Current Plan:

Create mask based on lane line color

Future Plan:

Segment anything via text prompt

Get location of the segmented thing

Go to that location

Hardware:

Nvidia Jetson Nano

EAI XL2 LiDAR

Dabai depth camera

iFlytek Voice Assistant/Google Assistant

Software:

Ubuntu 18.04

ROS1 Melodic

## Guides:

Agile X Limo GitHub

https://github.com/agilexrobotics/limo-doc/blob/master/Limo%20user%20manual(EN).md#7-depth-camera--lidar-mapping

Trossen documentation

<https://docs.trossenrobotics.com/agilex_limo_docs/index.html>

Indrorobotics documentation

https://indrorobotics.notion.site/Limo-fb4fa478d1524127a86eb06de742c902

## Startup

Long press power button to start

## Login to Ubuntu

Username: agilex

Password: agx

## Remote in via nomachine

Nomachine

Ip of limo: 172.18.19.2

Ip of limo 1323: 172.18.28.184

## ROS

<https://www.clearpathrobotics.com/assets/guides/melodic/ros/Practical%20Example.html>

Rosnode list

to visualize nodes:

rqt\_graph

Rostopic list

Rostopic echo

Echo only once:

rostopic echo -n 1 /topic\_name

Rosservice list

Rosservice call <service>

Rosmsg list

Rosmsg info <msg Type>

Save output to a file:

Echo > test.txt

Publish a new topic called /hello with topic type std\_msgs/String with a message “Hello Robot”

rostopic pub /hello std\_msgs/String “Hello Robot"

## Depth Camera

<https://github.com/orbbec/ros_astra_camera>

Astra\_camera package

wrapper

roslaunch astra\_camera dabai\_u3.launch

openCV test file:

Desktop/pythonTest.py

Topics:

/camera/rgb/image\_raw

Type: sensor\_msgs/Image

/camera/rgb/image\_rect\_color/compressed

Services:

/camera/get\_camera\_info

File location:

/home/agilex/agilex\_ws\_W24/agilex\_ws/src/ros\_astra\_camera/launch

Launch rviz:

Rviz

Add Image:

fixed\_frame → camera\_link

Image topic → your choice

Options:

RGB

Depth

IR

Point Cloud

Add depth cloud

Depth map topic → your choice

## LiDAR

Launch lidar

roslaunch limo\_bringup limo\_start.launch pub\_odom\_tf:=false

Bring up rviz visualization of lidar

roslaunch limo\_bringup lidar\_rviz.launch

Choose fixed frame

Add LaserScan Display

Choose /scan topic

## Basic Movement

roslaunch limo\_base limo\_base.launch

/home/agilex/agilex\_ws/src/limo\_ros/limo\_base/launch

forward

rostopic pub -1 /cmd\_vel geometry\_msgs/Twist -- '[0.5, 0.0, 0.0]' '[0.0, 0.0, 0.0]'

Turn left

rostopic pub -1 /cmd\_vel geometry\_msgs/Twist -- '[0.0, 0.0, 0.0]' '[0.0, 0.0, 0.5]'

rosrun rqt\_graph rqt\_graph

import rospy

import os

import sys

from geometry\_msgs.msg import Twist

pub = rospy.Publisher('/cmd\_vel',Twist, queue\_size=10)

rospy.init\_node('voice\_ctr\_node',anonymous=True)

def pub\_cmd\_msg(msg):

rate = rospy.Rate(10)

tick = 0

while tick <= 30:

pub.publish(msg)

tick = tick+1

rate.sleep()

cmd = Twist()

cmd.linear.x = 0.5

cmd.angular.z = 0.0

pub\_cmd\_msg(cmd)

## Lane Detection

/home/agilex/agilex\_ws/src/src/scripts/vision/launch

Lane\_detection.launch

### LaneNet

The Windows equivalent of the Linux command export PYTHONPATH=$PWD:$PYTHONPATH is:

set PYTHONPATH=%CD%;%PYTHONPATH%

MaybeShewill-CV/lanenet-lane-detection

### Segment Anything

MaybeShewill-CV/segment-anything-u-specify

C:\Users\orywi\Google Drive\School\Detroit Mercy\Office Assistant\segment-anything-u-specify-main\segment-anything-u-specify-main

python tools\sam\_clip\_text\_seg.py --input\_image\_path .\data\test\_images\test\_bear.jpg --text bear

### Lane Line Masking

Location on Ory’s laptop:

C:\Users\orywi\Google Drive\School\Detroit Mercy\Office Assistant\opencv

Start ROS and activate Dabai camera:  
roslaunch astra\_camera dabai\_u3.launch

Check stream via rviz:

Launch rviz:

Rviz

Add Image:

fixed\_frame → camera\_link

Image/ Image topic → /camera/rgb/image\_raw

Check stream via rqt\_graph:

New terminal → rqt\_graph

openCV test file:

Desktop/pythonTest.py

This will display a stream of masked images on the LIMO, showing only the lane lines. It will update as the robot is moved.

**Next steps:**

* Running the updated angle finding code on the LIMO.
* Using the angle information to control the LIMO to stay within the lane lines.

Topics:

/camera/rgb/image\_raw

Type: sensor\_msgs/Image

## Map course

On robot:

New terminal:

Needed to map

roslaunch limo\_bringup limo\_start.launch pub\_odom\_tf:=false

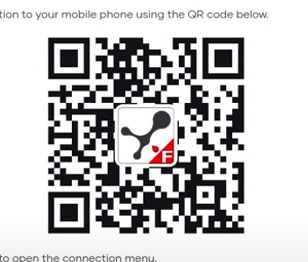
Launch mapping program

New terminal:

roslaunch limo\_bringup limo\_gmapping.launch

Drive around and map environment

Bluetooth remote control app:



### Save map

New terminal

cd ~/agilex\_ws/src/limo\_ros/limo\_bringup/maps/

rosrun map\_server map\_saver -f demoDay2

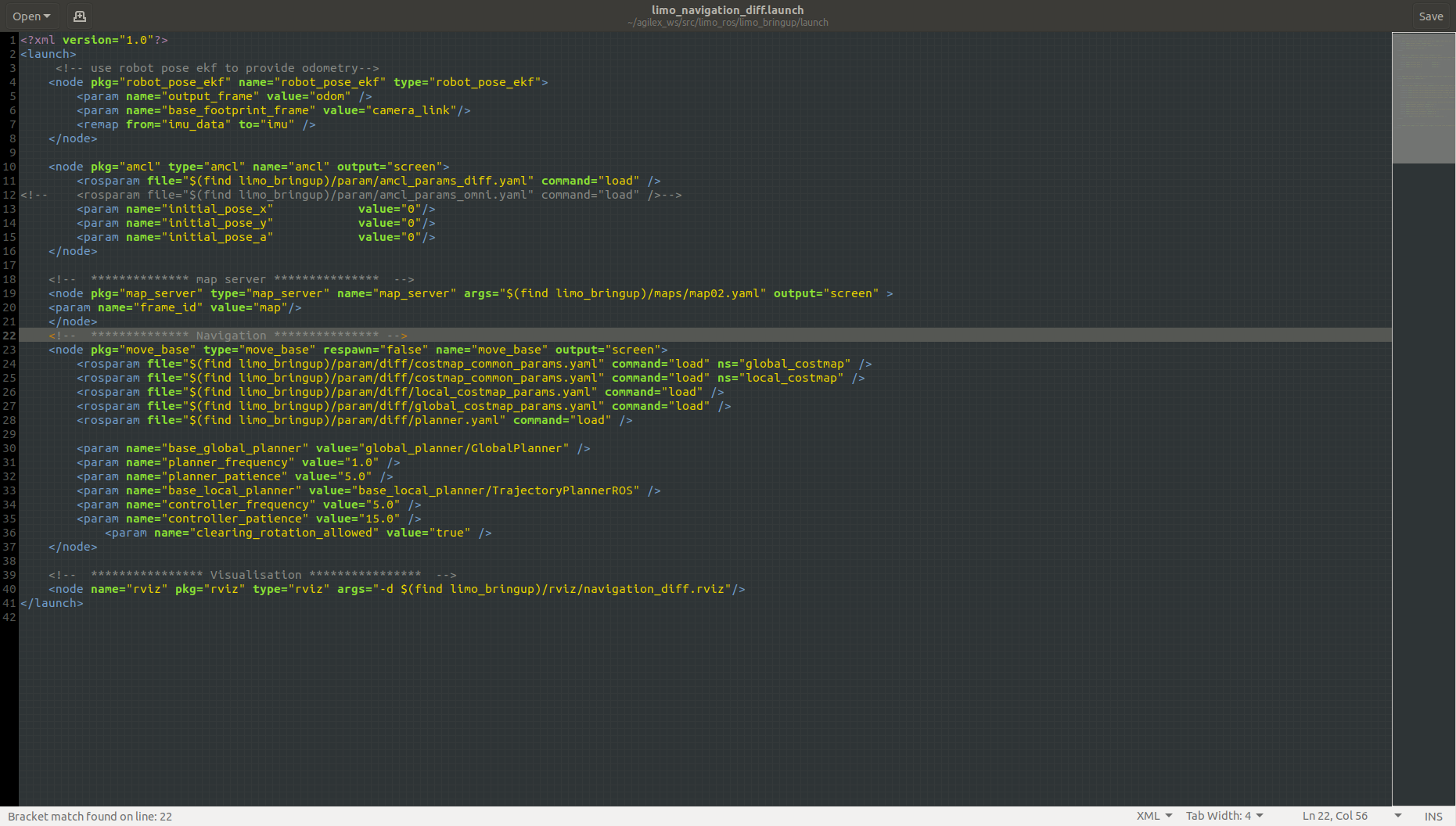
Stop mapping terminal

## Run navigation

### Change navigation file to point to created map

~/agilex\_ws/src/limo\_ros/limo\_bringup/launch/limo\_navigation\_ackerman.launch

Line 19



### Limit velocities

~/agilex\_ws/src/limo\_ros/limo\_bringup/param/ackerman/teb\_local\_planner\_params.yaml

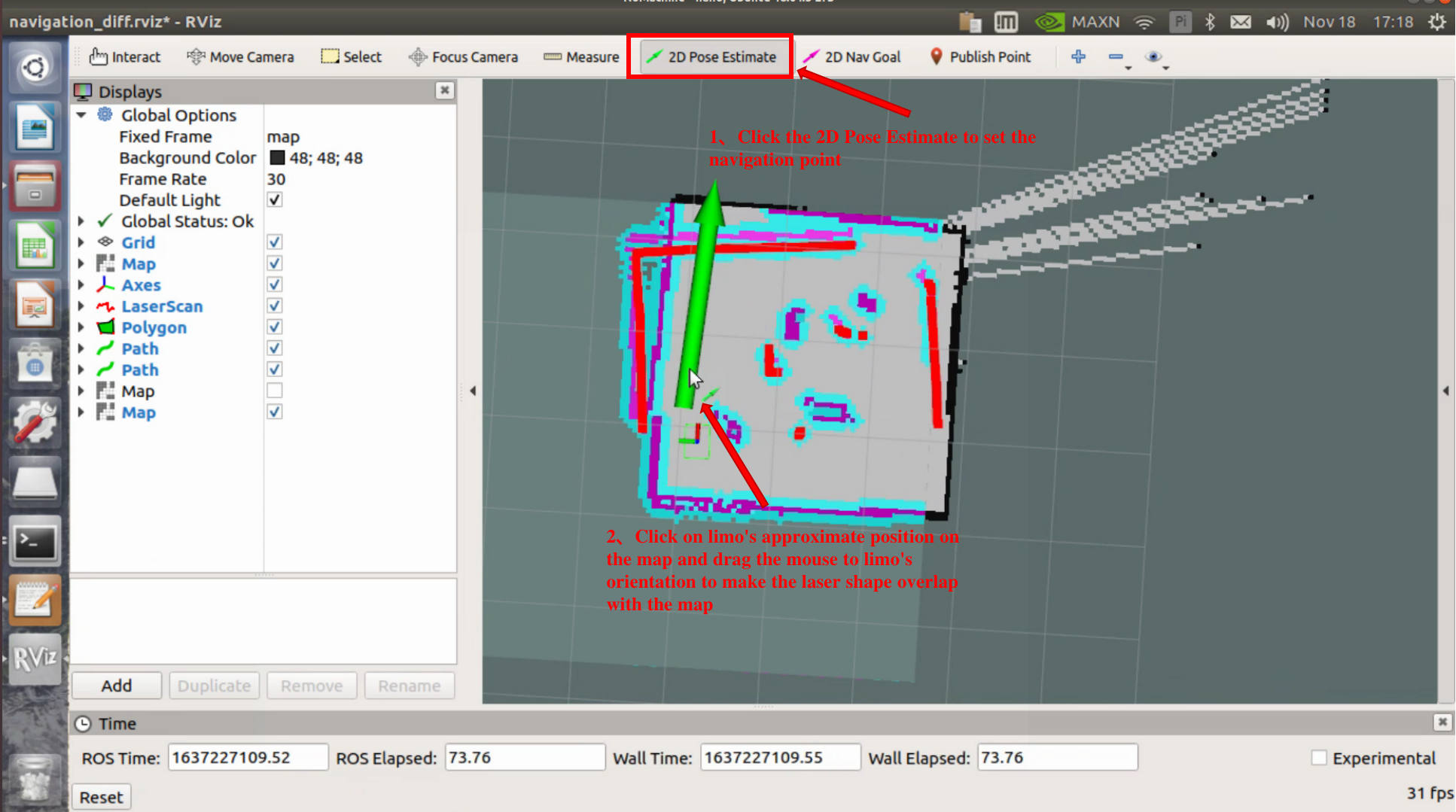
Max\_vel\_x

max\_vel\_x\_backwards

### Run navigation file

roslaunch limo\_bringup limo\_navigation\_ackerman.launch

### 2d pose estimate button to locate robot on map



### 2d nav goal button to tell robot where to go

Once phone app connection is turned off, the robot will move automatically

## VSLAM

Create map

roslaunch limo\_bringup limo\_start.launch pub\_odom\_tf:=true

roslaunch astra\_camera dabai\_u3.launch

roslaunch limo\_bringup limo\_rtabmap\_orbbec.launch

roslaunch limo\_bringup rtabmap\_rviz.launch

Map is saved on exit

Navigate

roslaunch limo\_bringup limo\_start.launch pub\_odom\_tf:=true

roslaunch astra\_camera dabai\_u3.launch

roslaunch limo\_bringup limo\_rtabmap\_orbbec.launch localization:=true

roslaunch limo\_bringup limo\_navigation\_rtabmap\_ackerman.launch

roslaunch limo\_bringup rtabmap\_rviz.launch

## Text Recognition

3 terminals:

Terminal 1:

roscore

Terminal 2:

rosrun vision detect\_node.py

Terminal 3:

rostopic echo /detect\_word\_reslut

## Traffic Light Recognition

Note: traffic light needs to be in front of camera for this to work

3 terminals:

Terminal 1:

roslaunch astra\_camera dabai\_u3.launch

Terminal 2:

roslaunch darknet\_ros yolo\_v3\_tiny.launch

Terminal 3:

roslaunch vision traffic\_light\_located.launch

## Voice Commands

roslaunch limo\_base limo\_base.launch

/home/agilex/agilex\_ws/src/limo\_ros/limo\_base/launch

## To turn off limo:

Shut down via ubuntu

Hold power button for a couple seconds til lights go off

## 

# Troubleshooting

Rviz map jumping around:

Stop “roslaunch limo\_bringup limo\_gmapping.launch” terminal.

# Conda

start miniconda terminal

add conda channel

conda config –add channels <new\_channel>

conda create -n gymenv

conda create -n myenv python=3.8

conda activate gymenv

to delete an environment:

conda deactivate

conda remove --name ENV\_NAME --all

conda install --file requirements.txt

libraries.io to see package dependencies without installing