Maximum Margin Path Planning

Qiong Wang
University of Pennsylvania
3401 Chestnut St., Philadelphia
qiong@seas.upenn.edu

Daniel Lee
University of Pennsylvania
3401 Chestnut St., Philadelphia
ddlee@seas.upenn.edu

Abstract

To plan path for human-like or vehicle-like robot is one of the most significant topics in modern robotics. In this paper, one Maximum Margin Planning (MMP) algorithm was developed to automatically plan the path for human or vehicle respectively given the aerial map for the environment.

1. Introduction

This project was accomplished for coursework in ESE 650: Learning in Robotics at University of Pennsylvania. The objective is to implement one Maximum Margin Planning (MMP) algorithm to plan the path for a given aerial map. The planner is then able to give prospective car or human paths based on map with respect to the features such as Lab color space, edges.

Imitation learning of sequential, goal directed behavior is always difficult. In MMP approach, the mappings from features to cost map was learned so an optimal policy in the MDP with these cost mimics the teacher or examples behavior. And the gradient descent was the main method to find the weights to combine the different features.

The environment data were provided from an aerial map of the environment. In this project, a MMP algorithm was implemented to plan the path for both human and vehicle given the data from an aerial map. The algorithm, the results are presented in details below.

2. Preprocessing of Map

At an input, the aerial map of Univeristy of Pennsylvania was preprocessed prior to the feature extraction. First, one median filter was used to remove the bad pixel assignments of the noise. Then, the MATLAB built-in functions such as `imdilate` and `imerode` were used to dilate and erode the image so as to expand the effective radius for feature extraction.

3. Feature Selection

For the feature selection, five features were mainly considered, road gray, sidewalk grey, green, other and edge. The method to do the color detection was the same as Project 1. First, color in Lab space were clustered and then find the best fit ones to form one binary indicator vector to label the color among green, road grey and sidewalk grey. The set of other was initially for the buildings such as detecting the square space in the map, but with insuffícent time here just put the pixels which are not green or the two greys to be the other. Here all features were binary, including the edges.

4. Gradient Descent

With feature matrix F after feature extraction, the total cost at each pixel was given by [1]

$$c = e^{\omega^T F}$$

Based on the cost map, for each example path, one optimal path was computed using the MEX function `dijkstra_path()`. The two paths should theoretically be the same, then we can update the weight at each timestamp based on this difference error between the two paths. According to Ratliff's method [2], the gradient can written as

$$g_t = \frac{1}{N} \sum_{i=1}^{N} q\beta_i((\omega_T F_i + l_i^T)\mu^\star - \omega^T F_i \mu_i)^{q-1} \cdot F_i \Delta\mu_i + \lambda\omega$$

, where the difference of the path is just the lattice difference for the two visitation vectors as

$$\Delta\mu = \mu_i - \mu^\star$$

Then we can update the weight as

$$\omega_{t+1} = \omega_t - \alpha_t g_t$$

Here the parameter α and λ are the learning rate and the regularization factor. Note that the learning rate decrease gradually according to the iteration number. And β is the scale number considering the length of the path, since longer path should have larger gradient. Besides, q is the norm number, here mainly using as $q = 2$. This method can converge after a set of iteration. The speed for running the gradient descent was quite slow for the trainning.

5. Results

In this project, the weights and cost map were trained with the MMP method. Here Table 1 shows the results of the final trained weights. The small value will lead to small cost map and will more likely to plan the path.

Table 1. Results of the trained weights

Features	**Pedestrian**	**Vehicle**
Road Grey	0.7690	0.0005
Sidewalk Grey	0.0030	0.8790
Green	0.0010	0.9260
Other	1.4590	1.4310
Edge	0.4400	0.4460

6. Conclusion

In this project, one MMP algorithm was implemented to plan the path given the aerial map of a specific location. The results varied rapidly, sometimes the vehicle robots can go on the sidewalk and even transverse the green area, but sometimes the vehicle can turns perfectly on the corner. The Figure 1 and Figure 2 shown the paths and cost map for the pedestrian and vehicle respectively. Besides, the speed was so slow when using `dijkstra_matrix` to get the cost-to-go map and path.

7. Acknowledgement

Thanks to Professor Lee, Alex and Zhuo. Thank you so much to prepare this great project.

References

[1] D. Lee. Lecture notes for ese 650: Learning in robotics, 2014.

[2] J. A. B. Nathan D. Ratliff and M. A. Zinkevich. Maximum margin planning, 2006. In Proceedings of the 23rd International Conference on Machine Learning (ICML06).

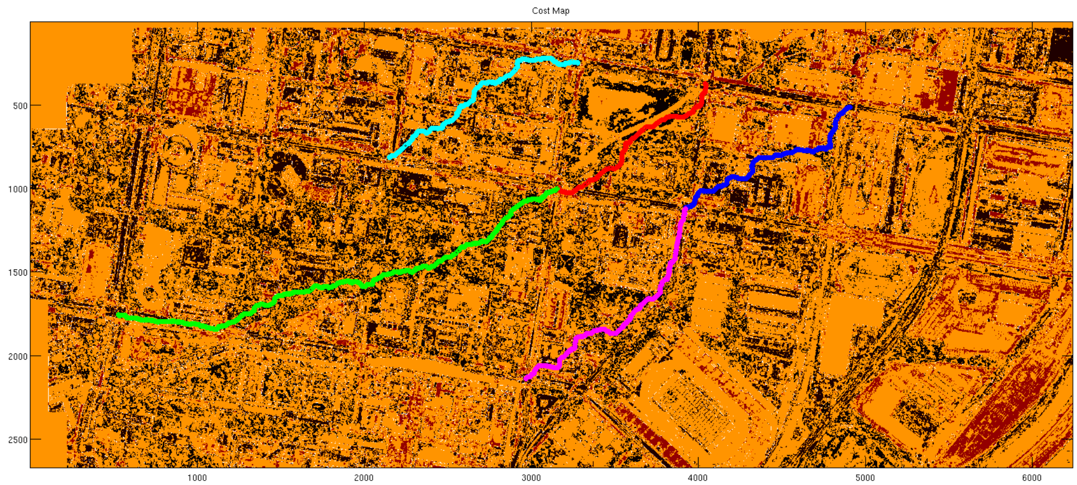

Figure 1. Cost map for pedestrian with some test paths

Figure 2. Cost map for vehicl with some test paths