Drone Challenge Overview

For our Senior Capstone, we were tasked with flying a drone through an obstacle course, finding a specific object, and reporting that location back to the user. Everything had to be done autonomously with computer vision as well. There was no real “customer” for this project, so we agreed to take this on as a research task.

Fall Semester

The fall semester was mostly about research. We wanted to find the best existing software we could to make our lives much easier. To actually control the drone, we found something called Robot Operating System (http://www.ros.org/). ROS is a tool that allows the modulor development of software packages and provides an easy to use interface for networking those nodes across a distributed system. In other words, it lets you create focused, self contained components to you robotic system and easily allow them to interact and work together. These modules are called nodes and the channels that they communicate on are called topics. We also looked into different ways to map where the drone was in the room. We found something called LSD SLAM (https://github.com/tum-vision/lsd_slam), which we assumed would do what we wanted easily. No real coding was done during this semester, but progress was still made since we found out what we could use and what we should avoid.

Spring Semester

When the spring semester started, we really started taking this project seriously. The CSE department bought a drone that we shared with another team, so we had a way to physically test the code we were writing. Matt DePero started working a lot with ROS and began to understand how to write and implement his own nodes. Cole Hoffbauer began working with the LSD SLAM libraries to see if we could correctly return maps of the rooms we were in. Chris Mabe focused on Docker (https://www.docker.com/) in order to get rid of the operating system dependency problems. Matt was able to implement a quick demo of what we could do with OpenCV and ROS, which is what is included in the video demo. This is a crude example, but it is not exactly the most important part of our project.

Finishing Up

Halfway through the spring semester, our group realized that the scope of our project was much too large. There was no possible way we would be able to implement everything in time, so we discussed how to proceed with our faculty advisor. We suggested that, instead of wasting time and focusing on a demo that ultimately wouldn't work, Chris could create a Docker image that future groups could use to reduce the startup time for the project from a whole semester to a week at most. He thought this was a good idea, so we switched our focus from developing a drone that finds objects autonomously to a development environment future teams could use. Chris was able to add everything we did into a custom image the next team can run on any operating system (which originally was a large hurdle because ROS is linux specific) in under 6 terminal commands. The CSE faculty seemed to really like the fact that, while we had a working demo that was fairly impressive, we took time to think about future groups and how we could help them get farther than we ever could. Due to this fact, they awarded us with “Best CSE Senior Design Project 2017”.