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

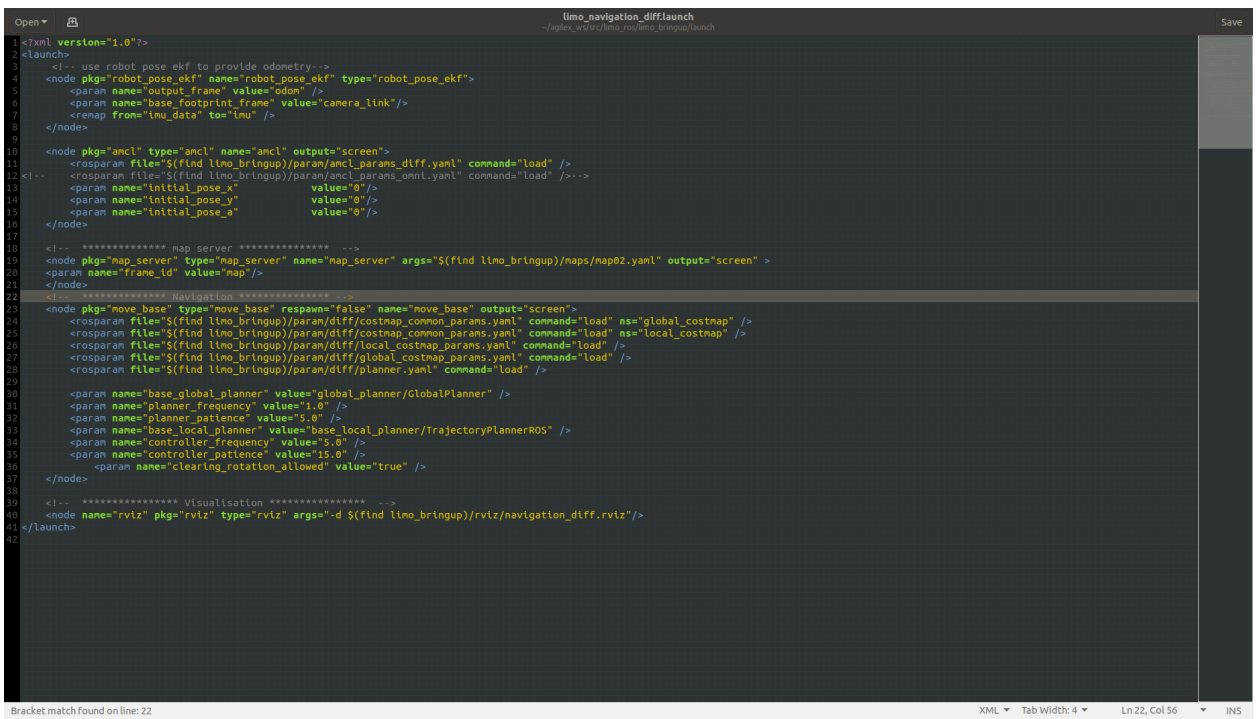
```
cd ~/agilex_ws/src/limo_ros/limo_bringup/maps/
rosrun map_server map_saver -f demoDay2

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
```



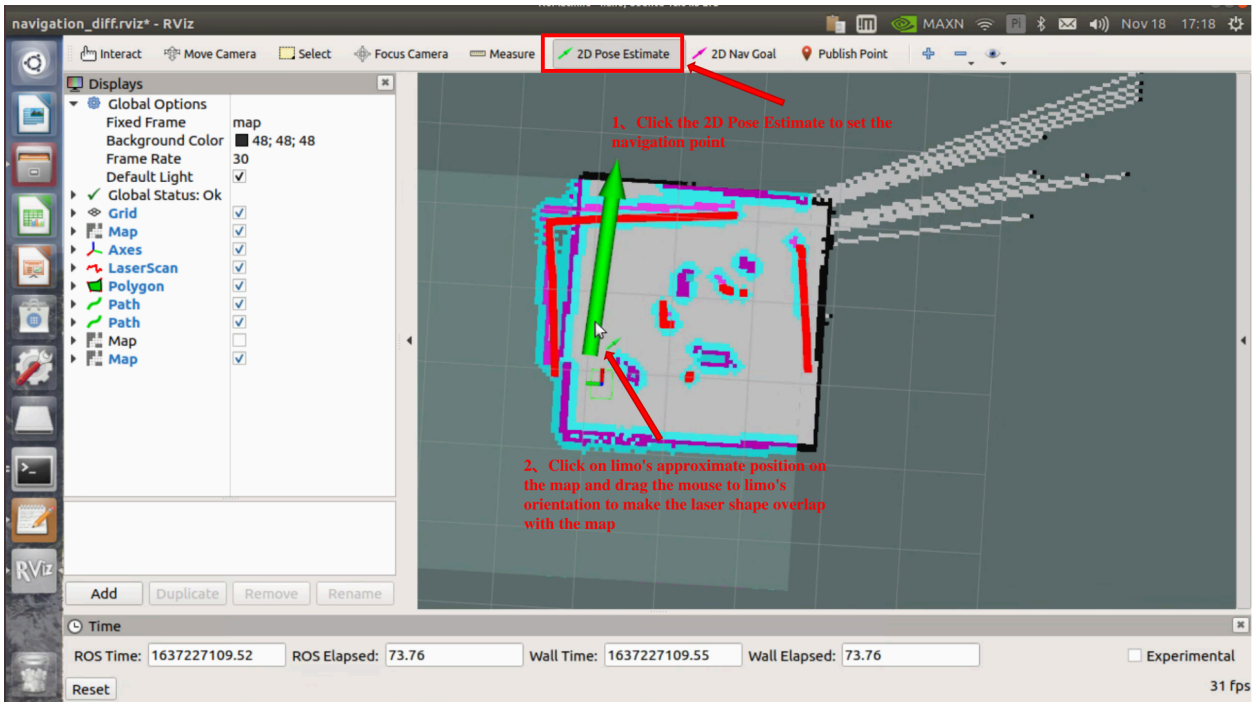
Limit velocities

```
~/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