**Design Document**

Please cite all sources (repositories, online articles, ROS wiki links, etc) you consulted/used for implementing your project.

Modules:

1. <https://github.com/IntelRealSense/realsense-ros>
2. <https://github.com/IntelRealSense/realsense-ros/issues/772#issuecomment-493132694>
3. <https://github.com/ros-teleop/teleop_twist_keyboard>

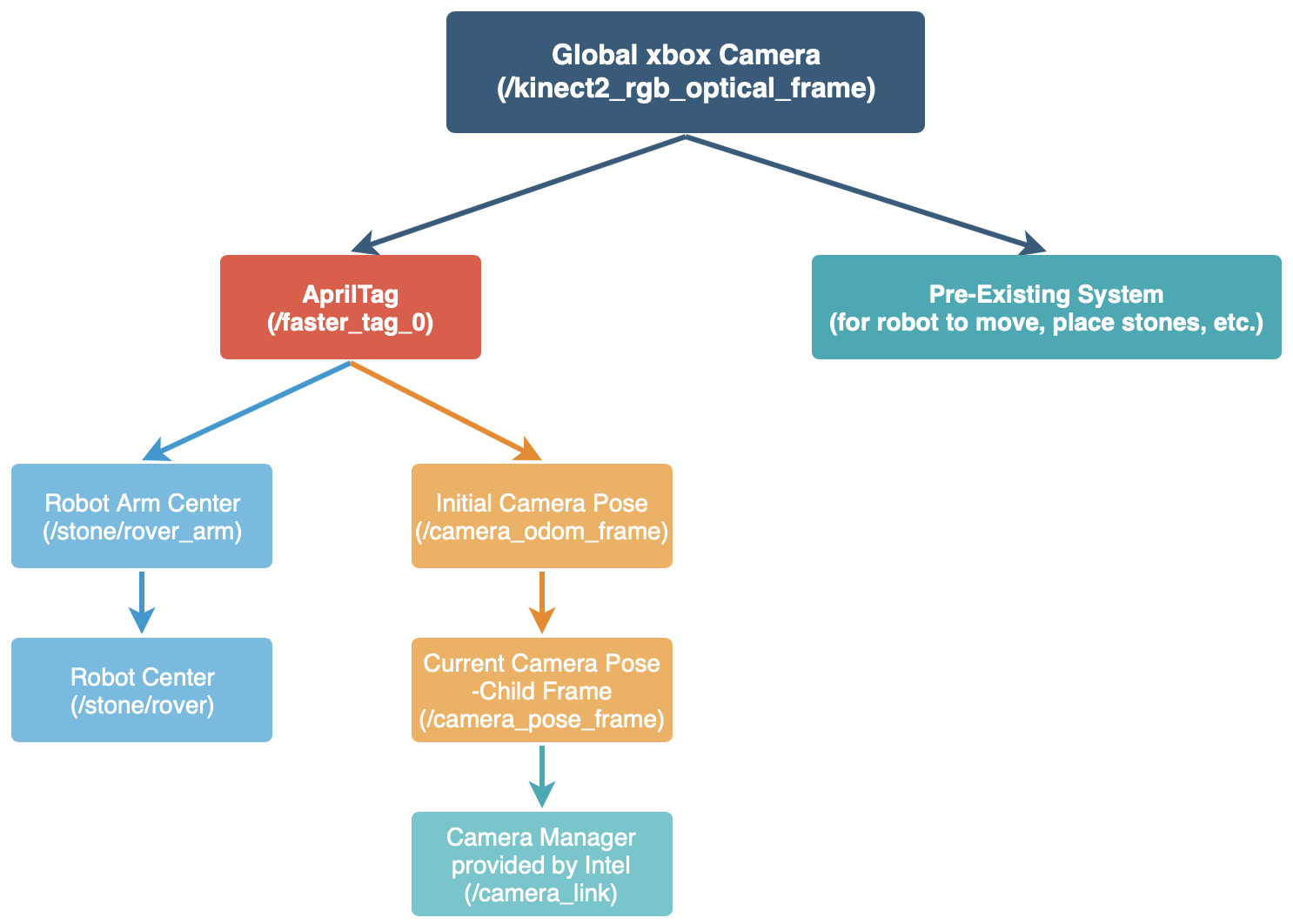
Intel Realsense module Official Website for SDK:

1. <https://www.intelrealsense.com/developers/?cid=sem&source=sa360&campid=2019_q2_egi_us_ntgrs_nach_revs_text-link_brand_bmm_cd_realsense-realsense_o-1lngr_google&ad_group=realsense%5Eus%5Erealsense%5Eexact&intel_term=%2Brealsense&sa360id=43700043668081492&gclid=EAIaIQobChMIm4-90MaA5AIVCSaGCh1oiAPVEAAYASABEgJHvfD_BwE&gclsrc=aw.ds>

Source Code:

1. Existing code (\*.launch, Transformation.py)
2. Existing code from the physics\_param\_analysis (for the analysis part)

ROS Wiki:

1. <http://wiki.ros.org/tf/Tutorials> (Every tutorial in 2. Learning tf -> Python)
2. Draw a structure chart that showcases the various modules in your project and how they communicate with each other. Draw the modules as ellipses and name them accordingly, with arrows between the various modules to indicate the information that they share. Identify ROS topics and ROS services, if any. (Refer to the lecture slides on ROS communication graph. [draw.io](https://www.draw.io/) is a good flowchart designer) [20 %]  
     
   Because I am dealing only with TF system -sending transformations back and forth- I have this TF tree. Some system are abstracted because I did not modify any modules from there.  
   
3. Identify the module(s) where you have contributed a considerable amount of new code to the project and give a brief description of what the code does. [10 %]  
     
   (a) ROS Wrapper for Intel® RealSense™ Devices  
   This is packages for using Intel RealSense cameras (D400 series SR300 camera and T265 Tracking Module) with ROS. Since I was using the T265 model, I had to install this wrapper code to launch the camera node. The topics contain accelerometer, gyroscope, fisheyes which shows the raw image of the camera, and etc. Through this information, I broadcast and send the transformation between the camera and the AprilTag.  
     
   (b) teleop\_twist\_keyboard  
   This module allows me to control the movement of the robot with the keyboard. I used this module to initiate such movements of the robot and acquired datasets that have position information.
4. Given there are various open-source implementations available for each module in your project, how did you go about selecting them? [10 %]  
     
   (a) ROS Wrapper for Intel® RealSense™ Devices  
   Once I decided to do the performance comparison between the AprilTag and the Intel Tracking Camera T265, I looked up the official Intel website where it introduces the tracking camera. After installing SDKs, the website guided me to download the wrapper code for ROS to utilize the camera along with the robot.   
     
   (b) teleop\_twist\_keyboard  
   In this case, I was initially thinking of using the rqt to control the robot’s movement. However, the instructor recommended me to use this, and it was way easier since it allows me to control the robot with the keyboard, while rqt is controllable by the mouse.
5. Which part of the project was the most challenging and why? [20 %]  
     
   I thought the most challenging part would be sending/broadcasting transformations because I had an experience of struggling with it before the project. However, the TF was not the problem; instead, it was somehow easy to implement once I take a look at the ROS wiki page where it has TF tutorials.   
     
   The most challenging part was gathering datasets and analyzing it. Since I did this project by my own, while acquiring datasets, I had to launch the robot, start the camera to record the robot’s movement, take a look at Rviz to see whether the camera frames are settled up well, move robot with keyboard, and take a look at data and check whether it is saved well or not. If I miss a single step, I had to restart everything again. Also, especially the dataset that has the trajectory of the unstructured environment was one of the difficult parts for me to gather the data. If I do not stack the rocks properly, the robot might flip, nor cannot climb it up. Thus, I had to try multiple times to have a stable unstructured terrain. Lastly, the plotting of datasets is not a difficult task; however, choosing which method to visualize the data was challenging. Because I had to compare the datasets that have an offset distance between the center of the AprilTag and the camera, even the naïve 3D-plot does not look intuitive without captions. Thus, I had to think of how to tackle this problem.
6. Were you able to meet the various milestones as stated in your project proposal? If not, why? [20 %]  
     
   I was able to apply the TF system and successfully interpret the current location of the robot with a single capture of the AprilTag in the beginning when we start the robot. Also, I was able to analyze the position and the trajectory taken from the AprilTag and the camera. However, I was not able to analyze the angles. I did not have enough time for the angles, and I had to spend more time on analyzing the positions and trajectories because of the unexpected result with the offset distance. It might become a more robust and precise analysis if I could compare the orientations of the robot as well. But if the position turned out to be bad, then there is no point for me to analyze the orientation of the robot. So, I put more weights on positions than orientations. Moreover, I could not publish the message due to time limitations. I unexpectedly spent too much time on position and trajectory analysis.
7. If you had more time, how would you extend this project? [10 %]  
     
   If I had more time, I would like to analyze the orientations as well, since the position analysis implies that the camera is much faster than that of the AprilTag’s. Also, I would like to publish message to avoid the positions to get lost. The last thing that I wanted to do was enable fisheye recordings on the camera and record the Rviz screen as well when I gather the datasets. Those videos might help in terms of analysis point, especially the trajectory, rather than having just a single video of robot moving.
8. Can you think of any ethical impacts future iterations of this project may have? If so how would you address them? [10 %]  
     
   From the wider point of view, this robot has an autonomous behavior of stacking/modifying the unstructured environment. Because the robot is looking for some suitable materials to add it to the utility structure, it might invade some natural species’ habitat and take some materials from there. Robots are not supposed to invade those, and should take materials from other places. In order to address this potential ethical impact, the robot needs to have another perception that avoids invading animals’ habitat -i.e. add perception about moving objects, avoid getting closer to those.

Survey Questions:

1. Do you have any specific comments on how we can modify the project portion of the course for future course offerings?  
   It would be great if we get to have more time to focus on projects. Also, the workload seems to be too much in the last few weeks of the semester; we need to prepare not only the final exam for the course, but also the project presentation as well.
2. Do you have any comments for future students who take the course?  
   It would be better if they are already familiar with ROS or take a brief look at ROS Tutorials page before they take the course. This will help them a lot in terms of programming assignments.