Stanford Racing Team's Entry In The 2005 DARPA Grand Challenge

Stanford Racing Team

Email: srt@cs.stanford.edu

Web: www.stanfordracing.org

Abstract

The Stanford Racing Team (SRT) has successfully developed an autonomous robotic vehicle capable of driving through desert terrain without human intervention. The SRT vehicle *Stanley* is based on a reinforced Volkswagen Touareg, equipped with a custom drive-by-wire system, a sensor rack, and a computing system. The vehicle is controlled through a distributed software system that uses inertial sensing for pose estimation, and lasers, vision, and RADAR for environmental perception. Sensor data is mapped into a drivability map, which is used to set the direction and velocity of the vehicle. A major emphasis of the SRT has been early development of a prototype end-to-end system, to enable extensive testing in authentic desert terrain.

1 Project Overview

The Stanford Racing Team (SRT) is Stanford's entry in the 2005 DARPA Grand Challenge. The SRT brings together leading automotive engineers, artificial intelligence researchers, and

[0]DISCLAIMER: The information contained in this paper does not represent the official policies, either expressed of implied, of the Defense Advanced Research Project Agency (DARPA) of the Department of Defense. DARPA does not guarantee the accuracy or the reliability of the information contained in this paper.

experienced program managers, to develop the next generation of self-driving vehicles. The SRT has develop a robotic vehicle dubbed “Stanley,” which has been selected as a semifinalist by DARPA.

The SRT leverages proven commercial off-the-shelf vehicles with advanced perception and driving systems developed by the Stanford AI Lab (SAIL) and affiliated researchers. The strong emphasis on software and vehicle intelligence indicates the SRT’s belief that the DARPA Grand Challenge is largely a software competition. As long as the vehicle stays on the road and avoids obstacles, commercial SUVs are fully capable of negotiating the terrain. The challenge, thus, has been to build a robust software system that guides the vehicle in the right direction at the appropriate speed.

The SRT software system employs a number of advanced techniques from the field of artificial intelligence, such as probabilistic graphical models and machine learning. Following methodologies described in [3], The SRT has also developed novel estimation and control methods specifically suited to driving at moderate speeds (35mph) through unrehearsed terrain. The software is housed in a state-of-the-art commercial off-road vehicle, appropriate modified to provide precision navigation under computer control.

From the beginning of this project, the SRT has placed a strong emphasis on in-field development and testing. Initial tests of a preliminary end-to-end system took place in December 2004. Since this time, Stanley has logged many hundreds of autonomous miles.

This article provides a high-level overview of the various system components, at a level suitable for broad public dissemination. Further material can be found on the team’s Web site, at

`www.stanfordracing.org`

The goal of the Stanford Racing Team is to to develop a vehicle that can finish the 2005 DARPA Grand Challenge within the allotted time. Through this research, the SRT also hopes to make driving safer, by advancing the state-of-the-art in vehicle navigation and driver assistance systems. The SRT believes that the technologies developed in this project can enhance the awareness of drivers and their vehicles, and enhance the safety of vehicular traffic.

2 Team Composition and Sponsorship

The SRT formed in July 2004, but continued to grow for the six months that followed. The team consists of approximately 50 individuals that include Stanford students, faculty, and alumni, and employees of the SRT primary supporters and other nearby research labs. The team's overall lead is a faculty member in the Stanford Artificial Intelligence Lab, a unit of Stanford's School of Engineering.

The team is comprised of four major groups: The *Vehicle Group* oversees all modifications and component developments related to the core vehicle. This includes the drive-by-wire systems, the sensor and computer mounts, and the computer systems. The group is led by researchers from Volkswagen of America's Electronic Research Lab. The *Software Group* develops all software, including the navigation software and the various health monitor and safety systems. The software group is led by researchers affiliated with Stanford University. The *Testing Group* is responsible for testing all system components, and the system as a whole, according to a specified testing schedule. The members of this group are separate from any of the other groups. The testing group is led by researchers affiliated with Stanford University. The *Communications Group* manages all media relations and fund raising activities of the SRT. The communications group is led by employees of Mohr Davidow Ventures.

The SRT is sponsored through four *Primary Supporters*: Volkswagen of America's Electronic Research Lab, Mohr Davidow Ventures, Android, and Red Bull. The Primary Supporters together with the Stanford team leaders form the *SRT Steering Committee*, which oversees the SRT operations. The SRT has also received support from Intel Research, Honeywell, Tyzx, Inc., and Coverty, Inc. Generous financial contributions were made by the David Cheriton, the Johnson Family, and Vint Cerf.

3 Vehicle Description

The Stanley vehicle is based on a stock Volkswagen Touareg R5 with variable-height air suspension. The Diesel-powered vehicle was selected for its fuel efficiency and its ability to negotiate

Figure 1: Stanley is based on a 2004 Volkswagen Touareg R5 Diesel. The vehicle is equipped with a number of sensors for environment perception and localization.

off-road terrain. To protect the vehicle from environmental impact, the vehicle is outfitted with custom skid plates and a front bumper. Fig. 1 provides images of the vehicle.

The Volkswagen Touareg R5 is natively throttle and brake-by-wire. A custom interface to the throttle and braking system enables Stanley's computers to actuate both of these systems. An additional DC motor attached to the steering column provides the vehicle with a steer-by-wire capability. Vehicle data such as the individual wheel speeds are sensed automatically and communicated to the computer system through a custom CAN bus interface. The Touareg's alternator provides all power for the various computing and sensing systems.

The vehicle's custom-made roof rack holds most of Stanley's sensors. For environment perception, the roof rack holds five SICK laser range finders pointed forward into the driving direction of the vehicle, a color camera which is also pointed forward and angled slightly downwards, and two antennae of a forward-pointed RADAR system. A number of antenna are also attached to the roof rack, specifically one antenna for the GPS positioning system, two additional GPS antennae for the GPS compass, the communication antenna for the DARPA emergency E-Stop, and a horn and a signal light, as required by the DARPA Grand Challenge rules. Three additional GPS antenna for the DARPA E-Stop are directly attached to the roof.

The computing system is located in the vehicle's trunk, as shown in Fig. 2. Special air ducts direct air flow from the vehicle's AC system into the trunk for cooling. The trunk features a shock-mounted rack that carries an array of six Pentium M Blade computers, a Gigabit Ethernet

Figure 2: Left: The computing system in the trunk of the vehicle. Right: The drive-by-wire system and the interface for manual vehicle operation.

switch, and various devices that interface to the physical sensors and the Touareg's actuators. It also features a custom-made power system with backup batteries and a switch box that enables Stanley to power cycle individual system components. The DARPA-provided E-Stop is also located on this rack, on additional shock compensation. A 6 degree of freedom (DOF) inertial measurement unit (IMU) is rigidly attached to the vehicle frame underneath the computing rack in the trunk.

4 Autonomous Operations

Autonomous navigation is achieved through a processing pipeline that maps raw sensor data into an internal state estimate. The internal state is comprised of a number of variables, relating to the vehicle's location, the workings of the various hardware components, and the location of obstacles in the environment.

4.1 Localization

At any point in time, the vehicle is localized with respect to a global UTM coordinate frame. Localization also involves the estimation of the vehicle's roll, pitch, and yaw angles. Stanley achieves its localization through an unscented Kalman filter (UKF) [1], which is a non-linear version of the Kalman filter. The UKF asynchronously integrates data from the GPS systems, the IMU, and the CAN bus, at a maximum update rate of 100 Hz. It utilizes a "bicycle model"

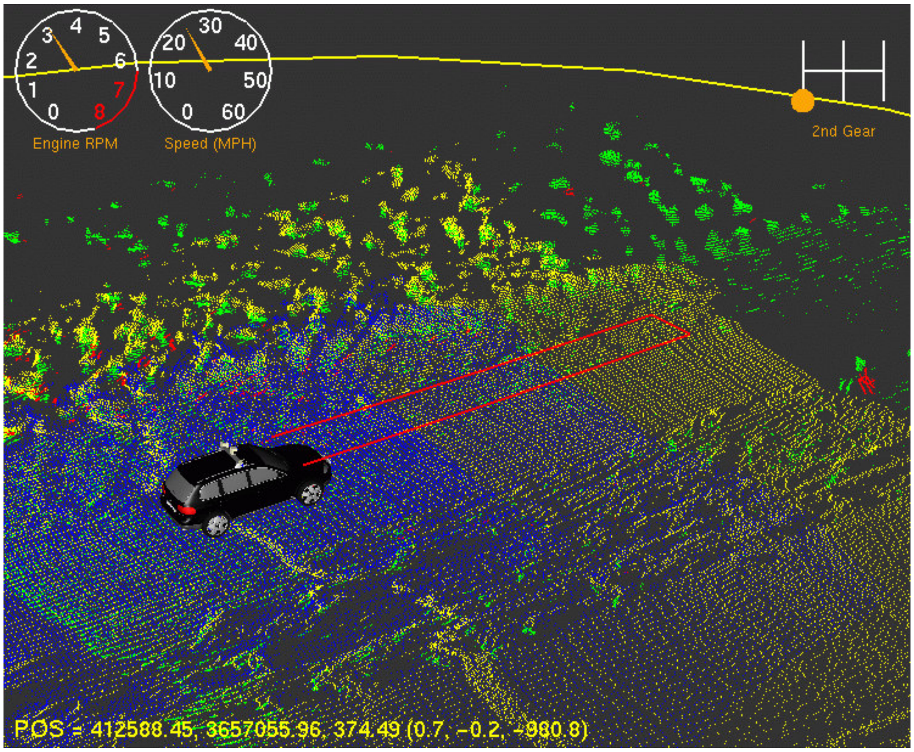

Figure 3: Laser data; see text.

for accurate position estimation during GPS outages. The output of the UKF are 6-D estimates of the vehicle position and Euler angles along with uncertainty covariances.

The localization module enables the vehicle to map the global RDDF file into local vehicle coordinates. To accommodate the residual uncertainty in the location estimates, the width of the RDDF corridor is dynamically adjusted in proportion to this uncertainty. As a result, the vehicle can accommodate moments of high position uncertainty.

4.2 Sensor Processing

Environmental sensing is achieved through the three different sensing modalities: laser, vision, and RADAR. Each of these systems is characterized by a different trade-off between range and accuracy.

The laser system provides accurate short-range perception, up to a range of approximately 25 meters. This range is sufficient for slow motion, but insufficient for the speeds required to win the Challenge. To enable faster motion, Stanley relies on two complementary systems, a camera and a RADAR system. The camera provides an enhanced range relative to the laser, and it captures denser data than each individual laser. However, the camera does not provide range data. The RADAR system provides range data for a range of up to 200 meters, but at a level of coarseness far inferior to the laser measurements.

The software system geo-references all raw sensor data by the UKF position estimates in global UTM coordinates. The laser data is continually analyzed for possible obstacles, defined as rapid elevation changes exceeding a height of 15cm. A temporal Markov chain is used to model the temporal information loss in the data acquisition process; and the Markov chain error terms are considered in the assessment of surface ahead. The specific functions involved in detecting obstacles are determined through a machine learning algorithm, which relies on human driving to acquire "training examples" of drivable terrain. See Fig. 3 for typical laser data. The coloring in this figure corresponds to different physical laser sensors.

The vision processing module relies on an adaptive filter to discriminate the road ahead from obstacles near the road. The filter classifies the terrain based on texture and color appearance of the desert terrain within the camera image. Using online machine learning, the vision module continually adapts to different terrain types, using near-range data classified by the lasers to determine the current best model of the road surface. This adaptation takes place at a rate of 8Hz. Rectification into UTM coordinates is achieved through a projective formula that makes an implicit planar world assumption.

The RADAR data is processed through a proprietary algorithm that identifies large obstacles in the environment. A temporal filter tracks individual singular obstacles over time, to reduce the false positive rate. RADAR data is mapped into the drivability map under a flat ground assumption.

4.3 Environmental Mapping

The data of all these three sensors is integrated into a single model of the environment, called the *drivability map*. Each cell in this 2-D map assumes one of three values: unknown, drivable,

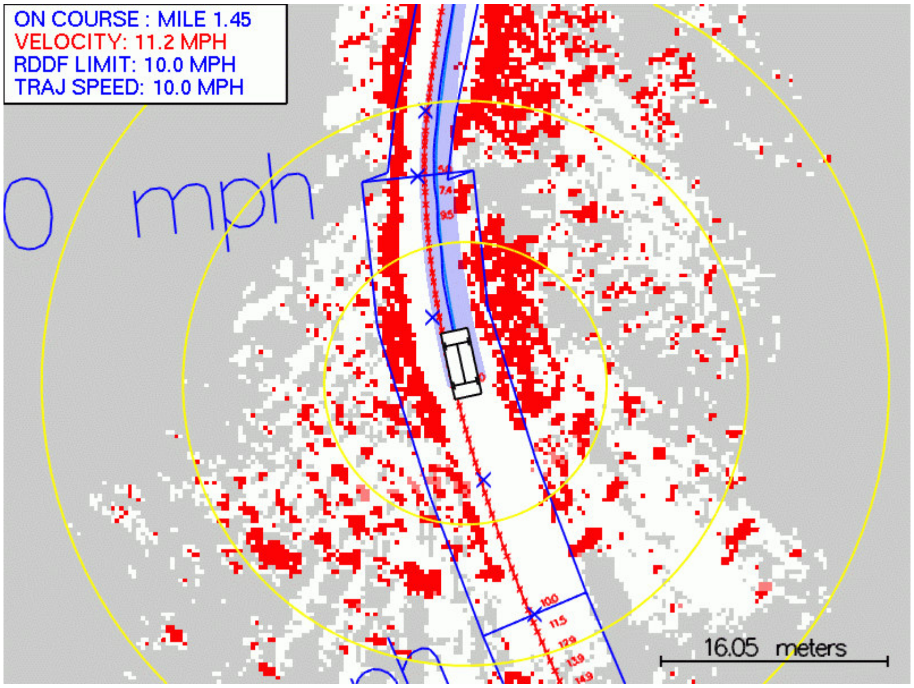

Figure 4: A typical drivability map.

or not drivable. The exact value is a function of the sensor data received for this cell. The map is referenced in global coordinates, though for computational reasons only a small window is retained at any point in time. The drivability map is updated asynchronously for the different sensor types, at rates that vary from 8Hz to 75Hz. As the vehicle moves, the map is shifted so as to always contain all cells within a fixed margin around the vehicle.

Fig 4 illustrates the drivability map. Shown there is the vehicle within its local environment. White grid cell correspond to drivable terrain; red cells to obstacles; and grey cells to unknown terrain. A rolling grid focuses the map on the relevant area around the vehicle.

To ensure consistency of this map, the sensors are periodically calibrated using data of dedicated obstacles of known dimensions. Calibration is an offline process which involves human labeling of sensor data. The calibration process adjusts the exact pointing directions of the individual sensors by minimizing a quadratic error, defined through multiple sightings of the same calibration obstacle.

4.4 Road Condition Estimates

In addition to the drivability map, the system also estimates a number of other variables pertaining to the general condition and structure of the environment. In particular, Stanley utilizes estimators of the terrain ruggedness, the terrain slope, and the left and right road boundaries. All of those estimates are implemented as low-pass filters on data directly derived from the sensor measurements. They are used to set the driving direction and the velocity of the vehicle.

The SRT robot also uses a detector for dead ends. While dead ends are generally unlikely to occur in the context of the 2005 DARPA Grand Challenge, they still may occur when disabled vehicles block parts of the road. The dead end detector is a high-pass filter on the drivability map; its main function is to initiate back-ups.

5 Vehicle Control

The state estimate are used to determine the three primary vehicle controls: the steering, throttle, and brake. It also controls the gear shifter.

The vehicle control system is implemented through three primary control systems, operating at different levels of temporal and spatial abstraction: a PID controller, a path planning algorithm, and a finite state automaton.

5.1 PID Motion Control

The PID controller accepts as input a reference trajectory provided by the path planning algorithm, and the vehicle state as provided by the Kalman filter. The PID controller generates steering and velocity controls that are executed by the vehicle. It is updated at a frequency of 20Hz.

The steering controller operates by minimizing the lateral offset to a desired trajectory provided by the path planer, with additional terms addressing steering wheel lag and vehicle drift. The velocity controller adjusts the brake pressure and the throttle position so as to attain a velocity commanded by the path planning module. The control module supports forward and backward motion.

5.2 Path Planning

The path planning module accepts as input the drivability map and the estimated robot pose, along with the corridor boundary from the RDDF file. The path planning module produces as output a reference trajectory suitable for vehicle control. This trajectory is determined by trading off five primary control objectives: The number of non-drivable cells along a path, the clearance to nearby obstacles, the nearness to the road center, the proximity to the adjusted RDDF corridor boundary, and the amount of lateral acceleration necessary to attain a given trajectory. By trading off these five different measures, the vehicle tends to identify paths that are safe to drive, within the RDDF corridor, and that maximize progress. Path planning takes place at a frequency of 10Hz.

The path planning module also sets the target velocity of the vehicle. The velocity controller runs at 10Hz. During every iteration, it generates a target trajectory that is communicated to the controller. The target velocity is obtained as a function of a number of criteria. Specifically, Stanley always assumes an allowable velocity according to pre-processed RDDF file, and it slows down in curves so as to retain the ability to avoid unexpected obstacles. The vehicle also adapts its velocity to the roughness of terrain, and to the nearness of obstacles. The specific transfer function emulates human driving characteristics, and is learned from data gathered through human driving.

To attain a suitable trajectory and associated maximum velocity, the RDDF file is processed by a smoother. The smoother adds additional via points and ensures that the resulting trajectory possesses relatively smooth curvature. The preprocessing then also generates velocities so that while executing a turn, the robot never exceeds a velocity that might jeopardize the vehicle's ability to avoid sudden obstacles. This calculation is based on a physical model of the actual vehicle.

5.3 State Automaton

The highest level of control is implemented through a finite state automaton (FSA). The FSA monitors the various state and road condition estimates to determine the principal driving mode of the vehicle. Driving modes include modes of forward motion, stopping, gear shifting, and backing up. The back up mode is used when the vehicle planner determines that all forward vehicle paths would result in a collision.

The FSA provides the highest level of vehicle control. It also implement the various steps necessary to react to a pause command by the DARPA team.

6 Software System

The various elements of the Stanley software system are all embedded into a large distributed architecture. The software is broken down into modules, each of which establishes an individual process on one of Stanley's computers. These processes are ran asynchronously on an distributed array of six Pentium M Blade computers. The clocks of these computers are constantly synchronized to ensure consistent time stamping. All inter-module communication is provided through the publicly available open source *Inter Process Communication* (IPC) package [2]. The IPC enables different modules to communicate via TCP/IP messages over the local Ethernet.[1] All software is written in C/C++. The operating system is Linux. Software verification is achieved with the help of code analysis tools developed by Coverty, Inc.

The software system possesses a number of data logging and display modules. Most of the sensor and control data is logged during major system tests. The visualization routines operate equally on live and logged data. The software also utilizes a centralized parameter server which ensures global consistency.

The software architecture also provides a number of safety and recovery mechanisms to accommodate component failure. A dedicated watchdog module monitors all primary hardware

[1]Written permission to use this publicly available software package was obtained from DARPA within the applicable deadline.

and software components for possible malfunctioning. It power-cycles hardware components and restarts software modules when necessary. As a result, the system can survive failures of individual modules and system components.

7 Vehicle Safety

Safety has been of utmost importance in the design of the vehicle system.

E-stop pausing is handled through Stanley's software system. When a pause command is issued, the FSA directs the vehicle to come to a prompt stop and shifts the vehicle into park until a run command is issued.

The disable command is connected to the vehicle engine control, bypassing Stanley's computing pipeline. A disable command results in brake actuation and a prompt shutdown of the engine. By directly connecting the disable mechanism to the Touareg engine system, malfunctioning of the computer pipeline cannot affect the functioning of this essential safety feature.

The vehicle is equipped with a siren and a strobe that fully comply with the regulations stated in the 2005 DARPA Grand Challenge Rule document. The vehicle is also equipped with two latching E-stop buttons.

Despite these modifications, Stanley remains fully street legal and can be operated manually. Switches mounted near the driver console enable a human operator to seamlessly transition between manual and computer-controlled operation, even while the vehicle is in motion. While this feature is not necessary for the actual Grand Challenge event, it ensures the safety of vehicle occupants during testing.

8 System Tests

Testing has played a major role in the development of the Stanford Racing Team robot Stanley. Primary testing areas include terrain in the Mojave desert, including parts of the 2004 DARPA Grand Challenge Course, a vehicle testing facility in Arizona and nearby public lands, and local terrain at and near Stanford University.

In the initial months from December 1, 2004, to July 28, 2005, testing took place within month-long development cycles that combined three weeks of core development with a week-long testing event in the Mojave desert. Since the beginning of August 2005, the system is being tested full time in Arizona.

From the very beginning of this project, the team pursued a sequence of milestones, most of which were met. The major milestones were as follows:

- December 1, 2004: First fully autonomous desert mile (achieved: December 1, 2004; the vehicle traversed the first 8.5 miles of the original 2004 DGC course before the autonomous run had to be terminated).

- February 1, 2005: Waypoint navigation at 35mph (achieved: February 13, 2005; the vehicle reached a top speed of 42mph).

- April 1, 2005: Five autonomous miles at an average speed of 25mph with full collision avoidance (achieved April 11, 2005, along an easy section of the 2004 DGC course).

- May 10, 2005: DARPA Site visit, which led to the selection of the team as one of the 40 semi-finalists.

- July 1, 2005: Autonomous traversal of the entire 2004 DARPA Grand Challenge Course, with the exception of public roads (partially achieved July 16, 2005; the team encountered a total of six failures, each at a level that would have been fatal in a actual race).

- August 15, 2005: 140 uninterrupted autonomous miles through unpaved terrain (achieved August 20, 2005, at an average velocity of just over 22mph).

Some of the testing is performed through a dedicated vehicle testing group. Since August 20 the emphasis has been on endurance testing of the integrated end-to-end system in realistic desert terrain.

9 Contact

Please direct all inquiries to the following address:

Stanford Racing Team, c/o Michael Montemerlo
Stanford Artificial Intelligence Laboratory
Stanford, CA 94305-9010
Email: srt@cs.stanford.edu
Web: www.stanfordracing.org

References

[1] S. Julier and J. Uhlmann. A new extension of the Kalman filter to nonlinear systems. In *International Symposium on Aerospace/Defense Sensing, Simulate and Controls*, Orlando, FL, 1997.

[2] R. Simmons and D. Apfelbaum. A task description language for robot control. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Victoria, CA, 1998.

[3] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.