Run loam:

roslaunch loam\_velodyne loam\_velodyne.launch

Run gmap

nvidia@nvidia-nx1:~/catkin\_ws$ roslaunch gmapping test\_vel.launch

Play bag for both:

rosbag play 2019-10-25-16-28-27.bag --clock

Run script for profiling:

Rosrun imu\_3dm\_gx3 util\_maP\_LOam.py

Rosrun imu\_3dm\_gx3 util\_maP\_gmap.py

<node name = "imu\_3dm\_gx3" pkg = "imu\_3dm\_gx3" type = "util\_map\_loam.py" respawn = "false" output = "screen" />

<node name = "imu\_3dm\_gx3" pkg = "imu\_3dm\_gx3" type = "util\_map\_gmap.py" respawn = "false" output = "screen" />

Run noise.py script in imu\_3dx\_gx

Orbslam3:

Install orbslam3: <https://github.com/UZ-SLAMLab/ORB_SLAM3>

Run orbslam3:

1. Download a sequence from <http://vision.in.tum.de/data/datasets/rgbd-dataset/download> and uncompress it.
2. Associate RGB images and depth images using the python script [associate.py](http://vision.in.tum.de/data/datasets/rgbd-dataset/tools). We already provide associations for some of the sequences in *Examples/RGB-D/associations/*. You can generate your own associations file executing:

python associate.py PATH\_TO\_SEQUENCE/rgb.txt PATH\_TO\_SEQUENCE/depth.txt > associations.txt

Execute the following command. Change TUMX.yaml to TUM1.yaml,TUM2.yaml or TUM3.yaml for freiburg1, freiburg2 and freiburg3 sequences respectively. Change PATH\_TO\_SEQUENCE\_FOLDERto the uncompressed sequence folder. Change ASSOCIATIONS\_FILE to the path to the corresponding associations file. ./Examples/RGB-D/rgbd\_tum Vocabulary/ORBvoc.txt Examples/RGB-D/TUMX.yaml PATH\_TO\_SEQUENCE\_FOLDER ASSOCIATIONS\_FILE

RRT

Cd /CUDA-RRT

make

./RRT

Depth Estimate:

Follow instructions from

<https://github.com/isl-org/MiDaS>

Fastdepth v2

Cd /darknet

Make

./Depth\_estimation\_multiple\_images.sh

OPtical flow estimate:

Follow instructions

https://github.com/sniklaus/pytorch-pwc