Sensors and Motors Lab

Rohan Thakker

Team C: Column Robotics

Teammates: Job Bedford, Cole Gulino and Erik Sjoberg

ILR02

Oct. 23, 2015

1) Individual Progress

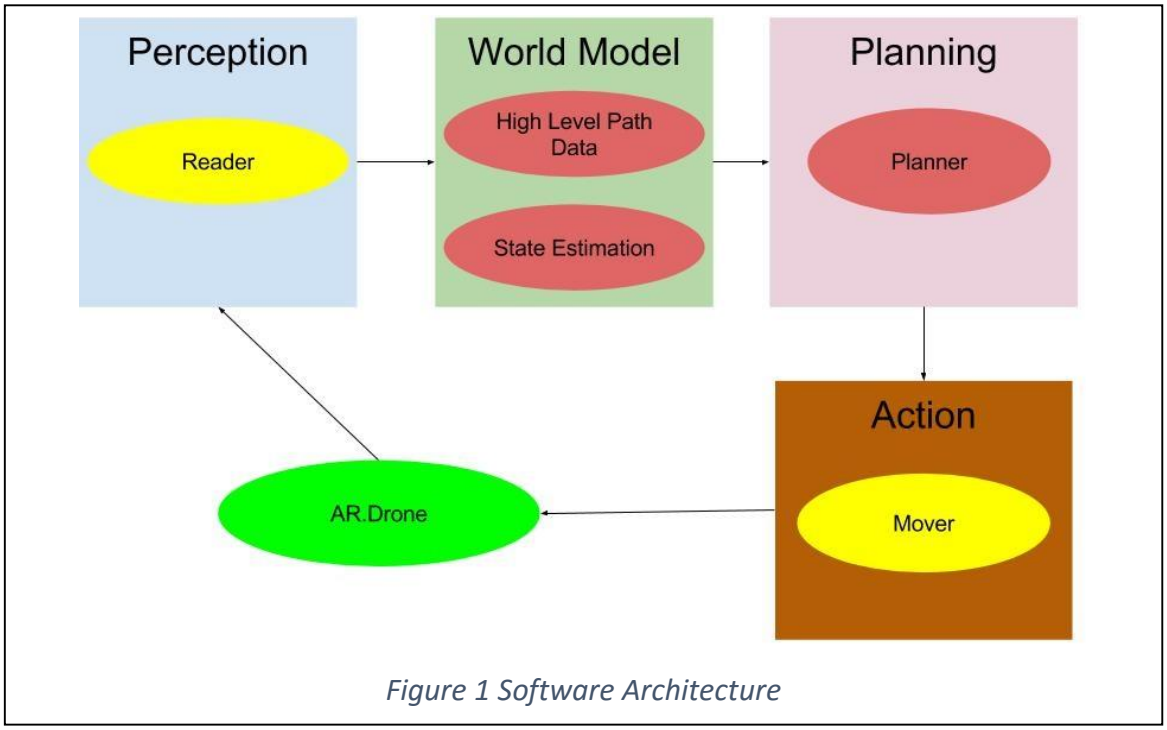

Figure 1 Software Architecture

Our goal was this week was to setup the software framework and low-level communication with the ARDrone platform. We plan to use the ARDrone to test our planning and vision algorithms before we use it to test on the IRIS+ (the platform we plan to use for the demo). The figure 1 shows the initial version of the software architecture. The reader and the mover nodes abstract implementation that are specific to the drone that we are using. Hence, when we shift to the IRIS+, the code of all the other nodes will be re-usable.

I was responsible to implement the reader node for the ARDrone. The reader node gets the sensor data from the drone and publishes to a topic that nodes like State Estimation can subscribe to. I used the ardrone_autonomy ROS package to achieve this. [1] shows the list of messages that are published by the ardrone_autonomy node. Currently, we are mainly using the estimates of the position and orientation.

I was also responsible for selecting and ordering the single board computer and quadcopter. We have already received the [2] minowbardmax (SBC) and expect to receive the IRIS+ quadcopter by next week.

2) Challenges

It was hard to choose the single board computer for our project because the variety of boards that are available. The ARM based ODroid is very popular among the quadcopter community. However we may later find some API that don't support ARM based architecture. Hence, we also ordered x86 architecture based MinnowBoardMax. It was a challenge to compare which one is better as there is no comparison information available on the internet. We resolved this issue by ordering both the boards, so that we can test

them ourselves. It has been a long time since I worked with ROS and C++, hence I had to spend extra time in reviewing those topics. Working on the ROS assignment did help with that.

3) Teamwork

Table 1

Job Bedford	**Cole Gulino**	**Erik Sjoberg**	**Rohan Thakker**
Design MOVER Node for AR.Drone in ROS	Risk Management Table	Set up ROS Framework and GIT Repo	READER Node for AR.Drone in ROS
Media and Visual Generation	Outline PDR tasks and work to be done	Work Breakdown Structure and Burn down chart	Selected and Ordered Quadcopter and SBC
	Set up website outline	Researched Robot Localization Packages	

The table 1 shows the tasks done by each of our team members. Erik helped me in the selection process of the single board computer and quadcopter.

4) Plan

We aim to accomplish the following things by the end of next week:

- Open loop control of the ARDrone (JOB)

Job will complete and test the mover node on the drone

- Develop GUI to visualize the state estimation (Erik)

Erik will develop the GUI that subscribes to the reader node and displays the state estimation values

- Research and finalize global position strategies (COLE)

We require a way to be able to estimate our position in the world frame. This can be done in multiple ways: by using an array of APRIL tags or using an overhead camera. Cole researching the available options and deciding on the best one for our project

- Select and order camera (ROHAN)
- Get familiar with Pixhawk code (COLE and ROHAN)

5) References

[1] http://ardrone-autonomy.readthedocs.org/en/latest/reading.html

[2] http://www.minnowboard.org/meet-minnowboard-max/