AC 2009-608: THE ROBOT RACER CAPSTONE PROJECT

James Archibald, Brigham Young University

James K. Archibald received the B.S. degree (summa cum laude) in mathematics from Brigham Young University, Provo, UT, in 1981, and the M.S. and Ph.D. degrees in computer science from the University of Washington, Seattle, in 1983 and 1987, respectively. Since 1987, he has been with the Department of Electrical and Computer Engineering, Brigham Young University. His current research interests include robotics and multiagent systems. Dr. Archibald is a member of the Association for Computing Machinery and the Phi Kappa Phi.

Doran Wilde, Brigham Young University

Dr. Wilde started his career as an electrical engineer in Oregon where he worked 12 years in industry doing microprocessor design. In 1990, he began graduate studies at Oregon State University, where he received his M.S. and Ph.D. degrees in Computer Science in 1993 and 1995, respectively. In 1995, he joined the faculty in the department of Electrical and Computer Engineering at Brigham Young University where he is currently serving as an associate professor. Dr. Wilde has taught a wide range of computer and electrical engineering courses and has been involved in new computer engineering course development. He has been actively engaged in research in the fields of computer arithmetic, application specific systems and architectures, and autonomous vehicles. Dr. Wilde is a senior member of the IEEE and is the father of seven children and eight grandchildren.

The Robot Racer Capstone Project

Doran Wilde, James Archibald
Brigham Young University

Abstract

This article describes a senior design project based on small vision guided autonomous vehicles that satisfies the longtime ABET requirement of a culminating design experience. The design and development of autonomous robots is well suited to capstone design projects because of the technical challenges in creating functional vision, control, and communication subsystems and integrating them into a working whole. The scope of this project is large enough that it requires a multidisciplinary team since no individual student can complete it working alone. Participants receive technical guidance as well as timely instruction on teamwork and project management. We describe how the Robot Racer project and the associated course are structured, detail the resources that are required, and present results that suggest that this culminating project is particularly effective in preparing our Electrical and Computer Engineering students for engineering practice and in giving students confidence in the application of their engineering knowledge.

Introduction

For several years, graduate students, engineering faculty, and selected undergraduates at Brigham Young University have been involved in projects developing new technology for small, autonomous vehicles, both ground- and air-based. Since student interest in robotics is widespread, and since the communication, vision, and control sub-systems for these vehicles are based on principles addressed in our undergraduate program, it was natural to create additional opportunities to expose our undergraduate students to the unique and challenging technical problems arising in this application area. In 2006, we began a new senior capstone project that builds on past successes with other robot-based senior projects, including a Robot Soccer project in which student teams built and programmed small robots to play soccer [1,2], and an Intelligent Ground Vehicle Competition project in which a robotic vehicle was autonomously driven through an obstacle course [3,4]. The new Robot Racer project focuses on designing vision and control modules so that an off-the-shelf RC vehicle can drive itself through a visually marked course using on-board vision processing. The project poses technical problems for our senior students that are both challenging and open-ended. For faculty instructors and advisors, the challenge is to provide enough structure that student teams are able to experience success with an expenditure of effort commensurate with the credit hours of the course they enroll in for the project.

In the next section, we describe the framework shared by all of the senior design projects in the department. We then discuss the rules of the robotic competition that determines the design goals and timeline for each participating team. That is followed by a discussion of our experiences in offering this class, including practical challenges and an assessment of how well the project meets the associated learning objectives.

Senior Project Framework

The Robot Racers project is one of several projects in which students in the electrical and computer engineering department can choose to participate. Examples of other projects (past and present) include: autonomous quad-rotor helicopters, ad hoc sensor networks, software-defined radios, free-space optical communication, and network-enabled MP3/MPEG-II players on field-programmable gate arrays (FPGAs). Each project is offered as a one-semester four credit-hour class, and students enroll in the section associated with the technical project they select. Students in all projects attend shared twice-weekly project management lectures that teach a systematic design process and teamwork skills. Students also meet for two additional hours each week with their faculty advisors for project-specific technical lectures and discussion.

In combination, the project management and technical components satisfy the long-term ABET requirement of a "major design experience based on the knowledge and skills acquired in earlier course work and incorporating appropriate engineering standards and multiple realistic constraints" [5]. The technical challenges inherent in each project are substantial; constraints arising from the physical characteristics of the operating environment combine with limitations on time and resources to create a design experience that far exceeds the scope and scale of design projects completed by students elsewhere in the curriculum. Each project lists a set of junior- or senior-level courses as prerequisites – each participant must have taken at least one of the listed courses, and teams are formed in such a way that each group has all essential competencies.

In addition to meeting the ABET requirement for a major design experience, our senior projects are to satisfy a variety of observable and measurable outcomes defined by the department. These are stated in terms of competencies that students should demonstrate in the course of completing assignments associated with either the technical or the project management tracks of the course. These are listed below.

Technical Track Competencies:

- Ability to take high level, customer requirements, identify from them the specifications a solution must meet, and formulate those specifications as an engineering problem.
- Ability to complete a significant culminating design that draws upon the knowledge gained in prerequisite courses.
- Ability to maintain a team portfolio documenting the design process.
- Ability to give an oral presentation describing team progress during a design review.
- Ability to produce a written report describing a technical design.
- Ability to give an oral presentation describing a technical design.

Project Management Track Competencies:

- Ability to establish and follow ground rules governing team behavior.

- Ability to understand the expectations from other members on a team.
- Ability to assign and accept leadership responsibilities on a team.
- Ability to define, understand and participate in team decision-making processes.
- Ability to understand the different skill sets each team member brings to a team.
- Ability to understand the expectations (regarding participation) of each team member.
- Ability to produce and maintain the necessary control documents on a design team.
- An understanding of the principles of leadership and project management.

Teams are formed the very first day of class, and they immediately begin to complete a sequence of assignments that apply principles of project management in solving the challenges faced in their own design. Students are taught a structured process for capturing customer needs, and each team produces a formal Functional Specification Document that maps customer needs to product specifications. They are also taught an iterative process of generating and evaluating concepts to be employed in their design, and teams create a Concept Generation and Evaluation Document that explains and supports their design decisions. Teams also produce a Project Schedule that requires them to define critical milestones, identify task dependencies, and assign people to tasks in an effective manner. Together, these three documents constitute the portfolio that documents the ongoing design experience of each team. (Near the end of the semester, a final report is added to the portfolio that describes their completed design.) In many projects, teams are required to create web pages that include the documents from their portfolios and project-specific design artifacts (e.g., software, drawings, and photographs). Team web sites are made publicly accessible at the end of the semester, and they constitute an important source of information for students participating in the same or similar projects in subsequent years.

Interim design reviews are scheduled for each team in the fifth and tenth weeks of the semester, and final design reviews are held at or near the end of the semester. Prior to the first design reviews, students receive instruction in the Project Management track on effective presentations. In each design review, the team is expected to present the key features of their design, summarize their progress, identify unsolved problems, and outline their schedule for future work. The audience for design reviews consists of instructors, teaching assistants, and, when they are able to attend, representatives from industrial sponsors. Typically, teams receive on-the-spot feedback on their design plans as well as their presentation. In projects based on design competitions, teams are not allowed to attend each other's design reviews so that creative innovations and solutions will not be publicly divulged.

Competencies associated with the Project Management track are accessed by team and individual assignments within that portion of the course. Underlying principles of teamwork and leadership are presented and discussed early in the semester, and students have ample opportunities to apply and explore these principles in their work together as a team over the semester. With few exceptions, all members of a team receive the same grade as determined by the overall team performance, so students are highly motivated to see the team succeed.

New projects come into existence and old projects are discontinued based on the interests and availability of faculty members. In practice, the department offers a wide variety of senior projects at each point in time, although the mix is constantly changing. Projects are often collaborative undertakings involving multiple faculty members, and some projects are formulated or defined in part by industrial partners who wish to see students develop knowledge and experience in a particular technical area. Corporate sponsors often provide funding or special resources and equipment that make a particular project possible.

The flexibility of our senior project framework is demonstrated by its ability to accommodate special projects when unusual opportunities arise. For example, one project was initiated by students wanting to leverage the interest and resources of a local company they had interned for; they explored the performance of prototype infrastructure supporting wireless internet access on local buses and light rail. A second project focused on creating a university-level entry in the DARPA Urban Challenge; external sponsors donated a Chrysler minivan, a drive-by-wire kit for the vehicle, and money for cameras, laser range finders and computers. Students were divided into groups to focus on real-time vision, control, and an intelligent agent for driving in traffic.

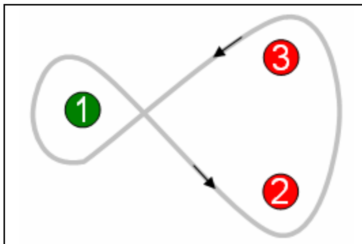
The Robot Racers Competition

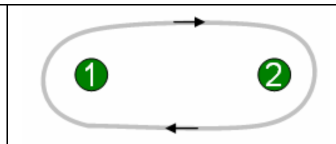
The semester-long assignments associated with the Robot Racers project focus on preparing teams for a final competition near the end of the semester in a public setting. The competition is held in conjunction with demonstrations and presentations from other senior projects, and the event often attracts a considerable audience of students, visitors, and media. Although most attendees do not fully understand the technical challenges involved, the concept of autonomous vehicles is intuitive and accessible, and the competition can be quite entertaining. The desire to perform well before the anticipated crowd – often including representatives from local news media – provides additional motivation to the students throughout the semester. Without fail, students learn important lessons in project management in their efforts to have their designs completed and implemented by the hard deadline of the final competition.

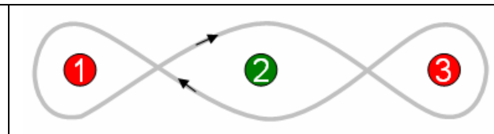
In the current format, competing vehicles are timed as they traverse a racecourse marked by vertical pylons of one of two solid colors. During competitive runs, vehicles must operate autonomously using only onboard equipment. A wireless link to a remote "basestation" computer is required, but only START and STOP commands can be transmitted to the vehicle during a competitive run. Between runs, teams can transmit anything they wish from their basestation to their vehicle to facilitate debugging or reconfiguration of their onboard systems. There are no restrictions on information transmitted from the vehicle back to the remote computer before, during, or after competitive runs. Teams often stream vision and state information from their vehicles to the basestation computer to provide visibility into the actions of the onboard control and vision systems.

Each racecourse is determined by the physical placement of pylons and a course description file – taken in combination. By convention, vehicles turn clockwise around pylons of one color (represented by green in the figures below) and counterclockwise around pylons of the other color (red in the figures). The course description file specifies a sequence of heading changes; each such change is the number of degrees that the vehicle must turn at the current pylon to face the next pylon, assuming it has approached the current pylon on a direct line from the previous one. The course description is thus equivalent to a sequence of line segments connecting the pylons in a specific order. Clearly, the vehicles cannot drive that path of linear segments precisely – they would collide with each pylon and they cannot execute turns instantaneously – so steering and navigation code must make appropriate compensations as the vehicles steer around each pylon.

Some examples of simple courses with their course description files are shown in the figure below. Note that heading changes are always positive for “red” pylons and negative for “green” pylons. By convention, each course is set up as a loop, so that the course description may be repeated from the beginning when the end is reached. Initially, the truck is placed facing the first pylon from the direction of the last pylon.







–240 120 120	–180 –180	180 –0 180 –0
a. An asymmetric figure 8	b. A clockwise loop	c. A double figure 8

As can be seen, any of a wide variety of interesting courses can be set up quickly and easily by placing a small number of pylons and creating a short course description. (In practice, a reasonable approximation of turning angles is sufficient.)

Courses must be constructed in such a way that the next pylon can be identified without ambiguity. In courses used for competition, the next pylon will always be the closest of the specified color (known by the sign of the heading angle) near the center of the image, assuming that the vehicle has made the appropriate heading change. Furthermore, maximum and minimum distances between pylons are agreed upon in advance; teams must consider these factors in selecting their turning radius and their algorithm to identify pylons in captured images.

Robot racers in the 2008 slalom style course

Our most common competition format is “time trials” with one vehicle on the track at a time racing against the clock. Teams take turns making runs in a round-robin fashion, and racing continues until an agreed-upon period of time has elapsed. The best time recorded by each team is recorded and used to determine the winner. When one team finishes a run – by successfully completing the course or by failing to navigate correctly – the next team has a maximum of 20 seconds to get their vehicle on the starting line and ready to run. If the time limit is exceeded, the turn passes to the next team in order. Not infrequently, teams will “pass” their turn directly to the next team when they are making a change to the vehicle or their software that cannot be completed in the available time. In general, this format allows every team to demonstrate its best performance on the course while minimizing the impact of a bad run.

As an alternative, in the crowd-pleasing “pursuit” format, two vehicles race on the same course at the same time; one starts at the beginning and the other starts approximately halfway around the loop (with an appropriately modified course-description file). Racing continues until one vehicle catches the other, or until one vehicle fails to navigate the racecourse correctly. The overall winner can be determined by a single- or double-elimination bracket involving all teams.

The “do or die” format is intended as a stress test for the navigation and control, and it rewards those designs and implementations that are the most robust. In contrast with the time trial format where teams typically fine-tune parameters between runs (increasing or decreasing their baseline driving speed, for example), this format prohibits system modifications between runs. Systems are configured and parameters are fixed, then the racecourse is set up and the course description is given to all teams, and then each team is permitted a small number of runs. The winning team completes the course in the shortest time, or comes closest to completing the course if no vehicle manages the entire course. Since this format emphasizes important aspects of the system that are critical in actual robotic applications, our final competition includes results from this format as a substantial component.

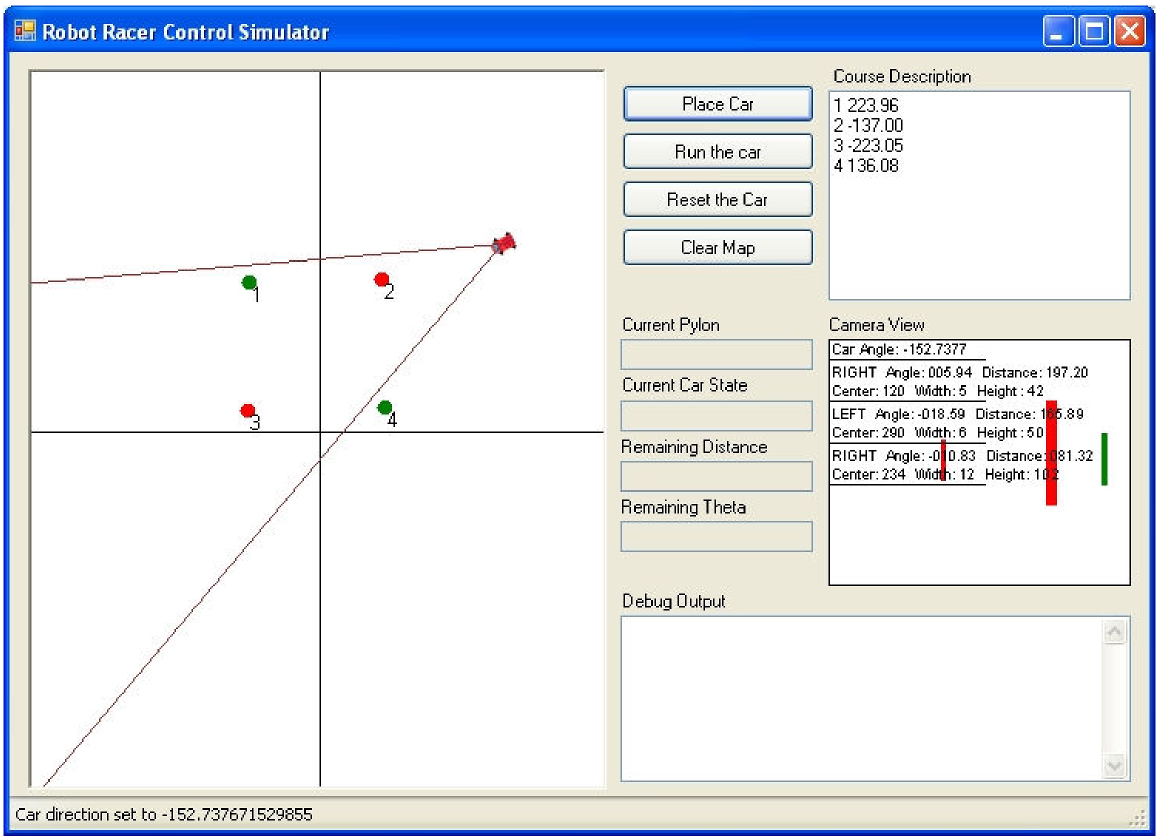
a. The 2006 track   b. Student placing his car on the track   c. A truck jumping off of a ramp

The pylon-marked racecourse format was chosen specifically to make it easy to set up and modify courses for testing. In our first year of this project (2006), red and green garden hoses were used to mark the side boundaries of a closed track (see figure above) and setting up and modifying courses was much more cumbersome. In that format, the vehicles had to keep at least one wheel in the track while avoiding cones and other vehicles in the track – both marked with orange – and navigating blue jumps in straight segments of the course. The change from continuous hoses to simple pylons makes it possible to do much more extensive testing in the limited space that is dedicated to this project; in our experience, the teams that do the most on-course testing tend to achieve the best overall performance.

As the vision and navigation capabilities of our teams increase over time, we anticipate evolving the competition so that it will continue to present a significant design challenge. In the next few years, we plan to add a “head-to-head” racing format in which the vehicles start side by side and overtake each other while avoiding collisions. When it is technically feasible, we will add fixed obstacles and jumps to the racecourse to further test the sensing and driving capabilities of each team, as well as making the competition more enjoyable to watch. In the long term, we hope to focus on applications that require exploration and mapping of unknown environments, such as autonomous search and rescue.

The Robot Control Simulator

In previous years, the development of the vehicle controller was handicapped until the vision system was working. This usually meant that control was done hurriedly in the last few weeks of the semester and controllers were not as well developed as they should have been. To help the student teams make progress on their controllers early in the semester, in 2009 we provided the students a control simulator that allowed control algorithms to be developed in parallel with the vision algorithms. Below is shown the users interface of the simulator.



The robot control simulator user interface

The controller was written in C sharp and included only stub code for the control module. The simulator provides the capability of placing pylons and the car on the field using the mouse, generation of the course description file, simulation of the movement of the car, and simulation of the scene that the vehicle-mounted camera sees. The students must write the controller that inputs information about the positions of the pylons in the simulated camera view and produces the desired velocity and turning angle for the car.

The introduction of the simulator had the desired effect and vehicle controllers were developed concurrently with the vision systems. Another positive, but unintended outcome was that the simulator established a de facto interface standard between the vision system and the controller that most teams have adopted.

Class Structure and Schedule

Senior projects are based on knowledge learned in junior and senior engineering courses. Prerequisite courses for the Robot Racer project include courses in real-time operating systems, embedded systems, control systems, C/C++ programming, and VHDL design. Each student enrolling in the associated senior project course must have competencies in one or more of these subjects. An explicit objective of the project has been to incorporate a wide variety of topics from several undergraduate classes so that students must work together in multidisciplinary teams to successfully complete the project. Fortunately, given the nature of robotics projects, this is easily accomplished.

The Robot Racer senior project course is offered once a year as a one-semester four credit-hour class. As noted above, students enrolled in the course meet with students from all other senior projects offered the same semester for twice-weekly project management lectures that teach general business practices, teamwork skills and a design process methodology. Robot Racer students also meet for two hours each week for technical discussions that give them technical guidance and that help them to become productive in their design efforts from the very first week of the semester. Faculty advisors and teaching assistants present technical information on vision and control algorithms and suggest different approaches that the teams should investigate. By the end of the first four weeks, all technical lectures have all been given, and the lab time is used for team meetings, team design reviews and practice competitions.

As the first order of business, students are grouped in teams of four or five students. From then on, all project assignments are done as a team. Each team organizes itself, chooses a team leader and team name, and partitions tasks among team members according to their interests and competencies. Each team creates a team web page for project communication and management. The Trac open-source project management system [6] is used by all teams. Trac is an enhanced wiki and issue tracking system that also includes an interface to the Subversion [7] file archiving and version control system. These web pages are password-protected during the semester so teams can keep their technology and design innovations private. At the end of the semester, the web pages are made publicly accessible and they can be used by teams in subsequent years.

In order to focus the teams and prepare them for competition, a set of specific milestones with due dates are assigned. In the first week, teams are formed. In the second week, teams must show a basic competency with the Xilinx EDK tool that configures the hardware and software on the truck. In the third week, they show that they can capture images from the on-board camera. In the fourth week, each team demonstrates a base station program and human interface that can communicate wirelessly to the car. In the fifth week, they demonstrate a working vehicle control algorithm using a simulation environment that is provided for them. In the sixth week, they demonstrate that they can color segment a video image to find the colored pylons. In the seventh week, they show that they have integrated their vision and control systems on their truck, and that it can drive a simple pylon course autonomously. Six weeks after that is the final competition.

In the interim, weekly practice competitions are held so teams can see what aspects of their designs and implementations need to be improved.

An innovation added in 2009 was a mid-semester Image Processing Competition, in which teams tested their computer vision algorithms against a challenging suite of images with pylons. (Examples are shown in the figure below.) Submitted programs analyzed each image and output – for the closest pylon of each color – the pixel coordinates of the pylon, its height and width, and the computed distance to the pylon. The output was compared with ground truth and scored for its overall accuracy. The emphasis for this competition was the vision algorithm rather than its implementation on the target platform, so any software package (e.g., OpenCV, Matlab) could be used. In its first year, the team with the highest score with just over one month to go to the final competition was treated to pizza at a local restaurant. As intended, the competition was successful in educating teams about the difficulty of devising an algorithm that correctly identifies pylons with varying lighting conditions and backgrounds, and it resulted in a significant improvement in the robustness and reliability of vision techniques employed on the vehicles.

Test Images for the Image Competition

Required Resources

Each team is provided with a complete and operational truck assembly, along with all resources required for the project. The total cost of provided equipment is about $2000 per team, not including labor or assembly costs. The specific hardware we use is summarized in the table below.

Hardware	Manufacture/Model	Description
RC Truck	Traxxas Stampede XL-5	1/10 Scale 2WD monster truck
Camera	Micron MT9V111 mounted on a custom PC board	¼ inch SOC VGA CMOS active-pixel digital image sensor
Speed Controller	DuraTrax IntelliSpeed DTXM1065-16T	7.2 to 8.3 volt DC proportional electronic speed controller
Encoder	US Digital S4 Encoder	Miniature optical shaft encoder
Wireless Transceiver	Aerocomm AC4490 Transceiver Module	900 MHz transceiver with serial interface 1200 to 15200 baud
Processor	BYU Helios processor board [8]	65mm x 90mm, 37g, 100 MHz Xilinx Virtex-4 FX20 w/ onboard PowerPC processor, 32 MByte SRAM, 1 MByte video buffer, 16 MByte Flash, USB 2.0 port, serial port, extension header.
Truck Interface	Custom "ground-based vehicle" daughter board	Digital compass, 2 camera ports, encoder port, servo port, wireless transceiver port.

In addition to the equipment listed above, teams are provided with two or three sets of batteries and a battery charger. Each team is responsible for the equipment provided and must strictly observe safety procedures for working with sensitive electronic equipment, particularly pertaining to electrostatic discharge protection. Hardware protecting the camera and the custom processor board must be in place each time the vehicle is operated.

Because the computational demands of real-time vision processing can easily exceed the capabilities of low-power embedded processors, providing sufficient processing power on-board small vehicles is a challenge. Critical to the success of our project is the custom Helios processor board [8] that uses a resource-rich FPGA (field programmable gate array) chip that includes a PowerPC processor in the FPGA fabric. The capabilities of the Helios board are extended by adding daughter boards (plugging into the extension header) that include application-specific sensors, actuators, and interfaces. With this crucial resource, teams can use a combination of custom software and hardware modules to achieve their target performance. For example, teams can implement edge detection or segmentation algorithms entirely in hardware, making results for an image available just a few cycles after the final pixels are obtained from the camera.

Each team is also provided with a computer preloaded with the Xilinx Embedded Development Kit (EDK) [9], along with some prewritten base packages to interface with communication ports, the custom camera board, and with the custom “ground-based vehicle” daughter board.

Open Ended Problems and Technical Challenges

There are many difficult and open-ended problems that students in the Robot Racers project must solve. Some of these are listed below.

- **Safety**: how do you regain control of an errant robot?
- **Vision**: how do you identify the type and position of a pylon or an obstacle from images captured by the on-board camera?
- **Vision**: how do you adapt to changes in ambient lighting?
- **Navigation**: how do you determine the desired path around a pylon and navigate the course?
- **Control**: how do you determine the desired speed and steering angle of the truck from the available sensor information?
- **Embedded Systems**: how do you perform an operation when a computer program running on the truck cannot do it in real time?

Of course, a variety of unanticipated technical problems and challenges such as equipment problems, hardware failures, and software bugs always seem to accompany a design project of this complexity. Solving these types of problems gives students invaluable real-world experience.

Results and Assessment

The evidence suggests that the overall senior project framework employed in our department is effective in accomplishing its intended objectives. Prior to our adopting this framework in 2000, students proposed their own senior projects, obtained the necessary equipment and materials, and found a faculty member to advise them. There were many problems with this approach. Students were often overly ambitious in defining their own projects and many were not successful in achieving their design objectives. Because most projects involved just one or two students, faculty resources were used inefficiently. Most critically, the majority of projects involved little teamwork and did not clearly meet the ABET requirement for a major design experience with multiple realistic constraints. The present framework makes better use of department resources, provides more opportunities for undergraduate mentoring, exposes students to more cutting-edge technical problems, and gives them valuable experience with a structured design and development process. Our department has undergone ABET accreditation reviews twice since adopting this framework, and both times evaluators complimented us on the strength of our senior project program.

The evidence also suggests that the Robot Racers project is effective in achieving both formally stated objectives and personal goals of the faculty advisors, including exposing students to intellectual challenges inherent in robotics, as well as attracting public attention to the discipline of engineering. First, we note that the project continues to be very popular with students, although they have many other alternatives. Of the 8 senior projects offered this semester (an unusually high number, 5 is more typical), the Robot Racers project has the highest enrollment with 29% of all senior project students. Over the past 4 years, 94 students have enrolled in this project course, or its predecessor. This total represents approximately one quarter of the graduates of our department over that period.

Second, the final competition associated with the project consistently attracts a substantial number of spectators, including potential students and the general public. We treat the event as a showcase of what engineers do presented in a way that everyone can understand and appreciate. Our industrial sponsors find this aspect of our project particularly appealing, as they also wish to enhance the public perception of the engineering profession. Local newspapers and TV stations often cover the competition, further extending the scope of its influence. We illustrate with a short exchange that took place on the local NBC affiliate at the end of a reporter's story – including over two minutes of video and interviews with participating students – about our senior project competition in 2008.

- Anchor 1: "Wow Sam, we're all reflecting on our college senior projects…"
- Anchor 2: "All the liberal arts graduates, huh? Go into science kids! That was impressive."
- Weatherman: "I can barely drive my own car, let alone have a car drive itself."

Third, the project consistently has a positive impact on the students who complete it. They report that job interviews take on a very different tone when they bring up their participation in this senior project. Interviewers ask about the technical design challenges, the experience of working together as a team, and the effort required to complete a demanding project with hard deadlines; all are topics that Robot Racer participants can talk about enthusiastically and authoritatively. We have also seen several students elect to pursue graduate study in topics related directly to the project, both at our institution and elsewhere in the US.

Conclusion

We have described a senior project framework that has been effective in providing students with an exposure to both technical topics of interest and a structured design and development process. The project is popular with students and industrial sponsors, and it has received strong approval from our ABET evaluators during two accreditation cycles. We have also described the Robot Racer project, a technical project within that framework that focuses on real-time vision and control for a small ground-based autonomous vehicle. We believe this project achieves a good balance between the desire to expose students to cutting-edge problems and technology, and the desire to have students experience success in solving a difficult design problem. The project is multidisciplinary and provides a good team experience in which each student can play an important role as a team member. The project continues to evolve, and we believe it will continue to play an important part in our curriculum for many years.

References

1. Archibald, James K. and Randal W. Beard, *Goal! Robot Soccer for Undergraduate Students*, IEEE Robotics and Automation Magazine, Vol. 11, No. 1, pp. 70-75, March 2004.
2. Archibald, James K. and Randal W. Beard, *Competitive Robot Soccer: a Design Experience for Undergraduate Students*, 32nd ASEE/IEEE Frontiers in Education Conference, Boston, MA, pp. F3D-14 to F3D-19, November 2002.
3. Archibald, C., E. Millar, J. D. Anderson, J. K. Archibald, and D. J. Lee, *A Simple Approach to a Vision-Guided Unmanned Vehicle*, SPIE Optics East, Intelligent Robots and Computer Vision XXIII: Algorithms, Techniques, and Active Vision, 60060J, Oct. 23-26, 2005.
4. Tippetts, B.B., K. D. Lillywhite, S. G. Fowers, A. W. Dennis, C. R. Greco, D. J. Lee, and J. K. Archibald, *A Simple, Inexpensive, and Effective Implementation of a Vision Guided Autonomous Robot*, SPIE Optics East, Intelligent Robots and Computer Vision XXIV: Algorithms, Techniques, and Active Vision, 63840P, Oct. 2006.
5. ABET, *Criteria for Accrediting Engineering Programs*. Available: http://www.abet.org
6. Edgewall Software, "Trac Integrated SCM & Project Management", http://trac.edgewall.org/
7. Tigris.org, "Subversion, an open-source version control system", http://subversion.tigris.org/
8. Fife, W. S. and J. K. Archibald, *Reconfigurable On-board Vision Processing for Small Autonomous Vehicles,* EURASIP Journal on Embedded Systems, Vol. 2007, Article ID 80141, 14 pages, 2007
9. Xilinx Embedded Development Kit (EDK), http://www.xilinx.com/ise/embedded_design_prod/platform_studio.htm