Project: Deliveryman Robot
-Final

Team member: Jie Wang, Zhenqi Li

Abstract:

Our project is designing a deliveryman system. In the real life, deliveryman usually deliver food to several people in one time. We are going to help them to manage an efficient way to deliver. How to make the path through one or more people be the shortest path? We decide to use algorithm to let the robot run the shortest path to reach to each people. And robot will be given several people's locations. Our design will calculate the shortest path which is based on these locations.

Equipment:

Ultra-Sound Sensor, camera
Robot,
A map which has marks.

Deliverables and Implementation Plan:

a. Create Street System - Lead: Jie Wang Deadline: 11/24
    1. Set the street shape
    2. Transfer the world to coordinate system.
    3. Add the obstacles and the location mark of customer.

    Final: To set the street shape, I set up a area which is 4*4 and mark them from 0-area to 15-area. And randomly select two subspaces as obstacles. For transferring the world to coordinate system, I use PIL library in Python which can know what the color is on each pixel. And also, I use a 4*4 matrix to represent the street shape. And if the program scan the subspace which has the characteristic color, we will mark that subspace as a obstacle. And in result, it will return that as 1-label. Empty space return 0-label.

b. Implement robot-side controller - Lead: Zhenqi Li Deadline:12/8
    1. Use robot's Serial port to verify the world coordinate runs correctly.(finished)
    2. Use sensor to detect the obstacles and rerun the algorithm to find the shortest path.(In final version, we use the camera and pictures to detect obstacles, because the sensor of sparki is not very accurate.)

Final: I use robot serial to verify that the robot runs correctly and the information between computer and robot can transform correctly and because the sensor of the sparki is not very accurate, I use the obstacles from the street system which is detect by the camera and draw a matrix of the street system.

c. Implement laptop-side controller - Lead: Zhenqi Li Deadline:12/18
    1. Run the appropriate algorithm to find the shortest path to deliver.(finished)
    2. Use Serial port to verify the algorithm is correct.(finished)

Final: for 1, I first use the Dijkstra Algorithm to find the shortest path and cost between each delivery point. Then I store the cost in a matrix as a parameter for the TSP algorithm, I use TSP algorithm to find the shortest path from start point to every delivery point and then go back to the start point. After I get the shortest path between these point we will go back to the path matrix and get the actual shortest path between these points and have the final path. Then I use the serial port to print the shortest path for the data we give, and verify the algorithm.

d. Combine Deadline: 12/20 -Lead: Zhenqi Li Deadline: 12/18
    1. Combine all the work together to finish the project.

I combine all the work together and create a demo for the final run.

Demo:

One day, a deliveryman robot who is working in a restaurant gets an order which delivers food to several people make an order. And the robot will use our system to find the shortest path. After the robot inputs the location of people, the system will automatically design the shortest path. When robot reaches to each location, he will open the grab and stay for a while. And then go to the next location. After finishing the delivery, he will directly follow the shortest path between the last people's location and the restaurant to come back then stop.

Final Demo:

We start and UPS store and grab all the packages and wait for the computer to give him the location where to send these packages and beep after it finish the setup. In the video we have obstacles at 6 and 9. And we want to send to 5 and 11, then the computer side will figure the shortest path to go around the two points then come back. Then transform the data to the robot. In robot side

we have a parser to translate the data to the actual move. When robot reaches to each location, he will open the grab and stay for a while. After that it will come back to the start location.