Directory update:

1. make sure map address in bash command are correct [map\_file:=/home/zianaiz/catkin\_ws/src/mie443\_contest2/maps/map\_1.yaml]

2. specific output address in contest2.cpp write\_result() [std::ofstream results\_file("/home/zianaiz/Desktop/results.txt")]

3. specific input address in contest2.cpp write\_result()

[templates\_file.open("/home/zianaiz/catkin\_ws/src/mie443\_contest2/boxes\_database/templates.xml")]

\*\* I can run gazebo so I don’t know if world file matchs with coords and map file.

\*\* once you open gazebo, verify the environment created in the gazebo with the map jpg.

\*\* let me know if theres any discrepancy between gazebo world and map.

Run 2 simulation cases:

1 case using world 1 and map\_1 and coords\_1:

- update gazebo\_coords.xml with coords\_1

- launch world:[roslaunch mie443\_contest2 turtlebot\_world.launch world:=1]

- load map: [roslaunch turtlebot\_gazebo amcl\_demo.launch map\_file:=/home/zianaiz/catkin\_ws/src/mie443\_contest2/maps/map\_1.yaml]

- run code: [rosrun mie443\_contest2 contest2]

2 case using world 2 and my\_map and my\_coords:

- update gazebo\_coords.xml with coords\_my

- launch world:[roslaunch mie443\_contest2 turtlebot\_world.launch world:=2]

- load map: [roslaunch turtlebot\_gazebo amcl\_demo.launch map\_file:=/home/zianaiz/catkin\_ws/src/mie443\_contest2/maps/my\_map.yaml]

- run code: [rosrun mie443\_contest2 contest2]

========================================================================

Compile:

- [roscore]

- [cd ~/catkin\_ws]

- [catkin\_make]

OpenCV (webcam): for image recognition test

- change kinect to webcam in imagePipeline.cpp: [#define IMAGE\_TOPIC "camera/image"]

- Enable Webcam: [rosrun mie443\_contest2 webcam\_publisher 0]

- run Han's image recognition contest2.cpp: [rosrun mie443\_contest2 contest2]

Simulation (stage): for movement, general test

- change webcam to kinect in imagePipeline.cpp: [#define IMAGE\_TOPIC "camera/rgb/image\_raw"]

- stage simulator with map and amcl: [roslaunch mie443\_stage turtlebot\_in\_stage\_contest2.launch]

- run code: [rosrun mie443\_contest2 contest2]

Simulation (Gazebo): for intergrated test

- change webcam to kinect in imagePipeline.cpp: [#define IMAGE\_TOPIC "camera/rgb/image\_raw"]

- launch world: [roslaunch mie443\_contest2 turtlebot\_world.launch world:=1]

- load map: [roslaunch turtlebot\_gazebo amcl\_demo.launch map\_file:=/home/zianaiz/catkin\_ws/src/mie443\_contest2/maps/map\_1.yaml]

- run code: [rosrun mie443\_contest2 contest2]

Execution:

- change webcam to kinect in imagePipeline.cpp: [#define IMAGE\_TOPIC "camera/rgb/image\_raw"]

- kobuki base: [roslaunch turtlebot\_bringup minimal.launch]

- load map: [roslaunch turtlebot\_navigation amcl\_demo.launch map\_file:=/home/zianaiz/catkin\_ws/src/mie443\_contest2/maps/my\_map.yaml]

- enable rviz: [roslaunch turtlebot\_rviz\_launchers view\_navigation.launch]

- run code: [rosrun mie443\_contest2 contest2]

Submission:

- Contest2\_Group\_<your group number>.zip

- mie443\_contest2

- README.md

a. Group Number.

b. The list of commands that we need to run in order to execute your group's code (as needed).

c. Any computer specific \*file path string\* that we need to modify for your code to run.

d. Where exactly the output file will be.

- Video\_file.mp4

- example\_scene\_output.txt

- Contest2 Report in PDF