SubjuGator 2002

Jason W. Grzywna, Jennifer L. Laine, Kevin Walchko, Ryan P. Dye, Ashish Jain, Nicholas Ivano
Michael C. Nechyba, Eric M. Schwartz, A. Antonio Arroyo

Machine Intelligence Laboratory
University of Florida
Gainesville, FL 32611

http://www.mil.ufl.edu/subjugator

Abstract

Graduate and undergraduate students at the University of Florida are in the process of modifying and testing an autonomous submarine, SubjuGator, to compete in the 2002 ONR/AUVSI Underwater Vehicle Competition. SubjuGator is designed for operation down to 100 feet, and can be quickly configured to optimize for mobility or speed. SubjuGator's body has mounts to support up to ten motors, each of which may be oriented in a multitude of directions. SubjuGator is controlled through a single-board Pentium3 based computer running the Linux operating system, which is interfaced to the motors through a microcontroller and to the camera through an IEEE1394 connection. On-board sensors include a digital compass, a fluidic inclinometer, and a pressure sensor. Additionally, mission specific sensors include a high-resolution progressive scan camera, and a sonar altimeter for height detection. In this paper, we first describe the mechanical makeup of SubjuGator. Next, we describe the electronic and processing hardware and the motivation for our electronic design. We then discuss the various on-board sensors, both mission-dependent as well as mission-independent. Finally, we comment on vehicle control strategies.

1. Introduction

The Autonomous Unmanned Vehicle Systems international (AUVSI) and the Office of Naval Research (ONR) are sponsoring the Fifth Annual Autonomous Underwater Vehicle competition to be held in San Diego at the SPAWAR facility July 31st through August 4th. A student team at the University of Florida is once again developing an AUV for this latest contest. SubjuGator has been revised and redesigned to meet the challenges of this year's competition.

The submarine this year must navigate within an underwater environment to identify targets. There are 18 targets total. Each target possesses a unique bar code, and a unique height. Points are awarded for passing through a start gate and for each target successfully identified.

In this paper, we first describe the mechanical makeup of SubjuGator. Next, we describe the electronic and processing hardware and the motivation for our electronic design. We then discuss the various on-board sensors, both mission-dependent as well as mission-independent. Finally, we comment on how we expect a typical competition run to proceed and how the subsystems on board SubjuGator will allow us to meet the mission goals.

2. Mechanical System

As a third-generation vehicle, SubjuGator embodies the lessons learned in four years of autonomous underwater vehicle (AUV) development. We considered several key design criteria, including the vehicle's hydrodynamics, its survivability in a salt-water environment, and its adaptability for different missions through easy motor reconfiguration and future sensor additions.

2.1 Body

The 36" long octagonal shape is composed of 0.25" thick aluminum plate and 0.5" thick square bar. A bulkhead on each end fastened with quick-release latches keeps the internals dry, while allowing access to the components from either end of the sub. Three hard-point rings are welded onto the frame (Figure 1) to strengthen the structure, provide mounting points for exterior sensors via blind-tapped holes, and carry all through-hull connections. The central hard-point ring also contains the cylindrical mounts for eight motors. The mount allows the motor's thrust to be positioned in line with the body, or perpendicular to it. With a mount on each of the eight faces of the sub, a multitude of motor configurations are possible, allowing the vehicle to be quickly adapted and optimized for a particular situation or mission. Figure 2 shows one configuration (a) optimized for mobility while the other (b) is optimized for speed and power. For the 2002 competition, we have chosen configuration (b) to maximize our speed.

2.2 Farings

The fore and aft flooded 14" farings provide a more streamlined flow around the vertical motors and the frame. Additionally, the farings offer structural support and protection to any sensor mounted within them. Both farings are open on the top and bottom to provide for upward or downward looking sensors. Moreover, the forward section of the fore cone is open for any forward-looking sensors.

2.3 Motors

All six motors are Motorguide Power Plus electric trolling motors with 6.75" diameter propellers. At 12V these motors provide approximately 22 pounds of thrust, and are fitted with custom O-ring seals that allow for a salt-water depth of up to 100 feet. Each motor is shrouded to prevent incidental blade contact.

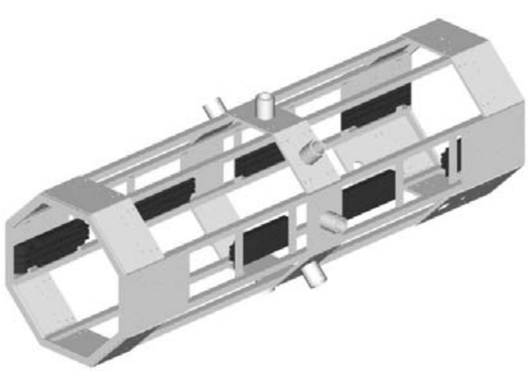

Fig 1. Body frame

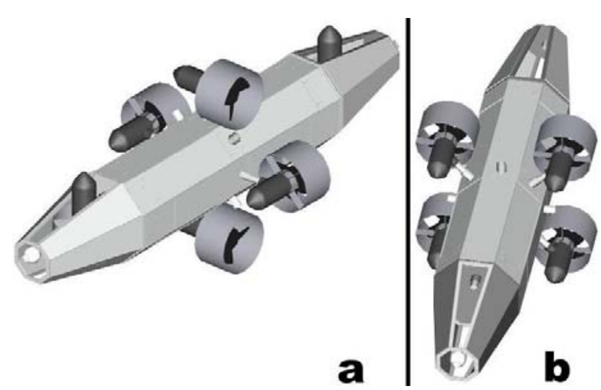

Fig 2. Example Configurations

2.4 Through-hull connections

All through-hull connections use Burton 5500 series sealed and molded underwater connectors. A kill switch is implemented with a Gianni hermetically sealed push-pull switch that disconnects power from the motors and initiates a software motor kill routine. A power switch is implemented with a Gianni hermetically sealed SPST switch.

2.5 Interior layout

Two shelves guided on delrin rails provide support for all the internal electronics and power. Batteries and high-power electronics are stowed in the lower shelf to provide a metacentric righting-moment, while the upper shelf houses the remaining electronics. Electrical connections terminate at connectors at the front of the sub for expedient removal of both shelves.

2.6 Exterior camera enclosure

Subjugator uses a custom built underwater vision system. To contain the camera and its connecting electronics, we have constructed an external forward mounted camera enclosure. This enclosure is mounted on the front nose of the submarine nominally pitched at 20 degrees forward, but is reconfigurable between zero and 40 degrees. It is constructed from a PVC compression fitting using a glass plate at one end, and a hose fitting at the other. The enclosure is connected to the internal cavity of the sub, and therefore of equal pressure.

For testing and extreme depths we are able to pressurize the internal cavity of the sub through the tubing connecting of the camera enclosure. This reduces the pressure gradient on the sub, and thus the chance of hull failure. Pressurizing the cavity has also assisted us in finding micro fractures in the outer casing of the submersible.

3. Electrical System

The electrical system of the vehicle is composed of a power system (batteries and motor drivers), computing resources (x86 processor, microcontroller) and the sensors that provide information about the environment to the vehicle.

3.1 Power supply

SubjuGator uses four Powersonic 12 Amp-Hour 12V sealed lead-acid batteries, three to power the motors, and a one to power the electronics. A Keypower DX250H DC-DC ATX power supply provides for all of the electronics contained within the submarine. This configuration allows for 2.5 to 3 hours of operational runtime.

3.2 Computing

The various tasks of the computing system on SubjuGator demand different approaches. First, the vision system and the main intelligence require a powerful processor to perform real-time decision making and analysis on the incoming sensor data. Second, the motor system requires a consistent and dependable output to control motor speed. To service these systems we chose the EEPD Pentium3 700MHz Envader embedded single-board computer, and the Motorola 68HC11.

3.2.1 68HC11

The Motorola 68HC11 is an eight-bit microcontroller unit with flexible and powerful on-chip peripheral capabilities. These include an eight-channel analog-to-digital (A/D) converter with eight bits of resolution, an asynchronous serial communications interface (SCI), and five output-compare timing output lines [5]. The A/D converter, together with the SCI system, interfaces analog sensors to the digital main processor. The SCI system also receives motor output specifications, which are fed to the output-compare lines to generate precise speed control for the motors. These signals are then fed into motor driver boards we designed to provide accurate high-current motor control.

3.2.2 Main processor

Top-level control is handled by an EEPD Envader single board computer. This Pentium3-based 700MHz board has 256MB of RAM, IEEE1394 (Firewire), USB, PC/104+, and runs Red Hat Linux 7.3.[2] We are using a PCMCIA adapter to interface our wireless ethernet card. All sensor information, gathered on one system, is evaluated, and consequent instructions are then issued to all subsystems.

3.2.3 Wireless system access

A communications interface between a base station and the vehicle utilizes a wireless ethernet (802.11b) connection with an 11Mb/s datapath. We are using ZoomAir 4105 cards. This allows secure shell, ftp, and simultaneous programmer access for parallel code development and debugging.

Testing of the sub is performed by remote operation through software running across the wireless link. By viewing the real-time sensor data, we can tune most aspects of the subs intelligence and control.

3.3 Navigational sensors

For even the most basic operation, an AUV must be able to maintain a heading, a depth and attitude. Sensors to allow this are present on almost all AUVs, regardless of any specific mission. We define these as navigational sensors.

3.3.1 Digital compass

SubjuGator uses a TCM2-50 compass from Precision Navigation. With a triaxial magnetometer, a fluidic inclinometer, and a microprocessor, this compass generates heading, tilt and roll information throughout its operational range.

3.3.2 Depth sensor

Depth measurements are gathered with a Measurement Specialties MSP-320 series pressure sensor. It is rated to 25 PSI with a rated accuracy of ± .25 PSI and outputs an analog voltage between 1 and 5 volts, which translates to a depth resolution of ±2 inches.

3.4 Mission-specific sensors

The competition task requires us to localize and read bar-coded targets. Also the heights of the targets must be determined along with the decoded bar-code number. We are focusing on vision and our underwater vision system to complete the mission. A sonar altimeter is attached as a supplement to the vision system both for confidence and collaborative height detection.

3.4.1 Sonar altimeter

We acquire height measurements with a Datasonics PSA-916 sonar altimeter. This model is modified to measure distances from 30cm to 100m with a resolution of 1cm over an RS-232 connection.

3.4.2 Underwater vision system

To accomplish underwater computer vision we have developed and constructed both hardware and software capable of capturing images and processing them completely inside our submarine. We are using a Unibrain Fire-i400 progressive-scan camera, capable of 640x480 resolution at 15fps. This camera has an interchangeable lens and interfaces to our embedded computer through IEEE1394 (Firewire).

The latest Linux kernels have built in support for plug-and-play Firewire devices. Using the digital camera libraries available for Linux we have written custom software for both frame grabbing and acquiring video.

The camera is mounted in an exterior enclosure as described in section 2.6.

We accomplish the detection, localization, and classification of the underwater targets using our camera and our computer vision algorithms [6]. Once a barcode has been detected we isolate that part of the image and use a variant of linear regression to classify and decode the data. We also take into account scaling and the many possible orientations of the targets.

4. Vehicle control and strategy

4.1 PID controller

As the submarine moves through the water, errors between the desired and current values of heading, pitch, and depth are controlled through a standard PID controller. The determination of the motor actuation values is based on the submarine's position and orientation divergence according to, where $m(t)$ is the motor value and $e(t)$ represents the error at time step t. The continuous equation is converted to its discrete-time equivalent and the errors are calculated from the difference between the current and desired heading, pitch and depth [3,5].

$$m(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \qquad (1)$$

The individual gains (Ki) are tuned through repetitive testing, at various depths and operating conditions. For each of our possible speed and depth range configurations, we maintain a separate set of control parameters. These parameters are determined through experimentation and simulation. We also use a PID controller to maintain the vehicle's pitch.

4.2 Kalman filter

The Kalman filter is an alternative way to calculate the minimum mean-squared error (MMSE) using state-space techniques. R. E. Kalman, a former graduate research professor in the Electrical Engineering Department of University of Florida, first developed the filter in 1960. An advantage of the Kalman filter estimator is that it is computationally efficient by recursively processing noisy data. It functions as a real-time estimator based on models of the systems, and the noise under test.

The Kalman filter estimates a process by using a form of feedback control. The filter estimates the process state at some point in time and then obtains feedback in the form of noisy measurements. The filter equations fall into two groups: time update equations and measurement update equations. The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate. The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations [1].

4.3 Arbiter

Each of the sensor analysis processes make heading, speed and depth requests to improve the position of the sub in relation to the targets. Due to the various strengths and weaknesses of particular sensors, and the occasional sensor anomaly, these requests may sometimes conflict. Therefore, we have implemented an arbiter, a rule-based algorithm specifically tuned for the competition environment, which is tasked with deciding on the next action for the sub, given the various, possibly erroneous, sensor inputs.

Our solution to identifying the underwater targets will logically proceed as follows. The sub will dive to a pre-determined depth and align on the proper heading to the validation gate. The sub will then traverse the distance to the gate and stop when this distance has been reached. At this point we will attempt to acquire targets. Relying on our vision system we will begin searching. We have a plan for both the targets on the outside edge of the pool along with the cluster of targets located in the center of the pool. The details of our solution will be divulged during SubjuGator's competition run.

5. Acknowledgements

We thank our sponsors: Harris Semiconductors, UF Dept. of Electrical and Computer Engineering, UF College of Engineering, International Transducer Corporation, Burton Electrical Engineering, Giannini, Torpedo Industries, Precision Navigation, UF Dept. of Chemistry Machine Shop. We would like to especially thank Todd Prox of the Chemistry Machine Shop for his time and expertise. Thanks to our faculty advisors who protected us from the paper storm that is inevitable with such a project. We also must thank the previous members who make up the legacy of Subjugator: Dave Novick, Scott Kanowitz, Scott Nortman, Scott Nichols, Patrick O'Malley, and Ivan Zapata.

6. References

[1] Rogers, R. M., "Applied Mathematics in Integrated Navigation Systems", Reston, VA: American Institute of Aeronautics and Astronautics, 2000.

[2] Matthew, N. and Stones, R. 2001. *Beginning Linux Programming, 2nd Edition*, WROX Press LTD.

[3] Dorf, C. and Bishop, R. 2001. *Modern Control Systems, 9th Edition*, Prentice-Hall, Inc.

[4] M68HC11 Reference Manual, Rev 3. Motorola. 1991.

[5] Control Tutorials for Matlab: PID Tutorial. http://www.engin.umich.edu/group/ctm/PID/PID.html. Carnegie Mellon. U. Mich. 1997.

[6] The Computer Vision Homepage. http://www2.cs.cmu.edu/afs/cs/project/cil/ftp/html/vision.html. Carnegie Mellon. 2002.