The University of Texas at Arlington RoboBoat

Tony Le, Ngoc Tran, Kristine Trujillo, Adebola Akeredolu, Koby Benefiel, John Benson, Michael Donaldson, Vanessa Duran, Fabian Estrada, Susan Kum, Anthony Madrid, Walter Mulflur, Leslie Rogers
Team Advisors: Manfred Huber, Brian Huff

Abstract
This will be the first time for the University of Texas at Arlington (UTA) RoboBoat team to compete in the AUVSI Foundation's RoboBoat competition. As a new team, much of the focus of the team this year has been on the design of the hardware for the competition challenges. To achieve its goal, the team has focused on a subset of the overall tasks and challenges, namely the channel navigation, the catch the ball challenge, the 'Rock, Paper, Scissor, Lizard, Spock' station, shoot through the hoops, and the return to dock task. To address these tasks, a new boat with a steerable front and rear engine was designed and built for maximum maneuverability. Similarly, a special rover for ball retrieval, a steerable pneumatic dart gun for shoot the hoops, and a pan-tilt hybrid heat-camera / vision system for the sign identification task were developed and built. The individual hardware components were developed by sub-teams and then integrated on the boat. Control software was developed using Windows and RoboRealm for the individual challenge components and Linux and ROS for the boat control and overall task integration.

Introduction
The RoboBoat competition is a student robot competition sponsored by the Association for Unmanned Vehicle Systems International (AUVSI) Foundation as a way to promote interest in autonomous vehicles research and development. In this contest, which requires fully autonomous vehicles, each team has to perform a number of mandatory tasks as well as a number of optional challenges, providing for a very challenging, multidisciplinary design and control problem involving advanced hardware and software.

For the University of Texas at Arlington (UTA) RoboBoat team, this is the first time to compete in this event and as a result much of the team's effort in this year has focused on designing and building hardware for the mandatory tasks and a sub-set of the challenges. To achieve this, the team is formed in a multidisciplinary from a set of students and faculty advisors from a number of engineering disciplines, including Mechanical Engineering, Industrial Engineering, Electrical Engineering, and Computer Science. Among the student participants, the majority are undergraduate Juniors and Seniors with a strong interest in robotics.

In this year of competition, the UTA RoboBoat team has focused its effort on the main boat design needed for the mandatory and navigation tasks with a strong emphasis on designing a flexible, highly maneuverable platform that can serve as the basis not only for this year but also for subsequent years of the competition. In addition, sub teams were set up that focused on three of the five challenges in this year's competition, namely: i) catch the ball, ii) 'Rock, Paper, Scissor, Lizard, Spock', and iii) shoot through the hoops. For these challenges, three additional hardware components were developed that are integrated on the UTA RoboBoat in the form of an autonomous rover, a pan-tilt hybrid thermal camera / vision system, and a steerable pneumatic dart gun. The systems for the challenges are designed to be largely independent, relying on their own sensor and control systems in order to allow for parallel development and robust integration on the RoboBoat platform. Similarly, the software components for the control of the individual hardware platforms is designed to be largely independent, coupling solely through the task and challenge sequencing mechanism integrated with the RoboBoat platform. This design allows software components to utilize different programming paradigms. In the case of the UTA RoboBoat, control of the challenge components utilizes RoboRealm [1] on Windows for image processing and higher-level control decisions while the main boat uses the Robot Operating System (ROS) [2] on Linux for navigation, control, and task sequencing. This hybrid approach permits to take advantage of the simple software development of RoboRealm for the challenge components and of the flexibility and simulation capabilities of ROS for the boat and integration. In the following, this report will present and discuss the different hardware and software components of the UTA RoboBoat.

The UTA RoboBoat

The UTA RoboBoat consists of four major components that had to be designed and implemented in this first year of the team's participation in the RoboBoat competition. These major hardware components are: i) the boat, ii) the rover for the catch the ball challenge, iii) the hybrid camera system for 'Rock, Paper, Scissor, Lizard, Spock', and iv) the dart gun for shoot the hoop.

The Boat

The main boat platform for the RoboBoat competition has to have a number of capabilities in order to be able to perform the mandatory and the navigation tasks as well as to support the three challenges that were selected by the UTA RoboBoat team for this year's competition. The main capabilities required here are: i) good maneuverability, ii) vision capabilities for navigation and docking tasks, iii) GPS capabilities for identification of challenge locations. To address these challenges, a new boat design was developed that is aimed at maximizing maneuverability, a set of sensors are integrated, and a control system is developed that allows for the efficient and modular control of the components as well as for an efficient integration with and coordination of the separate challenge hardware and software components described in later sections.

UTA RoboBoat Hardware Design

To optimize maneuverability in particular with aim at the required docking actions, the UTA RoboBoat team decided to design a holonomic boat platform that allows for movement in all directions. To achieve this ability, which is common on most larger real-world ships to allow for parallel docking, the boat was designed as a pontoon boat with two steerable motors mounted towards the front and in the rear of the boat. This design allows the platform to produce thrust in two different directions through the two motors, facilitating turning in place as well as sideways movement (although at reduced speeds due to a reduced stability of the boat in this direction). To facilitate the latter capability, the encased propellers of the boat are mounted partially below the support pontoons, avoiding sideways thrust to be captured (and thus eliminated) by the pontoons themselves. While this increases the complexity and the overall height of the boat and results in the propeller housings to be the lowest points on the boat (which makes transport and cart design more difficult), the advantages in terms of navigation flexibility were deemed to be sufficient to warrant the design. Figure 1 shows the basic boat design as well as the implementation of the motor mount and steering mechanism during boat construction.

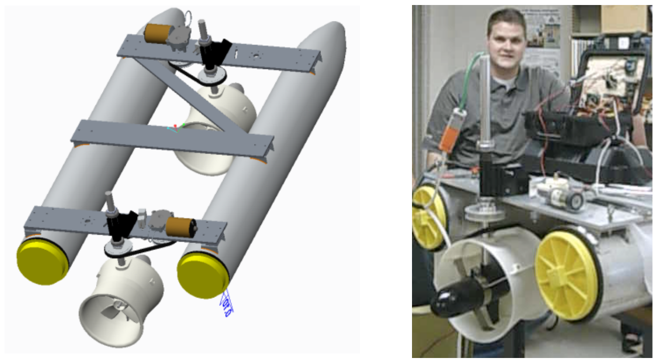

Figure 1: UTA RoboBoat holonomic boat design (left) and actual boat (right)

UTA RoboBoat Sensing, Control, and Computing Architecture

The tasks of the boat platform are to perform all the navigation tasks and to serve as the host platform for the challenge hardware components. As a result of this, the computing system associated with the boat will be used not only to control the sensors and actuators of the boat but also to coordinate the task and challenge sequence with the other, modularly designed hardware and software components. The overall sensing, control, and computing can be seen in Figure 2.

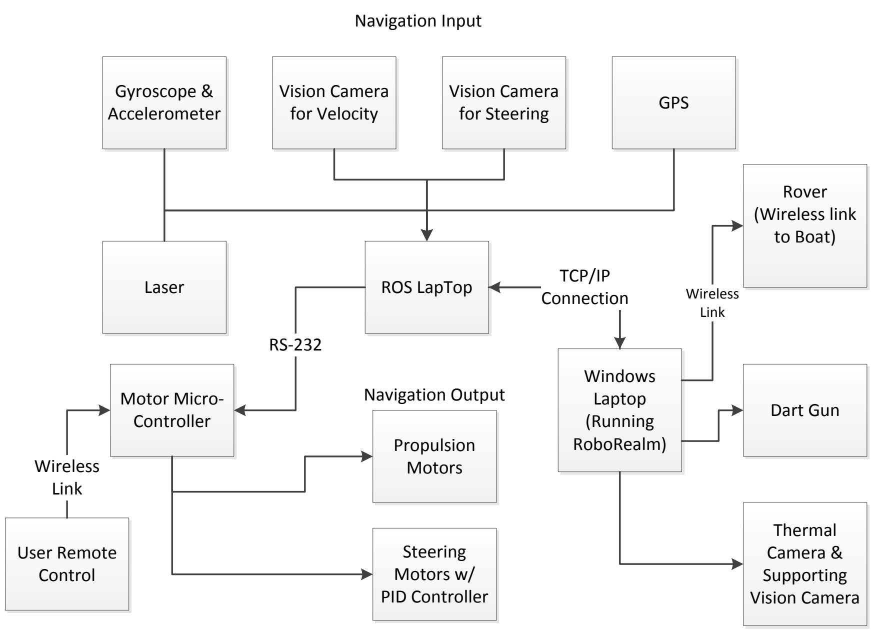

Figure 2: RoboBoat sensing, control, and computing architecture

This architecture is comprised of two laptops. One laptop, responsible for boat navigation, docking, and challenge coordination is running Ubuntu with the ROS [2] environment on top. For the navigation tasks, a static navigation camera as well as a GPS unit and compass are used (Note that the laser sensor shown in the diagram was not implemented on the actual boat due to time limitations). For the challenge platforms, as secondary laptop running Windows is used to

facilitate simpler development of the vision processing using RoboRealm [1], a simple vision processing and object tracking software with a relatively low learning curve. To achieve coordination between the different tasks and the challenges, a coordination process in the ROS system is used that communicates with the Windows computer (and the challenge processes) over a wired Ethernet TCP/IP connection.
The low level motor control on the boat is handled by a separate microcontroller connected over RS-232. This microcontroller is responsible for low-level PWM and PID control to drive the two large motors used for propulsion and the two smaller motors for steering (see Figure 1).

Boat Control Software

Low-Level Control: At the low level, it is important to be able to independently control the speed of the propulsion motors as well as the orientation of the two steerable engines to facilitate maximum navigation flexibility. To achieve this, the large motors are to operate in an open loop velocity control mode as the motors do not provide motor speed feedback (and because the information gained from better motor speed control was not considered sufficiently valuable to warrant the cost of the necessary feedback hardware). For the steering motors, on the other hand, PID controllers are implemented using feedback from a one to one encoder. This is necessary to keep the boat on the intended path. Figure 3 shows a diagram of the low level controls.

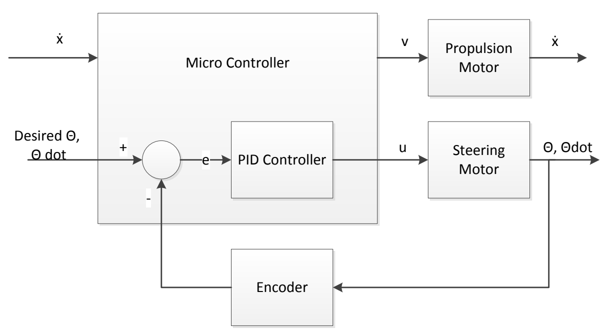

Figure 3: Low-level motor control

Navigation Control: For navigation control, three modules are implemented in ROS, one handling the boat dynamics [3][4], effectively translating desired movement directions into commands to the four boat motors, one to handle the navigation vision-based navigation for the sped and obstacle navigation tasks, and one for GPS-based open water navigation to reach the destination for the challenge tasks. For the vision-based navigation tasks, a visual servoing framework was chosen over path-planning due to its lower complexity and high robustness.
To be able to develop these capabilities while the boat was still under development and to allow for efficient testing, a simplified simulation model has been developed using the ROS/Gazebo system. While this simulation does not completely accurately reflect the complex dynamics of the actual boat platform, it useful to initial algorithm development and testing (see Figure 4).

The Rover

The rover for the UTA RoboBoat team was initially designed for the originally posted object retrieval task involving a tennis ball. To address this, many different design considerations had to go into the design of the autonomous rover: i) the rover has to fit onto the boat, ii) the rover had to be able to remove the ball from the deck, and iii) the rover had to securely grab the ball and leave no opportunity for it to fall out. In addition, attempts were made to keep the design as

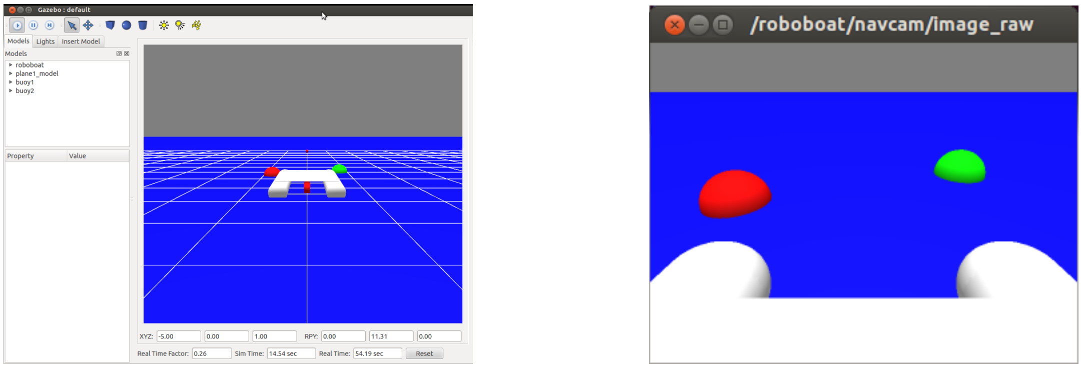

Figure 4: ROS/Gazebo simulation of the boat platform (left) simulated vision (right)

simple as possible. The rover is designed to leave the boat platform, drive down a ramp from the boat onto the ramp on the dock, find and pick up the disc, and finally return the way it came.

As the final rules replaced the ball with a Velcro-covered disk, small changes had to be made to allow the design to operate (as it had been designed around the task of having to lift a ball off of the dock) and were completed by adding a different grabbing surface inside the grabber.

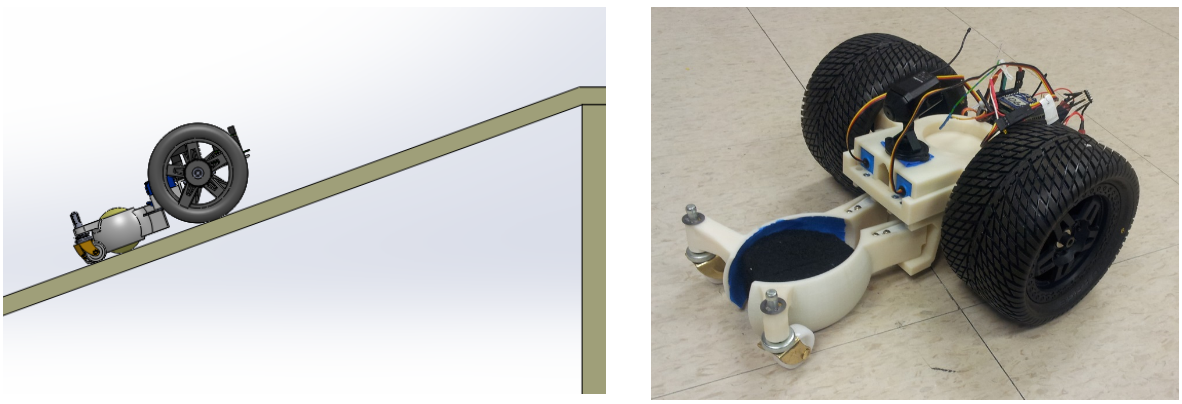
Figure 5: Rover design on ramp (left) and assembled rover modified for the puck (right)

Figure 5 shows the rover design and the assembled rover. The main components of the rover can be broken down into the frame and mechanical design, the electrical components, and the programming. Different team members worked on the different components, which were then integrated and assembled to make the rover.

Mechanical Design

The rover has large rear wheels, each powered by a brushless dc motor. This helps with the steering since each wheel can be controlled independently. Also, the large wheels help to deal with any bumps or dips the rover may face while disembarking the boat, retrieving the disc, or returning to the boat. The motors powering the wheels can give the rover a high speed, but can also be controlled by the electrical system to slow it down for accuracy.

A grabber arm design was used for the rover initially be able to pick up the ball but now serves to easily grab the disc. The grabbers will be open when the rover approaches the disc and then close over the disc. This will securely hold the disc inside of the grabbers of the rover. To adjust the grabber (which was designed for a tennis ball) to the Velcro-covered disk, it has been lined with Velcro to make sure the disc stays in the grabber. To ensure that the rover will not drag the grabber on the ground and to increase stability, casters have been put on the grabbers that swivel with the rover's movements. The grabber arms are powered by independent servo motors.

The main rover frame was designed around the electrical components, such as the motors, batteries, grabbers and electronics. This gave the frame an odd shape, but helped to secure the grabbers, motors and servos in place. Since the shape of the frame was irregular, a 3D printer was used to manufacture the frame of the grabbers. To add stiffness to the frame and the grabber arms, stiffeners were designed into the frame to strengthen the rover.

Electrical Components

To achieve its task, the rover needs many different electrical systems, including a camera and image processing to find the disc and navigate. To power the different systems, such as the camera and the motors, the rover also needs a power and motor control system.

For the camera, a small security camera is used that transmits the signal wirelessly from the rover to a receiver on the boat, where the image will be processed on the computer. The camera needs to a good transmitter because it has to transmit the signal from the rover on the dock to the boat which could be nearly 10 feet away. In addition, the camera has to be small enough to fit on the rover, but still give a good quality image so that the rover can find the disc. To power the camera, a separate 9 volt battery is used which can easily be connected and unplugged.

To run all the different motors and electronics, two lithium polymer batteries that can each deliver 11.1 volts are used. In order to avoid voltage drops from the motors, one battery is used to power the motors and servos, and the other battery powers the different electronics. Since the servo motors only take a maximum of 7.4 volts, two battery eliminator circuits (BEC) are used.

To help with the control of the drive motors, two H-bridges are used.

Programming

To control the rover for the challenge, two different programs are used to process the visual input and to control the motors. To control the motors, a program called NMITerm made by New Micros Inc. [5] is used. This program can be used to program the H-bridge with preset commands. Then, when a preset command is typed into the program on the computer, the H-bridge will let the predetermined voltage through to the motors, causing the rover to move, or the servos on the grabber to pick up the disk. The computer has a wireless USB transmitter that sends the command wirelessly to the H-bridge which has a receiver in it.

To process the camera IMAGE, RoboRealm functionality is used to find the ball in the image and send the output commands to the NMITerm program. RoboRealm can filter out colors, and calculate the center of gravity of the color component, all of which is used to help the rover find the ball. A diagram of how the programming and electronics are integrated is shown in Figure 6.

Here the camera obtains the image and transmits it wirelessly to a receiver which is connected to the computer. The computer then uses RoboRealm to process the image while running a Visual Basic programming script. The RoboRealm software will then send commands to the NMITerm on the computer, which will, in turn, transmit the commands wirelessly to the H-bridge on board the rover. The H-bridge then performs the commands by regulating the voltage to the motors causing the rover to move and perform the desired tasks.

Dart Gun

The structure of the gun was designed considering various constraints and requirements imposed by the AUVSI rules and by the limitations of other system components such as the servomotors and pneumatic actuation system. The goal of the design was to create a compact, nearly self-contained firing system and parts were selected and designed accordingly.

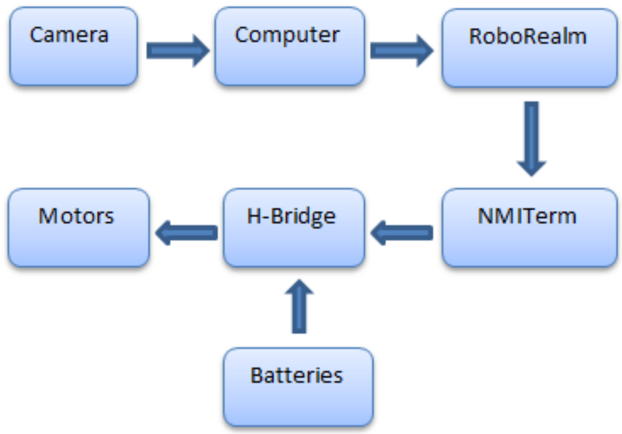

Figure 6: Diagram of the program and electronic interface

Mechanical Design

The primary restrictions that were considered, in terms of the structure of the firing system, was the total allowable height, weight and overall footprint. To include the gun on the boat for the shooting mission, the entire gun system could not extend to a height of more than about twelve inches from the boat deck; otherwise, the boat's overall height would be past the restrictions. Due to the overall boat weight constraints, the gun design was such that the least amount of weight was added from the gun's systems.

Though there is an overall height restriction, other dimensionality constraints were placed upon the gun as more subsystems were added to the whole design. An example of this is in the height of the yoke. The yoke is tall enough to allow for all of the desired pitching motion of the barrels without interfering with any other features like the camera mount and the bottom of the yoke or the back of the gun barrels and the valves. Another design consideration was regarding the control and firing mechanism. Due to the limitations on weight and range, the firing mechanism uses small CO2 cartridges controlled through an electronic valve. Together these considerations led to the gun design shown in Figure 7.

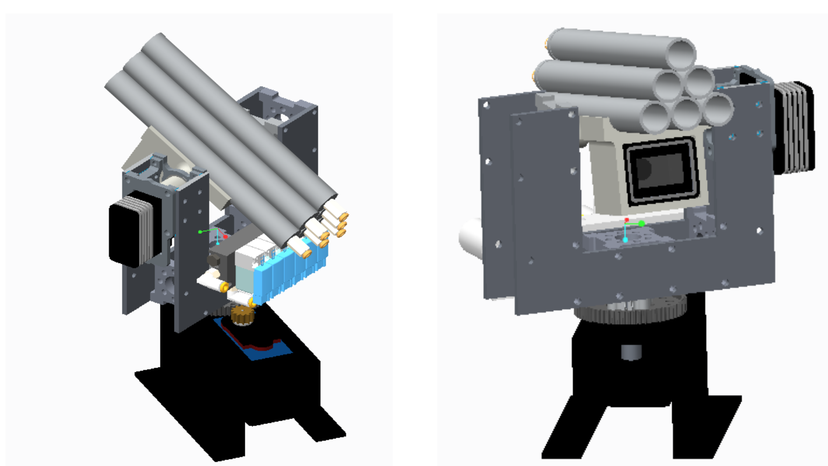

Figure 7: Dart Gun; back view (left) and front view (right)

Equipment Placement

The compact nature of the gun and the overall footprint became an important consideration due to the limited amount of space on the boat deck. Real estate with other challenge systems such as the rover became key in the optimal location of the gun on the deck. The placement of the gun w as plotted for the back of the boat to provide a clear shot at the hoops while leaving the front of the boat open for the rover and for the boat navigation camera.

The pneumatic manifold is placed on the back of the yoke. This reduces the need for unnecessary tube length between the valves and the CO2 canister. The manifold placement also eliminates the yaw motion resistance that additional tube stiffness would create for the gun. The CO2 canister and regulator will be located on the boat deck and have a single tube feeding CO2 to the valve manifold. Figure 8 shows the CO2 container and the solenoid used in the dart gun manifold.

Figure 8: CO2 container (left) and solenoid (right) used in the dart gun

Programming

To program the dart gun and the vision system mounted on it, a similar approach as for the rover has been taken except that the camera for the dart gun is directly connected to the Windows computer on the boat since it is static. Again, a microcontroller is used to perform the servo control for the dart gun orientation and RoboRealm software is used to find the hoops and compute the correct orientation for the dart gun to shoot the dart through the hoop.

Hybrid Thermal / Vision Camera

To address the requirements of the ‘Rock, Paper, Scissor, Lizard, Spock’ challenge, it is necessary that the designed system can recognize the signs of the signs as well as which of the signs is hotter than the others. In order to do this, the UTA RoboBoat team designed a pan-tilt system with vision capabilities in both the thermal and the visible range. To achieve this while staying within the weight and size requirements of the RoboBoat competition, a system has been designed around existing camera hardware.

Camera Hardware

Thermal Camera: The thermal camera selected was chosen because of its ready availability. Given the modular design built into the system, any camera with thermal capabilities should suffice. The camera in use is the DRS Tamarisk 320 series camera [6]. The camera’s specifications are largely unknown due to the secure nature of the company which manufactures the camera. The camera itself has a defined resolution of 320 pixels by 240 pixels (320x240) and outputs data in NTSC 480i format. From observation, it was also found to have a scaling threshold function built into it that is when an object is found to be hotter than other hot objects in view, the hottest object will appear the brightest and the others darker.

Visual Camera: The standard camera for the visual identification part of this project was chosen because it had been selected for the navigation task of the project as well and will be reused for this task. The reason for its selection also comes from the fact that RoboBoat's sister team is already utilizing this particular camera and certain code can be adopted from them to reduce the amount of new work needed. The camera selected is the Sony EVI-D70 camera [7]. The camera has matched resolution to the thermal camera, which is necessary for the image processing scheme to work. The camera also has pan, tilt and zoom functions.

Camera Mounting: Since the task requires the robot to both identify which target is hot in and what the target is, it both cameras have to operate at the same time and be calibrated with each other. To achieve this the cameras were placed in a manner such that both trained on the same point in space with minimal distortion due to parallax effects. This scheme was chosen for a variety of reasons. First reason is speed, by having both cameras being processed simultaneously. Second is simplicity, by having both cameras trained on the same point in space, there is no need to "remember" where an image was in space in order to move the other camera to it. Thirdly is efficiency, as will be explained further in the methodology of the final process, this setup can provide for a better rejection of false positives as it does not look for "hotspots" but rather processing hotspots seen. This allows the camera to be swept across a range processing only things of interest, rather than having to find hot points to process. Figure 9 shows the setup of the hybrid camera system in an evaluation setting.

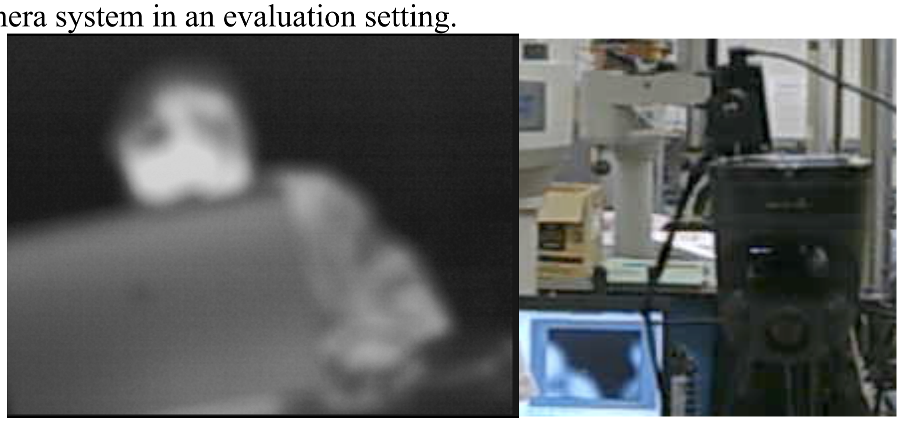

Figure 9: Pan-tilt hybrid thermal / vision camera setup (left) and thermal image (right)

Programming

As in the case of the other challenge components, RoboRealm software on the Windows computer is used to perform the image processing on both the thermal and the visual image. To keep things uniform both for proper operation and for modularity, the video size was reduced down to the maximum value of one camera. This means that the sizes of both cameras remain fixed to one value. This is vital due to the way that the images were to be processed. The value picked was 640x480. This gave ample resolution for processing.

Image Processing

To determine the hotter sign and identify its symbol, a number of image processing steps are necessary and implemented in RoboRealm.

Filtering and Thresholding: While the raw data provides a very good thermal picture of the environment, there are still far too many hot points to be effectively usable for the scheme devised for target identification. As such a threshold filter is applied to the raw data to discard outlier points. The minimum threshold value was found to be ideal at 150 per experimentation.

Center of Gravity Identification: In order to ensure that only the hot object is processed for image detection, everything not around the hot point is removed from the image. To identify what portion of the image should be removed, RoboRealm's built in Center of Gravity (COG) function that determines were the greatest collection and most intense value of pixels is located is used on the thermal image. COG also identifies a box which can include various percentages of pixels of the image. The default value of 80% was found to provide adequate framing.
Image Switching and Cropping: To identify the object, the image from the visual camera is cropped according to the COG box for image identification.
Image Identification: The final step in the Image processing pipeline is to take the cropped image and begin object identification on it. Again, RoboRealm has a specific function that uses pre-recorded images of the signs to do this. The function also produces a confidence value which can be used to further screen for false positives. The process also has the ability to detect items even if they are further away than the test picture is.

Conclusion

The UTA RoboBoat team was formed as a multidisciplinary student team covering various engineering disciplines. As this is the first year in which the team will compete, the design effort has focused on the design of a new boat for maximum maneuverability and a set of three challenges, for each of which a hardware component has been developed. To allow for the simultaneous development of all of the hardware component, the hardware and software was designed to be modular such that components can be evaluated separately before integration on the boat. This also allowed the use of different software platforms with Windows and RoboRealm used for ease of development on the challenge platforms, and Linux and ROS for power, integration support, and simulation abilities on the boat platform and for task integration.

Acknowledgments

The UTA RoboBoat team would like to acknowledge the generous support provided by the University of Texas at Arlington Research Institute (UTARI) and by QinetiQ North America, without which the team's participation in this competition would not have been possible. The team would also like to thank Drs. Robert Woods, Dan Popa, and Gian-Luca Mariottini for their invaluable assistance and guidance on different aspects of the development.

References

[1] RoboRealm Web Site, http://www.roborealm.com
[2] ROS Web Site, http://www.ros.org
[3] C. Sonnenburg. Control-Oriented Planar Motion Modeling of Unmanned Surface Vehicles. Virginia Center of Autonomous Systems. Blacksburg, VA, 2010.
[4] Jan Petrich. Improved Guidance, navigation and Control for Autonomous Underwater Vehicles: Theory and experiment. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2009.
[5] New Micros Inc. Web Site, http://www.newmicros.com/
[6] DRS Tamarisk Specifications, http://www.drs.com/products/RSTA/Tamarisk.aspx
[7] Sony EVID70 Specifications, http://pro.sony.com/bbsc/ssr/product-EVID70/