Intel Realsense Cameras to do SLAM

This project involves using Intel Realsense D415 and T265 cameras to achieve SLAM (D435 camera should also work with this). Specifically, this is done using **rtabmap**. SLAM using ocupancy mapping branch of realsense-ros instead of rtabmap discussed in excel file.

Getting Started

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes.

Prerequisites

This was done using Jetson Nano (Ubuntu 18.04.4 LTS) since Realsense was originally made to be run with Ubuntu and thus has easier setup.

Installing

If you have RealSense SDK and ROS installed and have set up a catkin workspace, skip to step 3.

1. Install Realsense

In order to install the RealSense SDK, we must first add the Intel repository. We will be using the tutorials from JetsonHacks, which uses convenience script to perform these actions.

- From a Terminal, clone the repository, and switch over to that repositories directory:

```
git clone https://github.com/JetsonHacksNano/installLibrealsense
cd installLibrealsense
```

- Install the sdk:

```
./installLibrealsense.sh
```

- Start realsense-viewer:

```
realsense-viewer
```

If you are having difficulty, follow the Jetsonhacks [tutorial](#) on installing Realsense until 7:30 min mark.

Make sure you get it running on the Realsense viewer before moving on to next steps.

2. Install ROS

- Clone repository on to the Jetson:

```
git clone https://github.com/JetsonHacksNano/installROS.git
cd installROS
```

- Run installROS

```
./installROS.sh -p ros-melodic-desktop
```

If you don't specificy package, then it'll install ros-melodic-ros-base by default.

- Setup Catkin Workspace

```
./setupCatkinWorkspace.sh optional-workspace-name
```

where optional-workspace-name is the name and path of the workspace to be used.

The default workspace name is catkin_ws. If a path is not specified, the default path is the current home directory. This script also sets up some ROS environment variables.

3. Install RealSense Wrapper for ROS

```
git clone https://github.com/JetsonHacksNano/installRealSenseROS
cd installRealSenseROS
./installRealSenseROS catkin-workplace-name
```

catkin-workplace-name is the path to the catkin_workspace to place the RealSense ROS package. If no catkin workplace name is specified, the script defaults to ~/catkin_ws.

4. Install the RTAB-Map ROS

```
sudo apt install ros-melodic-rtabmap-ros
```

This install's RTAB-Map's ROS package.

See here for the github repository and here for more info on demos/tutorials.

> 5. [Optional] Set the Camera Serial Number and Device Type

Setting the camera serial number in the launch files can help the software run more smoothly later in terms of identifying cameras etc. - Navigate to realsense2_camera launch folder. - Open the rs_rtabmap.launch file using vim or your favorite editor - Edit the default for **device_type_camera1** and **device_type_camera2** if they are not set to T265 and D4.5 cameras already. They should look like this:

```
<arg name="device_type_camera1"          default="t265"/>
<arg name="device_type_camera2"          default="d4.5"/>
```

- Edit the **serial_no_camera1** and **serial_no_camera2** arguments.

The serial number is 12-digits long and can be found on the cameras themselves.

How to Run

> 1. Navigate to your catkin workspace

Open a terminal window and enter

```
cd name-of-your-catkin-workspace
```

If you followed the jetsonhacks tutorials, the default name of your catkin workspace should be **catkin_ws**

> 2. Source the Devel

This step can only be done in the root catkin directory

NOTE: You must source the devel in every terminal window you open

```
source devel/setup.bash
```

> 3. Set ROS Network This step involves setting the ROS Network.

Note: This must be done in every window you use. See below.

- For first terminal window:

```
export ROS_HOSTNAME=localhost
export ROS_MASTER_URI=http://localhost:11311
roscore
```

- For all other terminal windows that you use:

```
export ROS_HOSTNAME=localhost
export ROS_MASTER_URI=http://localhost:11311
```

> 4. Navigate to realsense2_camera's launch folder

```
cd src/realsense-ros/realsense2_camera/launch
```

> 5. Run the rtabmap launch file

```
roslaunch realsense2_camera rs_rtabmap.launch
```

rtabmap doesn't always work on first run. You can try quitting and re-running again.

> 6. View the Map

Once RViz is running, make sure the Map checkbox is ticked under Displays (on the left half of the screen). If Map is not one of the displays, you can add it (button at the bottom left side)

```
Add -> By display type -> rviz -> Map
```

You can specificy what map topic you want. (grid_map, octomap_grid, grid_prob_map, etc). grid_map works quite well, but it is definitely worth experimenting.

> 7. [Optional] Add Other Displays

To add the D415 depth camera display:

```
Add -> By topic -> /d400 (what you named d415 camera) -> /depth -> /image_rect_raw -> double click image
```

That's it! Good Luck! See the below section for some tips on common errors and mistakes!

General Tips & Common Errors

This section will discuss potential and common errors.

1. control_transfer returned error, index: 768, error: No data available, number: 61

Generally, you can ignore this error (apparently the developers haven't removed this error).

2. ResourceNotFound: ___

For this error, you probably haven't installed what the error is pointing to (ex. realsense2_camera).

Ex. ResourceNotFound: realsense2_camera Fix: Run the following and try again

```
sudo apt install ros-melodic-realsense2-camera
```

If you are unsure of the install command, Google it! :)

3. roslaunch command runs but hangs almost immediately

This error generally has to do with ROS network settings. Might arise if you're working from home, new location, etc.

Make sure you've set the ROS_HOSTNAME, ROS_MASTER_URI, etc. See Step 3 under How to Run.

If you're still having issues, see ROS Network Setup.

4. Not seeing Map/Warning/Errors in RViz, RTabmap

If you're not seeing the map, make sure you've followed Step 6 above. Sometimes, it can be an error of the transform (TF)

Make sure the nodes and topics are as listed in the "Nodes and Topics" png file.

- "WARNING control_transfer returned error, index 768, error: No data available, number: 61" can be ignored

5. Other Errors

Here are some general tips:

- Quit the process and run again
- Wait a bit longer (sometimes it takes a while to get both cameras up and running)
- Source the devel in catkin root directory:

```
source devel/setup.bash
```

- Unplug cameras, replug, and try again
- Ignore Errors (some errors will fix itself or don't really impact the running of program)
- Look it up! There's a lot of documentation on errors people have faced and how they have fixed it! Hopefully some of these work :)