ROS Lab 6 Homework

Vipul Dinesh, 220929024, MTE-A-09

1. Place `model.config` and `model.sdf` in `~/.gazebo/models/red curve`

model.config

```
<?xml version="1.0" ?>
<model>
   <name>red curve</name>
   <version>1.0</version>
   <sdf version="1.7">model.sdf</sdf>
   <author>
       <name></name>
       <email></email>
   </author>
   <description></description>
</model>
```

model.sdf

```
<?xml version='1.0'?>
<sdf version='1.7'>
 <model name='red curve'>
   <model name='red line'>
     <static>1</static>
     <link name='link_ground'>
       <collision name='collision'>
         <geometry>
           <plane>
             <normal>0 0 1</normal>
             <size>0.1 3.2</size>
           </plane>
         </geometry>
         <surface>
           <friction>
             <ode>
               <mu>100</mu>
               <mu2>50</mu2>
             </ode>
           </friction>
         </surface>
       </collision>
       <visual name='visual_ground'>
         <cast_shadows>0</cast_shadows>
         <geometry>
```

```
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.5 5</size>
                </plane>
              </geometry>
              <material>
                <script>

<uri>file://media/materials/scripts/gazebo.material</uri>
                  <name>Gazebo/Red</name>
                </script>
              </material>
            </visual>
          </link>
          <pose>2.25 0 0 0 0 0</pose>
        </model>
        <model name='red line_0'>
          <static>1</static>
          <link name='link_ground'>
            <collision name='collision'>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.1 3.2</size>
                </plane>
              </geometry>
              <surface>
                <friction>
                  <ode>
                    <mu>100</mu>
                    <mu2>50</mu2>
                  </ode>
                </friction>
              </surface>
            </collision>
            <visual name='visual_ground'>
              <cast_shadows>0</cast_shadows>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.5 5</size>
                </plane>
              </geometry>
              <material>
                <script>
```

```
<uri>file://media/materials/scripts/gazebo.material</uri>
                  <name>Gazebo/Red</name>
                </script>
              </material>
            </visual>
          </link>
          <pose>-2.25 0 0 0 0 0</pose>
        </model>
        <model name='red_line_rotated'>
          <static>1</static>
          <link name='link_ground'>
            <collision name='collision'>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.1 3.2</size>
                </plane>
              </geometry>
              <surface>
                <friction>
                  <ode>
                    <mu>100</mu>
                    <mu2>50</mu2>
                  </ode>
                </friction>
              </surface>
            </collision>
            <visual name='visual_ground'>
              <cast_shadows>0</cast_shadows>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.5 5</size>
                </plane>
              </geometry>
              <material>
                <script>

<uri>file://media/materials/scripts/gazebo.material</uri>
                  <name>Gazebo/Red</name>
                </script>
              </material>
            </visual>
            <pose>0 0 0 0 0 1.570796327</pose>
```

```
          </link>
          <pose>0 2.25 0 0 0 0</pose>
        </model>
        <model name='red_line_rotated_0'>
          <static>1</static>
          <link name='link_ground'>
            <collision name='collision'>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.1 3.2</size>
                </plane>
              </geometry>
              <surface>
                <friction>
                  <ode>
                    <mu>100</mu>
                    <mu2>50</mu2>
                  </ode>
                </friction>
              </surface>
            </collision>
            <visual name='visual_ground'>
              <cast_shadows>0</cast_shadows>
              <geometry>
                <plane>
                  <normal>0 0 1</normal>
                  <size>0.5 5</size>
                </plane>
              </geometry>
              <material>
                <script>

<uri>file://media/materials/scripts/gazebo.material</uri>
                  <name>Gazebo/Red</name>
                </script>
              </material>
            </visual>
            <pose>0 0 0 0 0 1.570796327</pose>
          </link>
          <pose>0 -2.25 0 0 0 0</pose>
        </model>
        <static>0</static>
        <allow_auto_disable>1</allow_auto_disable>
      </model>
```

```
        </sdf>
```

2. Launch gazebo using the launch file, draw the path (insert→red curve) and place the robot on the path

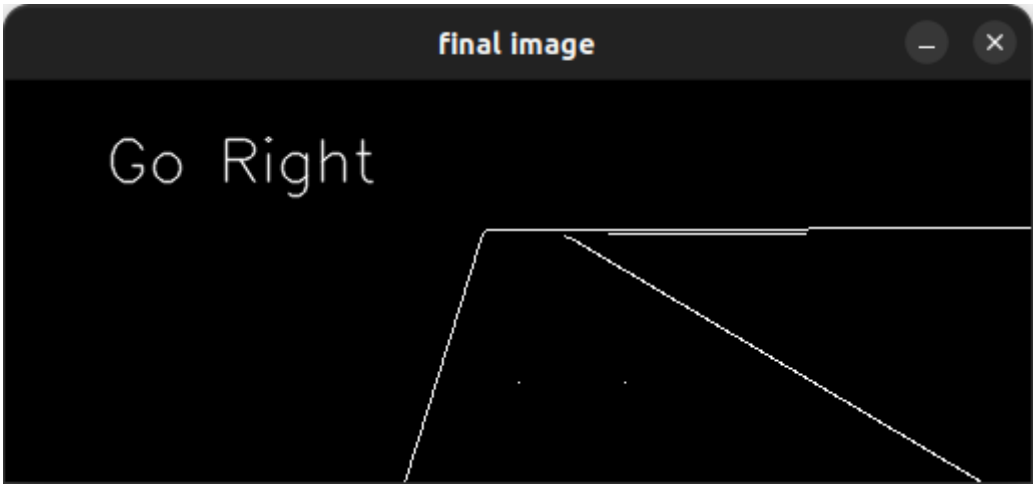
- `ros2 launch perception gazebo.launch.py`

3. Run the capture_image node, keep closing window to get a new image and press Ctrl+c to capture a satisfactory image of the road

- `ros2 run perception capture`

4. Run the extract_road node in vscode directly and modify values of `light_line` and `dark_line` to get proper masking

```
light_line = numpy.array([120,0,0])
dark_line = numpy.array([150,10,10])
```

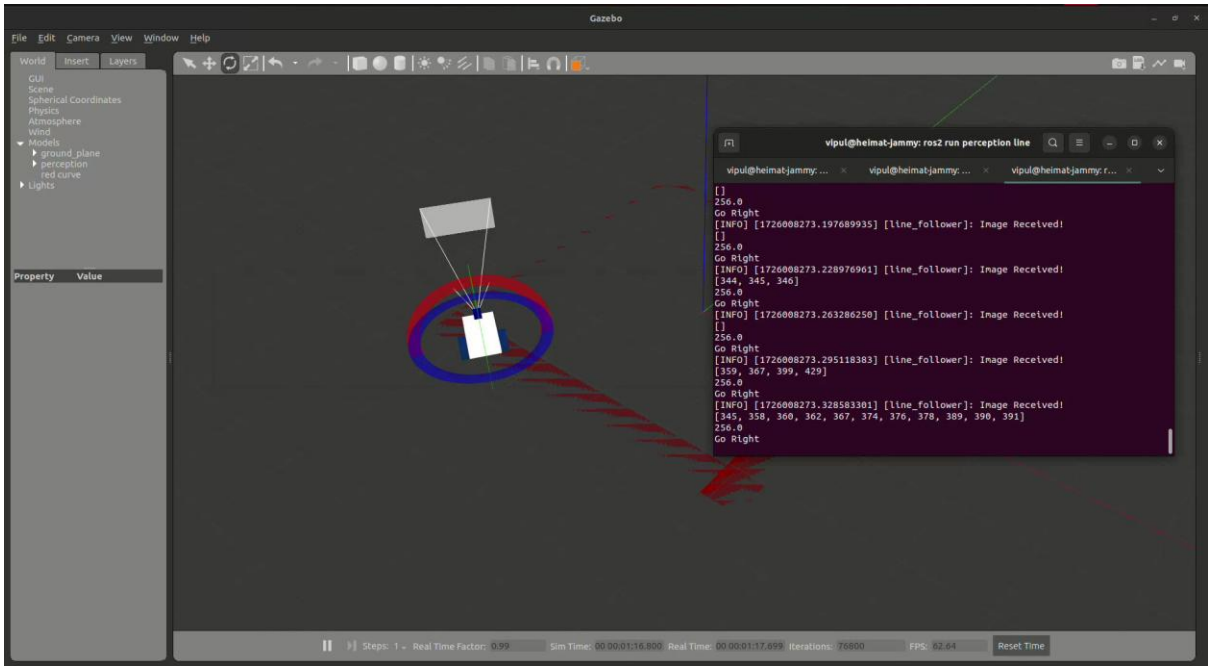


5. Modify the `line_follow.py` with the same values for `light_line` and `dark_line`. Then rebuild the package and source it again

- `cd ~/ros2_ws/`
- `colcon build --packages-select perception`
- `source ~/ros2_ws/install/setup.zsh`

6. Run the `line_follow` node (while gazebo is still running and red path is imported)

- `ros2 run perception line`


Gazebo
File Edit Camera View Window Help
World Insert Layers
GUI
Scene
Spherical Coordinates
Physics
Atmosphere
Wind
Models
ground_plane
perception
red curve
Lights
Property Value
vipul@heimat-jammy: ros2 run perception line
vipul@heimat-jammy: ...
vipul@heimat-jammy: ...
vipul@heimat-jammy: r...
[]
256.0
Go Right
[INFO] [1726008273.197689935] [line_follower]: Image Received!
[]
256.0
Go Right
[INFO] [1726008273.228976961] [line_follower]: Image Received!
[344, 345, 346]
256.0
Go Right
[INFO] [1726008273.263286250] [line_follower]: Image Received!
[]
256.0
Go Right
[INFO] [1726008273.295118383] [line_follower]: Image Received!
[359, 367, 399, 429]
256.0
Go Right
[INFO] [1726008273.328583301] [line_follower]: Image Received!
[345, 358, 360, 362, 367, 374, 376, 378, 389, 390, 391]
256.0
Go Right
Steps: 1 Real Time Factor: 0.99 Sim Time: 00 00:01:16.800 Real Time: 00 00:01:17.699 Iterations: 76600 FPS: 62.64
Reset Time