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: 6 nodes (anchors) and Crazyflie fitted with loco positioning deck[2]

The problem that I will be addressing is setting up the Crazyflie quadcopter for trajectory planning along waypoints. Solving the problem will pave the way for the control of single-motor drone in the future.

Experiencing with a quadcopter will go through the necessary step to program a single-motor drone to plan path following preset waypoints. Our goal is to obtain telemetry data only from onboard sensors. A technical report published by Mobile Robotics and Autonomous Systems Laboratory, Polytechnique Montreal outlines the process of motion capture system integration[3]. However, the study uses VICON listener which is too expensive and is based on external motion-capture cameras. Since we do not have onboard sensors available, an external system can still supply the necessary localization data for autonomous flight. There are a few options to choose from:
Our experiment can use the loco positioning system officially supported by Bitcraze as an intermediate step. The loco positioning system will provide the absolute position of quadcopters flying in the environment with an accuracy in the 10cm range. One important step is to set up the localization system for the drone. Bitcraze offers Indoor Explorer Bundle which includes 6 nodes 1 deck, 1 radio and a Crazyflie 2.0 . The bundle comes with 6 loco positioning nodes and a deck to be fitted onto the quadcopter. Bitcraze's website provides instruction on how to set up the loco positioning system.

The steps are outlined here[4]:
1. Update the anchor
2. Download the LPS configuration tool
3. Download the LPS node firmware
4. Update the node
5. Configure the node into anchors
6. Place the anchors into the room
7. Configure the anchor with the crazyflie client
8. Verify anchor position and estimated position

A ROS module available on Github[5] is needed to configure anchors and control crazyflies. The Python library provides functions and examples for autonomous scripted flying. However, the exact process of invoking the functions needs further investigation.

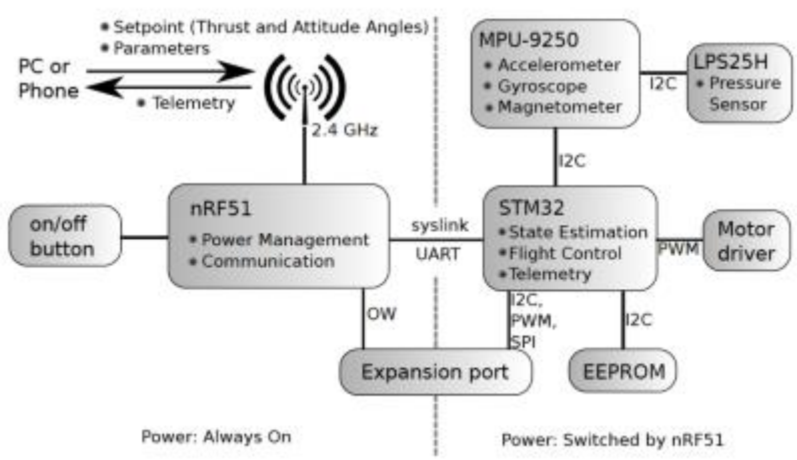

Figure 2: Components and architecture of Crazyflie 2.0[6]

Another solution is to use the webcam of a computer to acquire the position of the Crazyflie and possess the data with OpenCV and Xcode[7]. The approach has its limitation since the range of motion is limited to the visual field of the camera. Another problem is that the only resource that can be However, the approach is economical since it does not require any specific hardware other than an identifier to be attached to the Crazyflie.

Monthly Goals

October:
Finish the experiment with motors outlined by Carol's prelab
Set up ROS to work with the Python client and configure ROS to implement the controller node and determine how the controller code can be modified.

November:
achieve hovering at controlled height, and determine the localization system that will be installed.

December:
Apply the setup to crazyflies with fewer motors. Implement localization system to set up waypoints.

Weekly Goals

Week 1
Configure ROS to control Crazyflie with a controller. The goal is controlled flight and hovering when flying with a Xbox controller.
Week 2
Complete the experiment detailed in Carol's prelab.
Week 3
Implement altitude hold using the onboard barometer sensor. The crazyflie should be able retain its altitude in the air. A video will be recorded.
Week 4
Set up the webcam to capture the position of the crazyflie. Learn to use the relevant functions in OpenCV. The deliverable is presentation of the live position data from the camera.
Week 5
Use ROS and OpenCV to implement hovering. A video showing the hovering state will be the deliverable.
Week 6
Configure ROS to implement the controller node and determine how the onboard controller code can be modified. The end result should be screenshots and captured videos which show that we can locate and modify the controller.
Week 7
Use ROS to set up waypoints and program the Crazyflie to fly along waypoints autonomously. A video will be recorded.
Week 8-10
Modify the controller to accommodate loss of propellers. The goal is to achieve a stable hovering state and photos and videos will be recorded.

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. https://store.bitcraze.io/collections/positioning/products/indoor-explorer-bundle
3. Luis, C., & Le Ny, J. (August, 2016). Design of a Trajectory Tracking Controller for a Nanoquadcopter. Technical report, Mobile Robotics and Autonomous Systems Laboratory, Polytechnique Montreal.
4. https://www.bitcraze.io/getting-started-with-the-loco-positioning-system/
5. https://github.com/bitcraze/lps-ros
6. Hönig, W., & Ayanian, N. (2017). Flying Multiple UAVs Using ROS. Studies in Computational Intelligence Robot Operating System (ROS), 83-118. doi:10.1007/978-3-319-54927-9_3
7. https://bitbucket.org/hbd730/carzyflie-camera-control.git