ROS Squad 3 Code Tutorial

1.) Download ROS Kinetic. Documentation should be present on the wiki.

2.) Download and build OpenCV 3.2. Make sure to also build the extra modules from the `opencv_contrib` package, outlined in the third step of the online tutorial.

NOTE: If one can't find the mentioned package from the online documentation, a zip file should be found in the "Vision Testing" folder of the GTRI Drone Project shared google drive.

3.) Assuming the `catkin_ws` is setup for a computer with the ROS tutorials, execute `git clone https://github.com/GTRI-2017/Vision-Code.git` in the terminal within the `catkin_ws/src` directory.

4.) Use the command `rosrun drone_test detect` to run the main vision processing node. Drone camera feed should be visible on the laptop, if not make sure the proper steps for setting up the Parrot AR Drone were followed:

a. Turn on the drone and connect to it as one would connect to a regular Wi-Fi network.
b. Execute `roscore`
c. Execute `rosrun ardrone_autonomy ardrone_driver`
d. To switch between cameras execute `rosservice call /ardrone/togglecam`
e. Drone should be ready for running nodes

NOTE: If within an area crowded with wireless networks, consider changing the drone to a less crowded channel. Also make sure the camera is calibrated before running the detection. This is explained in the `detect.cpp` code. Video of camera calibration can be seen at the bottom of this webpage.

5.) Execute `git clone https://github.com/GTRI-2017/ARDrone-Master.git` in same directory as mentioned in step 3.

6.) Execute `rosrun ardrone_master mr3` to run the movement node. Drone should move in a search pattern and land once the ArUco tag has been detected through the bottom camera.

NOTE: In order for the code the work as in the demonstration video, make sure the vision node is run before the movement node. Also, make sure the video feed from the bottom camera is seen, not the front