THE GOLEM GROUP / UCLA

DARPA Grand Challenge Technical Paper

Contact:
Richard Mason
911 Lincoln Blvd. #7
Santa Monica, CA 90403.
mason@golemgroup.com

DARPA Grand Challenge 2005

We describe in general terms our approach to the DARPA Grand Challenge for Autonomous Ground Vehicles. Some of the distinctive elements of our approach are: fusion of horizontally- and vertically-sweeping ladar beams to disambiguate sensed terrain; texture-based image analysis to find the boundaries of an off-road trail; for localization, a Kalman filter utilizing steering properties of the vehicle; and for development, a powerful 4D visualization tool.

Choice of Vehicle Platform

Golem 1, our vehicle in the 2004 Grand Challenge, was based on a 1994 Ford F-150 truck with off-road suspension modifications. We believe this was a good choice of platform for several reasons. By design, the Grand Challenge route was well-matched to the capabilities of a commercial 4x4 pickup truck, such as the ones used by DARPA as chase vehicles. Also, we consider human drivability of the vehicle to be nearly essential for development purposes. The presence of two humans in the vehicle cab means that one can drive, or be ready to drive, while the other uses a computer. This adds an element of safety when testing systems that would otherwise be risky and inconvenient to test at high speed. Street-legal drivability of the vehicle is also very helpful logistically, especially for an urban team on a limited budget which has to travel some distance to suitable practice areas.

Reviewing the outcome of the 2004 Grand Challenge, we believe that generally speaking, vehicles capable of carrying at least two people did better than smaller vehicles, and vehicles based on commercial platforms did better than entirely custom-made vehicles. We felt this vindicated our choice of platform.

For the 2005 Grand Challenge, we decided to continue development with Golem 1, but also prepare a similar second vehicle for use in the Grand Challenge Event. Golem 2 is based on a 2005 Dodge Ram 2500. We chose a new model year truck, as opposed to another 1994 model like Golem 1, in order to get better mechanical reliability. We chose a three-quarter-ton Dodge Ram 2500, instead of another half-ton F-150, because the three-quarter-ton truck has a heavier suspension and we thought it would have a better chance of surviving certain accidents such as a rock striking the front axle. Otherwise, Golem 2 is similar enough to Golem 1 that technology can easily be transferred from one vehicle to the other. Apart from the benefits just mentioned, we wanted a second vehicle for redundancy (we were aware that several teams suffered serious vehicle accidents in the days leading up to the 2004 Grand Challenge) and parallelism (one vehicle can test obstacle avoidance while the other is being modified or repaired).

Both Golem 1 and Golem 2 use 35-inch Mickey Thompson Baja Claw Radial tires.

Processing

For development purposes, the Golem vehicles are run by laptop computers, including a set of Dell Latitude D810 laptops donated by Intel Corporation. The operating system is Fedora Linux. Any of the laptops can be inserted into either of the Golem vehicles and be used as the controlling computer. We have found this approach ideal for development, because individual team members can make changes to the software on their own laptops, take their laptops to a vehicle and test the results before sharing successful code changes with the rest of the team. Also, the system is highly redundant, so that if the laptop driving the vehicle were accidentally destroyed by an electrical short (as happened immediately prior to our DARPA Grand Challenge Site Visit) it could immediately be replaced by any of the other team members' laptops. The alternative approach of keeping a privileged computer or set of computers permanently mounted in the vehicles would, we think, reduce redundancy and make it less convenient for different programmers to work in parallel. We can make a post-development move from laptops to faster servers, if and when that seems desirable.

Our experience has generally been that we are not processor-limited and that a single laptop computer is sufficient to do all the tasks of processing laser range data, planning trajectories, and controlling the vehicle. The important exception is computer vision which does tend to be processing-intensive and is performed on its own processor, either another laptop or custom hardware such as the single board computer, based on a single Motorola PowerPC 7410 (G4) processor, supplied by Mobileye LTD. Communication with computer vision processors is by Ethernet.

We use the RS-232 bus to interface to most peripherals for reasons of cost, simplicity, and robustness. RS-485 is used for high-speed communication with Sick ladars. A block diagram of the architecture is illustrated in Figure 1.

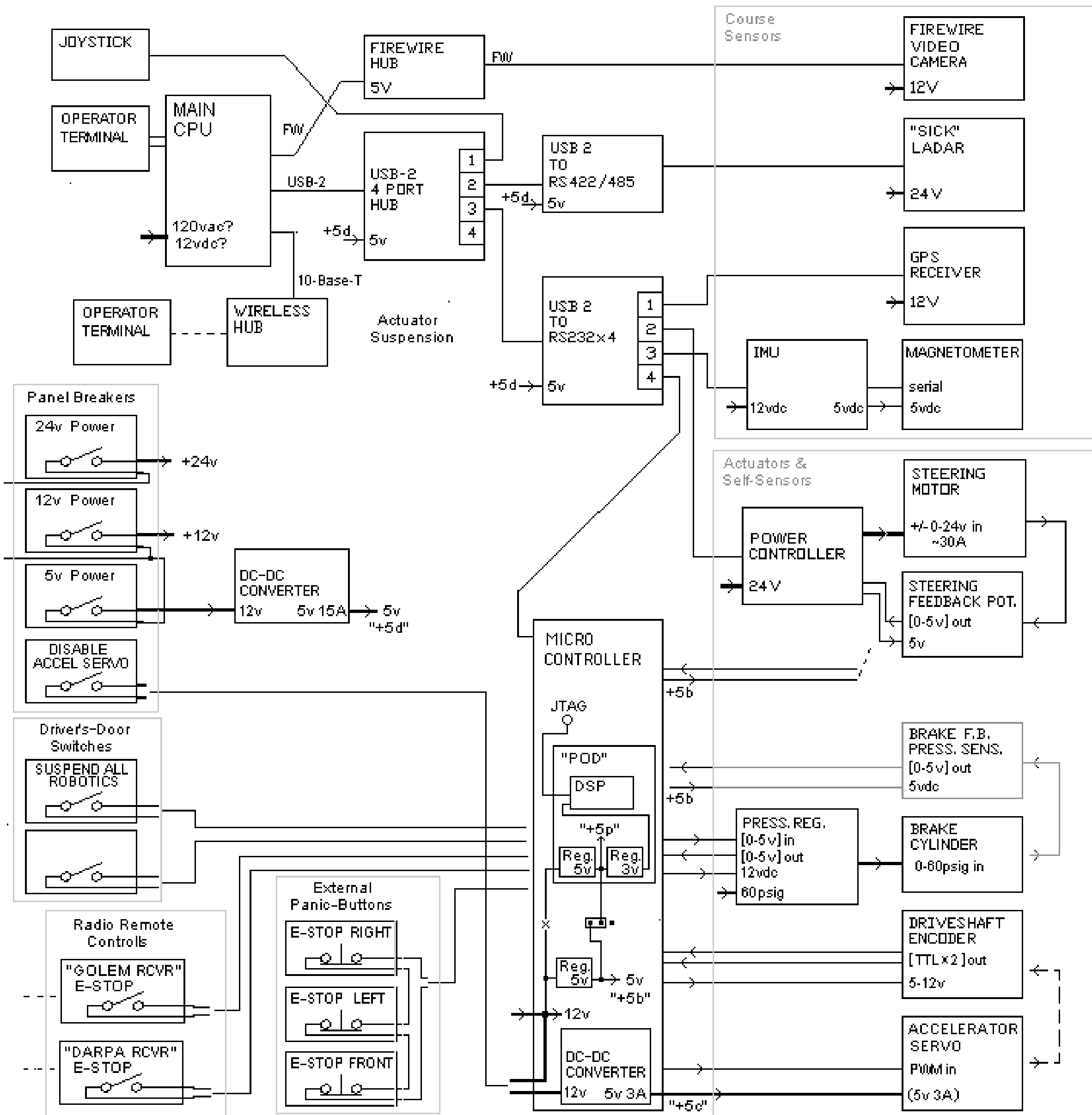

Figure 1. Architectural block diagram.

We devoted considerable effort to our visualization/control interface software, called "Dashboard." All sensor data is logged while the vehicle is running and can be examined by Dashboard in real time or replayed later. Some interesting features of Dashboard are: 3D visualization in space of the truck's location, heading, and wheel angle, the location of waypoints, ladar reflections, video imagery, inferred obstacles and trail boundaries, the planned route, and current and future planned speed; also the ability to pan, rotate, and zoom to different viewpoints; the ability to measure distances and angles between any points on the screen; and very importantly, the ability to scroll backwards and forwards in time when replaying a "movie" from logged data. In this way we can find the critical moments of a test run and visualize exactly what the state of the vehicle was at that time, what it sensed, and what decisions it made. This is very useful in debugging.

Localization

The Golem vehicles each use a NovAtel Propak-LBPlus receiver for GPS positioning with Omnistar HP correction. During our development process we also had good results using a Trimble AgGPS 114 receiver with Omnistar VBS correction. A C-MIGITS III inertial navigation system from BEI Technologies is used to track changes in orientation. During GPS outages we continue to track position using the C-MIGITS III and measurements of wheel rotation and steering angle.

The NovAtel Propak-LBPlus GPS has a nominal position accuracy of 20 cm, but under adverse conditions, this accuracy figure can become meaningless. For example, when passing into the shadow of a metal structure, we have witnessed sudden changes in reported position of over 100 meters. We use a Kalman filter which includes the steering properties of the truck in its physical model of the system to reject transient errors of this type. Under typical route conditions we estimate we can maintain a position accuracy of under 30 cm.

Terrain Sensing

Our sensing strategy is to use Sick laser measurement systems ("ladars") to detect significant positive and negative obstacles (roughly defined as too-abrupt changes of apparent ground elevation), and a Mobileye Pathfinder vision system to detect the edges of the road or trail. We have also experimented with using vision to classify objects detected by the ladars.

Each Golem vehicle has one or more Sick LMS-291 ladars, each arranged to measure the ground profile in a particular azimuthal direction, by sweeping its beam through a 90-degree arc of elevation (from approximately –60 degrees to +30 degrees elevation) at 75 Hz. At least one of these ladars on Golem 2 will be servomotor-actuated and movable to any azimuthal direction within a 180 degree arc, e.g., in the direction of a turn.

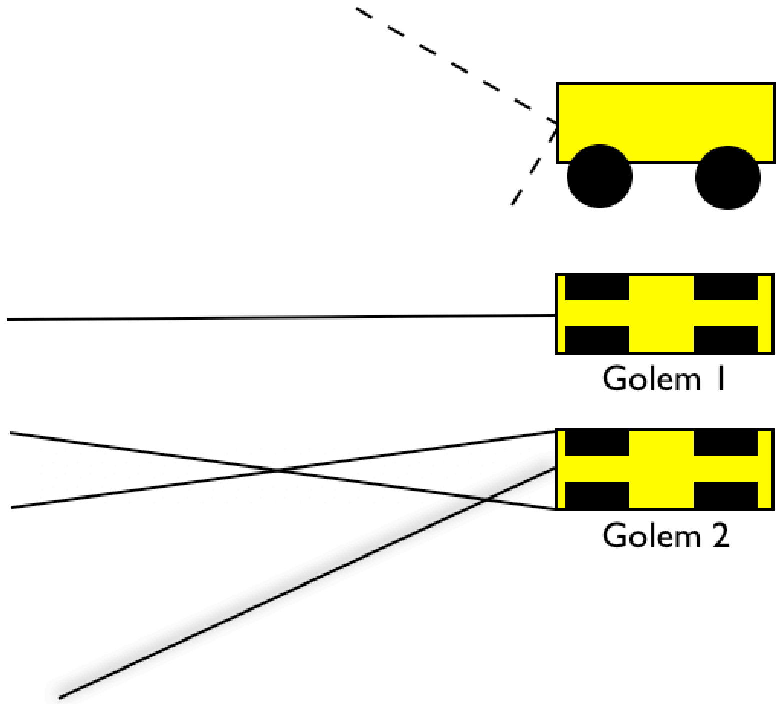

Figure 2. Arrangement of vertical-plane Sick LMS-291 ladars on the Golem vehicles.

The virtue of ladars used in this vertical-plane configuration is that the ground profiles are easy to interpret, and are not particularly prone to confusion due to rolling, pitching, or bouncing motion of the vehicle. (Of course, a six-degree error in pitch could make a marginally-traversable 27-degree slope appear to be a marginally-untraversable 33-degree slope, or vice versa. But away from the margin, a small pitch error should not radically change the interpretation of a ground profile.) It is reasonable to expect these ladars to detect significant negative obstacles at up to 20 meters, and significant positive obstacles up to 80 meters away. The disadvantage, of course, is that since each ladar looks in only a single azimuthal direction, instantaneous azimuthal coverage is poor and obstacles between the vertical ladar scan planes will be missed.

We also use a Sick LMS-221 which sweeps its beam in a horizontal plane through a 180-degree arc of azimuth. It can be supplemented by a Sick LMS-291 which sweeps a 90-degree arc in a horizontal plane at a different height. These horizontally-sweeping ladars are complementary in strengths and weaknesses to the vertically-sweeping ladars. A single horizontal ladar sweep provides good azimuthal coverage of positive obstacles, but on the other hand, a single horizontal ladar sweep is difficult to interpret in the absence of other information. The laser beam returns from a surface at a particular range, but is that surface an obstacle, or is it a traversable slope, or is it merely flat ground which has come into view because of the pitch and roll of the vehicle body? Our approach is to fuse the information from the vertically and horizontally swept ladars in order to take advantage of their complementary properties. With an idea of elevation profile derived from the vertically-sweeping ladars, we can make a more reliable interpretation of the horizontal ladar data. We have had success distinguishing between traversable slopes and non-traversable obstacles with this method.

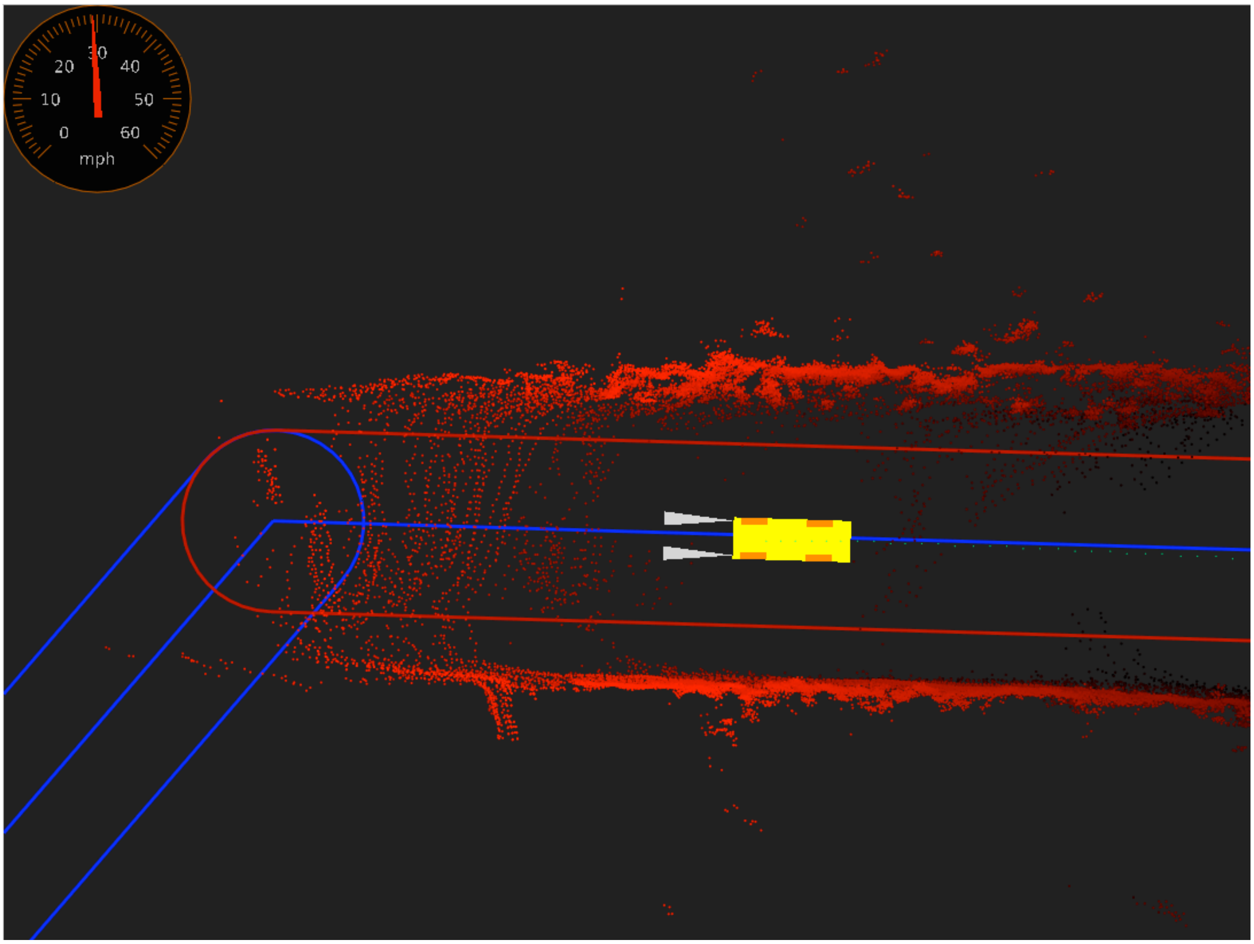

Figure 3. Top-down visualization of vehicle on route, with ladar reflections in red. The route ahead of the vehicle is slightly rough, but traversable. If the ladar data filter is working correctly, the vehicle will determine that the ladar reflections in front of it only represent a bumpy road surface, not an obstacle.

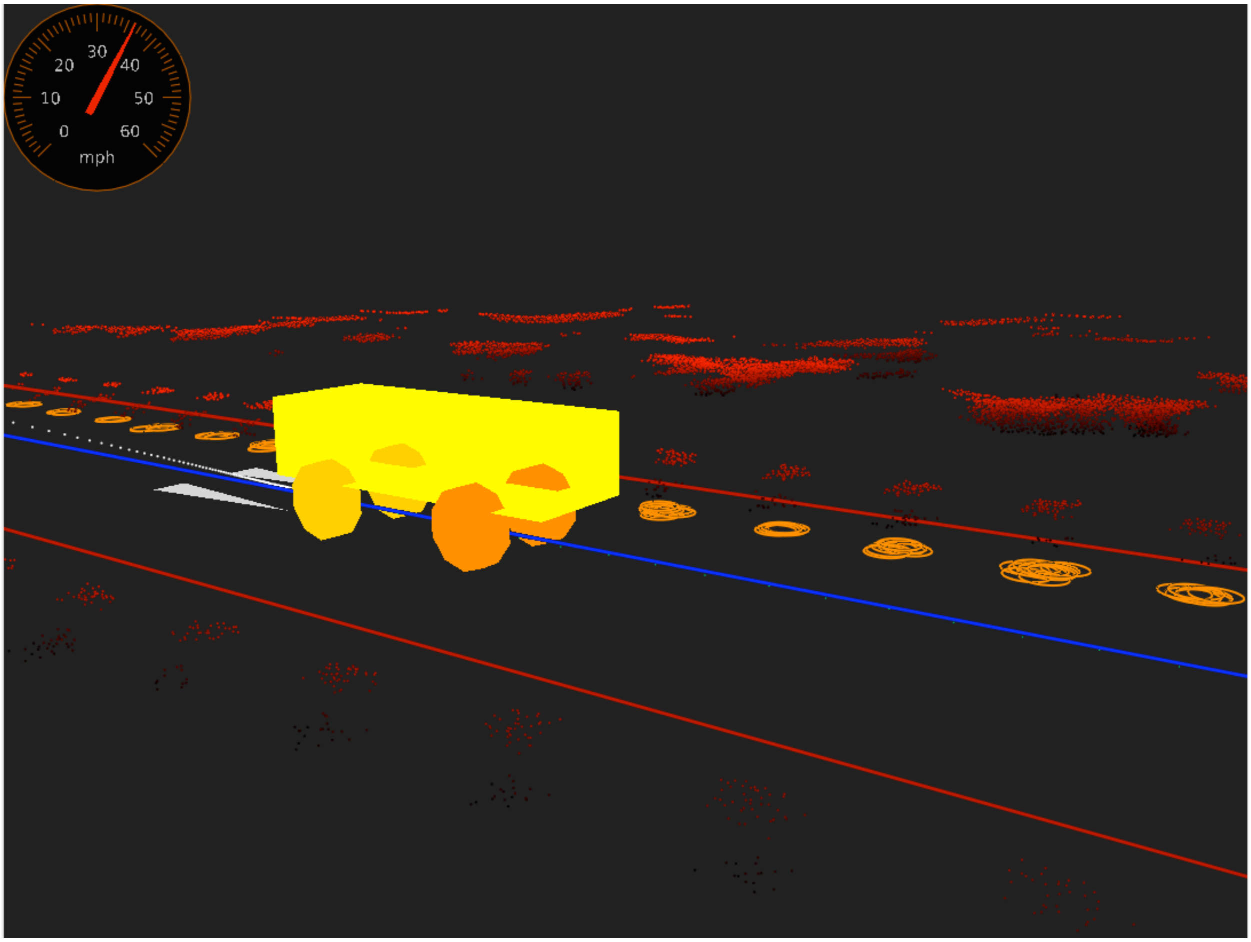

Figure 4. Perspective visualization of vehicle on route, with ladar reflections in red. The small clouds of ladar points, about three feet above the ground, are reflections from the posts of a guardrail. The orange circles on the ground underneath the guardrail posts are symbolic graphics indicating that the vehicle has identified non-traversable obstacles at those locations.

Mobileye's Pathfinder system is a vision system using images from a forward looking camera to detect paved road or unpaved trail boundaries using intensity contrast information and texture analysis. The system output includes the lateral distance to path boundaries, the relative position of the center of the path, vehicle heading angle relative to the path, and estimated pitch angle. The output also indicates cases in which there is no evidence for path in the image. In Paved Road mode, the system also detects vehicles in front of the host car, and output includes distance and direction to the detected vehicles.

The prototype system developed by Mobileye uses a miniature lipstick analog CCD camera with a typical 45 degree horizontal field of view for acquiring video images. As mentioned previously, the processing unit is a custom single board computer based on a single Motorola PowerPC 7410 (G4) processor. System input options include CAN and Ethernet. A keypad allows manual setting of various parameters. The color video input is captured at a resolution of 640x480 pixels. The images are processed at a rate of 10-20 frames per second, and the results are sent through the output channels at the same rate. The system has two running modes: Paved Road, and Off-Road. Switching between the modes is done by sending the appropriate message through the input Ethernet channel.

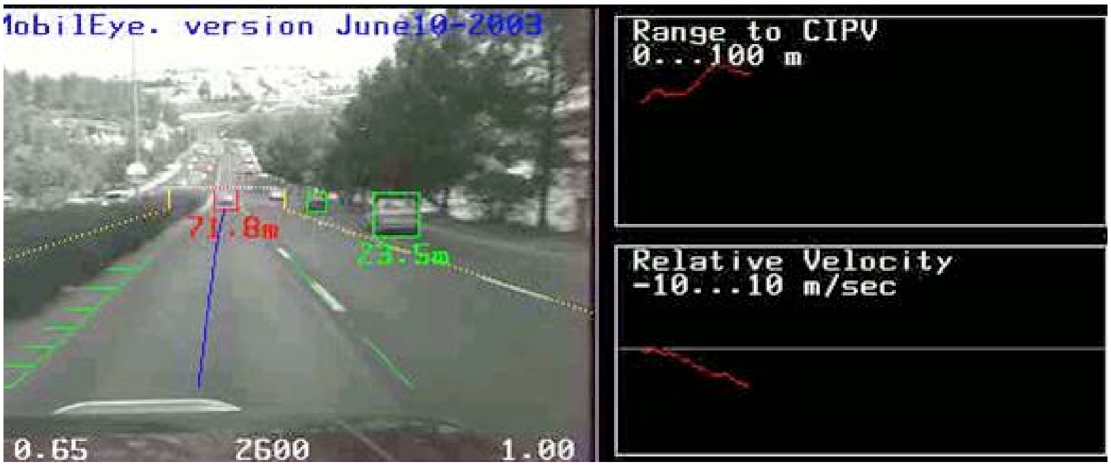

Figure 5. Mobileye Pathfinder system in Paved Road mode.

In Paved Road mode, the Pathfinder system detects and tracks vehicles on the road ahead providing distance, relative speed, and lane assignment of the vehicles ahead (whether in ego-lane or not). In addition, the system detects lane markings and measures and monitors distance to road boundaries on paved roads. These vehicle detection and lane detection algorithms are the same as those used in the Mobileye-AWS™, an advanced system for the automotive aftermarket which offers a suite of active safety applications for accident reduction. The Paved Road lane detection algorithm consists of four major steps:

1) Process the image with a few filters that enhance lane marks.
2) Assuming a flat road surface, stable lane width, and a linear lane model, predict the intersection point of lane marks. The result is an estimate of pitch and heading angle.
3) Given the intersection point calculated at step (2), calculate the coordinates of detected lane marks in the world frame. The result is lateral distance to the lane marks.
4) Compare the results from step (3) to lane position in previous frames. The system holds a few hypotheses about lane position, and outputs only the most trusted one.

The Paved Road algorithm also uses elements of vehicle detection and scene analysis to reduce ambiguities in lane detection. It has been tested on worldwide roads and can detect all types of lane markings: continuous, dashed, Bott's dots, and road boundaries. It is robust under varying illumination and weather conditions, shadows, rain, and night. The accuracy of lane position measurement is about 5 cm in good conditions.

Off-road trails are usually characterized by high-complexity textures, and do not allow the use of simple edge detection methods. The most effective way to detect off-road trail edge is by examining the textures in the image. The Off-Road algorithm consists of three major steps:

1) Detect path boundaries by searching for a difference between the texture on trail and the texture out of the trail. For robust performance, this method has to be combined with geometric understanding of the vehicle surroundings. The geometrical knowledge is achieved by combination of image stabilization, horizon detection, and calculation of the intersection point of the trail's left boundary and right boundary. This step is based on the Paved Road algorithm with the necessary modifications. It also combines feature tracking methods to assist in horizon detection. This method is effective on straight trails and shallow curves; however it fails on sharp curves and large elevation changes.
2) Detect path boundaries in the image without geometrical understanding of the surrounding world. The algorithm classifies the image into trail-like textures and non-trail textures, and finds the boundaries between trail and non-trail areas. The classification of texture types is based on a learning process done in advance during the algorithm development, not online. Therefore this method is only suitable for textures that were used in the learning process, and will fail on a trail with different characteristics. It will, however not be affected by sharp curves or large elevation changes, and so it is suitable as a complement to the method of step (1). The algorithm was tested on desert trails with high success.
3) For robust results, the results from the two approaches above are combined.

After detecting the road edge in the image, the system has to estimate the distance to it. In existing prototypes, the camera field-of-view does not include the area near the front wheel. Therefore, it is necessary to extrapolate the visible data in order to estimate the road edge position in proximity to the front wheel. On paved road, the extrapolation is usually good enough to allow 5 cm accuracy in the measurement of distance to road edge. On off-road trails the accuracy is expected to be lower than 5 cm and has not been tested at this point. The accuracy of distance measurement is also affected by pitch angle estimation that is necessary for transforming image coordinates to world metric. In off-road conditions it is more difficult to determine the pitch angle. The expected accuracy is roughly 20 cm in off-road conditions. An optional output includes the lateral distance between vehicle centerline to path boundaries at 6 meters ahead of the vehicle. Since the path boundaries at that distance are usually visible in the image, this output is more accurate.

Figure 6. Mobileye Pathfinder system in Off-Road Mode. Green lines indicate the detected edges of the trail.

Internal Sensing

A Hall sensor and a ring of magnets attached to the rear differential form a high-accuracy odometer that measures revolutions of the rear axle. The inferred measurement of vehicle speed is used for velocity feedback control. Either a potentiometer (Golem 1) or an absolute optical encoder (Golem 2) is used to determine the vehicle's steering angle.

Vehicle Planning and Control

A few times per second, and particularly if a new obstacle has been generated on the truck's expected route, the list of current obstacles is sent to the planner, which attempts to generate a continuous path for the center of the truck's front axle meeting all of the following criteria: avoid all ladar-detected positive or negative obstacles by a distance of at least one half-truck-width; avoid exiting DARPA-specified waypoint boundaries (although, should the situation somehow arise, it is permissible to cross waypoint boundaries in the other direction, back on to the route); avoid physically unachievable changes of curvature; and, if possible, remain within trail boundaries detected by the Mobileye system. Sensor events are associated with a latitude and longitude so that they can be mixed naturally with navigation information about the route. Obstacles are represented by point locations on the internal map, and trail and waypoint-corridor boundaries are represented as lines, but otherwise the collision-avoidance approach is similar. If a candidate continuous curve does collide with an obstacle or boundary, the planner perturbs the point of collision to find a new position which does not intersect any obstacles, then attempts to generate a new continuous path which passes through that position.

Once a collision-free path has been planned, the vehicle finds its lateral distance from the nearest point on the path and its heading deviation from that point on the path and uses feedback control via the steering to minimize those two errors. A velocity manager monitors the throttle and brake and increases or reduces the vehicle's speed to an appropriate level, taking into account upcoming turns as well as user-imposed speed limits. The control software takes account of the steering properties of the truck, including steering bias, understeer depending on velocity, and maximum achievable path curvature at a given velocity, all as measured in testing.

If an obstacle is detected in the vehicle's path, it will decelerate and search for a path around the obstacle. If the vehicle has somehow drifted outside the lateral boundaries of the route, it will keep going (unless e-stopped, of course) and should merge back onto the route while avoiding obstacles in the normal fashion. If the vehicle has somehow missed a waypoint, it will not back up but will continue going through the rest of the waypoints. We will try to make sure that the vehicle does not allow itself to bypass a whole section of course. If the vehicle becomes stuck or finds itself unable to proceed without hitting an obstacle or leaving the course, we have no special fallback plan; the vehicle will just be immobilized. We would like to have the vehicle shift into reverse and try to back up, but this is not implemented yet.

A person sitting at the driver station in the vehicle has a lever to mechanically disengage the steering servomotor from the steering column, after which the driver can turn the steering wheel normally. He can also apply the brakes normally at any time, and can kill power to the computer's throttle servo, and other systems, by throwing a switch. After these actions the vehicle is drivable in normal, street-legal fashion. This is the method by which the vehicle is operated in non-autonomous mode.

System Tests

We have tested the vehicle on ground of various difficulty, including a nearly-flat parking lot, a multi-level parking lot with sloping ramps, the El Mirage lakebed and nearby trails, grassy fields in the Rowher Flats recreational area, and last year's Grand Challenge course, especially Daggett Ridge. Some of the problems we encountered were the following:

- Traversable sloping ground was being misperceived as nontraversable. We solved this by gathering ground data with vertically sweeping ladars and using that to interpret horizontal ladar data.
- Poor estimates of vehicle heading during turns led to perceived obstacles being recorded in inaccurate locations, and the vehicle would later be confused by these mislocated obstacles. We solved this by improving the measurement of vehicle heading, and by allowing the discard of old, possibly bad obstacle information.
- We experienced some electrical short circuits which destroyed a cable, a motor controller, and a laptop computer. These nudged us into better practices of circuit fault protection.

We have tested the vehicle in moderate rain. Although the rain did introduce noise into the ladar measurements, our obstacle detection software appeared fairly robust to this noise.

Acknowledgements

We would like to thank our sponsors: Mobileye LTD; BEI Technologies, Inc.; Intel Corporation; Federal Signal Corporation; OmniSTAR USA; Kenyon Labs LLC; Prime Resource, Inc.; National Instruments; and Sick, Inc.

Special thanks also to Dean Vijay Dhir, Vice-Chancellor Roberto Peccei, Associate Dean Greg Pottie, Jim Stewart, Audrey Scates, and Tim Pfiffert of UCLA.

The Golem Group / UCLA team members include Dr. Richard Mason, Dr. Jim Radford, Joshua Arensberg, David Caldwell, Kerry Connor, Jeff Elings, Brian Fulkerson, Jerry Fuller, Izaak Giberson, Eagle Jones, Deepak Kumar, Dr. Maribeth Mason, Jason Meltzer, Dr. Brent Morgan, Jim Swenson, Robb Walters, and Matt Williams; Professor Stefano Soatto, Professor Emilio Frazzoli, Professor Robert M'Closkey, Dr. Michael Naderhirn, Chung Hsieh, Genny Pang, and Zahi Tarzi of UCLA; and Professor Amnon Shashua, Simon Polak, and Yaniv Alon of the Hebrew University of Jerusalem. For more information see www.golemgroup.com.