KITT (Knights Intelligent Turbo Trawler)
Robotics Club at UCF
Technical Design Report

Jade Zsiros
Boat Lead
jadezsiros@knights.ucf.edu

Matthew Kurtz
Team Lead
matthewkurtz@knights.ucf.edu

Wesley Fletcher
Quadcopter Programming
wkfletcher@knights.ucf.edu

John Millner
Quadcopter Programming
millnerjohn@gmail.com

Matthew Nipper
Computer Vision

Nicholas Califano
Quadcopter Mechanical Design
nickwcalifano@gmail.com

Nicholas Peters
Boat Electronics
npeters2014@knights.ucf.edu

Andrew Schroeder
Boat Perception
andrew.schroeder@knights.ucf.edu

Abstract

The Robotics Club at UCF refurbished and revitalized a boat used in prior years to compete at the International RoboBoat Competition. The boat's name is KITT (Knights Intelligent Turbo Trawler) and consists of two pontoons, four propellers, a large "egg" on the top of the vehicle that stores the electronics, and a "nest" that attaches to the top of the vehicle that secures the Unmanned Aerial Vehicle (UAV). The team used the Robot Operating System (ROS) native navigation packages to maneuver between the dock and the tasks, and we developed in-house perception algorithms to complete each task.

Competition Strategy

The team's competition strategy is based around the quadcopter challenge, due to its high point values and the team's interest to work on an aerial robot. The alterations to and programming of the boat are focused on the completion of the UAV and other advanced challenges.

Design Creativity

KITT's body was refurbished from a previous competition robot, so the design creativity of the body and hull was limited. A nest was constructed on top of KITT as a landing pad for the drone (called Lakitu). This landing pad has an Aruco marker and guard rail to ensure Lakitu stays attached to the boat until takeoff and after landing.

The programming of the boat was rebuilt from no legacy code using ROS. At the start, the robot uses the provided GPS waypoints to navigate to the Docking challenge using the ROS Occupancy_Map feature, and switches into a separate mode for processing the information for the specific challenge.

In this mode, KITT processes LiDAR scans to make progressively accurate estimates of the location of the dock (repeatedly self-checking). We take a slice of each LiDAR scan and randomly select 10 points at a time to find a line of best fit using linear regression. From thousands of lines, we seek the line that intersects with the most points and save it, checking it against each new scan.

Using the most recent scan of the dock, we take the remaining points and use random sampling to seek the attached buoys and extrapolate the locations of the three docks (**Figure 1**). Hydrophones are used to listen for the pinging dock, and after navigating to it, the boat backs away and releases the quadcopter, Lakitu.

Figure 1: Buoy Detector

Lakitu is constructed from Delrin and carbon fiber to reduce the weight of the UAV while maintaining frame integrity. Lakitu implements the Mavros package, which handles tasks such as sending commands to the motors, auto-stabilizing the copter to ensure stability, as well as navigating in straight lines when given a map and a target. This allows us to focus on the higher level tasks such as localization, landing, the state machine, and computer vision.

In order to accomplish various aspects of flight, we use a state machine that allows us to switch between preflight, takeoff, hover, navigation, and landing states. This allows for modularity and ease of debugging. If something is incorrect in one state, it does not affect how the other states operate.

To detect the seven segment display, Lakitu creates a bounding box around what it believes the display to be, and then it looks inside said box to determine which segments are activated (**Figure 2**). From this information, we can gather what numbers are displayed and relay that to KITT, who uses the value to seek the appropriate dock and navigate towards it.

Figure 2: Number detection

In order to land on the boat, we utilize Aruco markers on the landing pad, which give us the pose in reference to the camera indicating where to land. Then, Lakitu will interpret this pose and navigate and hover until it initiates a slow descent over the Aruco marker (**Figure 3**).

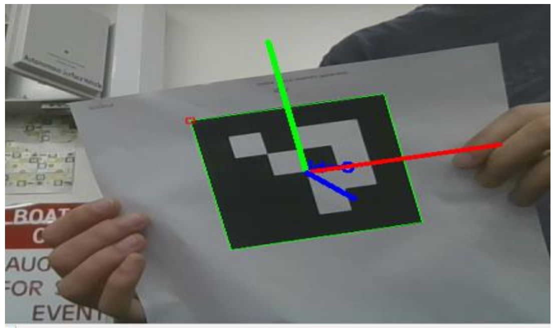

Figure 3: Aruco Marker

Experimental Results

During practice test flights of Lakitu, it was discovered that the high rate of spinning from the motors caused vibrations that shook loose the outer prop guards and the standoffs holding them in place. In response, the team decided that these guards should be removed for added safety.

To test the flotation of Lakitu, the team detached the electronic components from the frame and replaced them with a piece of metal twice their combined mass. The frame with weight was placed into the water repeatedly, from various angles and distances. When twenty consecutive dunkings proved the quadcopter buoyant, testing was complete.

In order to test the landmark detection, the team built a graphical output for our landmark detection program and simulated the course by bringing buoys and dock-shaped objects into our build space. We also put our boat on a cart and rolled it around the parking lot, ensuring it detected bollards as it rolled past.

Acknowledgements

For enabling the continual thriving of the Robotics Club at UCF, we would like to acknowledge the following individuals and groups:

The Institute for Simulation and Training, which provides us with our build space and continued funding and support.

Crystal Maraj, our faculty advisor, who acts as a liaison between our group and various outside groups, advises us, and keeps us on track.

The Peters family, for their generous financial contribution to the club this year.

For assisting in the completion of this project, we would like to acknowledge the following individuals:

Eddie Grekoski, the previous captain of our quadcopter team, initialized the group and designed the frame that is currently in use.

Stephen Sumner, the previous captain of our boat team, initialized the group and integrated the quadcopter and boat.

References

"PX4 Development Guide". Web. Retrieved from https://dev.px4.io/en/

"ROS Wiki: Documentation" Open Source Robotics Foundation. Web. Retrieved from http://wiki.ros.org/

"Spatial Reference Manual" Advanced Navigation. 2017. Web. Retrieved from http://www.advancednavigation.com.au/sites/advancednavigation.com.au/files/spatial_reference_manual.pdf

"Find a Linear Regression Equation: By Hand or in Excel". StatisticsHowTo. Web. 2018. Retrieved from: http://www.statisticshowto.com/probability-and-statistics/regression-analysis/find-a-linear-regression-equation/

L. Sanchez, A. Mahtani, E. Fernandez, A. Martinez. Effective Robotics Programming with ROS, third ed. Packt Publishing. 2016.

M. Gorelick and I. Ozsvald. High Performance Python. O'Reilly Media. 2014.

P. Norvig and S. Russell. Artificial Intelligence: A Modern Approach, 3rd ed. Pearson. 2009.

S. Thrun, W. Burgard, and D. Fox. Probabilistic Robotics, Intelligent Robotics and Autonomous Agents series edition. The MIT Press. 2005.

Appendix A: Component Specifications

Component	Vendor	Model/Type	Specs	Cost (if new)
ASV Hull	Prior UCF Roboboat team	Pontoon boat, hull made of fiberglass with aluminum supports.	Pontoons 48” long, 8” diameter. Egg 27” diameter, 6” tall.	N/A
Power system	Homebrew			
Motor controls	Robotech	SDC2130	https://www.roboteq.com/index.php/component/virtuemart/279/sdc2130-detail?Itemid=970	$190
CPU	Intel	i7 3770s	https://www.cnet.com/products/intel-core-i7-3770s-3-1-ghz-processor-series/specs/	$100
Teleoperation	Futaba	7C-2.4GHz	http://manuals.hobbico.com/fut/7c-manual.pdf	$170
Compass/IMU/GPS	Advanced Navigation	Spatial	http://www.advancednavigation.com.au/product/spatial#specifications	$3500
LiDAR unit	Velodyne	VLP-16	http://velodynelidar.com/vlp-16.html	$4000
Camera	Logitech	Webcam Pro 9000	http://support.logitech.com/en_us/product/webcam-pro-9000/specs	$90
Hydrophones	Teledyne	TC4103	https://www.m-b-t.com/fileadmin/redakteur/Ozeanographie/Hydrophone/Produktblaetter/TC4013.pdf	$1000 x 5
Aerial vehicle platform	Homemade	Homemade	5 hula hoops, plywood strip, spray paint	$6
Motor and propellers	Generic	Purchased before this team existed, type unknown.	Example: https://www.ebay.com/itm/36-Trolling-	$150 x 4

			Motor-50lbs-12V-Electric-Transom-Mount-Freshwater-Fishing-Boat-New/351265411795?hash=item51c90cd2d3:g:MuYAAOSw6lRaZuBK	
CPU- Quadcopter	Odroid	XU4	http://www.hardkernel.com/main/products/prdt_info.php	$59
Algorithms	RANSAC, SLAM	Homebrew, ROS respectively	To find the dock, we take a slice of each LiDAR scan and randomly sort those points into groups of 10. Using linear regression, we create a line of best fit for each of the groups of 10 points and see which line is closest to the most points in the cloud. The best line is saved and compared with the best line from the next scan, so that only the best line ever found is saved.	To find the buoys, we take a slice of each LiDAR scan and subtract points that are likely to be on the dock. We then sample 3 points at a time and transcribe a circle from each group. Circles with a buoy-like radius that intersect with many points are kept and compared with the circles in the next scan, so only the best buoys ever found are saved.
Localization and Mapping	Homebrew		To navigate between the dock and the obstacles, we use ROS navigation packages and occupancy grids. However, in each task, we use RANSAC with our LiDAR unit to create a library of landmarks important to each task, which are passed to the quadcopter when necessary.	
Autonomy	Homebrew, ROS			
Team Size	9	2 boat, 7 quadcopter		

Expertise Ratio	3:6			
Testing time in simulation	2 hours boat, 3 hours UAV			
Testing time in water	4 hours			
Inter-vehicle communication	Wi-fi			
Programming language	Python			

Appendix B: Outreach Activities

As the Robotics Club at UCF is not limited to only the Roboboat team, we have numerous opportunities for community outreach. Every semester, with our sponsor the Institute of Simulation and Training, we conduct demonstrations with our robots and simulations to local youth STEM-related groups in an event called STEM day held at UCF.

Also this year, we demonstrated our robots to three groups of Boys and Girls Club students in a program designed to inspire teens to pursue college degrees.

In order to facilitate our outreach efforts, we have a separate team called Demobot that builds robots of varying complexity and function that are intended to be taken to outreach events. These robots are designed for interactivity with crowds, particularly children, with enough novel functions to provide talking points for a demonstration. This Demobot team is rather freeform and has no deadlines, and is intended to help new and inexperienced members of our club learn robotics concepts from zero.