Project Proposal (Wei Dong)

Big Picture

Drones, mostly quadcopters have been increasingly commercialized. Quadcopters are often used for photography, film recording, and delivery. With 4 motors, a quadcopter can control its attitude and altitude precisely and hover at the same position. Drone swarm has not seen much use outside militaries. The research aims to develop a single-motor drone to reduce the cost and structural complexity. With only one motor and no rudder, the drone has to give up attitude control, since the center of mass of the drone needs to be off-center and the body will spin as a result of the reaction torque of the motor.

The general approach is to start with a Crazyflie quadcopter and reduce the number of motors attaching to the frame and modify the frame accordingly. Crazyflie is an open-source development platform that are commonly used in academic researches. The quadcopter by default uses a cascade controller that incorporate an outer loop (attitude controller) and an inner loop (rate controller). The first step is to configure the quadcopter to fly according to planned trajectory. The next step is to reduce the number of motors gradually to one. The project will base on a paper on control of quadcopters after stability loss and a paper on single-propeller drone[1]. If a single-motor drone could be built successfully, the potentially reduced cost and design complexity could bring down the cost of drone swarms. Individual drones will also be able perform tasks such as reconnaissance and internet connection.

Specific Project Scope

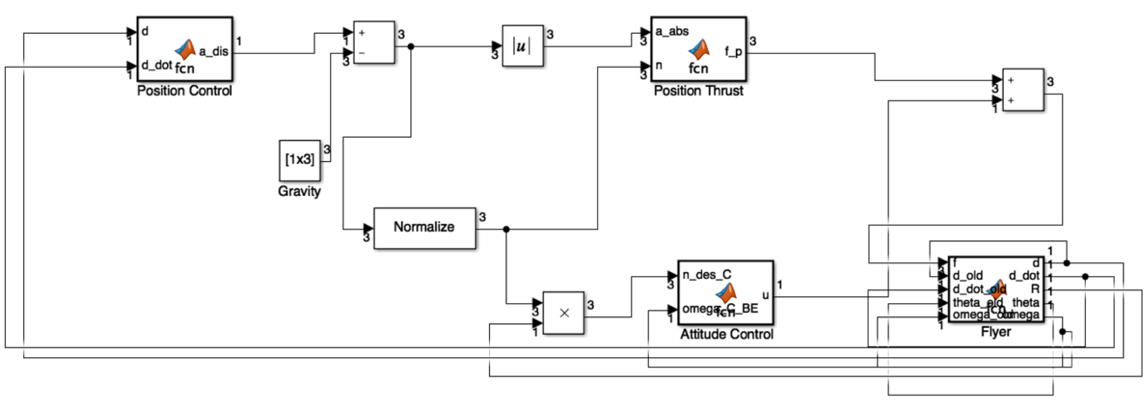

Figure 1: Simulink model of the single-motor flyer

During the past year, we have made some progress in setting up the hardware and simulation. However, we are still far from reaching the goal of having a single-motor drone flying in the lab. Since the summer time is less than 2 months, my goal is to tackle 2 issues: debugging the simulation model written during the last quarter and localizing Crazyflie using Kinect.

The simulation model is based on papers published by ETH[2], and utilizes a LQR controller for the attitude control. The controller loop used is a cascade control, with a position controller and an attitude controller. The desired thrust is calculated by adding the desired thrust from the two controllers together. Currently, simulation of the dynamics of the flyer poses the greatest problem. The design is a modified version of a quadcopter dynamics. The dynamics block has been fully implemented. However, there is still an error message that has not been resolved: “this function does not fully set the dimensions of output port 2.” The first step is to fix the dynamics block and then include linearization and LQR functions in the Simulink model.

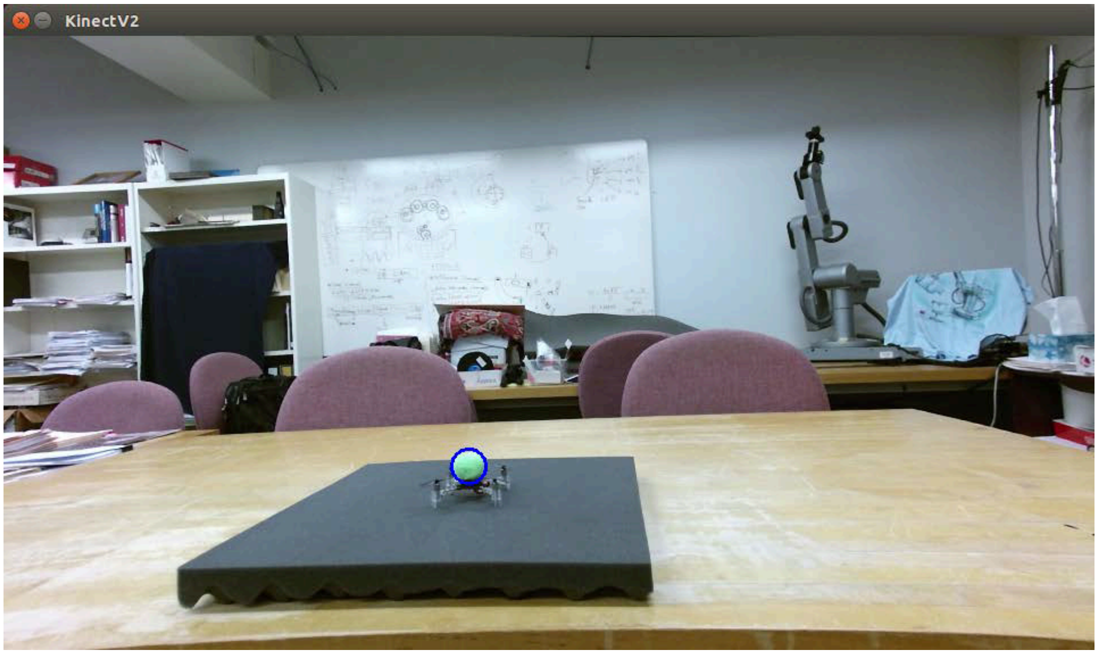

Figure 2: Kinect recognizes the marker attached to the Crazyflie (C. Fairchild, T. Harman, 2016)

Another part of the project is to finish localization of Crazyflie. We have spent months trying to configure a usable localization system. However, each method that we have tried has been buggy and unsuccessful. After installing ROS on the lab computer, we realized that we needed to write the script for image recognition ourselves. The steps involved are: tracking a marker with Kinect, detecting markers with OpenCV, and controlling Crazyflie with *crazyflie_autonomous* package. The purpose of this part is to establish a portable platform for autonomous flight and is not directly related to the single-motor flyer itself. This is a good opportunity of learning ROS and Kinect.

Monthly Goals

July

Finish debugging the simulation and set up the Kinect for image recognition.
Deliverable:
A Simulink model that can showcase the control of a single-motor flyer
A video showing that Kinect can correctly identify the marker attached to a Crazyflie

August

Implement autonomous flight, debug and fix any problems that could occur
Deliverable:
A video showing that a Crazyflie can fly autonomously

Weekly Goals

Week 2
Debug the dynamics block of the Simulink model
Deliverable: a model that can build and will free fall without the applied thrust

Week 3
Use linearization and LQR functions in Matlab to finish attitude control
Deliverable:
A complete model that can fly to a desired position

Week 4-5
Follow *ROS Robotics by Example* to set up Kinect and detect a marker attached to a Crazyflie
Deliverable:
A video showing that Kinect can correctly identify the marker attached to a Crazyflie

Week 5
Understand and go through the steps for autonomous flight (additional markers might be needed in order to find the yaw angle)

Week 6-7

Debug and resolve any issues
Deliverable: a video showcasing a Crazyflie performing autonomous flight

References
1. Mueller, M. W., & Dandrea, R. (2014). Stability and control of a quadrocopter despite the complete loss of one, two, or three propellers. *2014 IEEE International Conference on Robotics and Automation (ICRA)*. doi:10.1109/icra.2014.6906588
2. Zhang, W., Mueller, M. W., & Dandrea, R. (2016). A controllable flying vehicle with a single moving part. *2016 IEEE International Conference on Robotics and Automation (ICRA)*. doi:10.1109/icra.2016.7487499
3. Fairchild, C., & Harman, T. L. (2016). *ROS robotics by example: Bring life to your robot using ROS roboticapplications*. Birmingham, UK: Packt Publishing Limited.