|  |  |
| --- | --- |
| **Gazebo and ROS Installation** |  |
| Gazebo provides a 3d simulation environment with accurate physical modeling |  |
| Install python opencv:  Sudo apt install python-opencv=3.2.0+dsfg-4ubuntu0.1 | https://courses.dojofordrones.com/courses/1119413/lectures/24011936 |
| Install Gazebo9:  Sudo apt install gazebo9 libgazebo9-dev |  |
| Setup Ardupilot-Gazebo:  Git clone <https://github.com/dronedojo/ardupilot_gazebo>  Cd ardupilot\_gazebo  Mkdir build  Cd build  Sudo apt install cmake  Cmake ..  Make -j4  Sudo make install |  |
| Add the following environment variables:  GAZEBO\_RESOURCE\_PATH  And  GAZEBO\_MODEL\_PATH  To ~/.bashrc  Also add to ~/.bashrc:  Source /usr/share/gazebo-9/setup.sh  Remove GAZEBO\_MODEL\_PATH and GAZEBO\_RESOURCE\_PATH fom setup.sh |  |
| Install curl  Sudo apt install curl |  |
| Install Ros Melodic:  Sudo sh -c ‘echo “deb <http://packages.ros.org/ros/ubuntu> &(lsb\_release -sc) main” > /etc/apt/sources.list.d/ros-latest.list’  Sudo apt-key adv --keyserver ‘hkp://keyserver.ubuntu.com:80’  --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654  Sudo apt install rose-melodic-desktop-full  Add to .bashrc:  Source /opt/ros/melodic/setup.bash  Sudo apt install python-rosdep python-rosinstall python-rosinstall-generator python-wstool build-essential  Sudo apt install python-rosdep  Sudo rosdep init  Rosdep update | Install documentation: http://wiki.ros.org/melodic/Installation/Ubuntu |
| Install Gazebo-Ros:  Sudo apt-get install ros-melodic-gazebo-ros-pkgs ros-melodic-gazebo-ros-control |  |
| **Gazebo Setup** |  |
| Launch gazebo through terminal with command ‘gazebo’  Default launch behavior presents user with an empty gray world  To have an environment that can be interacted with a ‘world’ file must be launched with the program | <https://courses.dojofordrones.com/courses/1119413/lectures/24010631> |
| World files located in ‘ardupilot\_gazebo’ repo  To navigate to and show the world files available:  Cd courseRoot/ardupilot\_gazebo  Cd worlds  Ls  GAZEBO\_RESOURCE\_PATH environmental variable added previously allows the user to call on these world files without being in the worlds repo directly |  |
| Launch Gazebo with world file:  Gazebo iris\_arducopter\_runway.world  Loads up a world containing an environment and virtual iris hardware drone |  |
| Pay attention to the data trackers along the bottom of the screen. These will vary depending on the resources allocated on the virtual machine. Important ones to pay note are the ‘Real Time Factor’ and ‘FPS’ |  |
| If ‘FPS’ is low, around 3-4, shadows can be turned off in the ‘World | Scene’ menu and unchecking the ‘Shadows | True’ box |  |
| ‘Real Time Factor’ indicates the fraction of the speed that the simulated world is playing back to us. ‘1’ is normal speed. If RTF is low, it should become closer to ‘1’ when the SiTL vehicle is launched and connected to the virtual hardware drone in Gazebo |  |