

# **Progress review #4**

**Gauri Gandhi**

**Team G Robographers**

**Teammates:**

Rohit Dashrathi

Jimit Gandhi

Tiffany May

Sida Wang

**ILR #5**

**November 25, 2015**

#### **a. Individual Progress**

For the fourth progress review, I worked on creating the ROS nodes for detecting the pose estimates through the April Tags using the webcam and using them for moving the robot. I also worked on updating the team's website.

##### **I. Creating ROS node for detecting April Tags**

In the last progress review, I used the Michael Kaess C++ library to detect the April Tags using the laptop's camera. This time I worked on creating a ROS node to accomplish the same task using the webcam (the one attached to the pan tilt unit). First, I created a workspace for all the packages and nodes to be used in the project. Then I created a package for the april tag detection and created a node from the C++ source code for detection in the Michael Kaess library. But I could not link the header files and was getting a lot of errors in the code. Then I found the light-swarm/april\_tag repository on GitHub by Palashn. I modified the source code as per our requirements. The april\_tag\_node subscribes to the video stream from the USB camera and publishes the data obtained from the april tag on a new topic called april\_tags. The message type of the april\_tags is AprilTagList that creates an instance for the structure defined in the message named AprilTag. Both the messages are custom created messages. Then I created a node called Listener that subscribes to the rostopic april\_tags and displays the z coordinate (to be used for the distance estimation from the person of interest) using the ROS\_INFO\_STREAM. We will be using the z estimate for the navigation part in the Fall Validation Experiment.

After creating the source files, I created the launch file that runs the following nodes:

1. usb\_cam\_node in the usb\_cam package: to start the usb camera
2. image\_view in the image\_view package: to view the coming video stream
3. april\_tag\_node in the april\_tag package: to start the the node for the detection of the april tags

After launching this launch file, I could detect the april tags using the USB camera. To see the parameters defining the pose of the camera in the relative frame of the camera, echoed the rostopic april\_tags in another terminal. This gave the results as shown in figure 1.

To see the z coordinate, I ran the Listener node separately in a new terminal that gave me the output as shown in figure 2.

The april tag was detected using the webcam that I mounted on my laptop.

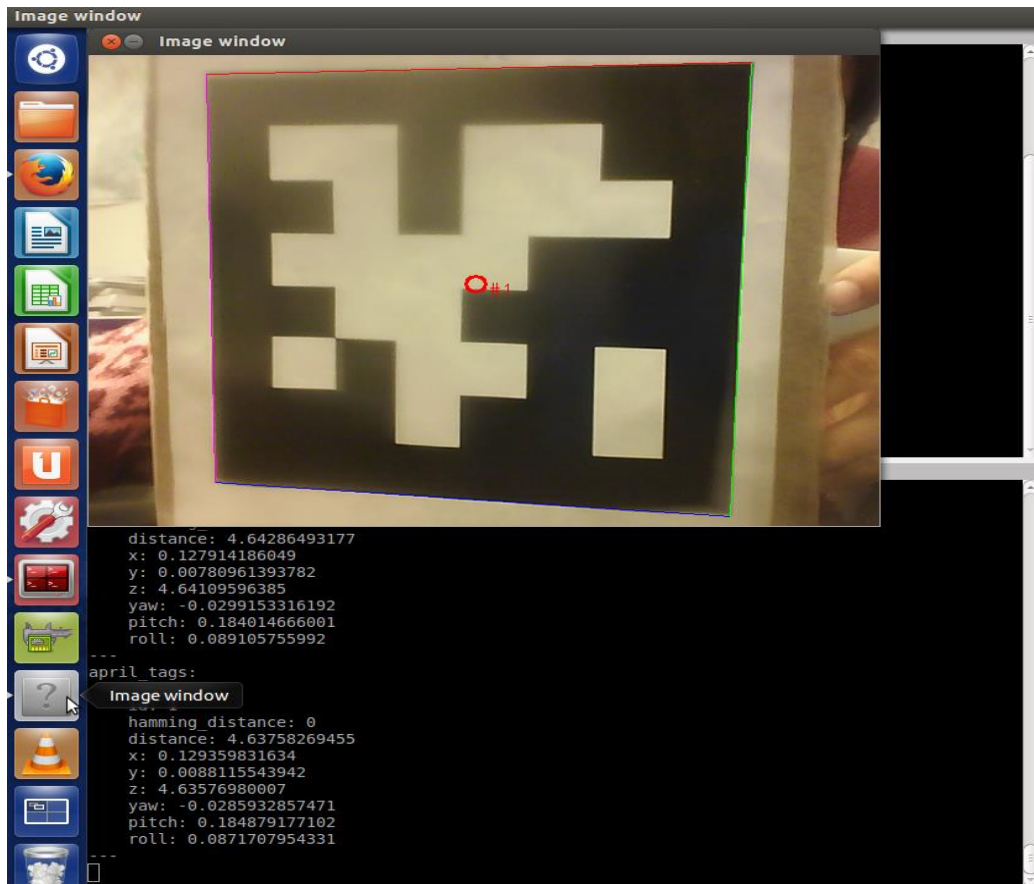


Figure 1 Detected April Tag and the displayed parameters

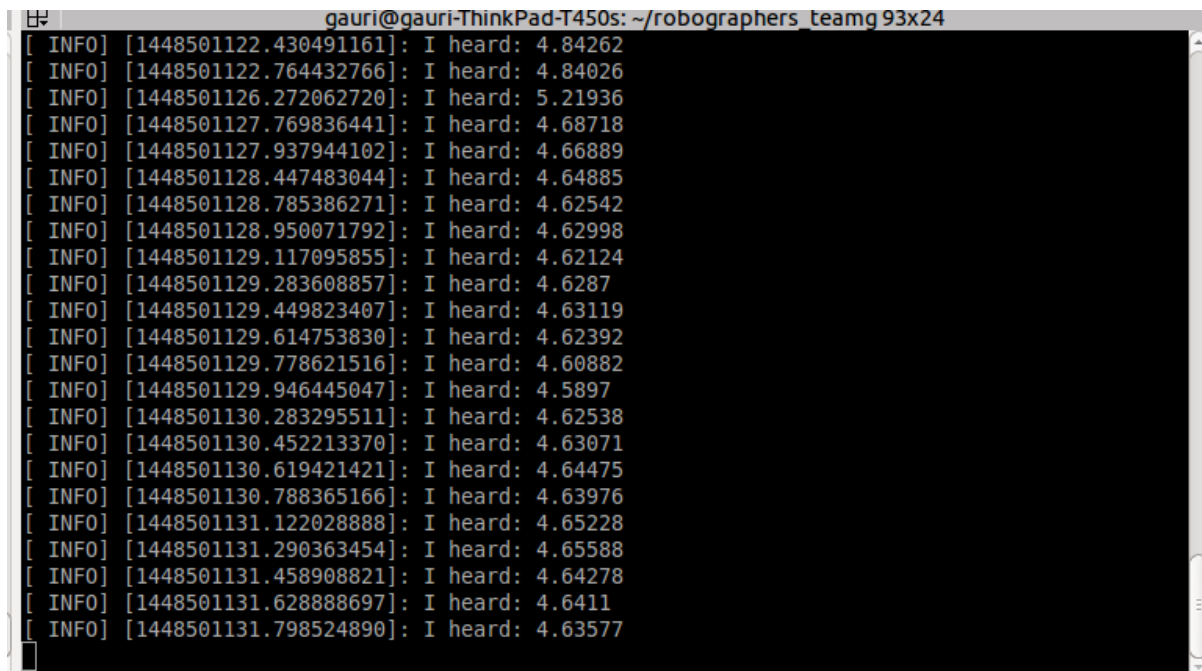


Figure 2 Displayed z coordinates after running the Listener node

## **b. Challenges**

- I. The first challenge I faced was in linking the header files with my main code in ROS. But the april\_tags repository that I found on Github cleared a lot of my doubts regarding the creation of the CMakeList file in the package.
- II. At first I tried to use the gscam package to start streaming from the usb camera. But every time I launched my launch file, it stopped because of that. Then, I tried to run the gscam streamer independently in another terminal. I set the environment variables for the USB camera. I found out that after 60-70 seconds of proper streaming of the live video input, the Ubuntu core was giving a segmentation error (may be due to some memory issues). While the laptop camera was giving the desired results. So I switched to using the usb\_cam package and changed its parameters to get video feed from the USB camera. This gave the desired results.

## **c. Team Work**

For this progress review, Rohit worked on manufacturing the elevation axis for the pan tilt unit. He completed this task successfully and mounted the complete pan tilt assembly on the turtlebot assembly. Sida was successful getting the pose estimate from the Intraface. She was able to send the position coordinates of the nose from the detected face to the serial port. Tiffany set up the connection between the obtained parameters from Sida's part and the Arduino to run the pan tilt servo motors. Jimit created a ROS node to autonomously click a photo from the incoming video stream. He also started working on writing a node to autonomously navigate the turtlebot from one point to another. Jimit and I worked together to update the website.

## **d. Future Plans**

For the Fall Validation Experiment, I will be working on the following tasks:

- I. Calibrating the usb camera to get the proper distance estimate from the April Tags.
- II. Obtain the transform parameters for sending the obtained distance estimate in accurate units to the navigation node.
- III. Merge the code for the autonomous navigation with the distance estimation code.
- IV. Update the team website.

Everyone in the team will be working on completing their individual tasks to make them compatible with other project tasks. All of us will work together to finish the integration tasks.

## **e. References**

1. Turtlebot ROS tutorials  
[wiki.ros.org](http://wiki.ros.org)

2. April tag C++ library by Michael Kaess  
<http://people.csail.mit.edu/kaess/apriltags/>
3. Palashn's repository on github  
[https://github.com/light-swarm/april\\_tag](https://github.com/light-swarm/april_tag)