Autonomous Vehicle class Final Project

윤 시 원

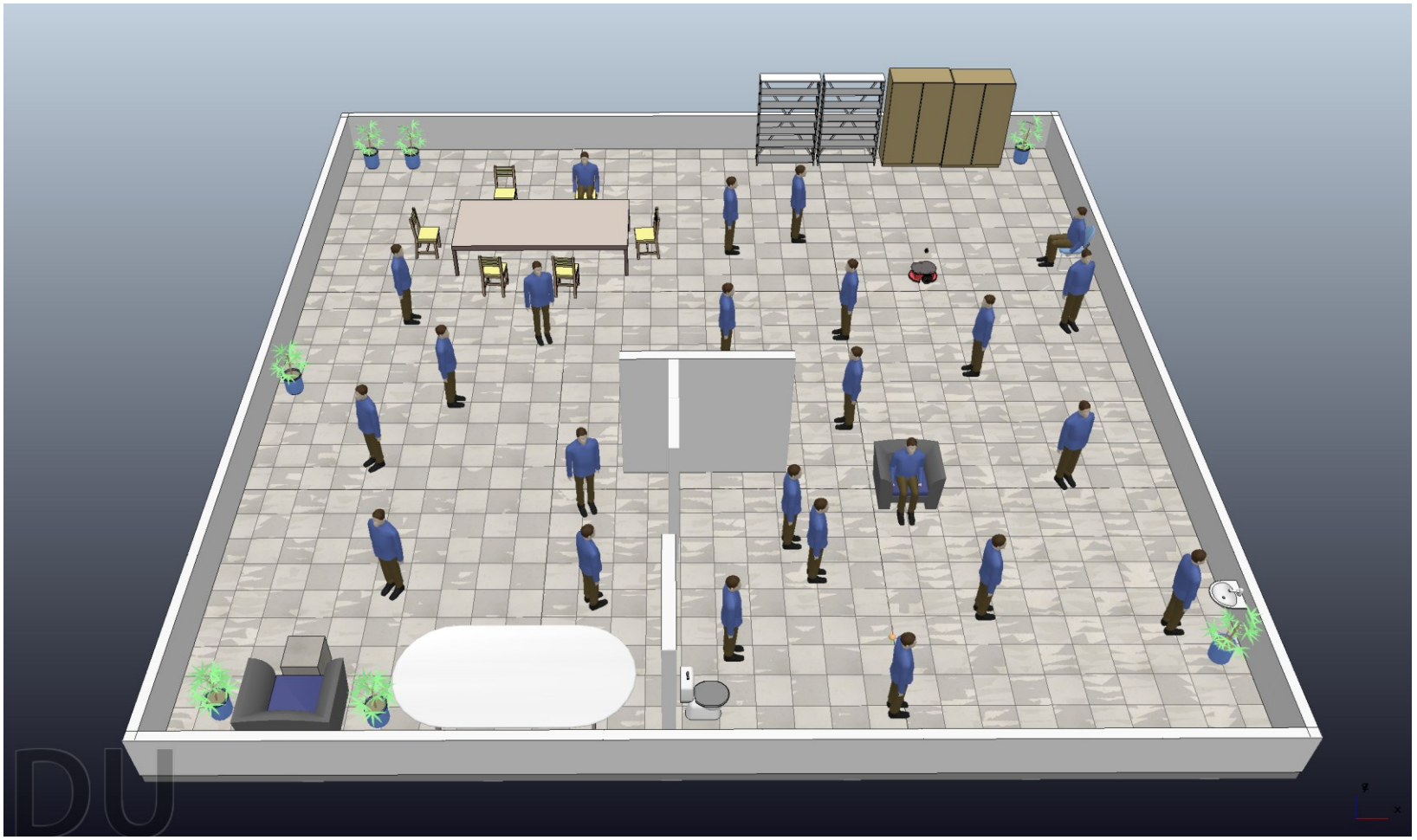
전자전기공학부 석사 과정

2023000853

Project Goals

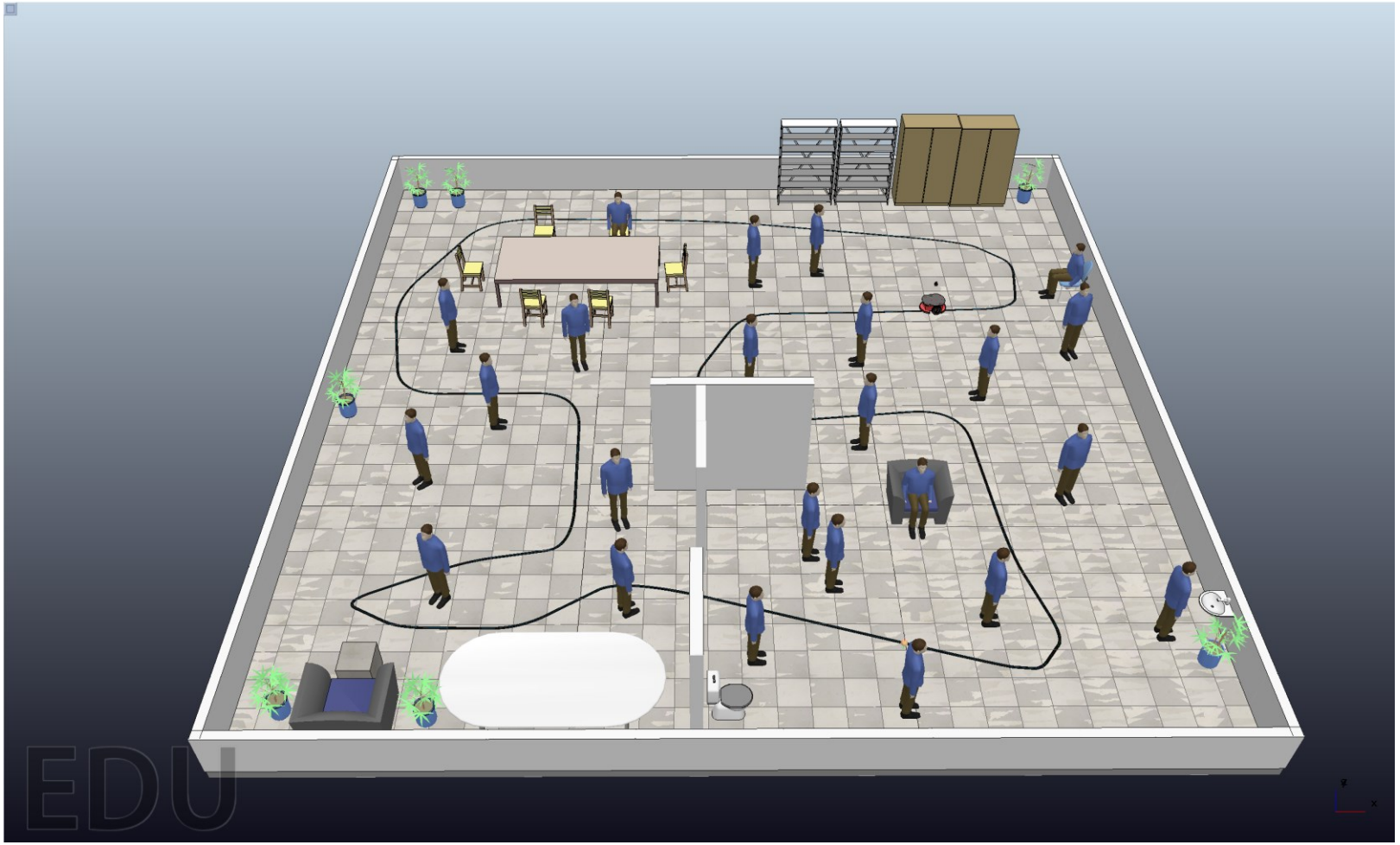
- Create a map using vrep
- SLAM(Simultaneous localization and mapping) to scan the map and plot grid map
- Find the optimal path on the grid map using the path planning algorithm

Map



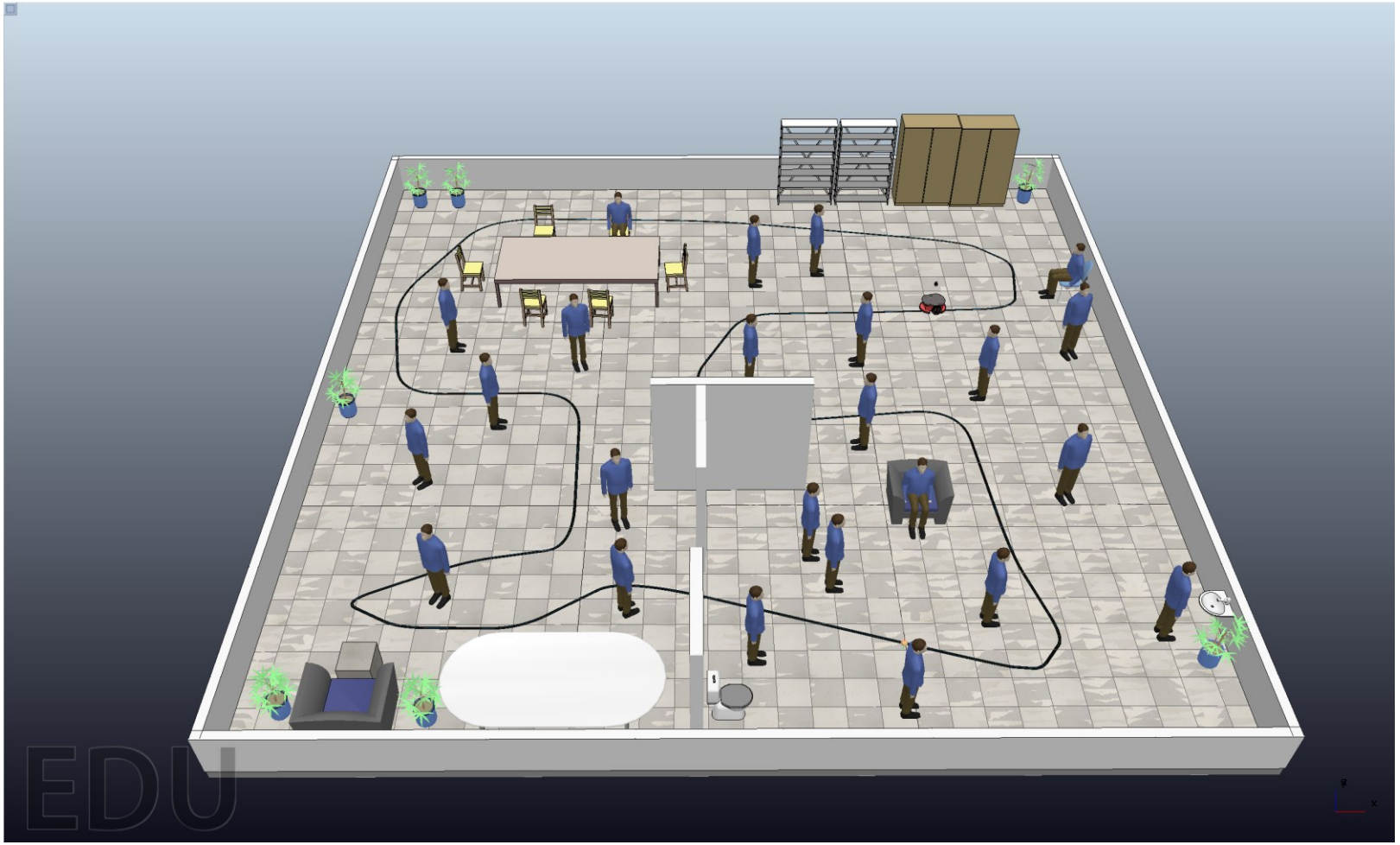
- Create a map similar to the indoor environment with many obstacles

Create Path



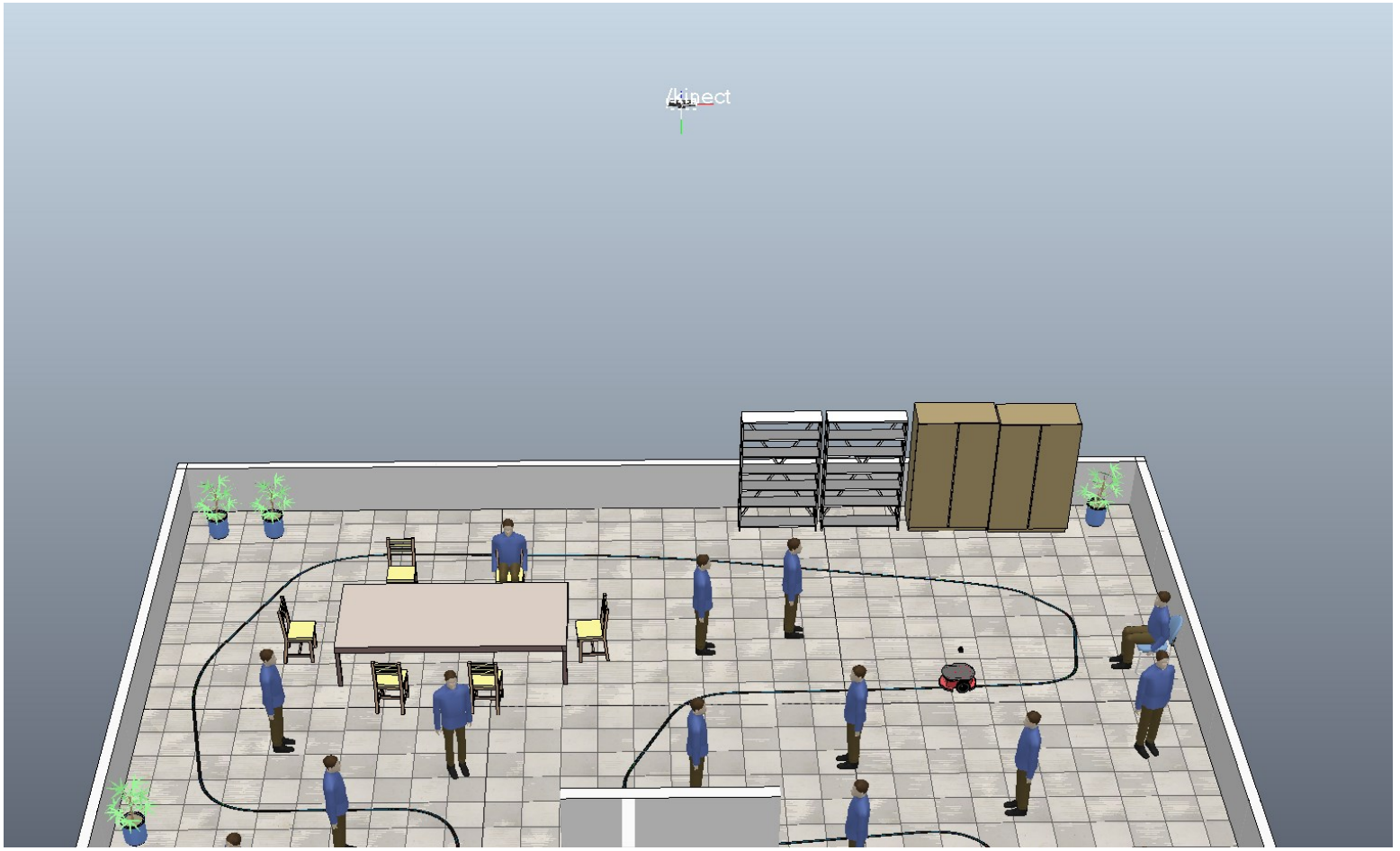
- Make the robot move along the path and scan the map.

Create Path



- Create the path to make the robot can scan all parts of the map.

Map



- To compare the results with the map scanned with the lidar,
  add a top view kinect cam with appropriate FOV and sensing distance

SLAM (Simultaneous localization and mapping)

- SLAM : Simultaneous localization and mapping

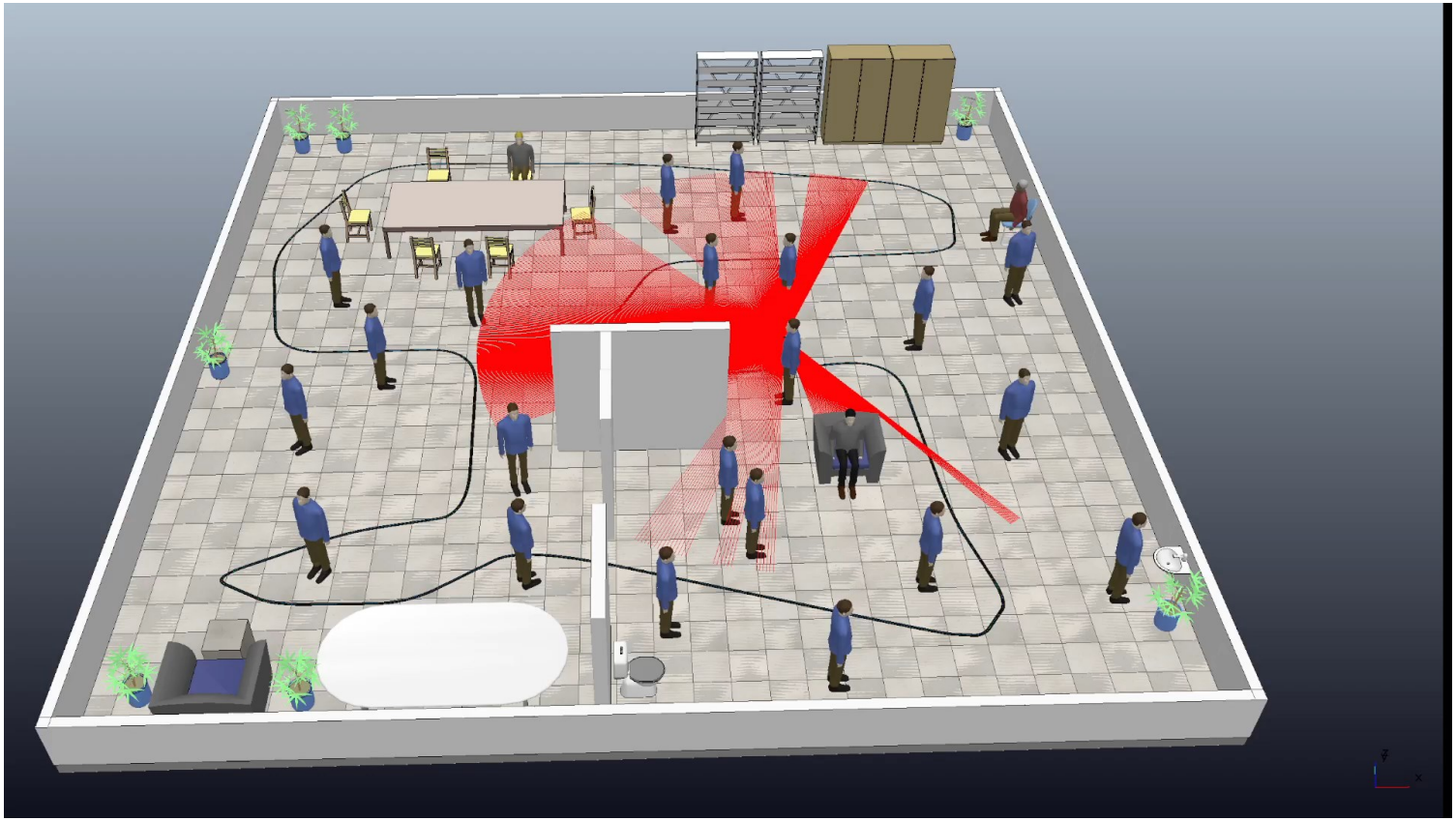
- Reason to do SLAM

  : To get a map when the map is unknown

- How to do SLAM in vrep and Matlab

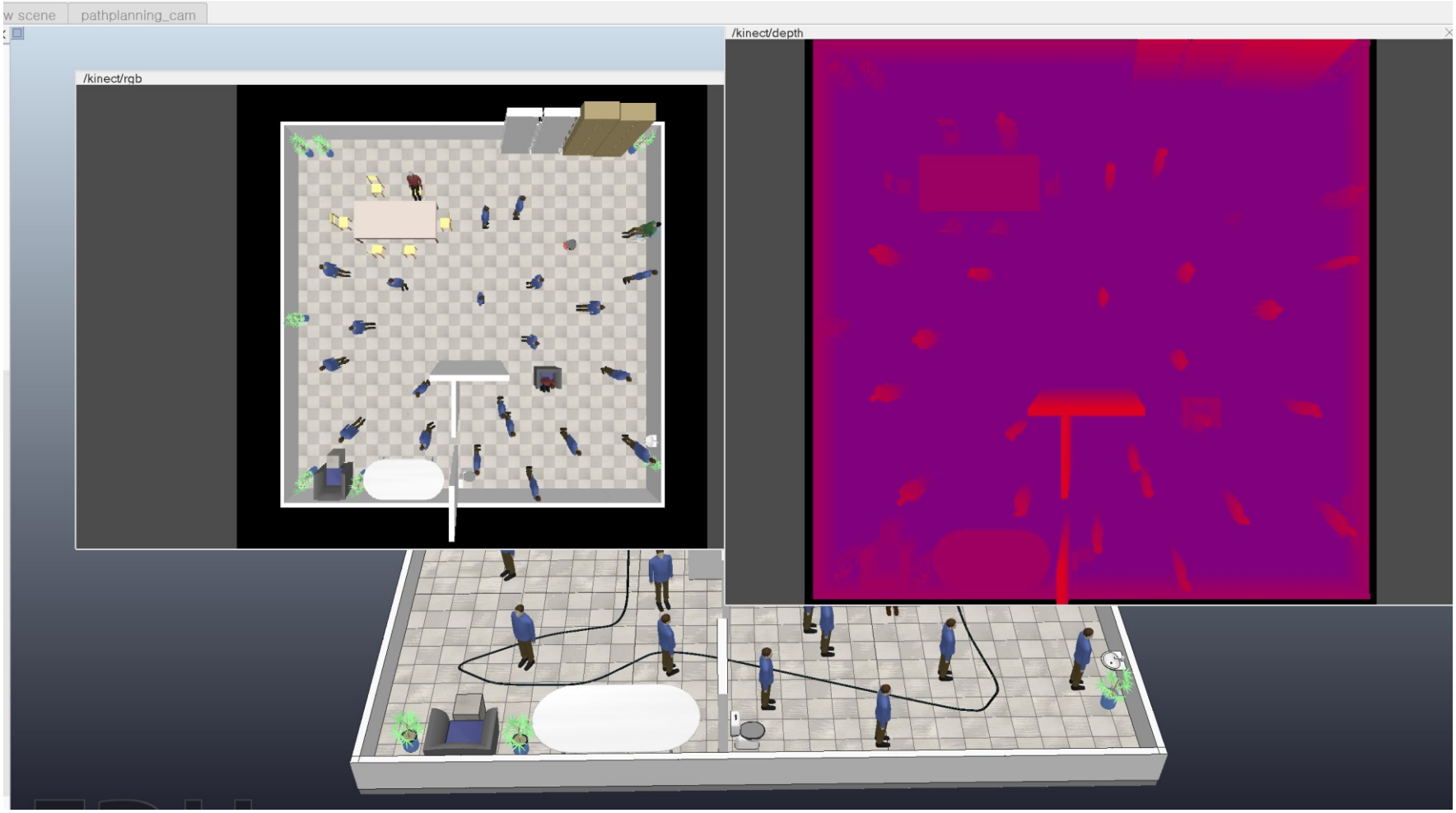
1. Attach sensors to mobile robots that can recognize their surroundings and objects (obstacles), such as LiDAR
2. Make a path so that the mobile robot can check every part of the map with a sensor (LiDAR)
3. Send the result data sensed by the lidar to the Matlab
4. Create a 2d grid map in Matlab based on the transmitted data

Result (SLAM)



- Succeeded to move the robot alog the path
- Failed to get LiDAR data.

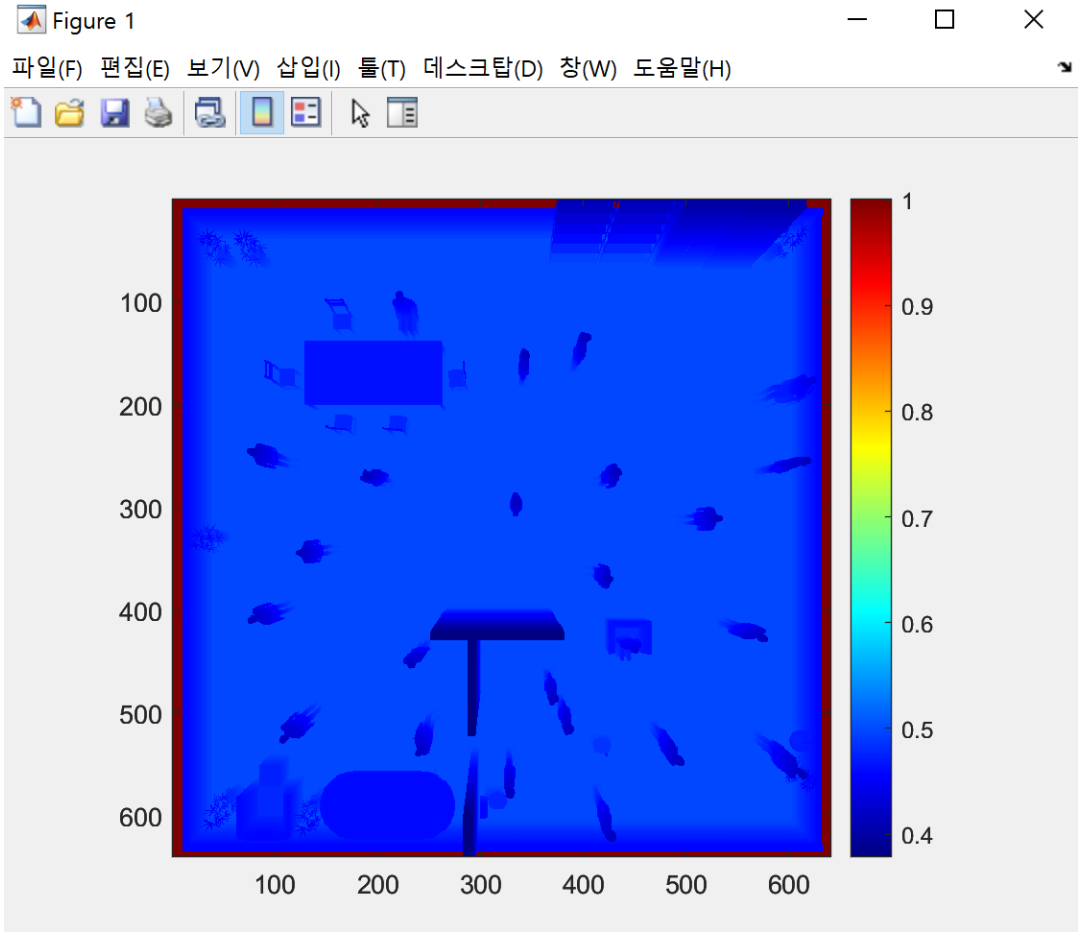
Result (Kinect camera)



Result (Kinect camera depth)

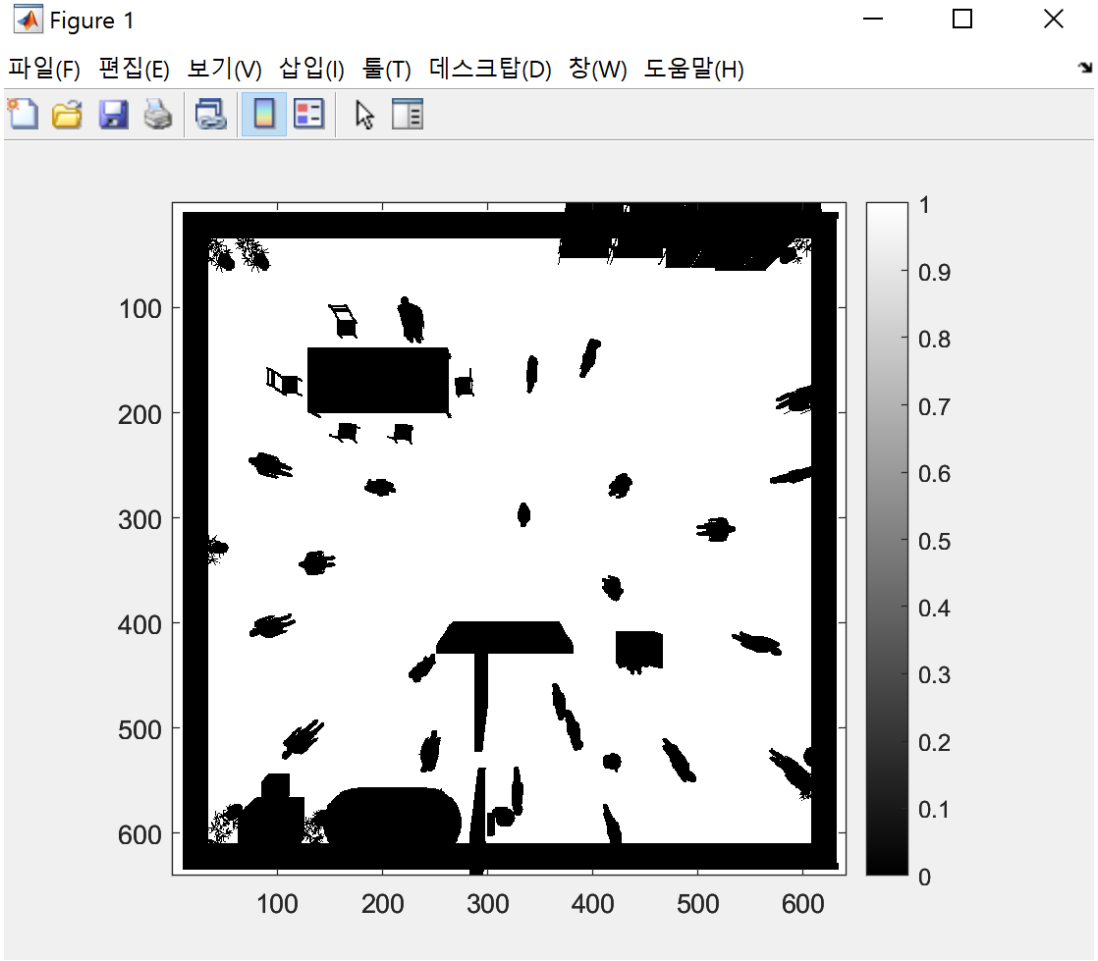
- Importing depth information into the kinect cam succeeded.

Result (Kinect camera depth)



- Can receive data from vrep to Matlab as above.

Result (Depth to Matlab gridmap)



- Made a binary grid map with 0 and 1 (range change, use threshold)

RRT*

- I tried to use path planning algorithm, RRT*

- RRT : Rapidly-exploring random trees

1. Common option that both creates a graph and finds a path. The path will not necessarily be optimal.
2. One of the sampling-based route planning methods.
3. Find a path by exploring point-wise spaces by randomly creating multiple sample points without dividing feature spaces into grids

RRT*

- RRT* (advanced version of RRT)

1. Optimized and modified algorithm that aims to achieve a shortest path, whether by distance or other metrics.
2. RRT + near neighbor search + rewiring tree operations

- Near neighbor operations

  → Finds the best parent node for the new node before its insertion in tree

- Rewiring operation

  → Rebuilds the tree within

Failure Analysis

- Tried RRT* with matlab, but it doesn't work.

- Use PlannerRRTStar → Failed to pass validation

1. Tried to change the start point and the goal point.
2. Simplified the map
3. Changed map to mat file

- If RRT* worked

1. Find the best route through matlab.
2. Check if the route could be implemented in vrep.
3. Try to simulate in vrep.