ESE650 Project 5: Cost Learning and Path Planning

Due Date: **4/5/2016 at 1:20pm** on Canvas, and in class

In this project, you will build a route planner for the Penn campus. First, you will learn a cost map of the UPenn campus based on a high resolution aerial photograph. Then you will use Dijkstra's algorithm and/or A* search to build a path planner.

Download: Now available at https://upenn.box.com/2016ese650-project5

Test Sample Release: at 4/5/2016 1:30pm in class

Upload: on Canvas

(1) Code (due 4/5/2016 1:20pm, <pennkeyID>_project5.zip)

: Please do not include the image, but include the final cost map, all feature maps, and learned weights in a .mat file. If you miss any of them, then there may be some penalty.

(2) Write-up (due 4/5/2016 11:59pm, <pennkeyID>_project5.pdf)

: Write the summary of your approach, result, and discussion.

: Make sure your report includes proper visualization of feature maps, the final cost map, and your result route on test samples.

Grading: Rubrics can be found on the Canvas assignment page.

*Clearly presenting your approach in the form of report and presentation, and having good algorithm performance are equally important.

Instructions and Tips

1. [CHECK DATA] You will find a large aerial image from Google Map and some Matlab mex files for running Dijkstra's algorithm and A* search.

2. [FEATURE SELECTION] First you should select a bag of features based on the map. Use your imagination to find sufficient and relevant features for the purpose of path planning. For example, the color of pixels, edges, geometric properties of regions, etc.

3. [GENERATE TRAINING SAMPLES FOR LEARNING] Next you should hand label some paths you would consider reasonable, and use that as the training set. The imitation learning method described in the paper "Learning to Search: Functional Gradient Techniques" can be implemented to learn the cost associated with each feature that you have selected.

4. [PATH PLANNING ON LEARNT COST MAP] Now you should be able to generate a cost map based on the feature layers and the cost that has been learnt. Use this to run Dijkstra's algorithm and/or A* search to find the optimal path. **Please make sure your algorithm can take two points by clicking, calculate the shortest path between them, and present the route it plans**.

5. [MODALITIES] The above training part can be performed under different modalities, walking vs. driving (nonholonomic motion). To this end, you need to gather multiple training sets of desired paths (for example, the driving route needs to stay on the road to get from Levine to Fagin Hall, but the walking route for the same endpoints can take Woodland Walk).

6. [CLASS PRESENTATION] For the presentation in class, you are expected to bring your own laptop or use the classroom computer. The projector has a VGA port and you may need a VGA adaptor for your laptop. You will voluntarily present your algorithm and the routes which it plans. The test samples which will be released in class prior to the presentations. Please use thicker lines to indicate your path for better visualization.