Kelly, A., Rander, P., Herman, H., Amidi, O., and Mandelbaum, R. 2002. Integrated Air/Ground Vehicle System for Semi-Autonomous Off-Road Navigation. *Proceedings of the AUVSI Unmanned Systems Conference*.

Stentz, A., Kelly, A., Rander, P., Herman, H., Amidi, O., Mandelbaum, R., Salgian, G., and Pedersen, J. 2003. Real-Time, Multi-Perspective Perception for Unmanned Ground Vehicles. *Proceedings of the AUVSI Unmanned Systems Conference*.

Sukkarieh, S., Gibbens, P., Grocholsky, B., Willis K., and

Durrant-Whyte, H. F. 2000. A Low-Cost, Redundant Inertial Measurement Unit for Unmanned Air Vehicles. *International Journal of Robotics Research* 19(11).

Talukder, A., Manduchi, R., Rankin, A., and Matthies, L. 2002. Fast and Reliable Obstacle Detection and Segmentation for Cross-Country Navigation. *IEEE Intelligent Vehicles*.

Thompson. A. 1977. The Navigation System of the JPL Robot. *Proceedings of the International Joint Conference for Artificial Intelligence*, pp. 749–757.

Thorpe, C. and Durrant-Whyte, H. 2001. Field Robots. Proceedings of the 10th International Symposium of Robotics Research, Lorne, Victoria, Australia, November. Springer-Verlag, London.

Vandapel, N. and Herbert, M. 2004. Finding Organized Structures in 3-D LADAR Data. Army Science Conference.

Vandapel, N., Huber, D., Kapuria, A., and Herbert, M. 2004. Natural Terrain Classification using 3-D Ladar Data. International Conference on Robotics and Automation, April.

Wellington, C. and Stentz, A. 2003. Learning Predictions of the Load-Bearing Surface for Autonomous Rough-Terrain Navigation in Vegetation. International Conference on Field and Service Robots, pp. 49–54.

Willstatter, R. and Stoll, A. 1913. Untersuchungen über Chlorophyll. Springer, Berlin.