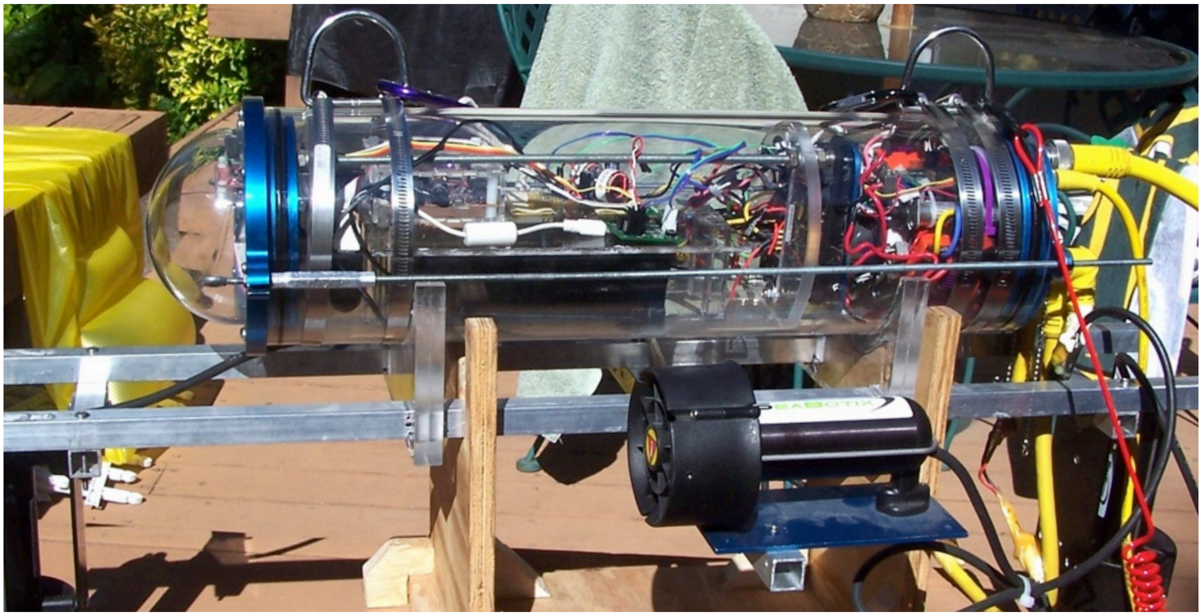
AMADOR VALLEY HIGH SCHOOL ROBOTICS CLUB

Brian Sherman, Rishabh Hatgadkar, James Heflin, Rishi Mukhopadhyay, Chris Covert, Kevin Chen, Anirudh Kandada, Ashutosh Shiledar, Vignesh Kalidas, Robert Holland, Brandon Lopez, Edward Wang, Daniel Naito



Abstract

The Amador Valley High School Robotics Club has constructed an autonomous underwater vehicle (AUV), named *Barracuda Mark IX*, for the thirteenth annual AUVSI International Autonomous Underwater Vehicle competition.

The internal components of the Barracuda are housed within a single, 6" diameter acrylic tube. The vehicle itself is propelled by two laterally mounted SeaBotix thrusters controlling speed and heading, and two vertically mounted SeaBotix thrusters controlling pitch and depth. A pressure sensor, inertial measurement unit (IMU), a pair of cameras, and a hydrophone array provide navigation and orientation data to the computer: a BeagleBoard, which provides autonomous control of the vehicle. The computer focuses on high-level mission control and image-processing tasks, while a new custom control board provides all low-level control over the thrusters and communications with the various sensors. This improves the reliability and responsiveness of the submarine's movement.

This year the team has focused on reliability and efficiency. By replacing our old Windows computer with a new Linux-based BeagleBoard, we have decreased processing overhead and greatly reduced power consumption. This computer vastly increases the Barracuda's ability for vision processing. The addition of a wireless tether and control system bootloader has allowed us to reprogram the entire system remotely, streamlining our development process. The software team has revised the mission control software to improve communications between the computer, control board, and various electrical systems, resulting in a smoother and more capable control system.

The 2010 Mission

This year the mission requires the submarine to first navigate through an underwater validation gate in order to demonstrate complete autonomous control. The submarine must then complete, in any order, a series of tasks, which it can locate by following an orange PVC path on the floor of the pool. First, the AUV must strike two out of three buoys or "life vests", colored red, green, or yellow, in a specified order. The AUV must then pass above a green PVC "hedge" suspended from the floor. The AUV then must "pick up a weapon": finding four white bins on the pool floor, identifying the red images of an axe, hedge clippers, a hammer, and a machete, then dropping two markers into the designated bin. Afterwards, the submarine must approach a 2x2 array of four windows - red, green, blue, and yellow - and shoot torpedoes, or a "crossbow", through a window of a specified color. Finally, the AUV must determine the position of an acoustic pinger and orient itself above the pinger, lower itself down and secure a PVC "counselor", and surface within the "police station", or PVC octagon floating above. Each vehicle has fifteen minutes to complete the mission, receiving points as it completes each obstacle. In addition, the vehicle receives bonus points proportional to the amount of remaining time in the run, and the weight below the weight limit.

Hardware

The *Barracuda Mark XI* consists of a hull, a frame, engines, and several other modular components. All external components are screwed directly into the frame (two square rods running parallel to the tube). The ability to attach components directly onto the tube with any size of screw gives us a greater flexibility in rapidly attaching new components to the frame, without relying on a separate attachment piece, such as ring clamps.

Tube and Frame

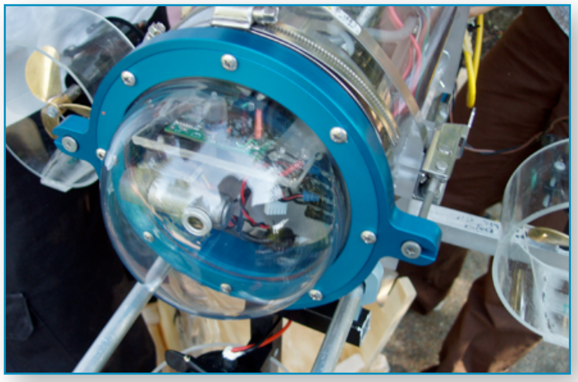
The hull of the *Barracuda* is a 20" long, cast acrylic tube that is 3/16" thick and has a 6" outer diameter. The frame is comprised of two parallel square aluminum rods that are 1/16" thick and are of 3/4" diameter. These run along to the bottom of the tube, and they are affixed to the hull through a series of acrylic cradles cemented to the tube. These cradles have holes through them that the rods of the frame can fit through, and the rods are secured in place by a series of pins on either side of the cradles. Two metal handles are attached to the top of the tube for transportation and deployment. Both ends of the tube have anodized aluminum end caps, secured to each, rather than the tube itself, to alleviate the strain on the hull integrity. These end caps are equipped with compression O-rings to create a water-tight seal.

Connector Endplate

All through-hull connectors are made on the rear end plate and use Brad Harrison Waterproof connectors. These connectors are IP67 rated and can handle up to 1000 PSI. These connectors are a major advancement from our original connections made via epoxied wires through the hull. The ability to rapidly remove and reattach these connectors without disrupting the water-tight seal of the hull adds to the modularity of the design.

Dome Endplate

The front end cap consists of a clear, cast acrylic dome. The acrylic material was chosen for its transparency and its durability during the machining process. The exceptional visibility it provides is necessary for our cameras to detect physical obstacles, and the dome shape of the endplate gives the cameras an improved field of vision. The flange of the dome is fixed between the front end cap and another metal ring. This assembly is secured by eight screws with compression O-ring.



Thrusters

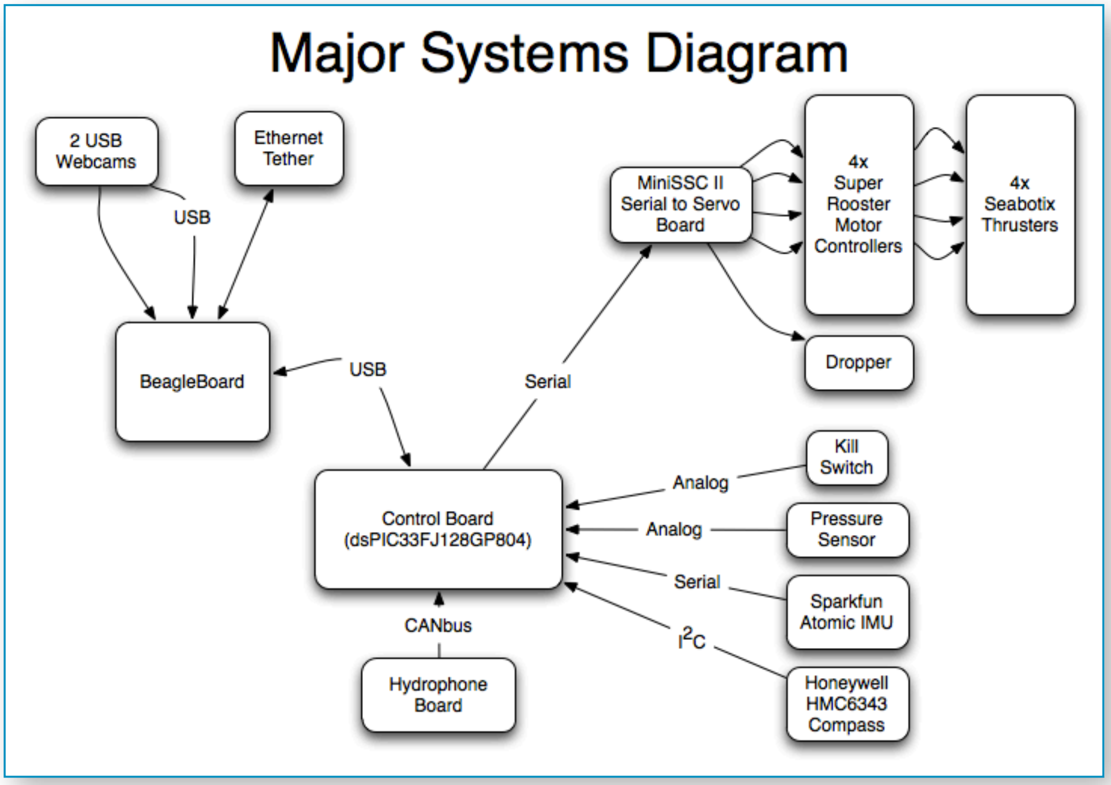
This year, for vertical and horizontal thrust, we are using Seabotix BTD150 thrusters with built-in shrouds. The power consumption of each thruster is quite low, only drawing 3.3 amps each at full power. This also limits the amount of magnetic distortion for the compass. The vertical thrusters control depth and pitch, and are mounted on aluminum bars affixed perpendicularly to the frame. The horizontal thrusters are mounted on an aluminum plate on the horizontal engine bar running perpendicular to the tube. They control the heading or yaw of *Barracuda*, as well as allowing forward and reverse propulsion.

Markers and Dropper

The marker are 3/4 inch diameter stainless steel spheres with our school logo (AV) engraved into them. They rest in a hollow acrylic tube mounted on the side of the horizontal engine bar. A small waterproof motor with one propeller blade is located to the side of this tube, and the blade prevents the markers from falling out of the hollow tube. The motor will turn when the proper location is detected, turning the propeller and releasing both of the markers.

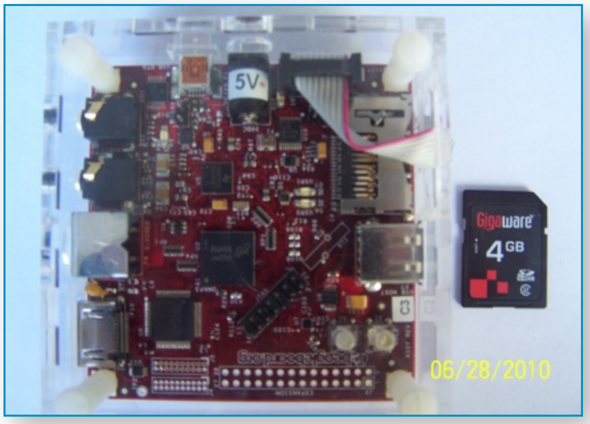
Torpedoes and Torpedo Launcher

The torpedoes are small acrylic bolts labeled with our school logo (AV). The torpedo launcher consists of two canisters of compressed air leading up to two small acrylic tubes, which house the torpedoes themselves. A quick-release valve separates the air canisters from the torpedo bays, and will be closed for the majority of the run. To fire the torpedoes, the vehicle will open the valve electronically, and the compressed air will propel the torpedoes out of the torpedo bays and forward towards the target.



Electrical Subsystems

The main tube is partitioned into two sections by an acrylic rackboard, which extends from the dome to the rear endcap. The BeagleBoard, USB hub, control board, hydrophone board, and power supply are on the top, and the batteries are located in the bottom. Elcon Rack Connectors allow the rackboard and rear endcap to securely attach, transferring data and power to the endcap and to any external peripherals or computers for tethering purposes. This also allows the rack to slide out from the submarine for testing and debugging.

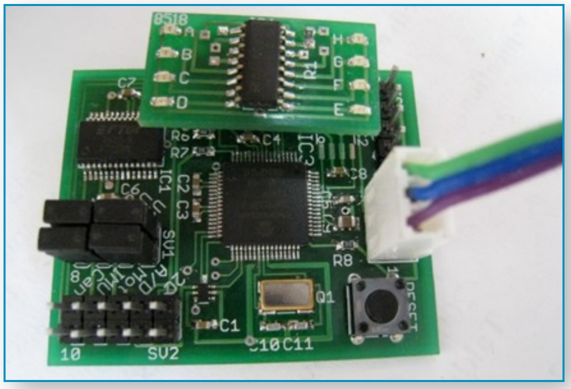


Computer (BeagleBoard)

The sub's computer is new this year, a BeagleBoard Rev C3. The BeagleBoard runs the Angstrom distribution of Linux on an TI OMAP3530 processor, which is fast, small and efficient. It runs on less than 5W, allowing it to run off of the main batteries without significantly decreasing Barracuda's run time. The code and operating system are loaded from an onboard SD card slot, and the remainder of the input and output is conducted through a single USB port, connected to a hub. We switched to the BeagleBoard from an ultraportable OQO computer because of its reduced power consumption, increased speed, and the versatility offered by using Linux rather than Windows. Debugging and troubleshooting can be done more easily either over the network, or via the HDMI port on the BeagleBoard.

Batteries

The batteries, situated in the center of the rackboard, supply power to Barracuda's thrusters and most of the internal electronics. Two 14.8V Tenergy lithium-polymer ion battery packs with a combined 12.7 Ah capacity can power the submarine in full operation for over 2 hours, allowing for longer run-time and less frequent charging. The battery packs contain protection circuitry which prevents over-discharging and over-charging of the packs reduces the risk of battery failure or damage to the submarine's electronics.



Control System

In order to enable the computer to focus on higher-level mission and image-processing tasks, we designed a new custom control board to handle low-level control and communications functions. The board uses a Microchip dsPIC33FJ128GP804 because of its low-power 3.3V supply, its fast 40 MIPS operation, and its high number of communications peripherals. The built-in analog-to-digital converters monitor the status of the kill switch and the pressure sensor to determine a depth measurement. A combination of UART, I2C, and CANbus configure, monitor, and communicate with the magnetic compass module, passive sonar system, serial-to-servo board, the gyroscopes and accelerometers on the inertial measurement unit, and the computer. An FT232R from FTDI Chip allows the control board to interface with the computer via USB, appearing as a serial port which can be accessed by the mission software. The dsPIC relays sensor information to the main computer, which sends control commands back to the microcontroller. Several PID processes on the dsPIC control the heading, depth, and pitch of the submarine.

A bootloader dynamically downloads updated code from the BeagleBoard upon reset, eliminating the need for a programming device and enabling the team to reprogram the control system remotely over the network. This streamlines and automates much of the development process for the entire submarine.

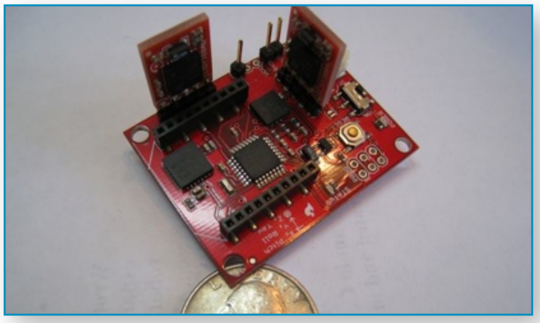
Serial to Servo Board

A Mini SSC board, an off-the-shelf serial-to-servo board made by Scott Edwards Electronics is mounted to the rear endcap. This board generates PWM signals that operate the Novak motor controllers in response to commands from the control board's UART module.

Tether and Communications

A communications link to the main computer is required to facilitate testing of the vehicle. This is provided by a wireless access point attached by tether, which can float along with the vehicle. The BeagleBoard is connected to the tether via the ethernet port on the USB hub. Via this network connection we can not only monitor and control the submarine, but also alter the mission code and even reprogram the dsPIC on the control board. Furthermore, the tether, when connected, enables a remote engine kill that replicates the onboard kill switch at the end of the tether.

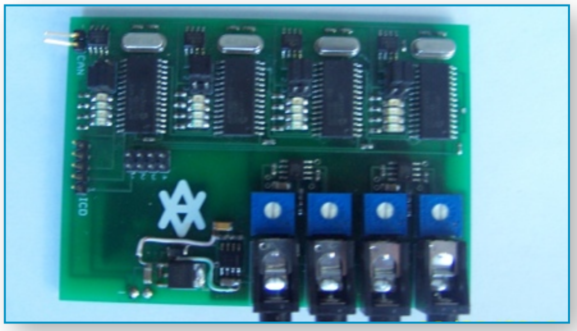
Navigation Sensors



The Barracuda queries a Honeywell HMC6343 3-axis compass module over an I^{2}C bus to measure its absolute heading, while compensating for the hard-iron magnetic distortion caused by the onboard electronics and ferromagnetic materials. However, the presence of constantly changing soft-iron magnetic distortion caused by the motors switching on and off necessitated an alternative method of determining tilt and heading, which is provided by a Sparkfun Atomic Inertial Measurement Unit. The IMU is populated with 3 ST Microelectronics LISY300AL gyroscopes and a Freescale MMA7260Q 3-axis accelerometer. This allows the submarine to determine its orientation independent of the fluctuating magnetic fields in and around the submarine.

Camera

The Barracuda uses two Dynex 1.3MP Webcams. These communicate over a USB 2.0 bus with the computer, and the video feed is read and processed with OpenCV. The downward-facing camera is used to detect the path and the weapons, while the forward-facing camera detects the start gate, life vest, hedge, and windows.



Passive Sonar

The array consists of four Reson model 4013 omnidirectional hydrophones mounted in a precise diamond array in an acrylic mold. The hydrophones attach to waterproof Brad Harrison connectors, using shielded wires to reduce noise. The signals are amplified by a pair of National Semiconductor LMH6646 dual operational amplifiers. The signals are received by three dsPIC30F4012's, which are Microchip PIC microcontrollers optimized for digital signal processing. The dsPICs will filter out the erroneous frequencies, and further amplify the signal. Each will calculate the phase difference between a pair of signals, and send the phase information over a CAN (controller-area network) bus to a fourth dsPIC. This fourth dsPIC uses hyperbolic positioning to find a directional vector towards the pinger. The same CAN bus is used to send the pinger's direction to the main control system. The speed, reliability, and modularity of CAN makes it ideal as the main communication bus between the microcontrollers and systems. By putting the system on printed circuit boards as opposed to the perfboard used in previous years, we can greatly increase the reliability of the system while decreasing its size and weight.

Software

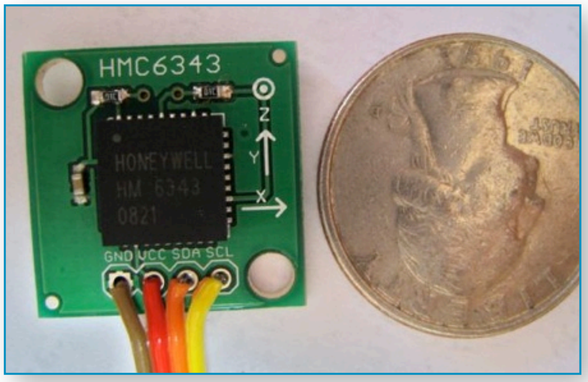
The BeagleBoard runs our own Autonomous Vehicle Intelligence software, which we call AVI 5.0. The shift from Windows to Linux this year has been accompanied by a change in programming language, with the bulk of the code being done in C++. We've updated AVI this year to include simple yet robust network interface, a graphical user interface run on a client computer, and flexible class design. This change in architecture and language has changed out development process dramatically, making it possible to quickly change and recompile AVI without ever needing to remove the submarine from the test pool.

Class Design

Rather than use strings to communicate between various classes, AVI 5.0 implements a class design in which a single mission controller class encapsulates all positional variables. This design allows the software to flexibly and quickly update the GUI, change desired heading and depth, and provide feedback to the dsPIC chip.

Feedback Control

Used at least indirectly by almost every high level control process on the vehicle, the feedback control allows the vehicle to adapt itself to changing external situations. This process implements a general Proportional-Integral-Derivative control loop, which can be run on any number of feedback inputs. Every feedback mechanism used by the system simply has to supply a configuration, an input value, and a set point. The feedback control process then produces a value from -1 to 1, indicating to the vehicle how it needs to change to reach the set point. The feedback control is used to control not only heading, depth, and pitch, which are handled by the control system PIC, but also for some of the high-level navigation tasks. The control loop itself includes several features, including derivative filtering (important for sensors that have noise spikes occasionally) and mechanisms to help prevent integral windup.



Speed Control

The control system sends motor commands to the a microcontroller on the serial to servo board. The board uses a simple serial protocol for communication and provides an interface to the motors; the PIC on the board has eight PWM servo outputs, which drive the Novak speed controllers and the marker-dropper.

Orientation Measurement

The control system combines the high-bias but low-noise integrated rotation data from the IMU's gyroscopes and the low-bias, high-noise data from the accelerometers and magnetometers on the Honeywell compass module using several Kalman filters, which estimate the current orientation based on knowledge of the previous known position of the AUV, the current sensor measurements, and knowledge of the dynamics of the AUV.

Mission Control

The Mission Control class contains the information on the state and current objective for the completion of the mission. Mission state is coordinated using shared variables, which other classes check to determine their behavior during a certain part of the mission. Once the controller has determined that one portion of the mission has been completed, it updates the mission state variable, and the other classes adapt their behavior to the new mission state.

Optical Analysis

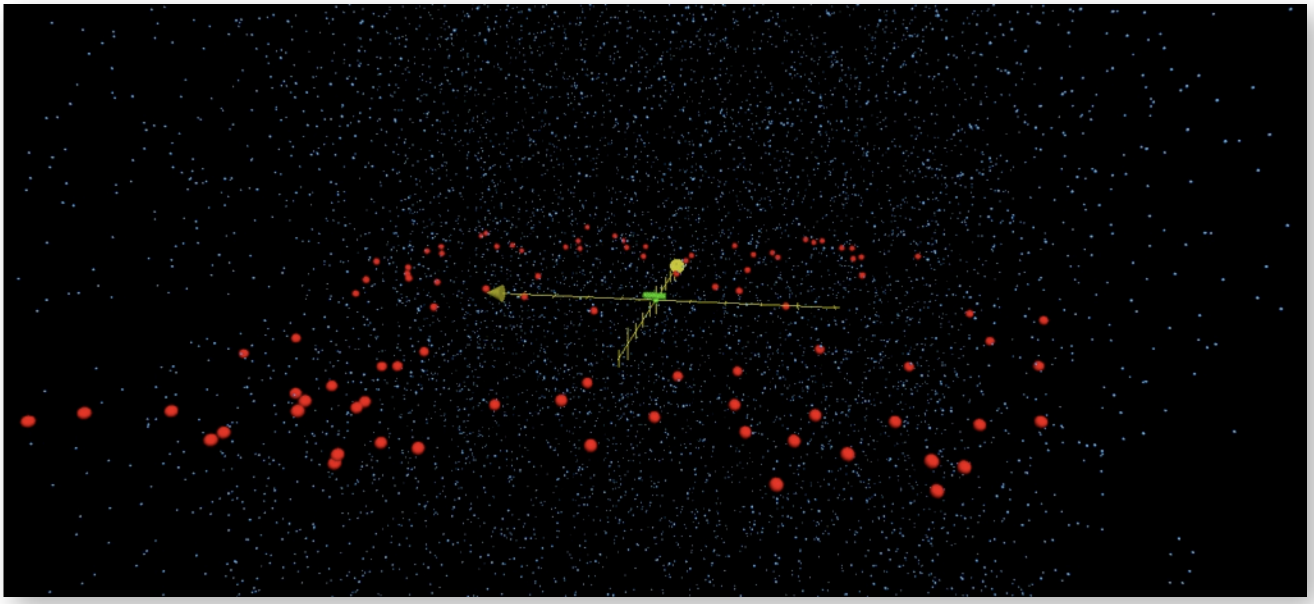
To make the navigation processes possible, it is necessary to have robust and effective optical analysis algorithms in place. As the code is now being written in C++, we have moved away from our old optical analysis programs, and have instead begun using the OpenCV library. This library provides a great foundation with which we can write custom filters and record video for later analysis. This allows us to develop highly efficient optical processing algorithms quickly and easily.

Beginning the Mission

Once the mission process is initiated, the process waits for the kill switch to be inserted. After insertion, the vehicle dives to mission depth and continues straight through the validation gate. During this time the optical processing code begins to search for the life vests (buoys). Based on the order of colors that we give it, it checks the forward camera for the presence of a circle of the first color within the image. The center of this circle is computed, and its distance from the center of the field of view computed. Any deviation causes a command to be sent to the control PIC to correct for the deviation. When successful docking is signaled by the sudden disappearance of the enlarged light source, the submarine then searches for and targets the life vest of the second color in the same fashion. Afterwards, all image processing is focused on locating the green "hedge". The passive sonar process will launch as well, and begin to time the intervals of the pings in order to ensure maximum accuracy.

Following the Path

With a second web camera facing directly downwards, we are able to switch between input sources to view the orange path that outlines the course. Our algorithms in OpenCV allow the computer to quickly filter out the orange color of the path, and apply hough transformation to find the coordinates and slopes of the path relative to the orientation of the submarine. The vision processing unit can then adjust the desired heading based on the coordinates that we calculate in our code.



Simulation of pinger location. Green is the submarine. Each dot is a simulated pinger location, red is a failed calculation, blue is a successful calculation. Success rate: 99.4%

Edge Finding

The first step in finding the bins is finding the edges. First to cut down on noise we normalize the image. Then we filter the image by color depending on the current target, i.e. black and red for the target bins. For the edges, we use the Sobel operator, which is effectively a midpoint approximation to the gradient of the image. The results are run through an intensity threshold, which provides a black and white image which has a higher contrast is more suitable for the Hough transform and also uses less CPU power to process.

Hough Transform

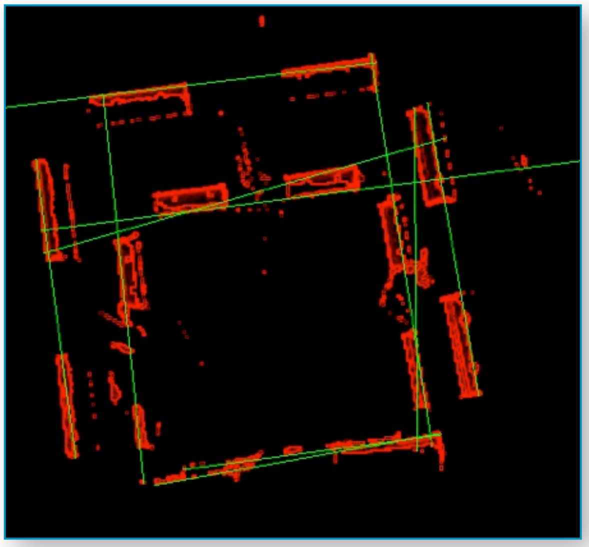
For target recognition, we use a process known as the Hough Transform. The goal of this process is to find a geometric description of lines within images, including possibly disjoint (dashed) lines, which makes it ideal for recognizing the edges produced by the target. The ideal output of a Hough transform on the box target image is a grid of 2 lines overlapping 2 other lines at right angles.

Dropping Markers in the Bins

While continuing to travel straight, the vehicle searches for the orange path and the target bins. When one of the bins is located the vehicle will attempt to station itself directly over the center of the target, using the center of mass image processing algorithm. Once the vehicle is positioned properly over the center of the bin, it drops both markers and proceeds toward the window.

Firing Through the Gun Nest

The vehicle resumes processing forward vision once the markers have been dropped. It then searches for an array of colored quadrilaterals which mark the presence of the window. Once the array is found, the vehicle slows down and orients itself in front of the quadrilateral of the specified color. The vehicle uses the sizes of the sides of the quadrilateral to determine which direction to face to



Edge detection/hough transform on an image of test quadrilaterals

maximize its chances of firing torpedoes through the quadrilateral. Once it is properly aligned, the vehicle will fire its torpedoes.

Navigating to the Police Station

After completing the marker drop, the computer signals the systems to begin analysis of the audio data. The dsPICs compute a bandpass filter for the audio, removing the majority of erroneous frequencies from the signal. Each dsPIC computes the XOR of the sign of the two signals, and averages the result. When the variance of the averaged XOR passes a threshold, a ping has been read. The average of the XOR related directly to the phase difference between the signals, which is sent, along with the time of the ping, to the main microcontroller. When the main dsPIC has received all of the phase difference information, it uses hyperbolic positioning formulas to calculate a directional vector to the pinger. This vector is sent to the control system, which combines it with depth readings to find the location of the pinger relative to the sub. Using the locational information provided by the board, the computer directs the submarine towards the pinger until the ping is read directly below the briefcase.

Retrieving The Counselor and Surfacing in the Police Station

Once the vehicle is positioned above the pinger, signified by a directional vector without an X or Y component, it proceeds to secure the briefcase by lowering the grabber onto it. After the briefcase is secured, the vehicle will surface and complete the mission.

Conclusion

The advancements made this year will allow our submarine, *Barracuda Mark IX*, to face the new challenges presented by the 13th AUVSI AUV Competition. This year our club focused on teaching the newer members mechanical, electronic, and programming skills. We also improved the development process, allowing for much quicker code turnaround time. The new computer and software provide a level of stability which will be greatly helpful during competition. We hope that these changes will make our submarine a much more competitive entry, and the transition of this year's team to the next a much smoother process.

Acknowledgements

We would like to thank our sponsors: Amerimade Technology, Advanced Circuits, PTSA, PPIE, FTDI, Microchip, HST, and C&C Group. We would like to specially thank all those who donated their time in order to help the team succeed; especially our advisor, Mr. Hall; Joel Soppet, who provided the club with advice, assistance, and donations; the Sherman's for use of their garage and pool, and all of our parents for their help.