Improved Interface & Modeling Environment for Search Task Optimization Progress Report I

Sponsored By: Northrop Grumman
Contact: Ryan Tintner

CSC 492 Team 13:

Michael Haag
Neil MacFawn
Samuel Munilla
Percy Tyndall
Todd Rothfuss

North Carolina State University
Department of Computer Science
Senior Design Center

February 7, 2007

Sponsor Background

Our primary sponsor, Northrop Grumman Corporation (NGC), is a global military contractor. They list their major business areas as, "Information & Services, Electronics, Aerospace, and Ships." Our project falls in as one of many attempts that Northrop Grumman has made to integrate autonomous systems into their military contracts. Other similar systems include autonomous missiles and the christening of the world's largest autonomous submarine last November 15th.

NG Contact: Ryan Tintner
Senior Engineer
895 Oceanic Dr. MS9035
Annapolis, MD 21401
e-mail: ryan.tintner@ngc.com
Phone: (687) 637-8210

Our secondary sponsor is the Underwater Robotics Club (URC) here at North Carolina State University. They were started in the fall of 2004 and their primary purpose for being is to create a machine capable of competing in the annual AUVSI Autonomous Underwater Vehicle competition.

URC Contact: Alex Stewart
Burlington Labs Rm 1130
e-mail: amstewa2@ncsu.edu
Phone: (919) 513-0638

Problem Statement

The original problem provided by our sponsor states that our mission is to work on the "development of a hardware independent communications API." However, it turns out after getting together with both of our contacts and our teachers the primary problem was rewritten. Our redefined mission is, "to aid in the ongoing development of the SeaWolf II and fill needs as defined by the Underwater Robotics Club and Northrop Grumman."

Project Goals & Benefits

The original project goals listed by the sponsor were fairly vague and as it turns out many parts had already been completed. Therefore, as per our new primary purpose, we sat down with all of our stakeholders and settled on a new set of goals.

Development of a more user friendly and intuitive GUI: The current GUIs display all of their information, besides collected digital footage, in pure text format. Though this information is highly accurate it is very difficult to internalize in a real time environment. Moreover, two separate GUIs are currently in use. The Telemetry GUI is used to monitor the SeaWolf and the Control group to impute desired headings and values. The new GUI will integrate the two previous systems as well as displaying most of the information with more dynamic visuals.

NG Benefits: NG sees the SeaWolf II project as a long term investment. During the course of development many demonstrations may be required. The current text based GUI means very little to those unfamiliar with each individual value being displayed. However, a professional grade visualization of much of the data would more accurately convey exactly what the SeaWolf II is capable of doing.

URC Benefits: Even to those who work directly with the SeaWolf II on a regular basis, much of the numerical data is confusing and difficult to accurately visualize. Therefore, as listed above, a new more ergonomic GUI will enhance the team's ability to test, use, and understand exactly what the SeaWolf II can do.

Universal Benefits: There is a great sense of pride invested both by the URC and NGC in the SeaWolf II. A GUI which displays all of the necessary information in an understandable, elegant, and beautiful format will enhance the usability and satisfaction for all users. Finally, a highly polished GUI is likely to garner more attention at events and competitions which is essential for the continued funding and thus longevity of the SeaWolf project.

Development of a simulated testing environment: This is not an improvement or expansion on any currently existing program or feature of the SeaWolf II. The purpose here is to create an ultra simplified testing environment that primarily allows for testing the efficiency of possible search algorithms needed to complete the AUVSI Autonomous Underwater Vehicle competition.

NG Benefits: Since the display for this particular simulator will be highly graphical it will serve many of the same functions as the GUI. This means that at demonstrations it will be possible to show a reasonable representation of what the SeaWolf II would do in at an actual competition without having to have the vehicle physically present.

URC Benefits: The new simulator will allow for algorithm testing even when the SeaWolf II is not working, making traditional testing impossible. Additionally, if this simulator can be expanded by future teams it may eventually allow the URC to test every function of the SeaWolf II without ever needing to enter a pool.

Feasibility Issues

There are no known feasibility issues at this time.

Resources Needed

Adobe Illustrator: In order to create an acceptably aesthetic GUI an art program will be needed to generate the base imagery.
3DS Max or equivalent 3D modeling package: In order to generate 3D models for the simulator a 3D design software will be required.

Requirements - Testing

Functional Requirements

FR 1:

FR 1.1 GUI Component: Artificial Horizon

Purpose: The artificial horizon component will provide the user with information about the current roll and pitch of the UAV.

Appearance: The artificial horizon component will consist of a circular display divided horizontally. It will be based on the artificial horizon found in most aircraft. The angle of the horizontal line will represent the current roll of the UAV, with a completely horizontal line indicating that the UAV is level. The vertical position of the horizontal line will represent the current pitch of the UAV, with a centered line indicating the vehicle is level.

Implementation: The artificial horizon component will be implemented in C# using built-in UI components and methods. Relevant data will come from the **roll** and **pitch** members of the **TelemetryDataClass** class. Using input from this class a horizontal line will be drawn on the component with the appropriate angle and vertical position.

FR 1.2 GUI Component: Compass

Purpose: The compass component will provide the user with the current heading of the vehicle.

Appearance: Like many of the other TGUI components, it will be based on designs found in aircraft. A semicircular window will display the current heading of the UAV in the center of the display. As the UAV changes heading the display will "roll" to indicate the new heading of the vehicle. Value for north, south, east and west will be represented by N, S, E, and W respectively. Appropriate hatch marks should be added between these four values to aid in navigation.

Implementation: The compass component will be implemented in C# using built-in UI methods and components. The compass will receive heading data from the **heading** member of the **TelemetryDataClass** class. Using this data the current heading will be displayed in the semi-circular display window.

FR 1.3 GUI Component: 3D Orientation Display

Purpose: The 3D Orientation display will provide the user with a 3D display of the craft showing the orientation of the craft in the water.

Appearance: The craft orientation display will give a view of the vehicle from a camera that has a static position relative to the craft. The camera will face in the north direction through the center of the vehicle.

Implementation: This will be implemented using a 3D model of the vehicle in obj format. The model will be read and a view of the craft using the telemetry data obtained by the sensors will be constructed.

FR 1.4 GUI Component: 2D top down view giving vehicle status

Purpose: The purpose of the 2D display is to give status information such as engine rpm, and engine direction.

Appearance: The 2D view will be a 2D top down view of the SeaWolf that will give the status information by text near the physical structure that the status information is about. In addition, the information will be given using colors appearing on the view of the SeaWolf.

Implementation: A 2D picture of the SeaWolf will be displayed. Text will be and colors will be displayed on top of this to give the information.

FR 1.5 GUI Component: Battery Gauge

Purpose: This will give the current amount of energy left in the battery.

Appearance: This will be represented as a colored bar. The size of the bar will give the amount of energy left in the battery.

Implementation: The battery state will be read from the SeaWolf. This value will be used to draw a bar in the GUI that will represent the amount of energy left in the battery.

FR 1.6 GUI Component: Vehicle Internal Humidity

Purpose: The purpose is to display the internal humidity of the SeaWolf on the GUI to make sure to much moisture is not near the electronic components.

Appearance: A bar will be drawn on the GUI that will represent by length the amount of humidity in the SeaWolf.

Implementation: The humidity will be read from the SeaWolf and a bar will be drawn in the GUI.

FR 1.7 GUI Component: Camera Display

Purpose: The Camera Display will show the images received from the two onboard cameras.

Appearance: The TGUI will show two boxes, a box to show the view from the forward-looking camera, and one to show the view from the down looking camera.

Implementation: The camera data will be received from the SeaWolf. This data will be scaled to the appropriate size. It will then be displayed in a box in the GUI.

FR 2:

FR2.1 Goal: Construct a graphical modeling system that interfaces with the vehicle's code to provide an out-of-water testing environment for search algorithms.

FR 2.2 Requirements:

- The system should provide environmental feedback that is usually gathered by the vehicle's sensors and move a virtual model of the vehicle through the artificial environment according to commands from the vehicle's search algorithms.
- This system will not include complex physics models
- The system should be expandable for future work with physics models, real-world environment modeling based on sensor input, etc.

FR 2.3 Implementation: The graphics for the virtual model will be built using OpenGL. An existing SolidWorks model of the SeaWolf II will be simplified so that it may be used in real time rendering. 3d models of obstacles and mission goals will be built from scratch. All models must be textured. Physics with the simulation will be grossly simplified. Turn and climb rates will be considered to be constant. Complex physics issues such as drag and buoyancy will be completely ignored at this stage.

Non-Functional Requirements

NFR 1: All code should integrate with existing code base that is written in C#.
NFR 2: GUI should be professionally presentable for international competition that vehicle is entered into.

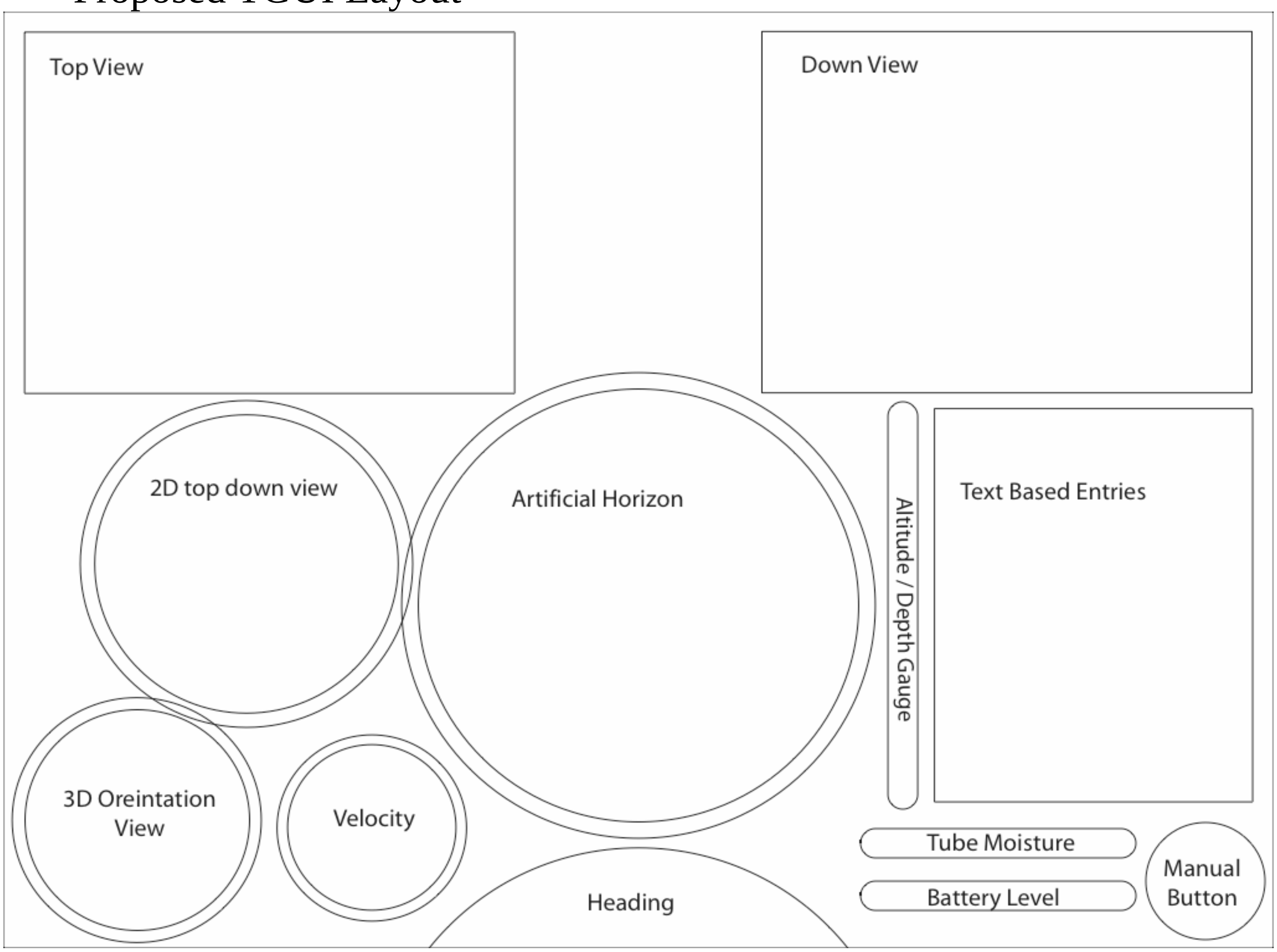
Design Choices of Interest: When planning the GUI the team decided that a graphical representation of the orientation of the SeaWolf Autonomous Underwater Vehicle (AUV) would give the viewer of the telemetry information a quicker and more intuitive way to assess the orientation of the AUV. Orientation data allows the operator to understand how the vehicle is performing programmed tasks and if a failure has occurred. This information can then be used to improve the programming, which specifies and controls the underwater vehicles motions and actions in the water, allowing an increase in the task success rate, under changing and difficult conditions.

The two main ways of displaying orientation data that were considered are two 2d views, or one 3d view of the SeaWolf AUV. The three 2d views would be: a top view, side view, and a view from the back of the AUV. The top view would display yaw information, the side view would display pitch, and the back view would display roll. By looking at these three views a viewer of the telemetry displays could assemble a mental 3d view of the current orientation of the AUV.

The single 3D view would simply display a 3d view of the SeaWolf from the view of a camera maintaining a fixed position relative to the AUV and always pointing north through the center of the AUV. The viewer of the 3D view would obtain an immediate understanding of the orientation of the AUV.

Although a series of 2D views would have been simpler to implement in the software, a 3D view was chosen as the view to include in the GUI because, the 3D view allows a quicker understanding of the orientation of the AUV.

Proposed TGUI Layout



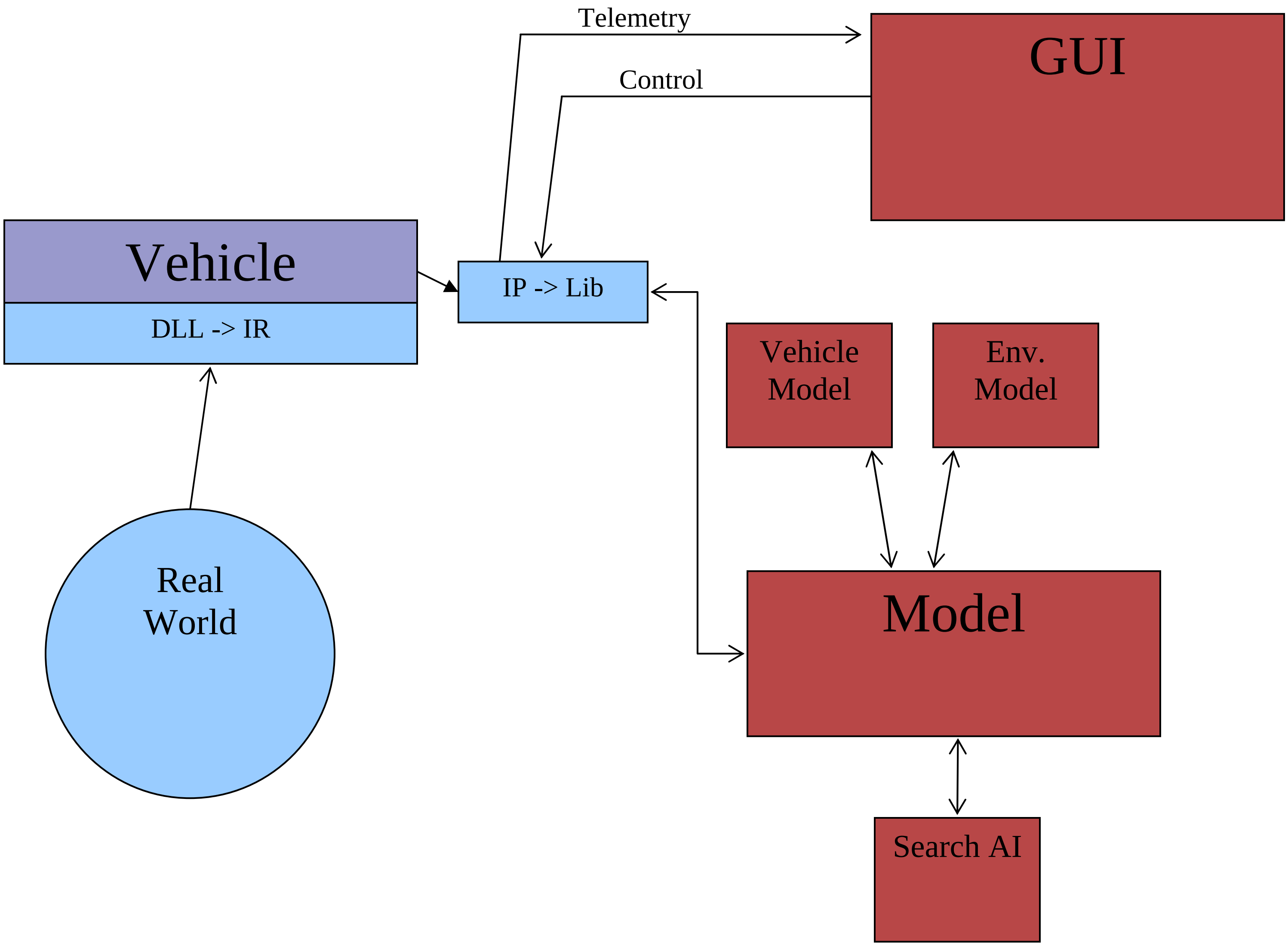
Top View: This space will display the video feed from the front facing camera on the SeaWolf II.
Down View: This space will display the video feed from the down facing camera on the SeaWolf
2D top down view: This gauge shows a colored top down view of the SeaWolf II. Values for the current angle of the engines and thrust values will be displayed in this section. Additionally non-responsive devices will be marked at their location by color shifts in the model.
3D Orientation View: This gauge is centered on the SeaWolf II and faces north. The aim of this gauge is to give the user a quick reference of the current orientation of the SeaWolf II.
Velocity: This gauge shows the absolute velocity of the SeaWolf II.
Artificial Horizon: Displays roll and pitch values. Additionally, located mission targets will be represented in this screen or by arrows pointing to in their direction.
Heading: Displays a compass which shows the current heading of the SeaWolf II.
Altitude / Depth Gauge: This gauge features a single fixed midpoint with moving horizontal bars above and below which simultaneously display the depth and altitude of the SeaWolf II.
Tube Moisture: Displays the internal moisture level inside of the SeaWolf II.
Battery Level: Displays the current battery level.
Manual Button: Engages manual control mode by activating two additional panels which slide in to the screen.

Plans for Control GUI Layout:
One of our project goals is to integrate the Telemetry and Control GUI's. In our current model when manual control is engaged two additional control panels slide into the screen.

Top Control Panel: This panel fits into the currently blank space between the two video displays and allows the user to enter target values for heading, depth, altitude, and thrust. Additionally it has the shutdown and reboot commands.

Side Control Panel: This panel slides in over the 2d top down view because it displays angle and thrust values for individual thrusters which is redundant to what the top down view displays.

Model Design



As shown, the GUI provides a way for the user to view telemetry data and provide control data to the communication interface that the vehicle talks to. While running normally, the vehicle receives real world data directly through its sensors and then communicates with the GUI.

When running the model, a virtual vehicle and virtual environment are passed through the model to search algorithms that control the virtual vehicle according to virtual data from the virtual sensors. All of this is shown to the user in a 3d model world that shows the virtual vehicle and virtual environment. Also, the model's virtual data is passed to the GUI for viewing and optional control.

Task Plan

Task	Team Member(s)	Date	Status
Oral Presentation I	Sam	February 05	Completed
Preliminary GUI Design	ALL	February 9	Completed
GUI Design and Approval	Michael	February 16	Completed
Build TGUI Components	All	February 23	In Progress
- Artificial Horizon	Sam		
- 3d Viewport	Todd / Neil		
- GUI Face Plate	Michael		
- Customizable Text Components	Percy		
- Depth / Altitude Gauge	Percy / Sam		
- Miscellaneous GUI Gauges	Percy / Sam	February 28	Not Started
Assemble TGUI Components	Percy / Sam	February 28	Not Started
Build CGUI	Percy / Sam	March 9	Not Started
Build Basic Model World	Todd / Neil	February 28	Not Started
- Show vehicle, water surface, tank bottom, and tank sides			Not Started
- Provide virtual depth, altitude, and velocity information for TGUI			Not Started
Add Obstacles	Todd / Neil	March 9	Not Started
Add AI Test Structure	All	March 23	Not Started