



Implementation of Image Processing and Reinforcement Learning in Path Planning of Mobile Robots

Nirmalya Roy¹, Rishav Chattopadhyay², Aditya Mukherjee³, Aritra Bhuiya⁴
B. Tech Student^{1, 2, 3, 4}

Department of School of Electronics Engineering^{1, 3}, Department of School of Computer Science^{2, 4}
KIIT University, Bhubaneswar, Odisha, India

Abstract:

Development and path planning in mobile robots is an exigent field of robotics. The objective of this paper is to show the use of Q-learning for navigation in indoor environments. Planning the shortest path from current state to the goal state using images captured from ceiling of the indoor environment. Captured image is tried to process through different image processing and machine learning techniques. Obstacles in the path of the robot is also tried to process by calculating Adaptive Gaussian Thresholding of the image captured. Position of the robot is tried to be tracked using template matching of CV. Q-learning techniques are applied to plan path of the mobile robot from current start to the goal state.

Keywords: Open CV, Image Processing, Mobile Robots

I. INTRODUCTION

In mobile robots, autonomous path planning is challenging as obstacle are presents. Mobile robots such as autonomous vacuum cleaners, autonomous drones face a big challenge in their path planning and to some extent they are inevitable. If the path are feed into the mobile environments, unknown obstacles makes path more tangled and cumbersome. In this type of mobile robots the path planned should be dynamic to avoid unavoidable obstacles which may block its path. The path planned by the system should be accurate and should able to detect obstacles. Initially range finder was used such as ultrasonic sensors, IR sensors and laser range finders to detect obstacles which tend to be a complex solution in terms of complexity of code used and accuracy.

Vision based technology is proposed in this paper that captures the whole map of the indoor environment and detects the objects in a single image capture. Earlier robots were programmed with consideration of each and every aspect which may act as a path or considered an obstacle, but this traditional method tends to make programs more complicated and less efficient. To overcome this problem some Artificial Intelligence should be given to the robot to overcome the problems and make the robot conscious about its environment. The image captured from the ceiling of the indoor environment is processed with Image Processing and CV technologies such as Low pass filtering, smoothening, contour mapping and condensation algorithm for object tracking. Reinforcement learning is used and Q-Learning algorithm is used for generating path of the robot. Q-Learning is a model free reinforcement learning technique. The algorithm uses best action policy which gives higher rewards to the action which is better to reach its goal.

II. IMAGE PROCESSING

The Image captured should be processed in order to remove features that do not require for generating path of the mobile robot.

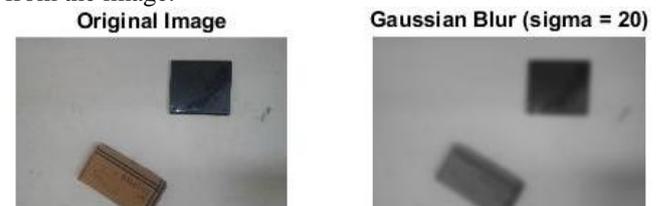
To detect the obstacles in the image the steps needed are:

1. The image is converted into gray scale image – Converting image into grey scale will reduce the information quality of the image and will help in further processing. Converting image into gray scale will also convert the image size to near about one third of the image.



2. Obstacle edges are detected with image thresholding – The main purpose of thresholding the image is to reduce data as only the structure of the obstacles is needed. The algorithm used here is Roberts Edge Detection which comprises the following steps:

2.1. Applying Gaussian Filter on the image with a sigma value of 20 to smooth the image that helps in removing the noises from the image.



2.2. Finding the intensity gradient of the image.

2.3. Applying non-maximum suppression to get rid of spurious response to edge detection.

2.4. Applying double threshold to detect potential edges.



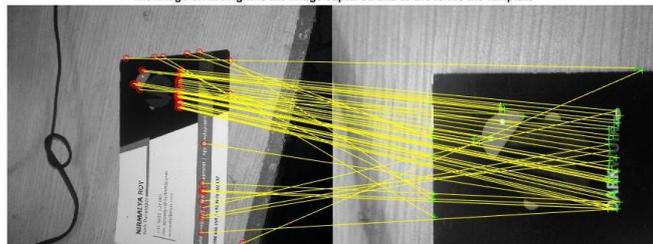
2.5. Tracking edges by hysteresis, i.e. finding the edges that are connected by suppressing all other edges that are weak or not connected to strong edges to detect the mobile robot template matching method of CV is used. It requires bulk of images of the mobile robot that matches with the image captured. This helps in finding the position of the mobile robot in the map.

III. TEMPLATE MATCHING IN IMAGE

Template matching is used in the system to detect the position of mobile robot in the image captured. Here histogram based template matching is used in the processing. In basic template matching, the template image is slide over the image and at each position degree of similarity is calculated between the template image and the main image which is centered in the position and has the same size as the template. Then the position with maximum similarity is considered as the region of interest. Histogram based template matching is useful since it is partly robust to rotation, deformation and partial occlusion. Let the template be T , $q = \{q_u\}_{u=1, \dots, m}$ denote the normalized m -bin histogram of T . Let the normalized m -bin histogram obtained at a candidate sub image centered at y in the current frame be p . The Bhattacharyya coefficient is selected as the similarity of coefficient is selected as the similarity metric is: $\rho(y) = \rho(p(y), q) = \text{Summation of root over of } p_u * q_u \text{ form } u=1 \text{ to } m$. A similarity map can be built by comparing the histogram q and $p(y)$, at each point of the current frame. The position with maximum $\rho(y)$ is considered as the position of the template of the main image.



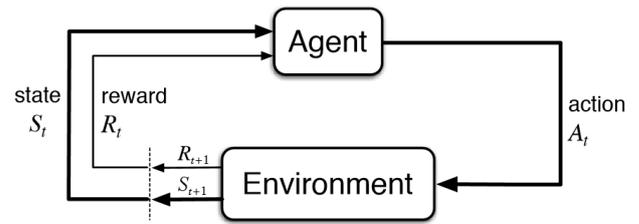
The image on the right is the image captured and to the left is the template



IV. PATH PLANNING OF MOBILE ROBOT

Q – Learning algorithm is used here. Q – Learning is a model free approach of reinforcement learning field. Reinforcement learning is a sub field of machine learning which focuses on how an agent ought to take action depending on the rewards to reach its goal. Reinforcement learning have five constituents namely agent, action, reward, state and environment. Here the

mobile robot is the agent which is a learner / decision maker. The actions are performed by the agent who is reflected as rewards. Based on the rewards the agent learns and takes further decision. State is the area where the agent is located and environment here is the image of the map captured.



4.1. The Q – Learning algorithm:

The steps in Q learning include:

4.1.1. Set the gamma parameter, and environment rewards in matrix R.

4.1.2. Initialize matrix Q to zero.

4.1.3. For each episode:

Select a random initial state.

Do While the goal state hasn't been reached.

- Select one among all possible actions for the current state.
- Using this possible action, consider going to the next state.
- Get maximum Q value for this next state based on all possible actions.
- Compute: $Q(\text{state}, \text{action}) = R(\text{state}, \text{action}) + \text{Gamma} * \text{Max}[Q(\text{next state}, \text{all actions})]$
- Set the next state as the current state.

End Do

End For

V. HARDWARE AND IMPLEMENTATION

The camera image is captured and processed with matlab. Matlab detects the position of the robot and obstacles present in the map. The goal position of the mobile robot is feeded into the matlab system. The matlab software generates the path planned by Q - Learning based on the processed image. The path planned is transferred to the mobile robot .

5.1. PC and Mobile Robot Communication

The steps for communicating PC and Mobile Robot are:

- 1.The IP address of the computer is assigned to a static IP.
- 2.Using XAMPP for creating a local server , a local server is created.
- 3.The websites on xampp/htdocs were available from the static IP address.
- 4.The path generated after image processing and path planning was displayed in the webpage which is visible from clients connected with the router.
- 5.The mobile robot connected to the router views the webpage on the ip address for getting commands time to time.

5.2. Displaying data on website

The path generated from matlab is send to the website with encoding in an url from matlab. Website uses PHP to receive the data using 'GET' method.

The data are the instructions for the robot for its movement , the are :

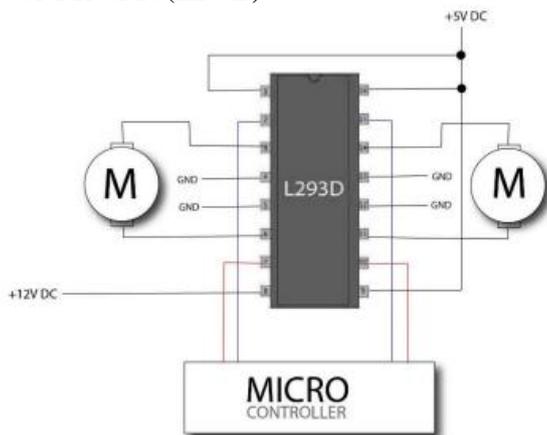
1. forward(distance)

2. left
3. right
4. back(distance)
5. rotate_left (angle)
6. rotate_right(angle)
7. stop

5.3. Hardware of the mobile robot

The mobile robot uses node mcu microcontroller with ESP8266 firmware for fetching data and generating signals for movement of robot. Node MCU was programmed in Arduino IDE. Generally DC motors runs on 12V while Node MCU microcontroller generates signal of 5 V, So we have used L293D motor driver IC for converting 5V signal into 12V signal. 2 DC motors are for movement of the mobile robots.

Motor Driver IC(L293D)



L293D is a dual H-bridge motor driver IC. The main function of this has been to amplify the current by taking a low-current control signal as input and gives back a high-current signal as output. The higher current signal is generally used to drive the motor. The motor driver ic has 4 pins as inputs(input 1,input 2,input 3,input 4) and in the output(output 1, output 2, output 3, output 4) it amplifies the current in the respective number output pin.

VI. CONCLUSION

In this paper, a mobile robot's tracking and path planning system has been shown using image processing, CV and Q – Learning. The image processing and the Q-Learning system is implemented in matlab, the mobile robots are communicated using Node MCU microcontroller. Using XAMPP a local server is created and the planned path is displayed in the webpage. Node MCU views the webpage and runs the motors accordingly.

VII. REFERENCES

- [1].D. Pandey and P. Pandey, "Approximate Q-learning: An introduction," in Proc. 2nd Int. Conf. Mach. Learn. Comput., 2010, pp. 317–320
- [2].M. Moll and L. E. Kavraki, "Path planning for minimal energy curves of constant length," in Proc. IEEE Int. Conf. Robot. Autom. Apr. 2004, pp. 2826–2831
- [3]. T. Dean, K. Basye, and J. Shewchuk, "Reinforcement learning for planning and control," in Machine Learning

Methods for Planning and Scheduling, S. Minton, Ed. San Mateo, CA: Morgan Kaufmann, 1993.

[4]. R. E. Bellman, Dynamic Programming. Princeton, NJ: Princeton Univ. Press, 1957. 5.

[5]. Watkins and P. Dayan, "Q-learning," Mach. Learn., vol. 8, no. 3, pp. 279–292, May 1992.

[6].R. J. Beattie, "Edge detection for semantically based early visual processing," Ph.D. dissertation, Univ. Edinburgh, 1984

[7].R. Courant and D. Hilbert, Methods of Mathematical Physics, vol. 1. New York: Wiley-Interscience, 1953.

[8].Youcef Mezouar and Francois Chaumette, "PathPlanning For Robust Image-Based Control", IEEE Transactions on Robotics and Automation Aug 2002

[9] W.F. Carriker, P.K.Khosla and B.H.Krogh, "Path Planning for Mobile manipulator for multiple task execution", IEEE Transactions on Robotics and Automation June 1991

[10]. O.Hachour, "Path planning of Autonomous Mobile robot", 15th WSEAS international conference on Systems, 2011