VISION BASED INDOOR NAVIGATION SYSTEM USING EXTERNAL CAMERA MOUNTED ON ROOM

G.S.U. Ahmadh[1], I.I.S. Premasri[1], A.N.F.M. Begum[1], M.F.M. Abdul Cader[1] and W.G.C.W. Kumara[2]

[1] Department of Electrical and Telecommunication Engineering, South Eastern University of Sri Lanka, Sri Lanka

[2] Department of Computer Science & Engineering, South Eastern University of Sri Lanka, Sri Lanka

umairahmadh@seu.ac.lk

ABSTRACT: *It is vital to have a very good path planning for self-navigating mobile robots. Having considerations of the environment and objects in the space is a key in achieving a safe and effective path for robots. In indoor environments, objects and the environment are unpredictable, unlike that of in the road navigation where there are standard guidelines and road symbols. To overcome this issue, existing applications provide solution to a certain extent. Indoor positioning systems based on Infrared, Wi-Fi and other similar techniques are existing in the industry. Also, there are solutions providing such systems with the use of camera on the robot. Although such systems deliver the positioning of the robot and thus generating a path based on it, they fail to have an overall view of the entire space so that the generation of the path is more effective. To overcome this, the paper discusses here, a solution of centralized vision based navigation system for indoor robots. Our solution takes a view of the entire space beyond the immediate surroundings to the robot which was the only concern of previous solutions existing. By such view, our solution can plan a path for the robot more effectively as it has all the objects in the space into consideration.*

Keywords: Robotics, Computer Vision, Path Planning, Indoor Navigation, Vision Based Navigation

1. INTRODUCTION

Path planning and navigation of autonomous robots and vehicles have always been one of the mostly researched areas in robotics and automobile industry. Giving the autonomous robots a power to navigate to their target destination without having a collision with other objects or navigating in a wrong path is a very essential task. Even though this is a common concern in all types of autonomous vehicles, Indoor robots are more prone to errors in navigation unlike in outdoor environments, because of their configuration space is not typical and also the obstacles are mostly unpredictable. The number of indoor robots like restaurant service robot, elder care robots and room cleaning robots is increasing to take their places everywhere from household to industrial workplaces. Although most of the existing indoor robots are manually guided or preprogrammed to follow a fixed path, nowadays autonomous navigation robots are coming into the market. To achieve the autonomy in navigation few of them have cameras to see their immediate surroundings and some have range sensors to detect immediate obstacles to improve their navigation. These navigations methods that utilize in-built camera or other sensor-based are mostly ineffective as none of them give the robots an insight of what's beyond their line-of-sight. Having such ability to do the navigation, based on the real-time condition of the entire configuration space would give an advantage to the system of being more effective and faster path planning and navigation.

In order to achieve such system, this paper presents an approach that will make use of cameras mounted on top of the room to get an oversight of the entire room and by utilizing them, the robot will be able to get a map of the entire room by processing the captured image and generating an obstacle map. From the map, the system will then be able to generate a path for the robot from its current destination to the target destination. For this purpose of path planning, a path planning algorithm named "Extended rapidly exploring random tree (RRT*)" has been chosen to be used for the purpose after exploring various available algorithms (Noreen, Khan, and Habib, 2016).

Rapidly exploring random tree (RRT) (Lavalle, 1998) provides a quick solution to generate paths with having no necessity of defining grids and nodes unlike in other shortest path algorithms like Dijkstra. Among all the probabilistic planning algorithms, RRT has become the most preferred algorithm for path planning due to its speed and simplicity. RRT generates a tree between starting and target nodes by creating nodes at random points and finding the nearest node and adding it to the tree. It creates and adds matching nodes to the tree until a branch of the tree meets the goal node.

The extended version of RRT named as RRT* provides an additional feature of Rewiring the tree. It gives the algorithm a possibility of recreating the nodes dynamically. The paper by Connell and Manh La (2018) is providing an explanation of how the RRT* algorithm can be optimized for dynamic obstacles in the environment. In this paper, our approach is to use RRT* in a real environment to navigate an indoor robot by detecting dynamic obstacles in the environment using external cameras. In addition to that range sensors will be mounted on the robot to detect immediate obstacles and to request for the regeneration of the tree if any immediate obstacles found.

1.1 Literature Review

There are several methodologies proposed in the literature about mapping and path planning. Mapping should be done in order to find the robot environment, which can be acquired by either constructing occupancy grid maps or landmark maps. To create the model of the environment, the information can be collected by either a variety of onboard sensors or external sensors such as surveillance camera and the external GPS system.

Depends on the status of environment whether it is known or unknown, the mapping technique will vary. For indoor mobile robot navigation, typically the environment is known. Li et al. (2014) presented a technique in which the experimental map was created using a vision-based system. In this paper, they used captured images using external webcams. Image processing technique has been taken with these captured images which include a greyscale, a background subtraction, a morphological image processing, and a connected component labelling technique. Then the coordinates were defined in the image domain (“Mapping Camera Coordinates to a 2D Floor Plan,” n.d.).

For the unknown environment, a well-known algorithm called Simultaneous Localization and Mapping (SLAM) is needed for many applications now. It is a process of creating a map which navigates the robot in its environment while using the map it generates. Milford and Wyeth (2010) presented a new system for integrating continuous Simultaneous Localization and Mapping (SLAM) with autonomous behaviors and a system of map maintenance. They experimented their results to demonstrate the use of the entire system in live delivery over a two-week period across two different office buildings.

In this paper, they used RatSLAM, a bio-inspired Simultaneous Localization and Mapping (SLAM) system to localize, map and navigate throughout the robot’s lifetime. They investigated the static mapping problem based on how the animals forage and identify their home with the change of activity of other animals, weather and season. Main components of persistent navigation and mapping system are core RatSLAM system, the navigation system, and robot task selection. The global experience map is created by RatSLAM which is a resource to plan paths to locations that are beyond the range of the robot’s sensors. The local obstacle map is a resource that captures the relevant aspects of the environment within the immediate range of the robot’s sensors.

Path planning is another important task of finding a continuous path that will drive the robot from the start to the desired target location. There are several studies found in the literature. Khanmirza et al. (2017) proposed a comparison of seven mostly used deterministic and probabilistic path planning algorithm which are A*, Dijkstra, Visibility Graph, Probabilistic Roadmap, Lazy Probabilistic Roadmap, Rapid-exploring Random Tree, and Bidirectional Rapid-exploring Random Tree in 19 different environments with different working conditions to evaluate the best path planning algorithm. Path planning techniques can be analyzed into three groups,

1. Deterministic algorithms: suitable in the case of a robot which has few degrees of freedom and deals with the 2D environment.
2. Probabilistic algorithms: preferable in the case of high dimensional space
3. Heuristic algorithms: deals with complex path planning problems.

The deterministic algorithm takes consideration with a weighted directed graph while probabilistic use the random sample points to get the nodes to move the robot. In this paper, they only presented the study of deterministic and probabilistic algorithms since the heuristic algorithm has been researched in several studies nowadays. The performance of algorithm is characterized by several objectives such as path length, the path smoothness, the runtime, and the planning success rate. By comparing the objectives to test the performance of each algorithm, dimensionless plots are used. In the visibility graph, shortest and smoothness path were received while in the Bidirectional Rapid-exploring Random Tree (B-RRT) achieved highest success rates. In recent research (Connell & Manh La, 2018), the Extended Rapid-exploring Random Tree (RRT*) is proposed with greater optimization compared with the general RRT algorithms.

2. METHODOLOGY

Our methodology is to have the following approaches

1. Capturing and processing the images of the room with cameras mounted on the top of the room,
2. Detection of objects in the room and classifying of moveable paths and obstacles,
3. Generating the path for the robot to its destination using the RRT* algorithm and
4. Regeneration of path when a new obstacle is introduced in the previous path

Based on these approaches an algorithm will be developed to implement this in an indoor system. The dataflow is shown in Figure 1.

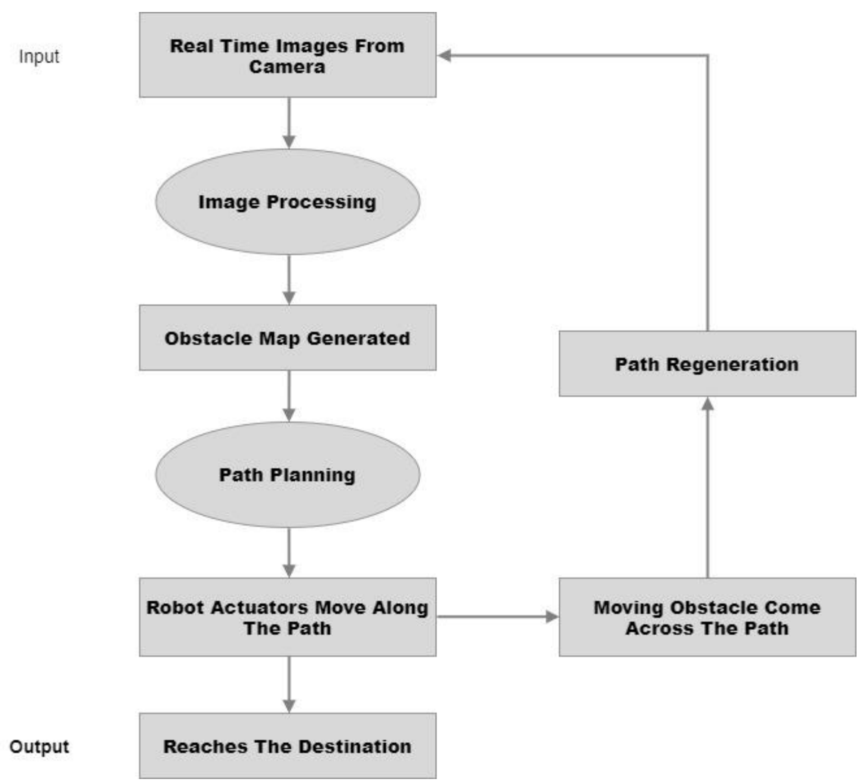

Figure 1: Data flow of the system methodology

2.1 Capturing and Processing the Images of the Room

A fisheye lens camera will be mounted on top of the room to cover a wide area. Images captured from the camera will be having distortions throughout the images. In order to get a fixed and a proper image calibration is needed. A standard method of calibration is to use a known size of the chessboard to get the camera intrinsic and extrinsic parameters and thus correcting the image distortions. Figure 2 shows the captured image with distortion.

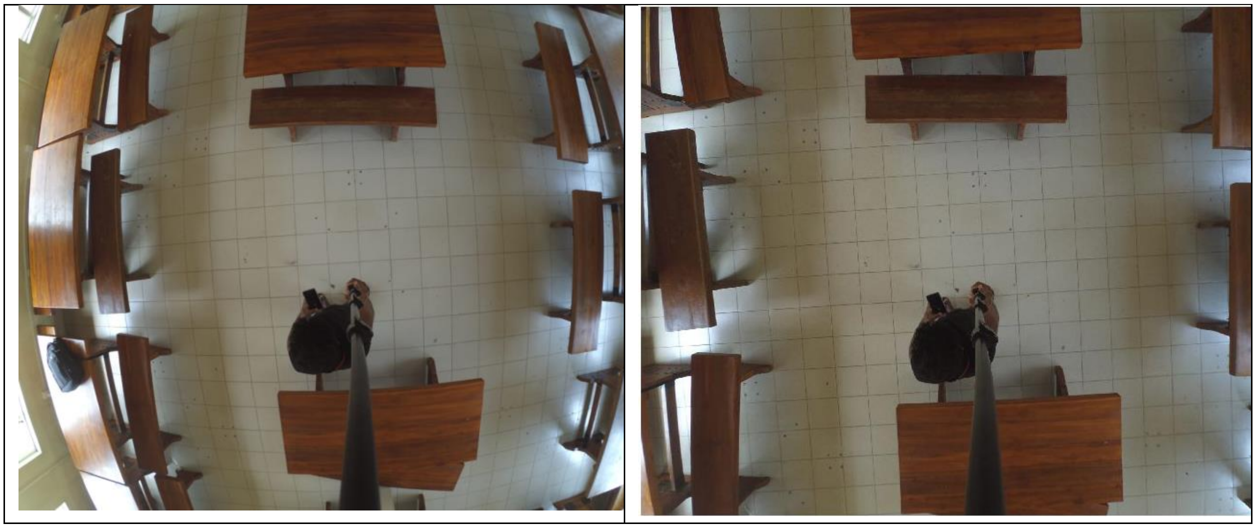

Figure 2: Captured Image of Cafeteria Which Has Distortion in the Edges (Left) and Image of the Cafeteria After Calibration (Right) (This Will Be Used for Further Processing).

From the calibration process, a matrix is generated and is used for further operations. After calibration, the above image shows the corrected and a plain image of the cafeteria. The calibration matrix will be applied to all the processed images throughout all further applications.

2.2 Detection of Objects in the Room and Classifying of Moveable Paths and Obstacles

Once the undistorted image of the room is achieved, then the objects in the room need to be identified in order to get a view of moveable path to allow the path planning algorithm to generate its paths in such a way it doesn't go through any obstacles.

There are two kinds of objects in the room.

1. Fixed objects like furniture and pillars
2. Dynamic objects like human

Firstly, the captured image on which there was no presence of any dynamic objects was run through image segmentation by clustering and then a cost map was generated and based on a threshold it was turned into a binary image marking objects as black and the moveable area as white (Figure 3).

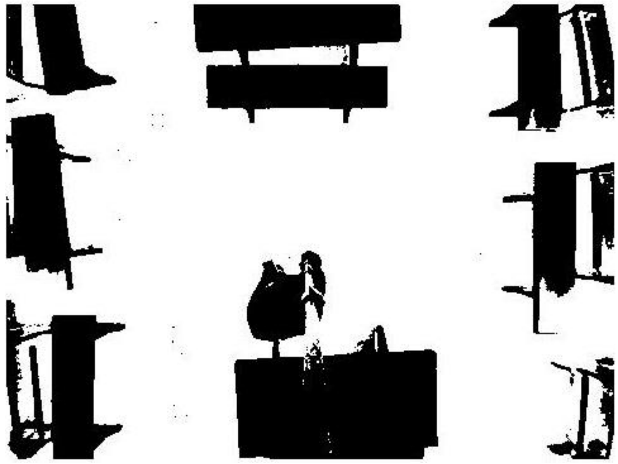

Figure 3: Detected Fixed Objects and Marked Them into a Binary Image.

But as this image was having inner white space between parts of objects which may be assumed by our path planning algorithms as a moveable path area, these spaces needed to be blacked out. For this purpose, detected objects were needed to be bounded into a contour.

And as in a real environment, there will be moving objects. In order to give the robot a path, all objects in the environment at the time of generating the path need to be considered. To do so, the reference background image (Figure 4) will be used for identifying moving objects on top of it. The same process which was used in the above background reference generation was used here on top of this reference for this purpose.

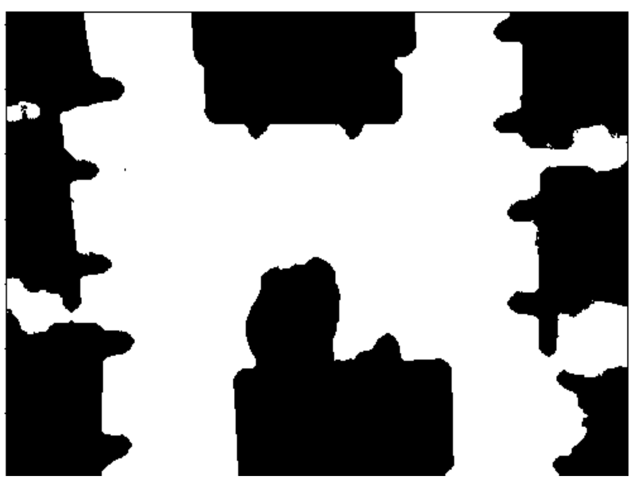

Figure 4: Image After Applying Contour

Figure 5: Image with New Objects on The Left and a Binary Image After Detection of All Objects in the Environment.

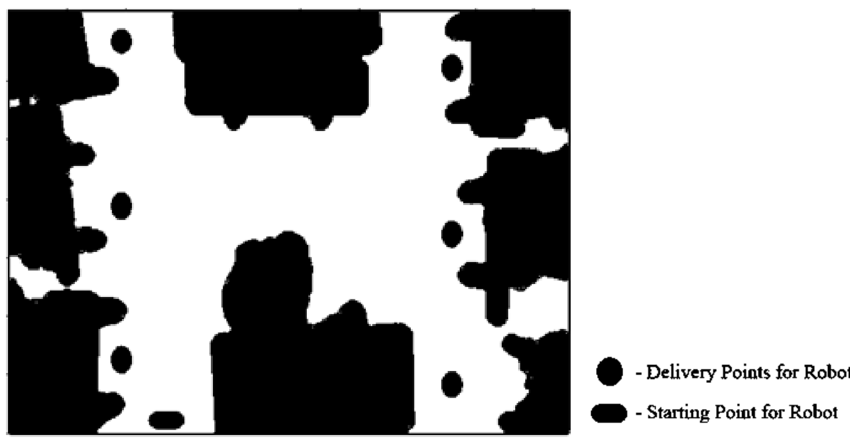

Figure 6: Starting Point and Possible Destination Points Are Marked on the Map.

2.3 Generating the Path for the Robot to Its Destination Using RRT* Algorithm

Once the objects and moveable area in the environment are detected, now the path needs to be generated. For this purpose, the starting and destination nodes need to be defined. In a real application, it should be definable by the end user. In this application, those nodes are defined and used to generate the path.

Using this map, according to the user desired destination, the robot will get the path generated from the navigation system. The navigation system uses Extended Rapidly Exploring Random Tree (RRT*) for generating a path. Even though this doesn't give an optimized path, this algorithm allows us to work with a dynamically varying environment where path generation needs to be re-done according to the moving obstacles.

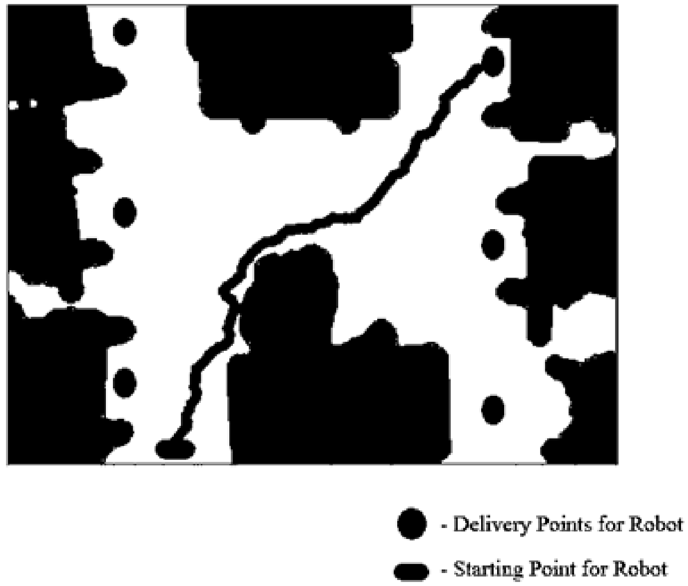

Figure 7: Image Showing the Generated Path from the Robot Starting Point and To One of The Possible Delivery Points

Once this path is generated, it is transmitted to the robot and thus the robot follows the path. A built-in range sensor in the front of the robot will keep looking for any sudden obstacles come to collide on it. To avoid the collision, the robot will stop moving if there is an object on its path. Once there is an obstacle on its path and it can't move forward, the robot will allow a waiting period to check whether the object is moving away from the path. If not, the robot will need a new path and it requests the system for a new path.

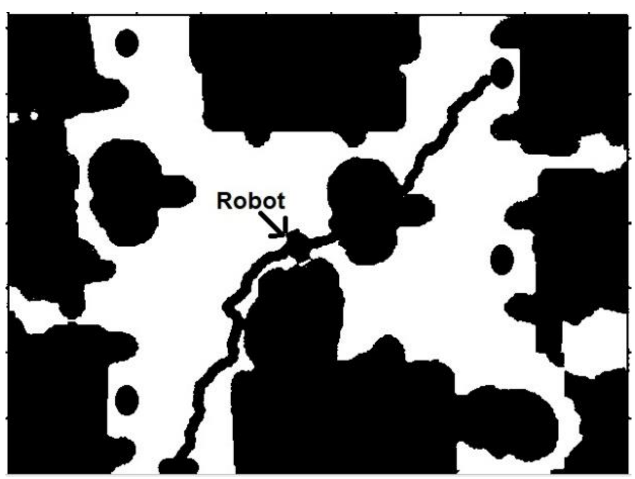

Figure 8: Robot Stops When There Is A Sudden Obstacle Comes on Its Path. Robot Waits and Request A New Path.

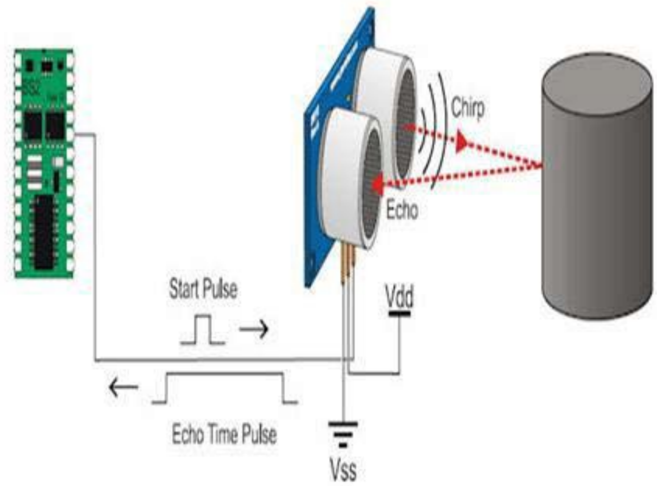

Figure 9: Ultrasonic sensor detects an immediate obstacle in the front (Anila, Saranya, Kiruthikamani, and Devi, 2017)

2.4 Regeneration of Path When a New Obstacle Is Introduced in the Previous Path

A regeneration of path is done by simply changing the start point to the current position of the robot and generating the path again based on current parameters. And the newly generated path will be transmitted to the robot.

When the request for a path generation is received by the system it goes through all the previous processes of detecting obstacles and generating the binary image of obstacles and moveable area. The following image (Figure 10) shows the regeneration of path.

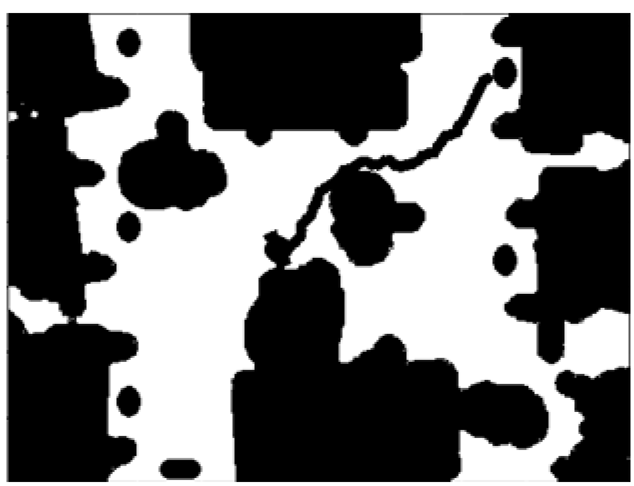

Figure 10: Newly Generated Path Based on the Current Environment Configurations.

3. CONCLUSION AND FURTHER IMPROVEMENTS

In this paper, the authors presented an external camera based indoor navigation system for indoor robots by mounting a camera on the top of the room to get an overall view of the room. With that, processed the images to get an obstacle map of the room and to plan a path for the robot to guide it to the desired destination. And and the authors found that most of the indoor service robots are using a fixed path to navigate in the environment and also a very few uses sensors built in the robot to navigate which is also not an effective method as it does not include the overall configuration space when path planning as the sensors have a limited line-of-sight. This methodology presented in this paper will be an added advantage over other existing systems in the industry as it provides solution for sudden objects that come across the path by allowing the robot to re-request a new path after a waiting time allowing to check whether the object is moving away from the path. In future, authors plan to improve this with optimizing the path and to include prediction of moving obstacles and improving the efficiency in indoor path planning process.

4. REFERENCES

Anila, D. S., Saranya, B., Kiruthikamani, G., & Devi, P. (2017). *INTELLIGENT SYSTEM FOR AUTOMATIC RAILWAY GATE CONTROLLING AND OBSTACLE DETECTION*. *4*(8), 8.

Connell, D., & Manh La, H. (2018). Extended rapidly exploring random tree–based dynamic path planning and replanning for mobile robots. *International Journal of Advanced Robotic Systems*, *15*(3), 172988141877387. https://doi.org/10.1177/1729881418773874

Khanmirza, E., Haghbeigi, M., Nazarahari, M., & Doostie, S. (2017). A Comparative Study of Deterministic and Probabilistic Mobile Robot Path Planning Algorithms. *2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM)*, 534–539. https://doi.org/10.1109/ICRoM.2017.8466197

Lavalle, S. M. (1998). *Rapidly-Exploring Random Trees: A New Tool for Path Planning*.

Li, I.-H., Chen, M.-C., Wang, W.-Y., Su, S.-F., & Lai, T.-W. (2014). Mobile Robot Self-Localization System Using Single Webcam Distance Measurement Technology in Indoor Environments. *Sensors*, *14*(2), 2089–2109. https://doi.org/10.3390/s140202089

Mapping Camera Coordinates to a 2D Floor Plan. (n.d.). Retrieved October 20, 2019, from https://zbigatron.com/mapping-camera-coordinates-to-a-2d-floor-plan/

Milford, M., & Wyeth, G. (2010). Persistent Navigation and Mapping using a Biologically Inspired SLAM System. *The International Journal of Robotics Research*, *29*(9), 1131–1153. https://doi.org/10.1177/0278364909340592

Noreen, I., Khan, A., & Habib, Z. (2016). *A Comparison of RRT, RRT* and RRT*-Smart Path Planning Algorithms*. 8.