14th Annual Intelligent Ground Vehicle Competition

Stereo Opticon

University of Missouri-Rolla

Robotics Competition Team

2006 Design Report

1.0 Overview

Stereo Opticon is the second generation three-wheeled, omni-directional robot entered by the University of Missouri-Rolla (UMR) into the Intelligent Ground Vehicle Competition (IGVC). The design aims to minimize cost, weight and size while maintaining a high level of performance on the course. Light-weight aluminum, low power electronics, stereovision cameras and an intelligent control system give this robot a competitive edge.

2.0 Innovations

The team decided early in the design phase to augment innovation with traditional, proven technology. Stereo Opticon is smaller and lighter than its predecessor, Optical Prime, but benefits from many of the same design innovations. The stereovision cameras, the tripod design, the hierarchical control design and the artificial intelligence system allow Stereo Opticon to turn in any direction smoothly, recognize relevant objects and efficiently navigate its environment. An AutoCAD render of the robot can be seen in Figure 1.

Figure 1: AutoCAD Render of Stereo Opticon

3.0 Design Process

Stereo Opticon was designed based on the 2005 entry into the IGVC but includes significant enhancements. Following the process in Figure 2 the team was able to envision the current robot.

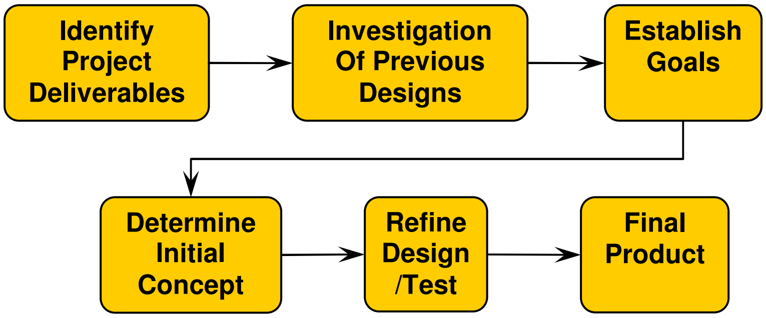

Figure 2: Design Process

Given that the rules for the competition are stated very specifically, it was easy to determine the project deliverables. For the primary competition the robot must be able to find its way through an obstacle course composed of construction warning objects (barrels, netting and cones) while staying between two white lines. The same robot must be able to run through a navigation challenge that is similar to the traveling salesman problem.

The team analyzed last year's design before commencing work on the current robot. The major limitations of the older model were found in the motors and control system, but it was also found that some software systems could be optimized and reorganized. In addition to this analysis, the team studied the successful robots from the 2005 IGVC, and created a list of possible improvements for the 2006 design.

This list of improvements was compiled into a statement of goals to make the new robot fit for the competition. The major goals were to minimize the size and weight, engineer an efficient control system and optimize the software from 2005.

Based on these goals, an initial concept was formed. Many different designs were considered, including changing the basic shape of the robot to a more traditional four wheeled design or a novel ball drive system. After careful evaluation of these design concepts, the team decided that a three-wheeled robot would work best.

Most of the design was implemented and tested first on the 2005 robot. Since it was already built and had similar systems, the control and vision issues could be resolved there. When it was time to build the new robot most components were already tested and evaluated.

4.0 Team Organization

The team consists of members from nine different majors including mechanical, electrical and computer engineering. The team is divided into a group responsible for the mechanical and structural components of the robot and a group responsible for the intelligence and controls of the robot. Each group has a project manager who is responsible for managing their section and reporting progress to the Vice President of the team. A group of officers is responsible for business and administration of the team as a whole. See Figure 3 for the hierarchy of administration. See Appendix A for a full list of team members. Both robots entered by UMR this year were designed and built by the same team.

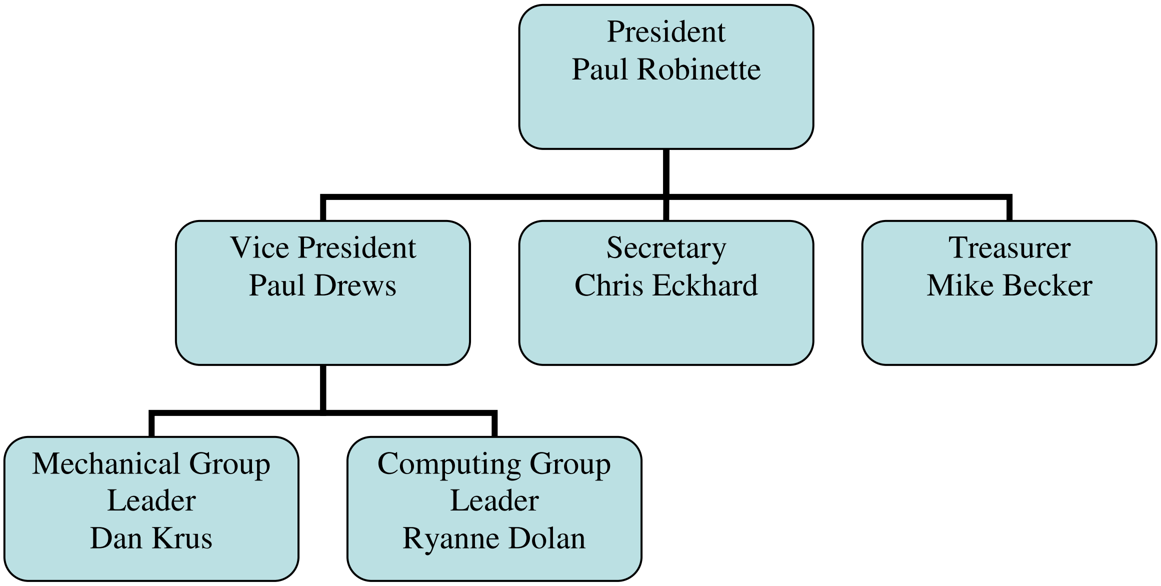

Figure 3: Organizational Chart

5.0 Mechanical Design

The basic three wheeled design developed for the 2005 IGVC maximizes stability while minimizing materials. Figure 4 shows an AutoCAD render of the 2005 robot used as a starting place for the design. This design was improved upon for the 2006 IGVC by replacing the steel tubing and lexan sheets with aluminum tubing and plates. The weight ratio of aluminum to steel is 1:2.9 while the strength ratio of aluminum to steel is about 1:1.3, making aluminum a much more weight efficient material while only costing about 20% more than steel.

The robot was also made smaller than the 2005 design so that it would have less trouble avoiding obstacles. The structural and mechanical components were designed in AutoCAD which allowed the team to make sure that the design was complete before manufacturing. The chassis is still hexagonal in shape as seen in Figure 5, but it was designed to transfer the bending and shearing forces into tension and compression forces to take advantage of the aluminum tubing's natural strengths.

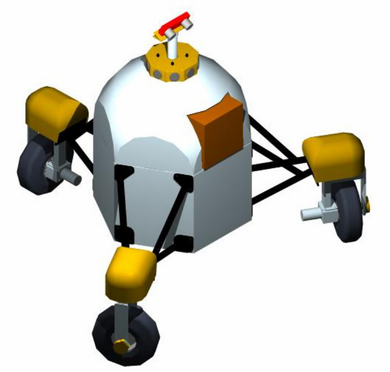

Figure 4: AutoCAD Render of 2005 Design

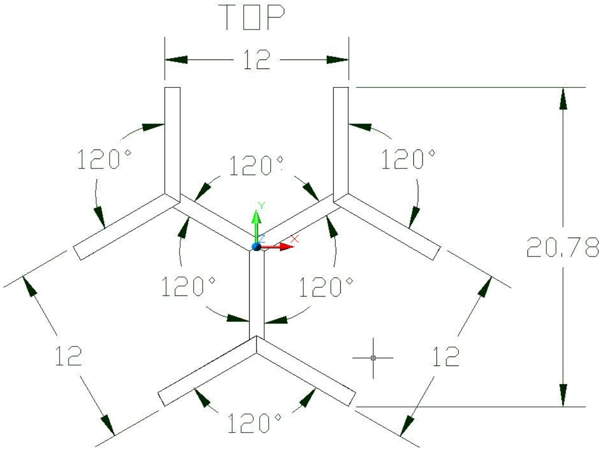

Figure 5: The Chassis

The robot is driven by three NPC 41250 gear motors mounted directly to the wheels. In deciding the motors, the team set goals for the motion of the robot including that it should be able to accelerate to 5 mph in 1 meter, and the robot should have a maximum speed of 5 mph. The ideal specifications for a motor that would meet these parameters would have a torque of 6.97 N-m, power of 123 watts and a maximum rotational speed of 17.6 rad/s. The NPC 41250 motors apply 6.79 N-m of torque, 117.3 watts of power and have a maximum rotational speed of 18.22 rad/s, which would result in a movement speed of 5.2 mph. The motors run at 24 volts, which will help them accelerate at 30% potential up a 15% gradient. The movement of the wheel is monitored by an optical encoder mounted on the motor axle which reports its data to the control system. Each wheel is turned by a Crouzet 808550 motor and the heading of the wheel is determined by a one turn potentiometer mounted to the rotation shaft. The wheel is limited to turning 270º in either direction so that the power and control wires do not tangle.

Each wheel pod is identical and can be removed efficiently. This allows the robot to be disassembled and reassembled quickly and easily. This also allows easy access to the components inside the robot and makes fabricating replacement parts simple.

6.0 Electrical System

All electrical components of the robot must receive their power in the correct voltage from a safe electrical system. The electrical distribution system is fairly simple and very safe. The batteries output 24 volts to the system, which converts to 5 volts for electronics, 18 volts for the computer and sends raw 24 volts to the steering and drive motors. Each of the three drive motors draw between 4 and 15 amps, while each of the three steering motors draws between 3 and 6.5 amps. The computer draws 3.6 amps, and the power draw from other electronics including the camera and encoders is negligible. The total minimum power draw of the whole system is 18.6 amps and 461 watts, while the maximum power draw is 53.6 amps and 1,265 watts.

Stereo Opticon uses 18 amp-hr, 12 volt, 13 lb lead acid batteries. During testing, two batteries are stored in the lower battery drawer, and two batteries are stored in the package spot resulting in 36 amp-hrs at 24 volts. This gives the robot over an hour of run time. During competition, only two batteries are stored in the battery drawer, giving the robot 18 amp-hrs at 24 volts, resulting in over half of an hour of run time. There is a 40 amp fuse for each motor and a 30 amp fuse for the whole electronics system and the computer. There is a switch just after the batteries so that the whole system can be turned off at the touch of a button.

7.0 Sensors

Sensors must be used by the robot to detect its environment. To keep the overall cost of the robot low a minimal amount of sensors were used. Stereovision cameras were chosen over a LIDAR system due to their low cost and ease of use. A LIDAR system would still require a vision component to detect the lines, so the problem was actually simplified by using computer vision. The Videre stereovision camera was determined to be the best fit for this competition since it can give accurate distance data up to 15 feet. The camera has trouble detecting objects within 2 feet of the robot, so low cost infrared rangefinders are used. Sonar sensors were implemented in the 2005 robot but in the redesign they were determined to be redundant.

Figure 6: Videre Stereovison Camera

The position of the robot is determined by the GPS receiver. If necessary, the wheel encoders can be used in conjunction with the digital compass to track the movement of the robot when a GPS fix in unavailable. When the robot starts in the Navigation Challenge it can use the digital compass to determine which heading it should turn to before the GPS has a chance to receive accurate data. The GPS is a Garmin GPS 18 5 Hz unit which has WAAS differential capability so that it can give the position within 1.5 meters. This should be accurate enough for the competition.

8.0 Software Design

Four main principles were used when designing the software for Stereo Opticon: modularity, rapid development, prototyping and hierarchical design. The software was designed to be modular so that each component could be developed and tested independently before the whole system was integrated. See Figure 7 for a diagram of the software. Rapid development was necessary given the one year design-build-compete schedule. Development time was reduced by using the simple syntax of interpreted languages (mostly Lua). High level logic was used to prototype algorithms quickly in simulation and in the field. Hierarchical design is necessary for a robot of this complexity; this can be seen in Figure 7 where high-level logic and low-level control systems are built out of simple modules. The major components of the software system are the controls, the computer vision, the model of the environment and the path planning. Given the similarities between UMR’s two robots, the software systems are nearly identical.

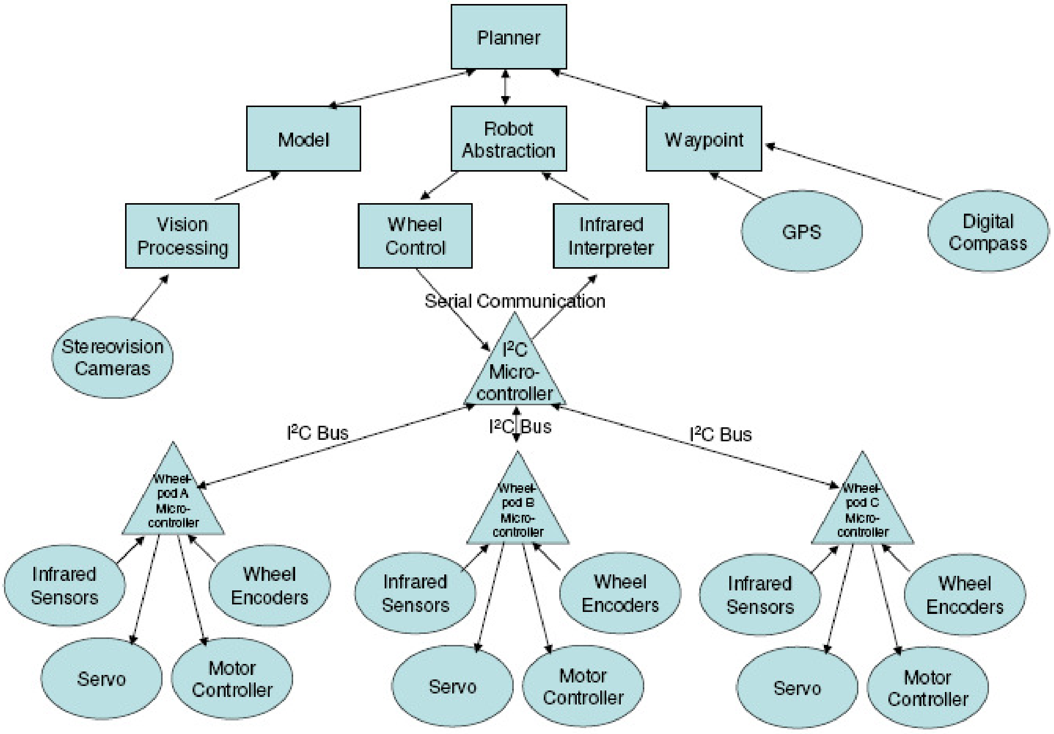

Figure 7: Software Organization Chart

The robot abstraction module provides an interface to each individual sensor and actuator on the robot. These interfaces are abstracted with simple commands that control the robot as a unit. This allows the planner to use simplified commands for navigation and also allows for interaction with individual components during testing.

The infrared interpreter module feeds data to the robot abstraction module so that the robot can react to obstacles too close for the cameras to see. The wheel control module takes high-level commands from the robot abstraction and converts these into simple movement commands to send through serial and I^2C buses to the wheelpod microcontrollers.

8.1 Computer Vision

Stereo Opticon uses stereovision to detect its environment. To interpret this data it uses a combination of the Small Vision System Library, the Open Computer Vision Library and in-house software developed at UMR. Instead of a feature based system as was implemented last year, a simple filter method was used for Optical Prime. Last year, the image was filtered for color but then objects were found and lines were detected. This was determined to be inefficient and was replaced with a system that does not waste time finding obstacles but instead finds regions of interest and sends them to be stored in the environment model. Since the robot only needs to pay attention to white and orange objects, it was logical to ignore everything else.

First, the image is filtered for white and orange colors. This creates regions of interest which correspond to lines on the ground and physical obstructions like barrels. Then the distance to each region of interest is found using disparity data from the stereovision cameras. Finally the model is updated with the three dimensional position of the region.

The performance of the cameras has been increased from 5 Hz in 2005 to 15 Hz in the current model due to these changes. This speed will help the robot to react quickly and efficiently. Below are two images from the vision system. Figure 9 is the raw image of a barrel in our shop and Figure 10 is the image after the filter. Notice how the filter ignores irrelevant components in the image and focuses on the orange barrel and the white board.

Figure 8: Raw Sample Image of Barrel

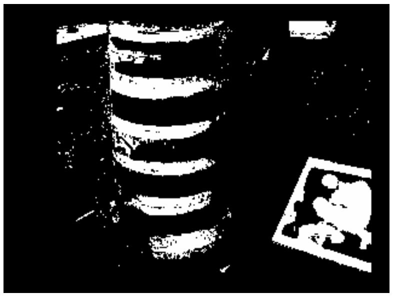

Figure 9: Image in Figure 8 Filtered for Orange and White

8.2 Environment Model

Using the data gathered by the stereovision cameras the computer develops a model of its environment. Since the computer only interprets data in the form of regions of interest, it creates a map of these regions and gives each a cost. The cost refers to how traversable it is for the robot. White lines are stored and avoided as if they were physical obstacles. The current model was simplified from an obstacle based system implemented last year to a cost map. This cost map takes in three dimensional data from the vision system and converts it to a two dimensional, pixel-based representation of the world. The three dimensional component was kept in order to make the model functional for other projects. After updating the cost map with new data the model is smoothed with a Gaussian blur so that the robot will see an increasing risk as it approaches a high cost area.

At the request of the planning module, the model will evaluate the cost along a linear path between any two coordinates. It does this by summing each pixel's cost with a weight towards pixels closer to the robot. This allows the robot to see nearer (and more certain) obstacle with a higher cost than obstacles at a distance.

The model updates each time the vision system receives a new frame, which has been measured at 15 Hz. Since the controls are able to receive commands at this rate also, the robot has a response time of 1/15 of a second. At 5 mph this corresponds to just 5 inches of travel for the robot. Below are two different screenshots of the model. Figure 10 is the model after it received data from the setup in Figures 8 and 9. Figure 11 is a simulation of many obstacles placed randomly around a robot within a 15 foot radius.

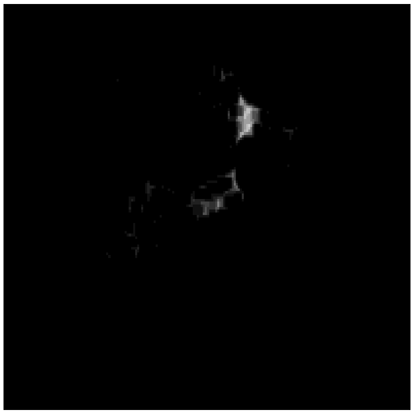

Figure 10: Cost Map From Figures 8 and 9

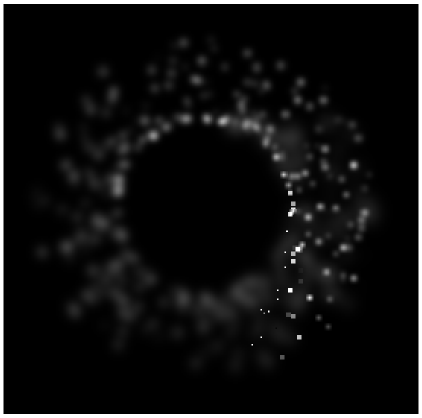

Figure 11: Simulated Cost Map

8.3 Planner

The environment model is used by the planner module to plot a course through all visible obstacles. The planner starts by asking the model to evaluate a linear path from the robot to a target. If the direct path to the target is obstructed, then the planner will bend the path until an unobstructed one is found. If no clear path is found then the path of least resistance will be chosen. The robot will follow this path while updating its cost map to see if any new high cost regions have been detected in front of it. If an obstacle is detected along the planned path, then the path will be recalculated.

For the Navigation Challenge the targets will be the waypoints provided. For the Obstacle Course the targets will be set at the furthest unobstructed point that the robot can see. This target will be reevaluated periodically to keep the robot progressing along the forward path. Since the only data considered is the forward view, the robot has a strong bias towards moving forward. Due to this implementation no line following is necessary; the robot will naturally find its way around the circle. It is possible to trap the robot in a corner or make it turn around and drive the wrong way; however, this situation is highly unlikely in practice. The system should avoid any traps that it can see, so the robot will not be caught in any dead-ends that can be seen from 15 feet away.

8.4 Controls

Once an appropriate path is chosen the control system is responsible for moving the robot using low-level commands sent to the microcontrollers on each wheel pod. The control system for the IGVC robot uses an innovative distributed processing design that allows the use of inexpensive nodes connected to an inter-robot bus instead of a traditional centralized controller. This design provides several important advantages. It is modular, allowing simple component interchange and replacement, inexpensive due to distributed processing, and organized to use a single serial bus instead of multiple wires.

Each node consists of a Phillips 89LPC938 microcontroller module (Figure 12) and supporting connecting hardware. The 8051 module was designed by the team and incorporates programming hardware for the 8051 and simple connections to a serial bus and I^2C bus. This module is inexpensive, in-system reprogrammable, and can easily be swapped between wheel pods to facilitate quick hardware debugging and replacement.

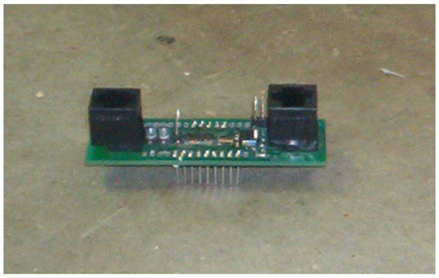

Figure 12: Microcontroller Board Designed by UMR for IGVC

Since these modules are small, easily programmable and utilize a simple I^2C bus, a single 4-wire bundle can carry power and data from a central controller to different nodes on the robot. This greatly simplifies wiring complexity and makes designing and debugging a much easier task.

Because these modules can be easily integrated into different parts of the robot and connected via a common bus, much less powerful, less expensive, and less complex processors can be used. Instead of having one powerful central computer trying to run 6 feedback loops in real time, this task can be offloaded to several less expensive slave processors that are controlled from a central node.

In our current robot, there are 4 separate modules controlling the low-level drive systems of the robot. On each wheel pod, there is one microcontroller module that controls the steering motor, drive motor, and infrared range finders. Robot Power's Open Source Motor Controllers are used to directly control the motors' speeds. The microcontroller communicates with these and the servos by generating pulse width modulation signals. The microcontroller modules communicate via an I^2C bus to another microcontroller module. This fourth module acts as the interface between the wheel pod controllers and the central PC. This architecture allows for simple and flexible wiring within the robot and efficient distributed processing.

9.0 Integration

It is not a trivial process to combine several sets of sensor data with many outputs to motors and servos. Using the inherent advantages of our modular and hierarchal design the task was greatly simplified. First, the data is taken into the computer via the vision, waypoint and robot abstraction modules. Next, it is processed into a cost map of the world around it. Then, a course is plotted through the least costly areas so that the robot can travel safely to its target. Finally, the control system is told where to move the robot.

10.0 Safety

In addition to the safety of the electrical system, the robot has two emergency stop buttons. One is mounted next to the camera within easy arm reach and the other is a wireless device. Each of these is connected to the disable input on the motor controller. When the input is triggered the motors will stop. This will halt the robot in just a few feet, even at 5 mph. Using the microcontrollers, each motor is hardware limited to 5 mph.

11.0 Vehicle Cost

UMR has presented a low cost design for the 2006 IGVC. The total components for this robot were initially predicted to be around $8000. The final total cost is calculated as $5080. Due to donated components, the team actually paid almost $1000 less than this. In addition to the monetary cost, over 3000 man-hours were necessary to design and fabricate Stereo Opticon.

Component	**Cost**
Drive Motors	480.00
Steering Motors	200.00
Aluminum/Steel	400.00
Misc Gears and Fasteners	300.00
Wheels	50.00
Cameras	1400.00

Computer	700.00
Electronics	500.00
GPS	200.00
Batteries	150.00
Power Distribution	50.00
Paint and Aesthetics	50.00
Motor Controllers	600.00
Total	*5080.00*

12.0 Conclusion

Stereo Opticon has many advantages over the more traditional robots seen in the IGVC. The three-wheeled design is more efficient than a four wheeled system and allows for smooth omni-directional behavior. The stereovision cameras detect obstacles easier and with a lower cost than a LIDAR and camera system would. The inexpensive and distributed control system allows the robot to move efficiently. The model and planner are optimized for the competition and are much less computationally intensive than last year’s implementations. All of these are integrated into a robot which will be highly effective in the 2006 IGVC.

Appendix A: Team Roster

First Name	Last Name	Major	Grade Level
Robert	Adams	Mechanical Engineering and Biological Sciences	Freshman
Ryan	Arlitt	Mechanical Engineering	Freshman
Kent	Barnett	Electrical Engineering	Junior
Mike	Becker	Mechanical Engineering	Senior
Ben	Bethge	Physics	Junior
Dave	Brown	Computer Science and Electrical Engineering	Junior
Ryanne	Dolan	Computer Engineering and Computer Science	Junior
Paul	Drews	Electrical and Computer Engineering	Junior
Matt	Duncan	Mechanical Engineering	Sophomore
Chris	Eckhard	Computer Science	Senior
Ramaprasad	Eshwarahalli Lakshminarayana	Mechanical Engineering	Graduate Student
Kerry	Geisz	Mechanical Engineering	Senior
Steven	Gullen	Mechanical Engineering	Sophomore
Jia	Guo	Mechanical Engineering	Graduate Student
Venkatesh	Hariharan	Manufacturing	Graduate Student
Matt	Hawkins	Mechanical Engineering	Senior
Mike	Hibbeler	Mechanical Engineering	Senior
Igor	Izyumin	Electrical Engineering Computer Science	Junior
Chris	Jacobsen	Applied Mathematics	Senior
Daniel	Krus	Mechanical Engineering	Graduate Student
Joel	Logue	Electrical Engineering	Junior
Joe	Lombardo	Electrical Engineering	Junior
Amanda	Luellen	Electrical Engineering	Freshman
Matt	Marsh	Computer Engineering	Freshman
Ryan	Meuth	Computer Engineering	Graduate Student
Justin	Miller	Computer Science	Senior
Phillip	Ponzer	Computer Engineering	Junior
Casey	Porta	Mechanical Engineering	Junior
Jeff	Pretz	Electrical Engineering	Freshman
Paul	Robinette	Computer Engineering and Physics	Junior
Zachary	Royer	Materials	Junior
Stuart	Salvador	Mechanical Engineering	Freshman
Gerard	Sequeira	Electrical Engineering	Graduate Student
Chris	Solomon	Aerospace Engineering and Business Administration	Senior