EMBRY-RIDDLE AERONAUTICAL UNIVERSITY

SmartACE ASV

Steve Seyler, Sid Menon, C. Randy Breingan, John Li, Santiago Iglesias

6/11/2012

This journal paper describes the design and functionality of the Autonomous Surface Vehicle designed and built by Embry-Riddle Aeronautical University for their entry into the 5th International RoboBoat Competition hosted by AUVSI and ONR. The Embry-Riddle team has come up with new and innovative designs in order to complete the challenges in the competition such as an improved hull, flexible motor mounts, robust and reliable buoy navigation, and an Android based sub-vehicle.

1 Introduction

The Robotics Association at Embry-Riddle is pleased to present *SmartACE*, a new ASV system designed to compete in the 2012 RoboBoat "The five card draw" challenge. The goal of the *SmartACE* team is to complete the three mandatory tasks (strength, speed and channel navigation) and also to complete the tasks posed by of each the four challenge stations (poker chip, jackpot, cheater's hand and hot suit). To achieve these goals, the *SmartACE* team developed innovative approaches to solve the challenges specific to this competition. The final design of the *SmartACE* ASV includes a stable, maneuverable tri-hull boat platform, a safe and reliable power and propulsion system, an array of sensors and an on-board processor. The system also includes specialized tools, including an autonomous sub-vehicle, a button bumper, a thermopile infrared sensor, and a water cannon~~,~~ for completing the four challenge stations.

2 Hull Design

SmartACE ASV is designed to outperform the previous platform (Seagle 4.0) in both stability and turning speed. It features a tri-hull configuration with the center hull bearing most of the load and the two exterior hulls providing stability. The two outside hulls draft one inch less water than the center hull to lessen drag and to provide counter forces that resist pitch and roll. The leading edge of the center hull is angled thirty degrees relative to the water for both hydrodynamics and for docking against the ramp specified in the poker chip challenge.

Propulsion is provided by a pair of Seabotix thrusters. The thruster mountings are designed to be attached to the outside hulls and to be adjustable along their length. The wide separation of the thrusters improves the turning rate and hence the maneuverability of the vessel in tight quarters. The ability to adjust the thruster location along the length of the outer hulls allows the location of the center of thrust to be controlled to fine tune the performance as vehicle subsystems are developed and modified. In the current design, the thrusters are 23.5 inches apart versus 16 inches in the previous design. This feature provides an additional 6.1 lb-ft of turning torque with the identical Seabotix thrusters. The test results presented in Table 1 confirm the theoretical calculations. *SmartACE* is lighter and more maneuverable than its predecessor, Seagle 4.0.

Table 1: Performance Comparison

Vehicle	Total Weight	Forward Speed	Reverse Speed	Time to Complete a 90° Turn	Time to Complete a 180° Turn	Static Thrust
Seagle 4.0	54.6 lbs	3.3 ft/s	3.1 ft/s	2.6 s	5.7 s	12.9 lbs
Smart ACE	46.1 lbs	3.3 ft/s	3.3 ft/s	1.1 s	2.1 s	12.9 lbs

3 Sensor Suite

A collection of four sensors is used to navigate the buoy channel, locate challenge stations and complete challenge tasks. The goal in selecting sensors was to provide necessary perception, but to avoid extreme complexity. Figure 1 depicts a flow diagram of how data is collected through the system. The sensors on the left hand side send data to the onboard processor. The output is sent to the motors and in the case of the Cheater's hand challenge, the output is sent to the water cannon as well.

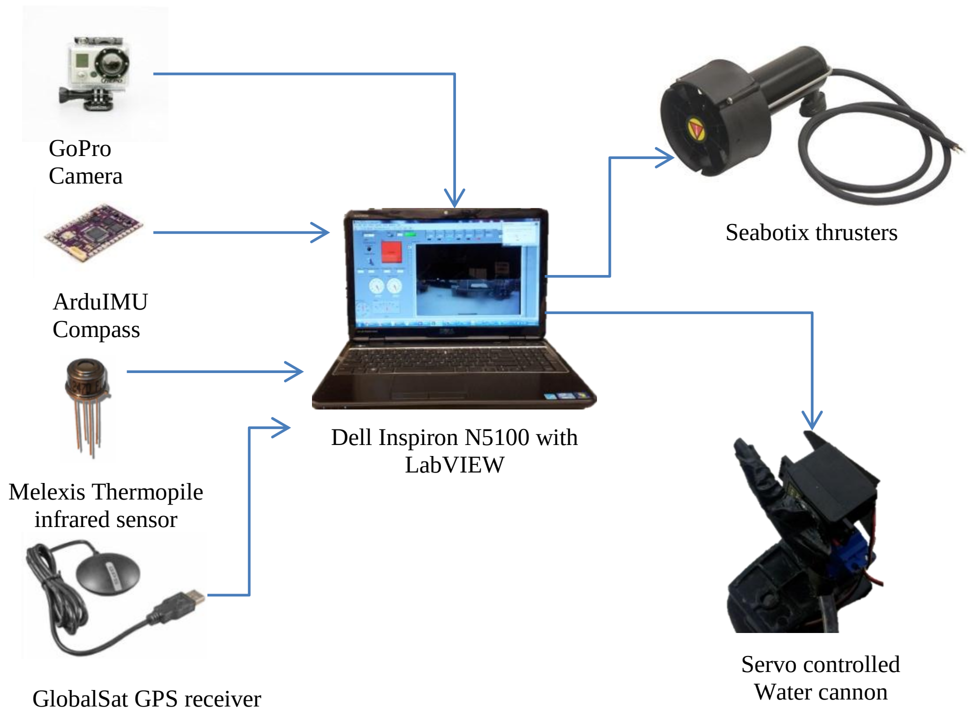

Figure 1: Sensor diagram

4 Buoy Identification & Navigation

4.1 Buoy Identification

The primary difficulty associated with navigating the buoy channel course is unambiguous identification of the buoys using computer vision and image processing. The system must work reliably, regardless of lighting conditions, specular reflections from the water and background noise.

To find buoys, images are processed using the hue-saturation-luminance (HSL) color model. Instead of extracting an entire color plane before doing additional processing, a range of HSL values is used to help separate the buoys from the image. Each color buoy will have a unique set of HSL values that vary slightly depending on the lighting conditions. By comparing images of the same buoy under different lighting, a range of HSL values are constructed for the four buoy colors (red, green, yellow and submerged white). Searching for these ranges cuts down on the number of steps used in the image processing and provides a simple way to adjust the processing to identify buoys.

4.2 Vision Algorithm

Vision processing is performed using the NI LabVIEW graphical programming environment. The LabVIEW Vision Toolkit was used to construct a basic algorithm. The

Vision Toolkit allows the user to apply different filters and particle analysis tools to the GoPro camera image while displaying the processed image. This provides a step-by-step look into the image processing and makes it easy to adjust the filters and troubleshoot. Once satisfied, the algorithm developed in the Vision Toolkit can be imported into the primary navigation and control algorithms, also developed in LabVIEW.

4.2.1 Buoy Detection

The algorithm begins with applying a color threshold. This threshold only allows pixels that fit within the empirically determined range for each buoy color to pass the filter. Using this threshold produces a binary image. Pixels that do not fit within the HSL range are zeros, and pixels that match the filter are labeled as ones. In the LabVIEW image processing, the pixels marked as zeros are black in the binary image, and pixels that are ones are red. When a buoy of the target color is found in the image, there will be a high concentration of red pixels in its location. The HSL ranges used in the threshold are shown Table 2.

Table 2: HSL Ranges used in buoy detection algorithm

	Red	Green	Blue	Yellow
Hue	0	80	122	25
	16	130	255	60
Saturation	160	75	100	90
	255	255	255	255
Luminance	40	25	30	135
	130	130	152	255

Typically, an image with a buoy will have large "blobs" that have passed through the filter and also smaller positive particles that passed the threshold that are not buoys. To remove these false positives, a particle filter is applied twice in order to remove small particles. The binary image is scanned with a nine pixel by nine pixel block, with the center pixel being the one of interest. If less than half of the surrounding eight pixels are the same of the center picture, then the value of the center pixel is defined as a zero. This means that pixels with very few adjacent pixels that have also passed the HSL threshold are converted to zeros and removed from the binary image. This step successfully leaves only the buoy of interest in the binary image.

The third and final step in the algorithm is a particle analysis that determines the center of the buoy, based on the average position of all of the pixels that have passed the filter. Determining the center point of the buoy allows for it to be placed in the vehicles frame of reference. Performing this algorithm for all of the buoys simultaneously creates a set of points on the image, representing the center of the each of the buoys that make up gate. Averaging these centers generates a point in the center of the gate, which can be used by the navigation and control algorithms as a drive point.

An example of these steps is shown in Figure 2. An image with both a red and a green buoy is analyzed with the binary output of each step shown.

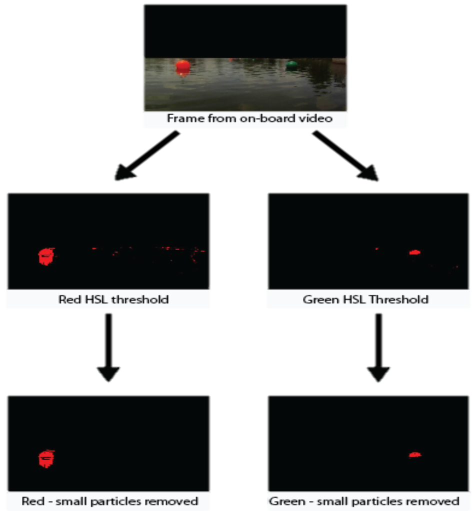

Figure 2: Buoy detection algorithm with binary outputs of each image filter.

To navigate through a given buoy gate, the vehicle attempts to align the center horizontal pixel of the image with the center horizontal coordinate generated by the vision algorithm. When a yellow obstacle buoy is present and simply driving through the center of a gate is not possible, a new drive point is determined by choosing the largest gap in the channel: either between the green buoy and yellow boy or between the red buoy and yellow buoy. This new drive point is then used to navigate. For either situation, the algorithms are performed for each frame received from the camera. This means that the vehicle is always reacting to its new position and the position of the buoys. The competition course does not contain any hard turns or trap situations in the channel, so the vehicle can navigate purely reactively. This makes the navigation extremely simple, streamlining the process and saving computing power for other competition tasks.

5 Challenge Stations

To navigate from the end of the channel to the Challenge stations, *SmartACE* uses a combination of GPS waypoints and compass headings from the blue buoy marking the end of the channel. The vehicle is capable of navigation with just a compass, to adapt to changing course layouts.

5.1 Target Identification – The Cheater's Hand

The target identification for the "Cheater's Hand" secondary task works off of the basic steps used in the buoy detection algorithm. Images are taken from the vision system and a filter is applied that produces a binary image that only allows pixels that fit within a designated HSL range. The second step, just like the buoy detection, applies a filter twice that removes small particles. The same nine by nine pixel technique is used.

The third step is where the target identification begins to vary. The image is scanned for a specific pattern within the binary image. The pattern it looks for is shown in figure 3. The scale of the pattern is ignored and the algorithm only looks for the pattern. The pattern matching also picks up shapes that are not the actual blue box. In order to filter out the other patters, the pattern with the highest score (closest match to the original pattern) is kept and everything else is discarded.

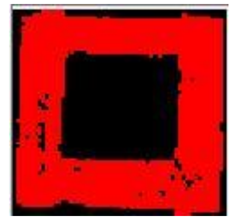

Figure 3: Pattern template

Once the square has been detected, a motor algorithm similar to that of the buoy navigation is used. However instead of maneuvering between two buoys, the algorithm tries to move the vehicle so that the blue box is centered with respect to the camera. As soon as the blue box is detected, the servos of the water cannon begin aiming the cannon at the blue box. As the boat moves closer to the blue box, it slows down proportionally and once it reaches a distance of 5 feet the motors shut off completely. At this point, the water pump is turned on and it oscillates in the vertical direction so as to ensure that the target is hit. Once the camera detects the raised flag, the state changes to GPS navigation and the vehicle navigates to the next waypoint. Figure 4 shows the original image of the blue box and the final result of the image after it has been processed. Figure 5 shows an image of the boat firing the water cannon at the target during testing.

Figure 4: Mock-up of “cheater’s hand” target (left) and the binary output (right) of the image processing developed to find the target.

Figure 5: Cheater's hand testing

5.2 Target Identification – The Jackpot

The logic behind the completion of “The Jackpot” station is similar to that of the buoy channel navigation. Images are pulled from the GoPro video feed and filters are applied that produce a binary image that only displays pixels that fit within a given HSL range. When the vehicle gets in the vicinity of the e-stop buttons, the vision system searches for the red buttons by applying an HSL filter to the images and navigates in their direction, with the center of the e-stop button set as the drive point. To determine which button to press, the vision system runs a secondary analysis, searching for the submerged white buoy. Testing at Embry-Riddle has shown that a white buoy submerged six to twelve inches underwater in a pond or lake environment is still visible from the surface. The vision analysis runs an HSL filter first to determine possible buoys, followed by a pattern match to eliminate false results. When the correct e-stop button is determined, the boat drives to the e-stop and turns right once it is close enough. The distance at which it turns is determined by the size of the e-stop in the image from the camera. This allows the button bumper to sideswipe the e-stop button. Figure 6 shows an image of the boat sideswiping the e-stop during testing.

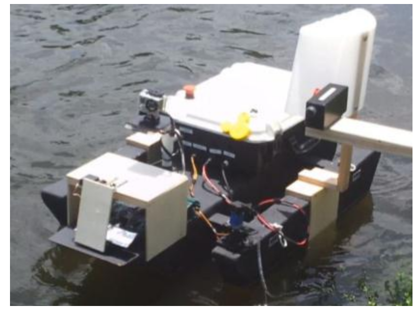

Figure 6: The SmartACE pushing the e-stop button

5.3 Target Identification – The Hot Suit

Completing the “Hot Suit” begins with the vehicle detecting the signs and navigating towards them. Closing in on the signs will allow the vision system to determine the suit on the sign by applying an HSL filter, followed by a pattern comparison. This set of image processes determines the position of each of the four suits. To determine the temperature of the targets, the servos used to aim the water cannon for the “Cheater’s Hand” are used to aim the infrared sensor. The vehicle scans each of the four signs and finds the raised temperature using the Melexis thermopile sensor. The sensor is aimed at the target and the voltage output is read using the same ArduIMU used to read the compass heading.

The suit is recognized using the camera and the GPS location is determined using the boat’s current GPS coordinates, the heading of the hot suit with respect to the boat and the approximate calculated distance of the suit by scaling the observed suit in the camera image. The GPS location is then reported back to the ground station through the onboard computer and transmitted using the required TCP protocol.

5.4 Poker Chip

In order to complete the poker chip challenge, there is a sub-vehicle on board. The sub-vehicle was designed and built using VEX Robotics kit and uses an Android computer to navigate itself onto the dock and find the poker chip. The sub-vehicle is contained on the front of the boat within a special housing. The front door of the housing is servo actuated. When the boat reaches the dock, it will open the door and the Android device will begin navigating the sub-vehicle to the chip when it visually recognizes that the door is open. The sub-vehicle has a front panel covered in Velcro so as to pick up the chip. Once the chip is retrieved, the sub-vehicle will return to the boat and wirelessly signal the

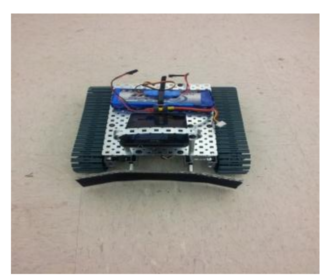

Figure 7: Sub-Vehicle

onboard system of its return. Figure 7 shows the sub-vehicle with the Android device mounted.

6 Testing

The algorithms were tested in several stages. Preliminary testing was performed on the computer, where the filters and other vision processing steps were performed on test images and logged video. The primary method of analysis was examination of the binary image outputs. The filters were tweaked until they produced useful results for all lighting conditions and reasonable distances.

Once the filters were proven successful on the test images, they were implemented into a test code that included active motor and servo control. Initially the code was run with no hardware attached. The signals that would be sent to the motor controllers were monitored as log videos were run. This was done to monitor the outputs and how they changed based off of the drive point determined through the analysis of the video. Hardware was then hooked up for hardware-in-the-loop testing and the thrusters and water cannon servos were actuated based off of log video. The actions were monitored and vehicle reactions were verified. Once this was confirmed, the log video was switched out for a live feed from the GoPro camera in the lab. The vehicle was stationary on a cart, and buoys were held in front of the camera and moved manually to generate responses by the vehicle to “navigate” through them. While one team member moved the buoys, another monitored the LabVIEW front panel and thruster power/direction to confirm that the vehicle was indeed attempting to move in the correct direction.

The final stage of testing was performed on the water. A pond next to the Embry-Riddle ICI Center was outfitted with buoy gates. The vehicle was placed on the water and remote-controlled in various orientations before being switched into autonomous mode and allowed to navigate the mock channel. The vehicle was monitored visually during the run, and the logs were post-processed to check for performance accuracy. The video logged during the on water tests can be run through the test code discussed in the earlier test stages to see the exact values that the vehicle was outputting.

Video was logged in the morning, mid-day and early evening as well as several different weather conditions including sun, overcast and rain to provide an extensive catalog with lighting conditions covering all possibilities at competition. This data was used to build HSL filters that detect buoys in all of the conditions without the need of adjustment. Failure due to changing lighting conditions has plagued the competition entries of years past, so creating an algorithm that works regardless of the weather was an important design requirement set by the team this year.

7 Results

One of the main objectives of the project was to create an algorithm that successfully identifies buoys of different colors, determines their location relative to the vehicle, and navigates through a channel of the buoys without missing a single buoy gate. This algorithm was designed with the goal in mind of being able to run the code in any weather condition without alteration.

Examples of the algorithm's success during a cloudy and rainy condition and a sunny condition are shown in Figure 8.

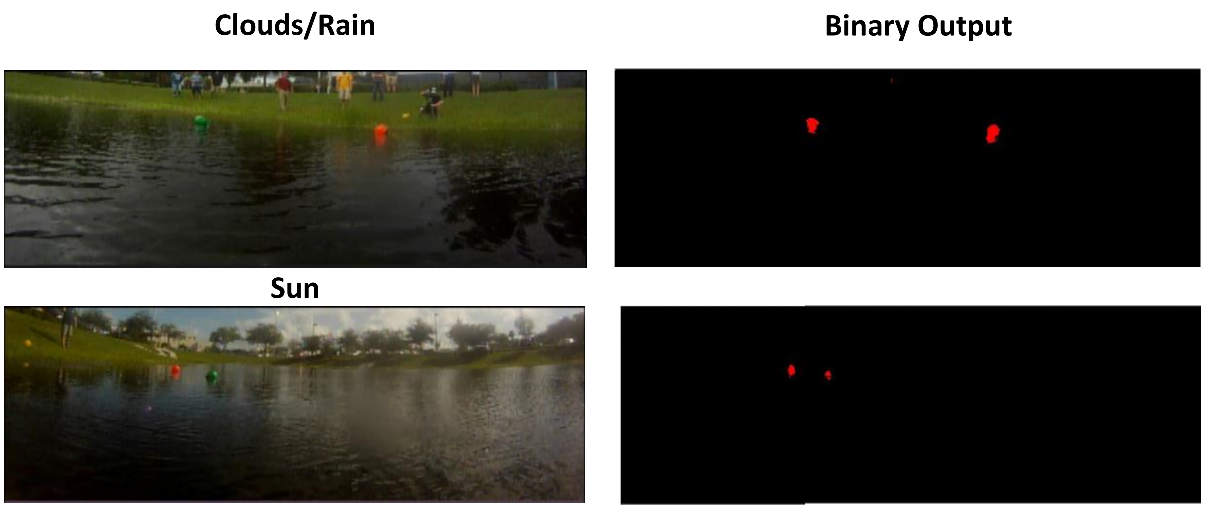

Figure 8: Raw video and binary output of buoy detection algorithm for red and green buoys in cloudy weather and sunny weather.

The images show that the algorithm is capable of finding buoys in extremes of both sunny and cloudy weather. Glare during sunnier conditions makes less of the surface of the buoys detectable, but the buoys are still visible at a distance farther than any that will be encountered at competition.

The video logs taken during different weather and lighting conditions were used to determine the success rate of the algorithm. Frames where buoys were visible and within 50 feet (maximum distance in competition) were run through the test algorithm and the binary output was examined. If the channel marker buoys were visible in the binary image and had an assigned pixel coordinate, then the frame was considered a success. If the buoys were not visible in the binary image or if the coordinate location marked an object that was not a buoy, the frame was considered a failure. The results are shown in Table 3.

Table 3. Table of the success of the algorithm in detecting buoys.

		Frames	Detection	Success Percentage
Morning	Sun	100	89	89%
Mid Day	Sun	100	92	92%
Late Day	Sun	100	91	91%
Mid Day	Clouds	150	146	97%
Late Day	Clouds	150	143	95%
Total		600	561	94%

The results show that the algorithm detects both buoys 94% percent of the time when both are present in the image. The algorithm is successful 96 percent of the time when the buoys are under overcast skies. Detection in sunlight is less successful but still occurs around 90 percent of the time. Performance during autonomous tests where this data was collected shows that the success rates above are high enough to navigate the successfully navigate the buoy channel. If only one buoy is seen, the boat navigates to either the left or right side of it depending on the color of the buoy and the direction that the boat is travelling.

The algorithm developed for the blue square detection is not as successful as the buoy detection algorithm. Frames from test video were taken at distances from 5 to 25 feet. The water cannon and pump setup on the vehicle has a maximum range of 16 feet, but being able to detect the square from further out will help to line up the vehicle before it has to fire the cannon. The success of the algorithm at the various distances is shown in Table 4 below.

Table 4. Table of the success of the algorithm in the blue square target.

	Frames	Detection	Success Percentage
5 Feet	40	39	98%
10 Feet	40	37	93%
15 Feet	40	32	80%
20 Feet	40	25	63%
25 Feet	40	14	35%

The algorithm performs very well within the first 10 feet, before dropping off significantly over the other 5 feet increments. Since the water cannon cannot reach targets beyond 16 feet, the plan for competition is to have the vehicle within 10 feet to shoot water at the target, so the lower detection rates at range will not be much of a factor.

8 Conclusion

SmartACE has shown through lab tests and on-water test runs that it is capable of attempting and successfully completing the tasks in this year's RoboBoat competition. Through rigorous testing and improvement based on previous years' entries, the entry for the 2012 RoboBoat competition is the most reliable and capable system delivered by Embry-Riddle to date.

9 References

2012 Roboboat Competition Final Rules,
http://higherlogicdownload.s3.amazonaws.com/AUVSI/fb9a8da0-2ac8-42d1-a11e-d58c1e158347/UploadedImages/2012%20RoboBoat/RoboBoat_2012_final_rules.pdf

Acknowledgements

The Smart Aces would like to thank Dr.Charles Reinholtz, Dr.Patrick Currier, Dr.Eric Coyle, Nancy Do, Shanice Jones, Robert Roeder and Karl Ritz for their support on this report.