Beohawk: Autonomous Quadrotor

Ryan Goldstein, Keith McKay, Yujia Zhai, Russel Stewart, Arti Hira, Christopher Li, Reese Mozer, Divakar Singamsetty
University of Southern California

ABSTRACT

In this paper we introduce a Micro UAV system that can explore an unknown indoor space without the assistance of GPS in order to compete in the 6th IARC mission. The robot receives sensor measurements from cameras, sonar, accelerometers, gyroscopes and an infrared depth sensor. Using Simultaneous Localization and Mapping, it handles the data probabilistically and generates a map of the environment, which is used for avoiding obstacles and generating waypoints. The mechanical construction of the vehicle, low-level control, and sensor communication are also discussed in the paper.

INTRODUCTION

Problem Statement

For the 6th International Aerial Robotics Competition, an autonomous aerial vehicle under 1.5 kilograms must enterer an unknown office floor through a window, localize itself according to the environment, identify and pick up a black USB drive, and return to a handler within 5 or 10 minutes, depending on covertness. The robot may also be intelligent enough to identify mission-related features, such as signs above rooms, or observe and avoid obstacles such as surveillance devices while traversing the hallways.

Conceptual Solution

The high-level solution of the mission that will be integrated into the mission planning algorithm is as follows:

1. Take-off and localize robot with respect to the window.
2. Wait for blue LED on security camera to go out of view and proceed through window.
3. Explore office while avoiding obstacles.
4. Identify surfaces above the ground and search for USB drive.
5. Pick up USB drive, drop the dummy drive, proceed towards exit.
6. Exit through window and land.

Figure of Overall System Architecture

The USC Aerial Robotics Team (ART) purposes a quadrotor design of its aerial robot vehicle, *Beohawk*, which consists of four motors located symmetrically at each end of an X-shape frame with fixed-pitch propellers attached. Various sensors, the on-board computer, and other facilities are located at the center of the frame. An off-board computing station is dedicated to the majority of computational tasks; it receives data from the on-board computer, processes the data to determine future movements of the robot, and publishes operating commands back. The communication is done over a standard 802.11n network. Both computers run mission-specific programs written by our team. These programs run on the Robotic Operating System (ROS) and take advantage of many packages and tools made available by public contributors. Utilizing inertial and visual data from various sensors and processing the data with state-of-the-art algorithms, we hope to perform stabilization, navigation and mission planning on the quadrotor to complete the challenge. Figure 1 shows the basic system architecture of the quadrotor. The quadrotor's low level control including stability, attitude control, altitude control, and position control are all performed by the low-level control board. This board is Arduino based and has sensor inputs and motor outputs. It also receives radio-control signals that allow control to be over-ridden by a human pilot.

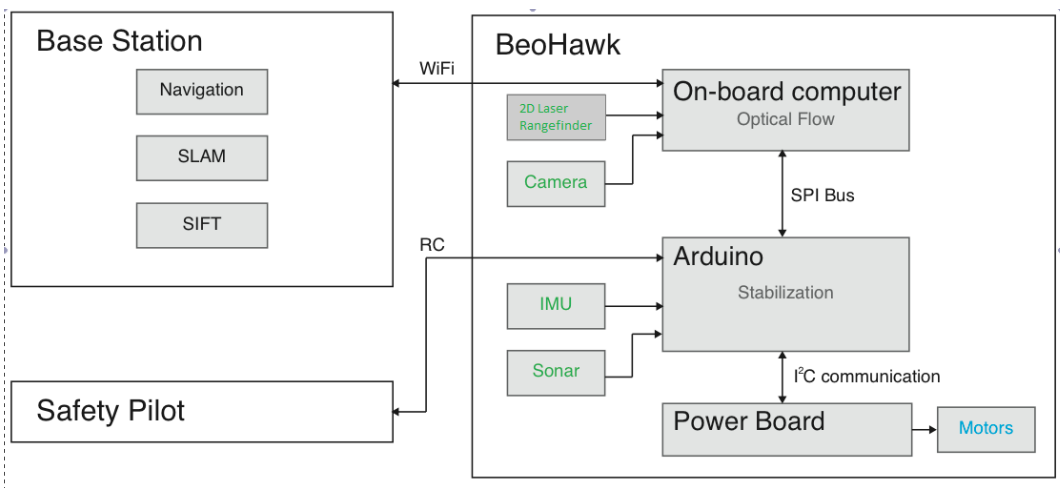

Figure 1: *General architecture of the Beohawk control system.*

Yearly Milestones

This is the USC Robotics Society's second appearance at IARC. Continuing on our progress from last year's vehicle, our goal for the vehicle we bring to the 2012 competition is fully autonomous navigation through the window and exploration of the office space. Our hardware team has developed the fourth iteration of the frame, making improvements in durability and weight reduction. The software team has implemented algorithms for control gain optimization, optical flow, and SLAM. Our electrical team has designed a customized circuit board to replace the commercialized alternatives (such as Arduino board). Our post-competition work, regardless of the outcome, will focus on integrating higher performance electronics, obtaining more accurate models of the vehicle's dynamics, and improving the high level mission control algorithm.

AIR VEHICLE

Propulsion and Lift System

Beohawk has four rotors, each an equal distance from the vehicle's center, attached to four brushless outrunner motors (figure 2). Two opposite motors spin clockwise and the other two spin counter-clockwise, which generates a net torque of zero. Therefore, by increasing the speed of a pair of rotors spinning in one direction and decreasing the others, a yaw motion can be produced. A quadrotor is an ideal vehicle for autonomous indoor aircraft due to the power efficiency and vertical takeoff and landing capabilities [1].

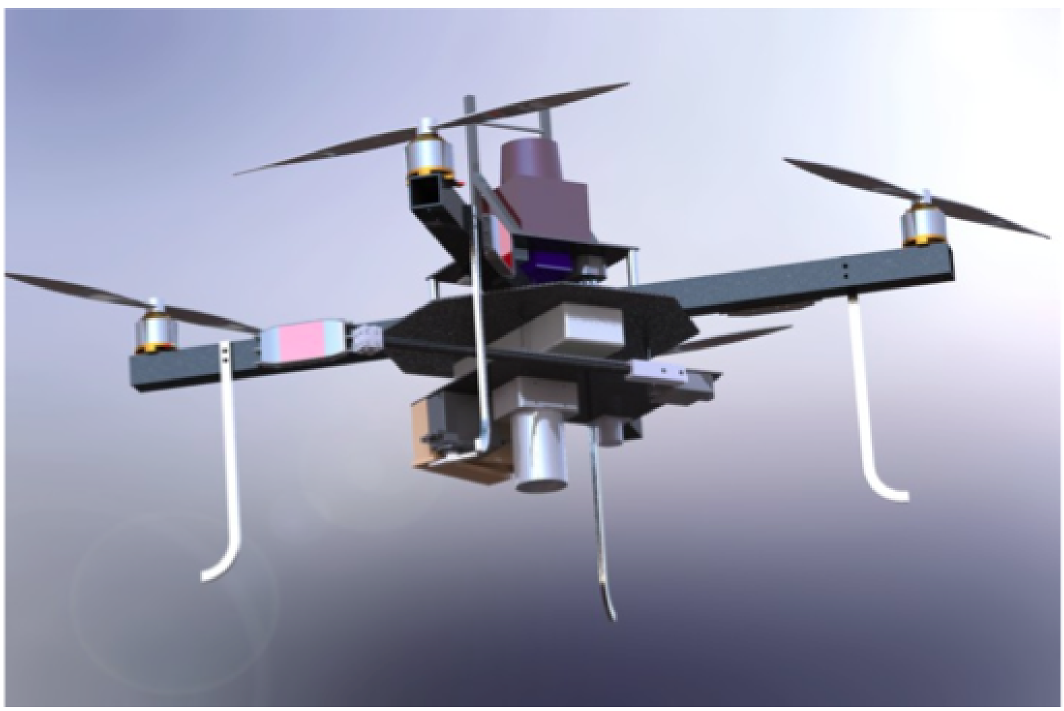

Figure 2: *CAD render of the 2012 Beohawk design.*

Guidance, Navigation, and Control

Stability Augmentation System

The quadrotor vehicle is a statically unstable system that must be controlled by a stability control system. The attitude and rate of change of the vehicle are measured by accelerometers and gyroscopes. Additionally, the vehicle's altitude and drift from original position are measured using a sonar and an optical flow sensor. With this combination of sensors, the vehicle is able to hold a set location without significant drift.

Several sensor measurements have been proven to have strong, reliable relationships with the position (translation and/or rotation) of the robot. Monitoring their changes can give us very useful knowledge about how much the robot drifts from its original position. We use the direction-cosine matrix algorithm (DCM) to provide an optimized result of the real-time orientation of the robot. This algorithm utilizes the three-axis linear acceleration and angular velocity measurements from the onboard inertial measurement unit (IMU). Since linear acceleration readings from the accelerometer are more accurate in the long term while angular velocity readings from the gyroscope tend to drift through time, the DCM algorithm uses acceleration information to correct the gyroscope readings, thereby giving almost drift-free information about the robot's orientation [2]. In addition, we also use a downward-facing camera to calculate the optical flow of salient, time-independent visual features. By doing so, we are able to figure out the motion trajectory of this camera, a technique called "structure from motion". A sonar sensor is also mounted on the bottom of the quadrotor. It measures a much more accurate altitude reading, and is used to improve the results of the optical flow algorithm. The combination of the DCM algorithm for the IMU and the structure-from-motion technique for the visual camera allows us to keep track of robot's linear and angular position in real time.

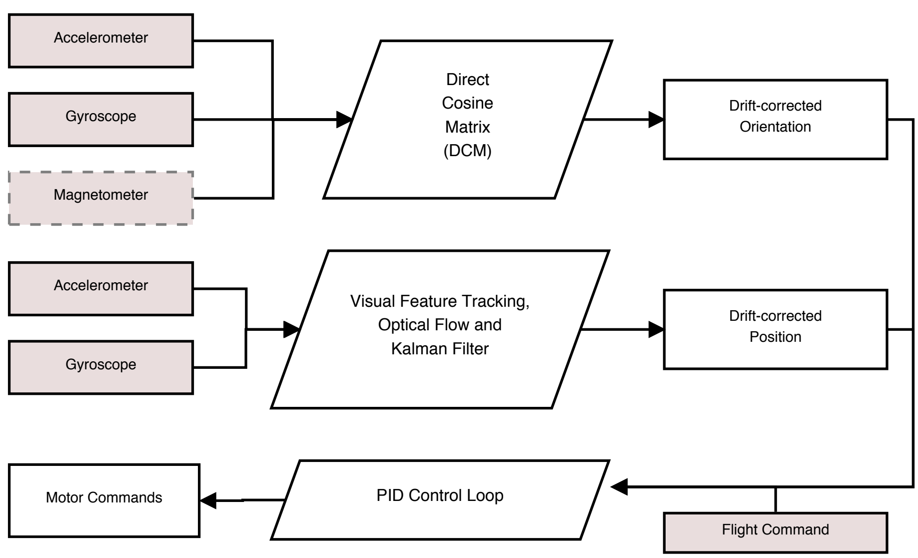

Figure 3: Feedback of drift information into the PID loop.

Finally, this system uses a PID control loop to handle the stabilization issue given these position-related measurements (figure 3). This control loop constantly updates the motor speeds based on the difference between perceived and desired attitude and position. The system also uses an Extended Kalman Filter to correct the position estimation from optical flow algorithm in order to get rid of unpredicted noises in visual feature recognition and matching. The positional error generated by the optical flow algorithm is used in another PID control system to control the position of the vehicle. In fact, all vehicle movements are perceived as errors in position instead of as goals.

The control system then applies the appropriate motor commands to correct the positional error and move to the next location.

Simultaneous Localization and Mapping

Our main navigational tool on the quadricopter is a laser range finder. The laser range finder provides our control system with information about the distances of surfaces in a horizontal slice of the environment. Because the competition will provide a building with a large amount of vertical symmetry, information about the distance to walls, tables, or other obstacles in the plane of the quadricopter is nearly sufficient for intelligent navigation. However, scans captured at different times during the quadricopters trajectory will not necessarily lie in the same plane. Therefore, we use the current estimate of the rotational pose provided by the gyro and the directed-cosine matrix calculation to project all of the received data points from the laser range finder onto a plane parallel to the ground.

Once we have access to information about the distances of objects in the horizontal plane, we use the information for 3 primary purposes. First, we observe features in the data received through the laser range finder to map key elements like the walls and doors in our environment. Second, we assume that many of these high-level features will be immobile, meaning that our observations can be used to localize our position within the environment. Third, consecutive scans will contain largely redundant information, but we can assume that changes will be primarily due to changes in the translational and rotational pose of our robot. Therefore, we developed an algorithm to estimate motion from data in consecutive laser scans.

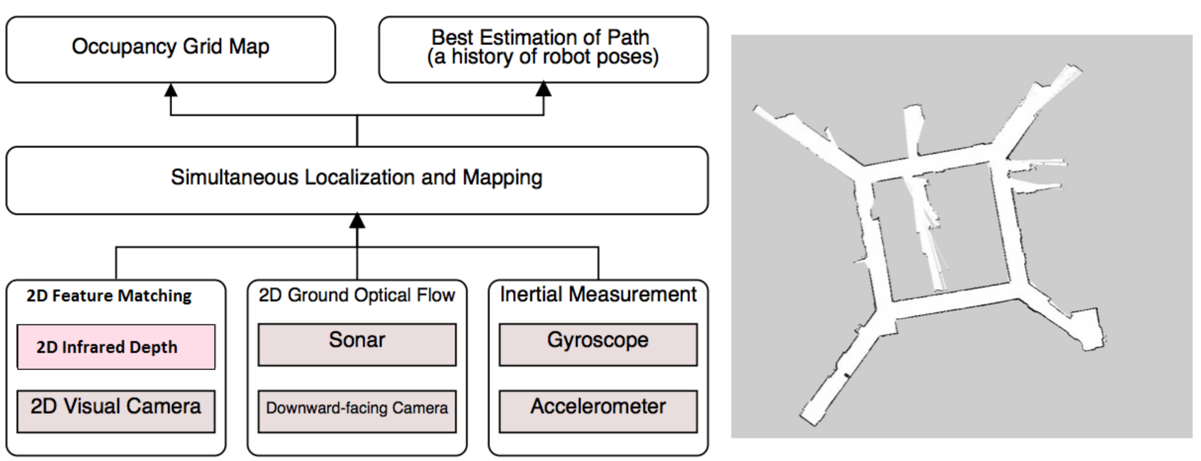

Figure 4: *Left: The inputs and outputs of SLAM. Right: Our robotics lab as seen through the SLAM algorithm.*

We achieve 3 three purposes using 3 separate algorithms. The first and second work by deconstructing each scan of the laser range finder into lines using a customized implementation of the

Hough Transform for the discrete data points. We then assume that the linear patterns we observe in the laser data will represent the boundaries of our environment. Using these patterns, the first algorithm maps these lines into a global blueprint of the building. The second algorithm will then allow comparison of the lines in a current scan to this global blueprint, enabling localization. Finally the third algorithm enables transient motion estimation without needing to access lines. Much as an optical flow algorithm can be used to infer motion without identifying robust visual features, our laser flow algorithm estimates translation and rotation without observing higher level features. These three algorithms combine to give our quadricopter localization, mapping, and short-term odometry, providing the necessary prerequisites for an intelligently controlled robot.

Navigation

With the simultaneous localization and mapping algorithms running on the laser range finder, we are able to continuously update a 2D model of our environment. Because our environment is an abandoned building, we reduce the problem of finding the USB slot to that of exploring a 2D maze. From this perspective, our cognitive control algorithm has the task of supplying movements that will make gradual progress towards finding the USB. For this year, we have not completed the computer vision task of reading the signs that indicate the identity of each room. However, we are still hopeful that our robot will be capable of finding the USB room by a more nave exploration algorithm with the primary goal of navigating into previously unobserved environments.

Figure of Control System Architecture

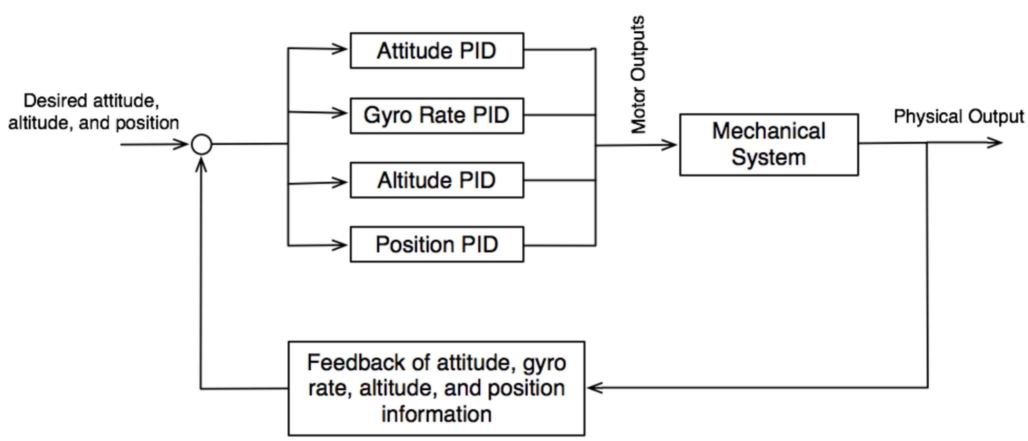

Figure 5: *Control system architecture.*

Figure 5 shows a general diagram of the control system architecture for vehicle control and stability. The feedback control system uses measured attitude, gyro rate, altitude, and position information to correct the vehicle's position and orientation to the desired position and orientation. The PID control system relies on optimized gains that were found through an iterative tuning procedure called Twiddle.

PID control systems are used for all control movements because of their simplicity and effectiveness. All commands sent from the navigation computer are interpreted as errors relative to the current state of the vehicle. The PID control system is a simple way to reduce such errors and can therefore be used to control all aspects of vehicle movement.

Target Identification and Threat Avoidance

Since the USB is a black object located on a white table surface, an edge or contrast detection algorithm will easily identify the USB as a salient feature. While the 3D visual sensor may not be accurate enough to tell the dimension of the flash drive, it can sense the much larger box containing the USB and therefore helps in target identification.

Identifying the signatures on the doors to different rooms is a tough task because arabic characters are difficult to identify with traditional OCR technology. Furthermore, the quadrotor may look at the signature from a perspective, resulting in an affine transformation of the signature image. *Beohawk* uses a scale/rotation invariant feature detection algorithm called "SIFT" to tackle this problem. The algorithm tries to match the picture it sees and the template in the database by trying numerous cases of different transformations. This algorithm has been proven successful in matching pictures regardless of scale, rotation or gradient constraints [3].

Flight Termination System

Because the RC controller communicates directly with the Arduino, the safety pilot can terminate the operation of the quadrotor in the event it is needed. One channel of the RC transmitter is connected to a switch that transfers control from the computer system to the human pilot. In an emergency situation where there is no chance for the safety pilot to land the vehicle, the motors can be disarmed and shut off at any time using a specific joystick movement.

PAYLOAD

Sensor Suite

Inertial Measurement Unit

The sensor board is based on the commercially available "Ardupilot Mega" which provides an Inertial Measurement Unit or IMU. The sensors are analog MEMS-based sensors that provide high sensitivity, high bandwidth measurements used to calculate attitude of the vehicle. This sensor board was used because of its commercial availability and extensive community support and software.

2D Visual Camera
Using a downward facing camera, the quadrotor runs an optical flow algorithm on the x86 processor to reduce drift. A second, front-facing camera captures pictures for reading the signs above the doors. The images are published through the wireless network to the base station, which runs SIFT to detect the Chief of Security's office via the sign above the door. The front camera is also responsible for detecting the blue security light. The bottom camera is used to identify the flash drive.

2D Infrared Depth Scanner
We have installed a Hokuyo laser rangefinder on the quadrotor as a novel solution to the SLAM and navigation problem. This device serves as a depth sensing scanner, which provides a 2D depth image at 10Hz with a field of view of 270 degrees. Because of the pitch and roll movements of the vehicle, these scans give us 3D depth information, so, using the inertial data at the time of each scan we extract a 2D slice parallel to the ground. This sensor is mounted on the top of the vehicle with its field of view above the plane of the propellers.

Sonar and Barometer
The quadrotor uses a barometer and an ultrasonic rangefinder pointed towards the floor to measure its altitude. Due to the presence of ground obstacles and other surfaces (tables, chairs, ect...) the data from both sensors is compared in order to identify when the vehicle is flying over said surfaces, calculate its true altitude, as well its altitude above the surface.

Power Management System
Beohawk is powered by a 7.4V 5000mAh Lithium-Polymer battery pack, which provides enough energy to run at least ten minutes. A power regulator board controls the battery and motor speed. The power board receives motor commands over I^2C communication and sends them to the motors using PWM. In the event of low power, the base station initiates a controlled descent to avoid damage to the quadrotor.

Communications
An IEEE 802.11n network is established between an on-board computer and a ground station. This network is time-synchronized and supports functionalities on both machines to communicate with each other through the ROS message system. While camera sensors are well supported by ROS, inertial sensor data and motor command have to go through serial communication. Since serial communication has not been implemented in ROS distribution so far, we developed a node that provides a protocol for exchanging messages safely through a serial bus, which greatly improves the reliability of whole system. We use a RS 232 based connection between the on-board computer and ArduPilot to communicate the motor commands as well as receive position/orientation

information from the various on-board sensors. The motor commands are sent to speed controllers that convert the PWM signals into the proper voltages to control the speed of the motors.

OPERATIONS

Flight Preparations

Before any flight is performed, batteries must be charged and an able human operator must be available to assume the control of the RC controller. For competition, the following checklist must be completed.

Competition Checklist

1. Inspect and test hardware
2. Launch ROS nodes
3. Check software status
4. Ensure RC link works
5. Test hover in place
6. Start mission control

Man-Machine Interface

The base station displays *Beohawk's* current mission and status inferred from the sensor data returned by the quadrotor. If the link is terminated, *Beohawk* will hover in place until the connection is restored or until the human operator assumes control. The human operator has an RC controller that communicates directly with the Arduino.

RISK REDUCTION

Vehicle Status

Beohawk transmits battery information, sensor information, and current objective to the base station. Based on these statistics, the vehicle can take action to save itself prematurely and to avoid transfer of technology into enemy hands.

Shock/Vibration Isolation

In the 2011 design, we use 1/4 inch square carbon fiber tubes with uni-directional fibers, attached to a square supporting structure made of carbon fiber tubes in order to adequately remove vibrations while keeping the weight low. The square tubes with uni-directional fibers turned out to be too brittle to withstand shocks from landings. So, in this year's design we are using 3/4 inch carbon fiber tubes with woven fibers, which allows us to eliminate the supporting square structure without compromising vibration isolation. In addition to this, we are changing the material of the landing gear to aluminum due to its higher modulus of elasticity, which will absorb more of the shock from landings.

EMI/RFI Solutions

Because our sensors and electronics are not very sensitive to the interference from motors and wireless communications, we have not had to worry about EMI or RFI. In a more demanding environment, all electronics could be shielded in conventional EMI/RFI shielding cases.

Safety

A substantial effort was put into designing the vehicle to be as safe as possible to the developers and the end user. During the testing phases, propeller guards were used to protect the propellers from bystanders. Once the quadrotor could fly autonomously and avoid obstacles, we removed these guards to reduce weight for the competition. The continuously updating costmaps in the navigation package ensure that the quadrotor won't get too close to walls or nearby people. The safety of all electronics and sensors was also taken into account when designing *Beohawk*. As a result, all of the electronics are mounted in between carbon fiber plates, with the sensors on the bottom being protected by the landing gear.

Modeling and Simulation

Hardware was modeled in SolidWorks. All electronics were modeled to ensure that they would fit in the final design properly. SolidWorks files were converted to machine readable CNC code that was used to produce many of the pieces on the vehicle. Several revisions of the vehicle were completed in CAD software before a final version was chosen for production.

The lift characteristics of various motor/propeller combinations were modeled using a thrust stand consisting of strain gauges wired to a wheatstone-bridge circuit. Models of thrust versus power consumption were made to select the optimal motor/propeller pair.

Testing

We have done continuous testing on *Beohawk* as we developed it. Testing follows a stepwise testing procedure where each component is tested individually and then as part of the complete system. For instance, the quadrotor was first flown under pilot control with only a simple attitude control system. Other sensors, such as sonar, optical flow sensor, and 2D depth sensor were added individually and fully tested on the platform before integration. Mission testing will be performed in a similar way, working to accomplish each goal individually before testing the complete mission.

For our first flight tests we built a tether out of PVC pipe that restricted the movement of *Beohawk* to only half a foot horizontally and a few feet vertically. Next, we attached an RC receiver to

Beohawk and test flew it with an RC controller until our stabilization software functioned properly. Further tests of *Beohawk* in its autonomous state were done in the hallways of our laboratory at USC, where it had the chance to avoid walls and recognize offices. In addition to this we built a window out of PVC pipes and construction paper to test the first step of the mission.

For software, individual unit tests were done to ensure the proper functioning of SLAM and vision algorithms. SLAM was prototyped in MATLAB and re-written in C++ while the mission control was written in Python and has a unit test suite as well as a behavior test suite.

CONCLUSION

The *Beohawk* quadrotor is an example of the collaboration of students from many different engineering disciplines. The rigorous demands of this project require significant effort from the mechanical, electrical, and software members of our team. As a result, the final product is a true team effort that could not have been completed by any individual member alone. The USC Aerial Robotics Team will continue to develop the software and hardware needed for the August 2012 competition in an effort to create a robotic solution to a unique problem.

References

[1] Pounds, P.; Mahony, R., Corke, P. (December 2006). "Modeling and Control of a Quad-Rotor Robot". In the Proceedings of the Australasian Conference on Robotics and Automation. Auckland, New Zealand.

[2] Phuong, N., et al. (February 2009). "A DCM Based Orientation Estimation Algorithm with an Inertial Measurement Unit and a Magnetic Compass". Journal of Universal Computer Science, vol. 15, no. 4

[3] Lowe, D. G., "Object recognition from local scale-invariant features", International Conference on Computer Vision, Corfu, Greece, September 1999.