Turtlebot4 SLAM

운영체제의 실제

안인규 (Inkyu An)

Turtlebot4 – Sensor Data

- RPLIDAR A1M8
  - Topic name: /scan (type: sensor_msgs/msg/LaserScan)

- OAK-D-Pro
  - Topic names:

Topic name	Message type	Description
/turtlebot4/oakd/rgb/preview/image_raw	sensor_msgs/msg/Image	컬러(빨강-녹색-파랑) 이미지 스트림.
/turtlebot4/oakd/rgb/preview/image_raw/compressed	sensor_msgs/msg/CompressedImage	압축된 컬러 이미지 스트림.
/turtlebot4/oakd/rgb/preview/camera_info	sensor_msgs/msg/CameraInfo	카메라 내부 파라미터(초점거리, 왜곡 계수 등) 정보.
/turtlebot4/oakd/stereo/depth	sensor_msgs/msg/Image	깊이(depth) 이미지 스트림: 각 픽셀에 depth value가 담긴 이미지 형태.

Turtlebot4 – SLAM

- Run SLAM (recommended to run synchronous SLAM on a remote PC to get a higher resolution map):

```
$ ros2 launch turtlebot4_navigation slam.launch.py
```

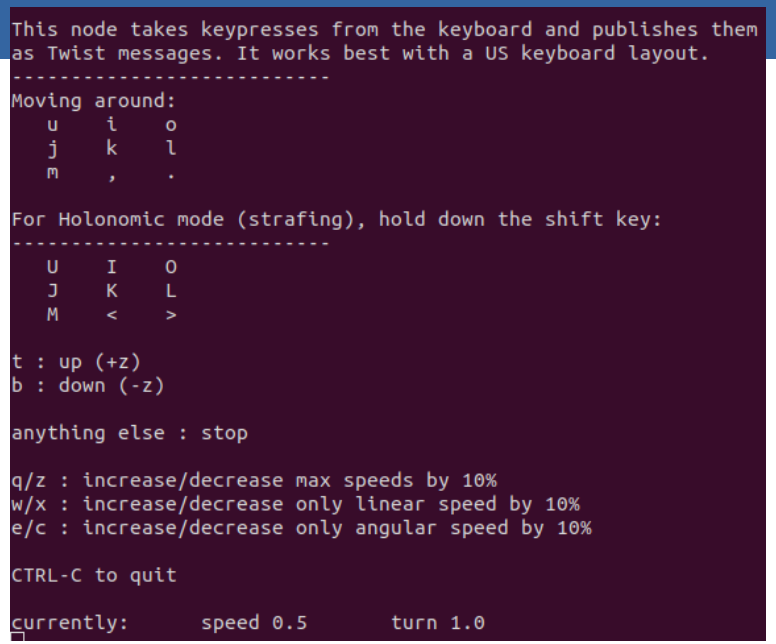
- To visualize the map, launch Rviz2:

```
$ ros2 launch turtlebot4_viz view_navigation.launch.py
```

Turtlebot4 – SLAM

- The simple way to get our robot driving is to use a keyboard application on our PC:

```
$ sudo apt install ros-jazzy-teleop-twist-keyboard
$ ros2 run teleop_twist_keyboard teleop_twist_keyboard --ros-args -p
stamped:=true
```



- Can save the map:

```
$ ros2 service call /slam_toolbox/save_map slam_toolbox/srv/SaveMap "{name: {data: 'map_name'}}"
```

Turtlebot4 – Localization

- Open a terminal and launch:

```
$ ros2 launch turtlebot4_navigation localization.launch.py map:=map_name.yaml
```

- To visualize the map, launch Rviz2:

```
$ ros2 launch turtlebot4_viz view_navigation.launch.py
```

SLAM Toolbox

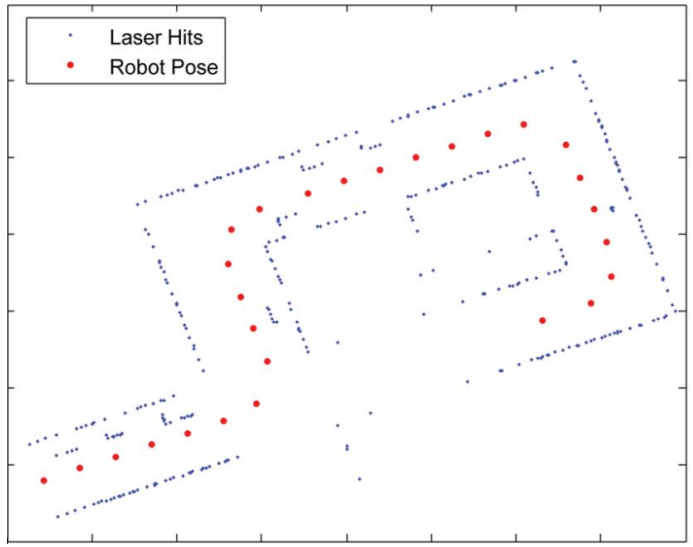
- ROS 2 officially supported 2D SLAM package
- Provides **Lifelong SLAM** capability (map update and reuse) → Pose graph를 저장해야 함
- SLAM Mode:
  - Offline SLAM (sync SLAM)
  - Online SLAM (async SLAM)
- Map 구조
  - PGM file: Occupancy grid map (Image)
  - YAML file: SLAM 설정 정보

- 흑백 이미지 (occupancy grid)
- 픽셀 값:
- 0 : 장애물 (검정)
- 254~255 : 자유 공간 (흰색)
- 205 : 미탐색 영역 (회색)

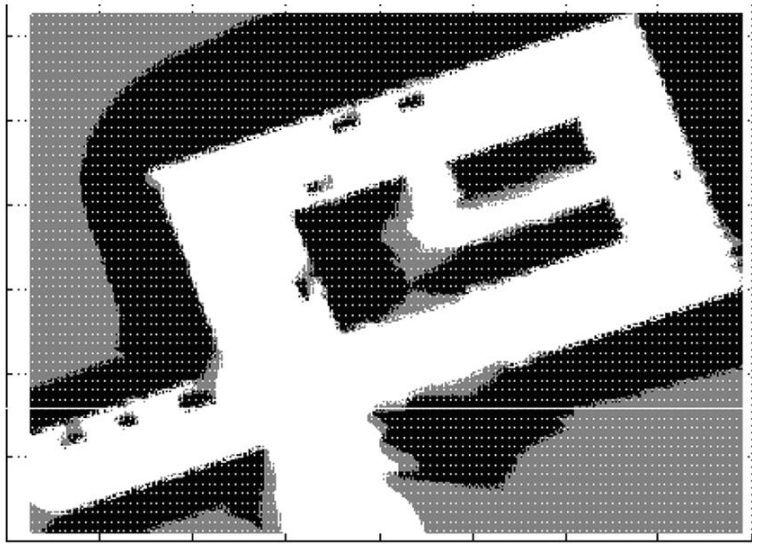
```yaml
image: my_map.pgm
resolution: 0.05
origin: [-10.0, -10.0, 0.0]
occupied_thresh: 0.65
free_thresh: 0.20
negate: 0
```

Occupancy Grid Map

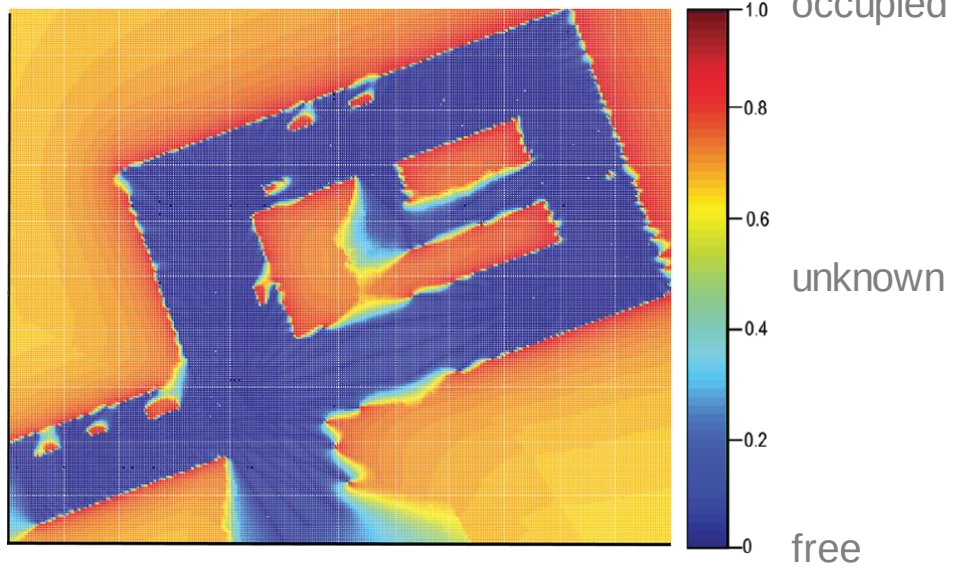
- An occupancy map represents the existence of surroundings, given sensor measurements and their poses



Sensor measurements and their poses



Occupancy state



Occupancy probability

Image: O'Callaghan and Ramos, Gaussian process occupancy maps, IJRR 12

Occupancy Grid Map

- Update the map based on Ray Casting
- Utilize the log-odds:
  - $l = \log \frac{P(A)}{1-P(A)}$, $P(A) = \frac{1}{1+e^{-l}}$
  - $p = 0.5 \rightarrow l = 0$
  - $p > 0.5 \rightarrow l > 0$
  - $p < 0.5 \rightarrow l < 0$
- Every cell is initialized with $p = 0.5$
  - If hit, $l = l + 0.619$
  - If hit, $l = l - 0.619$

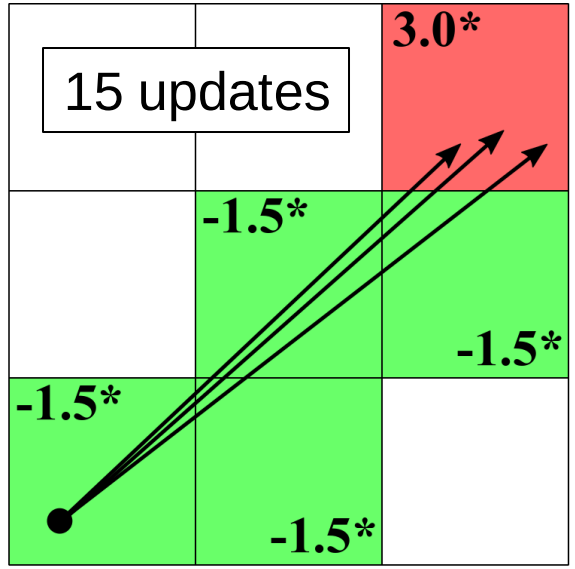


Image: Kwon's Ph.D. Desertation