# A review on Motion Planning techniques

Keith .M. Machina
*Department of Computer Science*
*Kent State University.*
Kent, Ohio, USA
kmachina@kent.edu

Justin Dannemiller
*Department of Computer Science*
*Kent State University*
Kent, Ohio, USA
jdannem6@kent.edu

***Abstract*—Stretching out your arm, whether right or left, to grab your TV remote might seem like a simple task which sometimes, we as humans do subconsciously, without even paying attention but to a robot, the case is different. Path planning and path finding more often than not, is a task that is not simple in the mind of a robot. The robot has to calculate an efficient way to calculate the path to the desired object and at the same time move to the desired state in an efficient manner something which makes the topic motion planning a subject of interest. This paper will strive to go over the various approaches that are currently being used in motion planning in the field of robotics.**

***Index Terms*—Qlearning, reinforcement learning.**

## I. Introduction

Robotics has recently grown to be among the hottest topics in the field of computer science with its application ranging from industry to industry. Some of the incredible robotics applications that have transformed the landscape include: surgery operation in medicine, quality assurance in manufacturing industry and cultivator operator in agriculture. Looking at all these applications, the field of robotics aims to improve processes by reducing error and human intervention, hence the term automation. Making processes autonomous is quite vital as it introduces efficiency by gets rid of redundancy and reducing cost. It's important to note that robotics is a multidisciplinary and borrows concepts from fields such as math and physics to bring these ideas to life, concepts such as kinematics that have been discussed in this paper review. With automation being a driving factor, this paper will review some of the state of the art motion planning techniques that are being applied and clearly demonstrate how they are used in practice.

## II. Computer Vision-Based Path Planning for Robot Arms in Three-Dimensional Workspaces Using Q-Learning and Neural Networks

In this paper, the goal is to determine how a robot arm will move to grab an object, commonly referred to as path planning, using two techniques: neural networks and Qlearning. This section will go over the two techniques to discuss how the goal was achieved using this two techniques. The very first step that was used in this path planning technique was localization which is basically determining the position of the object in the given space, basically object detection. During this process, details on the precise location of the target object and obstacles around it are extracted. To successfully do this, the 'You Only Look Once' (YOLO) version 4 model was used to extract the x,y, and z planar coordinates of the target object as well as other objects that are within the region of interest. A neural network with 4 hidden layers each layer containing the following: 16,32,64,16 neurons was the used to extract spatial coordinates of the objects detected within the region of interest with the planar coordinates being input for the neural network. The outputs of this model were the x,y,and z spatial coordinates of this object. Given a 3D space of 50x50x50, the discrete cells were reduced to 8x8x8, and the space is divided into small size grids to simplify the comaprison of current state and goal state. The state space at each time was presented as follows:

$$S = (cell1, cell2, cell3, \ldots\ldots celln * m * 1) \quad (1)$$

$$A = (Up, Down, Left, Right, Forward, Backward) \quad (2)$$

It is worth noting that other object recognition models such as RCNN, Fast RCNN, mask RCNN could be in place of YOLO, it was implemented because it is often faster compare to other models.

Next step was to use a technique called Qlearning to determine the possible series of actions that the robot can take given the position of the target object. Qlearning is a reinforcement learning technique serves as guide in intelligent system to enable the system achieve maximum reward. Qlearning was used to determine a series of robot arm movement such as up, down, right, and left, specifically arm movement that maximize the reward of the intelligent system. Due to the constant movement of the robot arm, the reward was constantly changing. The reinforcement learning was set up in the following way, 50 points were gained every single time the robot arm got to the target and -100 everytime the arm got to an obstacle. Penalty for all other options was set to -1. The goal was to maximize the score every single time. Qlearning being a greedy approach, meaning that it always chooses an option that maximizes the reward, proved to deliver better results compared SARSA algorithm which stand for state action reward state action Final step in this process was a process referred to as angle finding. In angle finding, the main task was to transform the series of actions produced by Qlearning (up, down, right, and left) into a sequence of joint angles. This step would enable the robot arm to move

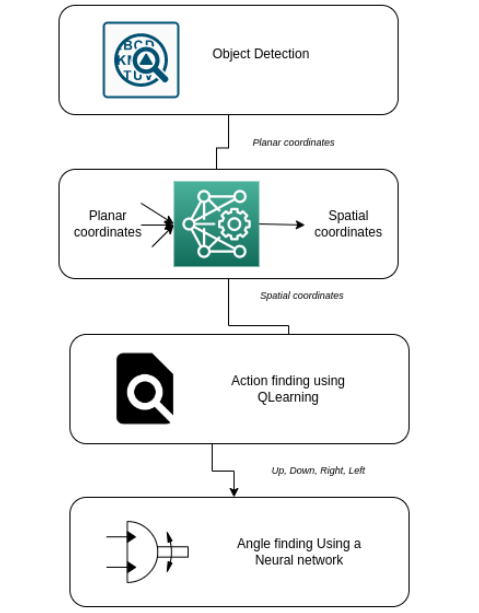


Fig. 1. An Overview of the approach

directly to the object. Inverse Kinematics is a strategy that can be applied in this step but was ruled out due to its low speed caused by the enormous mathematical calculations that have to be done. This second neural network took output of the Learning process as its input and then outputting the joint angles between each series of actions therefore the robot doesn't have to make 90 degrees turn between each series of movements.

### A. Experiment

To test the ideology stated above, a robot arm with six degrees of freedom, two cameras, a 3D workspace and the objects representing start, obstacle and target were used. The cameras were installed on the robot arm pointing in the direction of the region of interest. As the experiment began, a snapshot of the 3D work space containing the target and obstacle was fed into the YOLO model, where the model outputted coordinates of both target and obstacles that existed in the 3D work space, then converted into spatial coordinates by the first neural network. These coordinates were then consumed by Qlearning reinforcement learning algorithm as its input to determine the series of steps, then the second neural network to determine the angle joint between each two actions that can be taken by the robot arm. Upon successfully generation of joint angles, which was the last step, the robot, slowly and carefully followed the path to the target object successfully receiving an award of 50. The advantage of this approach was that its was much faster compared to using methods such as inverse kinematics and SARSA algorithm.

### B. Limitations

During the experiment, only one photo was taken and fed into the model. This might not have been enough, it would have been better if the the cameras were set to take multiple photos, feed those into the model so as to perfectly capture objects of the image from different angles hence improving the models ability to extract precise coordinates.

## III. Object Grasping of Humanoid Robot Based on YOLO

In this paper, the aim was to achieve autonomous movement of the robot arm by by taking advantage of a depth camera and leveraging on a state of the art object detection model to identify object within the region of interest. The depth camera is a specifically used in the experiment in order to extract information an distance of objects from the actuator, the robot arm. Also the paper provides a partial solution to dealing with uncertainties that tag along when grabbing an object. Such uncertainties include: shape of the object,pose of the object, and contact point with the actuator and the quality of the object. To bring this idea to fruition, the following three steps were followed: Object perception in the RGB image, Object segmentation from the depth image, and kinematics of the arm. The section below will discuss the aforementioned steps. In the first step, object perception in RGB image, the goal was to properly understand the environment, a step known as localization and using a depth camera, the images of the region of interest were taken. The state of the art 'You Only Look Once' model was invoked to carefully extract the convolution features of the environment image fed into the model in real time. The detected objects of interest would then be marked using a bounding box. It is worth noting that alternative object detection models such as R-CNN and Fast R-CNN can be also be applied in this step. The Second step, object segmentation from depth image was harnessed to specifically extract target image pixels, a process regarded to as image segmentation. It is basically separating an image from it background.To successfully separate the image from the background, pixel values of both RGB and depth that were larger than 1 were set to 0 using the following formulas fro both RBG and depth pixels.

$$RGB(p) = 0, depth(p) > 0, RGB(p), other,eq \quad (3)$$

$$depth(p) = 0, depth(p) > 0, depth(p), other,eq \quad (4)$$

Next was to the adopt RANSAC (Random Sample Consensus), a mathematical approach for estimation, was applied to determine important points of the detected objects and fit those points on a plane. The plane fitting equation is defined as follows:

$$ax + by + cz + d = 0 \quad (5)$$

From that above equation, a are position parameters while d is the distance from the actuator, the robotic arm, to the object

The last step was determining the kinematics of the robotic arm, complex process as it entail calculating by what degree each degree of freedom will move. Inverse kinematics was applied in this experiment and was applied the following equation.

$$Jt(JJt + y2I) - 1e \tag{6}$$

The following is an explanation of the inverse kinematics formula. J represents the Jacobian matrix which is a matrix comprised of all first-order partial derivatives obtained by performing differentiation on an equation. JJT The term refers to the jacobian matrix multiplied by the transpose of the Jacobian matrix. Represents the scalar value that regulates the angle by minimizing to a certain degree. I represents Identity matrix. Represents the difference between the current position and the desired position.

### *A. Experiment*

The following were the specifics of the experiment An imnov robotic arm was used. A KINECT II depth camera was used. A vision detection software was running on a controller PC. It was linked to the KINECT II. When a predefined target appeared in the region on interest, the software identifies the category and sends the location and size of the target to the robot arm for it to grasp the target.

## CONCLUSION

The field of motion planning still has a lot of untapped potential and has a long way to due to slow speed of growth of the robotics industry. However, the current motion planning techniques are able to given the output as expected but a lot more can be done. This paper reviewed some of the motion planning techniques that used advanced technologies such as deep learning and computer vision. These are: using Qlearning and neural network, and use of object detection and inverse kinematics.

## REFERENCES


[1] Bugday Bugday, M.; Karali, M. Design optimization of industrial robot arm to minimize redundant weight. Eng. Sci. Technol. Int. J. 2019, 22, 346–352. [CrossRef].

[2] Korayem, M.; Madihi, M.; Vahidifar, V. Controlling surgical robot arm using leap motion controller with Kalman filter. Measurement 2021, 178, 109372. [CrossRef].

[3] Park, H.C.; Ahn, K.H.; Min, J.K.; Song, J.-B. 5 DOF Home Robot Arm based on Counterbalance Mechanism. J. Korea Robot. Soc.2020, 15, 48–54. [CrossRef].

[4] Chen, L.; Yang, H.; Liu, P. Intelligent robot arm: Vision-based dynamic measurement system for industrial applications. In International Conference on Intelligent Robotics and Applications; Springer: Berlin/Heidelberg, Germany, 2019.

[5] Redmon, J., et al. You only look once: Unified, real-time object detection in Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.