Path Planning of a Mobile Robot in Outdoor Terrain

S M Haider Jafri, Rahul Kala[1]

Citation: S.M.H. Jafri, R. Kala (2015) Path Planning of a Mobile Robot in Outdoor Terrain. In Intelligent Systems Technologies and Applications. Advances in Intelligent Systems and Computing, vol 385., pp 187-195, Springer, Cham.
Final Version Available at: https://link.springer.com/chapter/10.1007/978-3-319-23258-4_17

Abstract In this study, we discuss the path planning of mobile a robot using an aerial image. Many times mobile robots are to be deputed to go to far off lands on a mission over uneven outdoor terrains. The aerial image available either as satellite images or produced by aerial drones can be used to construct a rough path for the navigation of the mobile robot. First Gaussian Process Bayesian classifier is used to classify the different classes of terrain. Next each class is associated with a cost denoting the cost of traversal of a unit distance in that particular domain. The costs account for the energy costs, risk of accidents, etc. These numerical values corresponding to each location are called as a costmap, and that array of costmap is passed to the A* algorithm which is a graph search algorithm. The A* algorithm gives the optimal path. The final result is shown in the form of the path over the aerial image with different resolutions and samples of the image.

Keywords: Robot Motion Planning, Outdoor Robotics, A* algorithm, Graph Search, Gaussian Process Bayesian classifier

1 Introduction

Robotics is a highly multidisciplinary area of research that attracts people from different domains with a common aim to make the robots operate autonomously. In the area of mobile robotics the most challenging problem is how to navigate from one place to another, using less memory and time with no collision from the obstacles. To fulfill this requirement hardware engineers are working very hard to

---

S M Haider Jafri, Rahul Kala

Indian Institute of Information Technology, Deoghat, Jhalwa, Allahabad (U.P), INDIA, Emails: sayedhaiderjafri6@gmail.com, rkala001@gmail.com

get processors with very high processing speeds and large amount of capabilities, but after all, hardware areas have lots of limitations. This necessitates the design of algorithms which work well with limited resources.

Path planning [1-2] is a big challenge in the area of mobile robots as it requires complete information about the operational area in the forms of a map. Through this map the robot can navigate from the initial position to the goal position by avoiding collision while going through the shortest path. In path planning it must be ensured that the algorithm always returns the path if it is available in the map, takes less execution time and the returned path is optimal.

Many times mobile robots are asked to travel on long missions across geographical locations in outdoor, hostile and diverse terrains. It may be suited to send robots on such missions, not requiring humans to travel for prolonged hours in uncertain terrains, further risking their life. In either case it is important to plan the route of the robot. Unlike planning for autonomous vehicles [3] where a structured road-network graph is readily available, here all one may have is a satellite image or an aerial image taken by some drone deputed for surveillance. The images are usually of low resolution and or noisy. The absence of a structured road structure makes navigation a challenging problem. The purpose of this paper is to design a route planning algorithm in such a context.

In this paper it is assumed that an aerial image of the navigation area is available. The source and the goal is assumed to be known. Terrains are marked on the aerial image using a Gaussian Process Bayesian classifier [4]. Each class is associated with a cost denoting the relative expense of navigating through a region of that class. The costmap is reduced in dimensionality by the application of bilinear interpolation. The costmap hence generated is used by A* algorithm [5] to get a path from the source to the goal.

The idea of path planning for mobile robots is not new, a lot of work has been done on using grid cells, but less on real world imagery with a different algorithms. Lee's algorithm [6], was based on breadth–first search algorithm, it was a complete algorithm and guaranteed to find a goal, if it exist but also highly inefficient in terms of memory and time. In the 'Quadtree' cell decomposition approach given by Kambhampati and Davis [7], the map was decomposed into quarter cell. If the decomposed cells are mixed with the obstacle and free cells, it will again be decomposed till a completely free space cell is obtained. After the decomposition, a graph based search algorithm was applied to find the optimal path.. Alexopoulos and Griffin [8] used visibilty graph assuming that all obstacle are polygonal in shape.

Izraelevitz and Carlotto [9] used planning on aerial image. They used a Laplacian operator to detect the road from a map by assigning some threshold value. If the aerial image is noisy then it becomes difficult to detect the road class. To overcome this problem they operated on the whole map by dividing in small regions with an assumption that the small regions are linear. Sofman et al. [10] showed the notion of guiding the ground vehicle through path planning based on an aerial image. They classified the image using neural network, created a costmap, and applied D* algorithm for path planning.

Gaussian process are also widely used in robotics. O' Callaghan et al. [11] classified the operational area of robots into occupied and not occupied. To analyze the capability of Gaussian process classification Bazi and Melgani [12] also classified using the remote sensing image.

Section 2 describes the problem statement and the solution design. Section 3 describes the technique for creating a deterministic costmap. In section 4 we show how to use the deterministic costmap by A* algorithm for path planning . Section 5 presents the simulation results and analysis. Section 6 gives the conclusion remarks.

2 Problem Definition and Solution Design

The problem is to navigate a robot from a specific start point (*S*) to a specific goal point (*G*). We are given an aerial image *I* of the navigation area. It is assumed that the different types of terrains *C* are known in advance i.e. shrubs, obstalces, roads, fields, forests, etc. Each terrain C_i is associated with a cost denoting the expense of navigating through that region, say $cost(C_i)$. Each pixel of the image (x,y) may be representing any of the known terrain classes *C*. The data are assumed to be noisy. The result is a trajectory of the robot τ: $[0,1]\rightarrow R^2$, $\tau(0)=S$, $\tau(1)=G$, which represents the approximate path to be followed by the robot.

In this paper, we use a graph based search algorithm A* [8] to find the best path in a static environment (τ). The novelty of this work that we apply the A* algorithm in the real world by using aerial images of the operational area. To apply the A* we have to pass a costmap, *map*: $R^2\rightarrow R^+$ to the A*. The costmap denotes the relative cost (denoted $cost(x,y)$) of navigating a unit distance in the area (x,y). Since A* algorithm is a single objective algorithm, all factors contributing to the overall path cost need to be integrated to a single cost value per cell. The factors may include robot safety, fuel economy, navigation speeds, etc. The A* algorithm uses the costs indicated by the costmap as step costs which contribute to the overall path cost of the different competing paths.

To construct costmap first we have to divide the operational area available as an aerial image (*I*) into grid cells $I(x,y)$ and then classify each grid cell into either of the different terrain classes (*C*). Each terrain (C_i) is associated with a deterministic cost value, $cost(C_i)$. Here we have used the Gaussian Process Bayesian classifier [4] to classify the operational area with different number of samples and compared the final result in each sample.

A* algorithm is highly sensitive to resolution. A large resolution results in better results, however the computational expense increases exponentially. Hence we apply bilinear interpolation to reduce the dimensions of the original image and increase execution speed of the software. The overall process is shown in Figure 1.

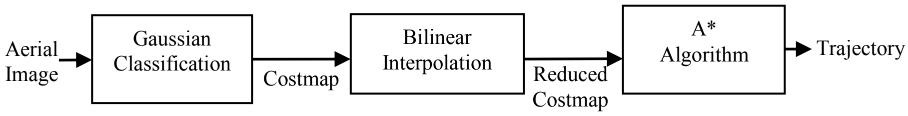

Fig. 1. General Working Methodology

3 Deterministic Costmap using Gaussian processes

To create a costmap of aerial image we consider each pixel of the image $I(x,y)$ as one of the grid cells of the operational area. This operational area is classified in either of the different terrain classes C. We assign a scalar cost to each terrain class ($cost(C_i)$) and ultimately the grid ($cost(x,y)$). This cost states the traversing cost of a robot in a particular terrain. As an example if a cell belongs to obstacle class, which the robot cannot traverse, the cost associated will be infinite. Gaussian Processes Bayesian classifier is used to map every pixel to a terrain class and ultimately the cost.

Bayesian classifier is a statistical approach to classify the different classes using the Bayes theorem. To classify the terrain we need to select some samples from each terrain and on the basis this feature we classify into pre-defined classes. This serves as the learning data for the Bayes classifier, based on which the classifier model is made and used for testing of unknown data. Let RGB denote the intensity value of the terrain classes and variables C_i denote the different terrain classes.

Where the symbols have their usual meanings. The model for our specific problem is given by equation (2).

$$P\left(C_i \mid RGB\right)=\frac{P\left(RGB \mid C_i\right)\times P\left(C_i\right)}{\sum_i P\left(RGB \mid C_i\right)\times P\left(C_i\right)} \tag{2}$$

Here $P(C_i|RBG)$ is the posterior probability of occurrence of class C_i given the intensity level of the pixel as RGB. $P(RGB|C_i)$ is the likelihood of getting RGB intensity level for the class C_i. $P(C_i)$ is the prior probability of getting the class C_i. $\sum P(RGB|C_i)* P(C_i)$ is the total probability of evidence RGB.

Any given input associated with intensity value of RGB is labeled by the class having the largest probability of occurrence. From equation (2), we calculate the posterior probability of pixel with respect to each class and find maximum between them.

Intuitively, Gaussian process [4] extracts the spatial structure from operational area to predict the map. The Gaussian process requires prior knowledge of operational area to calculate the covariance function. The covariance and mean of sample data are required to calculate the likelihood (probability distribution) of the grid cell. The likelihood probability is given by equation (4).

$$P(RGB \mid C_i) = \frac{\exp\left(-\frac{\|RGB - \mu_i\|^2}{2\sigma_i^2}\right)}{\sqrt{2\pi\sigma_i^2}} \tag{4}$$

Here RGB is the input vector, μ_i is the mean RGB intensity value of the class C_i, σ_i is the covariance of sample points of class C_i and $\|.\|$ is the Euclidian norm.

Correspondingly, the value of prior is given by equation (6).

$$P(C_i) = \frac{|samples(C_i)|}{\sum_i |samples(C_i)|} \tag{6}$$

Here $|samples(C_i)|$ is the total number of samples corresponding to class C_i. Using these equations, we calculate the probabilities of each grid cell, compare these probabilities, find the class with the highest probability, label the grid and assign the cost to produce the costmap used by the A* algorithm.

3 Graph Search

The A* algorithm [5] is a graph search algorithm to find an optimal path between a pre-specified source and goal. It is a resolution optimal and a resolution complete algorithm, meaning that both optimality and completeness can only be guaranteed for the current resolution of operation. The higher resolutions result in better paths, however with an alarmingly high computational costs. The algorithm maintains a frontier of nodes which are to be expanded, and a closet set of nodes which have been expanded. Initially the frontier consists of only the source. At any iteration the node with the lowest cost metric is taken from the frontier, expanded and added to the closed set of nodes. All neighboring nodes generated in expansion are added to the frontier if they are not present in closed set of nodes and not already present in the frontier with a better cost value.

The costmap generated from section 3 is first compressed to reduce its resolution using a bilinear interpolation technique. Each grid in the smaller resolution costmap is a state of the A* algorihtm. Each state is assumed to be connected to the 8 neighbouring states by an edge. This makes the complete search graph.

The A* algorithm is associated with three costs. The historic cost $g(n)$ of a node n is the cost incurred in reaching the state from the source. The heuristic cost $h(n)$ is an estimated cost to reach the goal from the node n. This heuristic value is not the actual value of the cost to the goal which can only be computed by computationally expensive searches, it is rather the estimated value. The correctness of the estimate are not guaranteed. The A* algorithm is optimal only when the heuristic is admissible or the heuristic function always returns an estimate lesser than the actual cost to goal. The total cost $f(n)$ is the sum of historic $g(n)$ and heuristic $h(n)$ costs and estimates the cost incurred to reach the goal from the source via node n. The costs for a node n generated from a parent n' are given by equations (7-9).

$$g(n) = g(n') + \|n - n'\|.\mathrm{cost}(n) \tag{7}$$

$$h(n) = \|n - G\|.\min_i(\mathrm{cost}(i)) \tag{8}$$

$$f(n) = g(n) + h(n) \tag{9}$$

Here ||.|| is the Euclidian norm. *cost*(n) is obtained from the cost and denotes the cost of traversing a unit step in the region.

As we consider our operational area as the an array of grid cells, so we can say that each node of the above graph a grid cell. The path computed by the A* algorihtm is in the form of the low resolution costmap representation. The path is scaled up to the original costmap. The path is printed over the aerial image.

4 Results and Analysis

In order to test the algorithm, we took a large number of images from [13-15]. To save space, the results to only one such scenario is discussed. The original image is shown in Figure 2(a). The figure shows an aerial image of an airport area. As it can be seen the scenario consists of obstacles which cannot be traversed by the robot whose cost is taken as infinity; well-built roads which are the easiest and safest to travel around associated with cost 1; and grasslands which can be traversed but with some difficulty and have a cost of 10.

4.1 Results of Classification

We used Gaussian Process Bayesian classifier to classify the terrain in different classes. We classified the input aerial image in three classes, i.e. road, grass and obstacle. We randomly sample out 30 samples per class to train the classifier. The variance and mean of each class were noted and used to calculate the likelihood and the prior probability, which was used to classify all points in the image by computing the posteriors. The classified image is shown in Figure 2(b). It can be clearly seen that most of the pixels of the image were correctly classified by the algorithm.

From Figure 2 it can be seen that even though the classification was largely corect, there were some errors with one class being confused with the others due to the noise in the image. These are uncertainties in each class, but if we increase the number of samples from each class we can avoid this uncertainty by compromising with its execution time. The results for higher number of samples are shown in Figure 2(c-d). We can see that the uncertainty decreases with an increase in the number of samples. It can be seen that the uncertainty decreased, as in grass class there are very few marks of black and white and the same condition for road class and obstacle class.

(a) **(b)** **(c)** **(d)**

Fig. 2. Classification Results (a) Input Aerial image, (b) Classified image in three classes, with 30 samples per class black represent obstacles, gray represent roads and white represent grass, (c) 50 samples per class (d) 100 samples per scale.

After classification of the operational area, we make a deterministic costmap denoting the cost of each classified class in the original image. The costmap is of the same dimensionality as the operational area. Each pixel in the image is mapped to a class and correspondingly to a cost which is pre-assumed for every class. The cost map is used by the A* algorithm.

4.2 Results of A Algorithm*

The costmap is used by the A* algorithm for the generation of the path. First the resolution of the costmap needs to be reduced in order to get results in small computational times. The original image is of resolution 512×512, which is converted into a lower resolution costmap using bilinear interpolation. The lower resolution costmap is then used for planning the path of the robot using A* algorithm.

Experiments are performed by using different resolution settings. In general a higher resolution is associated with a larger computational expense and better paths. Correspondingly the factor of sample size as studied above also plays a similar role, larger number of samples resulting in better classification and better paths, at the same time resulting in higher computational times. To best study the parameters 3 resolutions were considered which are 25×25, 50×50 and 100×100. For each of these, 1 sampling sizes per class were considered which are 30, 50 and 100. The paths are plotted for each combination of parameters and are shown in

Figure 5. Figures 5(a-c) show the results for a resolution of 25×25 with 30, 50 and 100 samples respectively.

From Figure 5 it can be seen that low resolutions result in coarser cells of the costmap, visually appearing as unit grids with larger cell sizes. This forces the robot to take large unit steps and the path of the robot cannot be finely tuned by the A* algorithm, hence implying sub-optimality. Similarly larger number of samples result in slightly better classification and hence the paths change a little.

(a) (b) (c)

Fig. 5. Experimental results (a) Resolution: 25×25 with 30 samples, (b) Resolution: 50×50 with 50 samples, (c) Resolution: 100×100 with 100 samples.

Based on Figure 5 it can also be seen that the lower resolutions do not sight the narrow road region connecting the airbase area to the runway area, which should be easier for the robot to navigate also resulting in a smaller path cost. This is the narrow corridor problem in the case of the A* algorithm, wherein the A* algorihtm cannot sight the narrow corridor because the corridor width is larger than the size of the unit grids. But if we increase the resolution of our costmap, this narrow corridor problem can be avoidable as shown in Figure 5(a-c).

The larger resolution, however compromises with the speed of execution. This execution speed is also an important factor, therefore we cannot make the resolution very high. The resolution settings require a tradeoff between the opposing factors of computational time and path optimality. Figure 6 shows the execution time with respect to an increase in resolution.

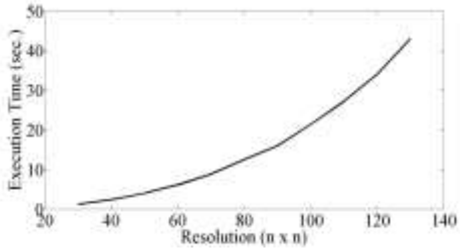

Fig. 6 Plot of execution time of A* algorithm v/s resolution (*n*×*n*).

5 Conclusions

In this paper, we presented an approach of constructing a deterministic cost map using an aerial image and path planning using that costmap. The Gaussian process classifier was used to classify the aerial image into different classes. On the basis of classification we produced a costmap for each of the terrain classes. To make a better costmap, it is required that classification be good. It was shown that the costmap improves on increasing the number of samples drawn. This costmap was used by a graph search algorithm (A*) as the unit cost of each node and the graph search algorithm created the best suitable path for mobile robots.

In the future, we will generalize the graph search to accommodate for cost uncertainities, making the search on a probabilistic cost map. We will create a probabalistic cost map stating the probability density function of cost of each node, on which the A* algorithm will operate. Coventional A* algorithm cannot handle probabilities, which is a challenge to address. Further we need to create a dataset of different terrain types to eliminate the need of a human to tag initial few terrain types used for classification. Better modelling of the terrain costs needs to be done. The approach needs to be extended to physical testing on robots.

References

[1] L. E. Kavraki, P. Svestka, J. C. Latombe, & M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 4, pp 566-580, 1996.

[2] R. Tiwari, A. Shukla, R. Kala, *Intelligent Planning for Mobile Robotics:Algorithmic Approaches*, IGI Global Publishers,Hershey, PA, 2013

[3] R. Kala, K. Warwick, "Planning Autonomous Vehicles in the Absence of Speed Lanes using an Elastic Strip", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 14, No. 4, pp. 1743-1752, 2013

[4] C. E. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

[5] N. J. Nilsson, *Principles of Artificial Intelligence*, Tioga Publishing Company, Palo Alto, California, 1980.

[6] C. Y. Lee, "An Algorithm for Path Connections and Its Applications", *IRE Transactions on Electronic Computers,* vol. EC-10, no. 3, pp. 346-365, 1961.

[7] S. Kambhampati, L. Davis "Multiresolution path planning for mobile robots," *IEEE Journal of Robotics and Automation,* Vol. 2, No.3, pp. 135-145, 1986.

[8] C. Alexopoulos, P. M. Griffin.,"Path Planning for a Mobile Robot", *IEEE Transactions On Systems, Man, And Cybernetics*, Vol. 22, No. 2, pp. 318-322, 1992.

[9] D. Izraelevitz, M. Carlotto, "Extracting Road Networks From Low Resolution Aerial Imagery", In *Proceedings of the SPIE Visual Communication and Image Processing Conference,* Cambridge, MA, 1988.

[10] B. Sofman, J. A. D. Bagnell, A. T. Stentz, N. Vandapel, "Terrainclassification from aerial data to support ground vehicle navigation," Technical Report No. CMU RI-TR-05-39, *Robotics Institute, Carnegie Mellon University*, Pittsburgh, PA, 2006.

[11] S. O'Callaghan, F. T. Ramos and H. Durrant-Whyte, "Contextual Occupancy Maps using Gaussian Processes," In *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, pp. 1054-1060, 2009.

[12] Y. Bazi, F. Melgani, "Gaussian Process Approach to Remote Sensing Image Classification," *IEEE Transactions On Geoscience And Remote Sensing*, Vol. 48, No. 1, pp. 186 - 197, 2010.

[13] D. Dai, W. Yang, "Satellite Image Classification via Two-layer Sparse Coding with Biased Image Representation," *IEEE Geoscience and Remote Sensing Letters*, Vol. 8, No. 1, pp. 173–176, 2011.

[14] G. S. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun, H. Maitre, "Structrual High-Resolution Satellite Image Indexing", In *Proceedings of 2010 ISPRS, TC VII Symposium, Part A: 100 Years ISPRS - Advancing Remote Sensing Science*, 2010, Vienna, Austria

[15] W. Yang, " High-resolution Satellite Scene Dataset", Signal Processing and Modern Communication Lab, Wuhan University, Accessed: 18th September, 2014, Availale at: http://dsp.whu.edu.cn/cn/staff/yw/HRSscene.html