Mobile Robot Navigation System Vision Based through Indoor Corridors

[1]Hairol Nizam Mohd Shah,[2]Lee Zhong Chen,[3]Zalina Kamis, [4]Azhar Ahmad, [5]Mohd Rizuan Baharon
[1,2,3,4]Center for Robotics and Industrial Automation
Faculty of Electrical Engineering
Universiti Teknikal Malaysia Melaka
76100 Durian Tunggal, Melaka, Malaysia
[5]Department of Computer System and Communication,
Faculty of Communication and Information Technology,, Universiti
Universiti Teknikal Malaysia Melaka
76100 Durian Tunggal, Melaka, Malaysia
Email: hnizam@utem.edu.my

***Abstract*— Nowadays, industry has been moving toward fourth industry revolution, but surveillance industry is still using human in patrol. This will put this industry in risk due to human nature instincts. By using a mobile robot with assist of vision sensor to patrol can bring this industry to a new level. However, the indoor corridor navigation will become a big challenge to this method. The objective of this project is to develop a navigation system using vision sensor and navigate the mobile robot in indoor corridor environment. To perform this operation, a control system though the WLAN communication develop to guide the movement of mobile robot. Besides that, corridor following system with vision sensor that using Sobel edge detection method and Hough transform to getting the vanish point is needed to help the robot to safely travel in the corridor. Both systems can be using MATLAB to be execute and link with the mobile robot through WLAN connection. This system can be analysis the corridor condition base on different feature and can decide to drive the mobile car in the direction that given. The image capture by mobile robot can be stream to MATLAB in real time and receive a feedback in short time.**

***Index Term*— Hough transformation, line extraction, vanish point, edge detection**

I. INTRODUCTION

In Malaysia, there is a massive amount of money that had stolen by the security guards who secure the building. This case is occurred on election night 2018 at Prime Minister's Office. Former Prime Minister claimed that a total of RM 3.5 billion is stolen by 17 security guards [1].

From the above incidence, there are a potential hazard in hiring a human with unknow personality to secure our estate. Therefore, it is required to replace the human patrol with machine. This machine can be a robot with vision sensor, which can analysis the area of patrol and record all the data obtained for future reference. This method not only overcome the problem above, but it also increases the quality of the patrol system. This system can provide a powerful evidence in form of image or video rather that a testimony of a security guard.

To overcome that problem, human patrol is used to secure a place. Human is used to guard an area of place with the assist of the CCTV. Besides that, human also required to patrol the area in a random or fixed time and route frequently. This will give them permission to assess the area and it will be having some risk in it. This is due to the nature of human being that is greedy.

Hence, the human patrol task needs to be digitalized. To making this solution achieve, the advantage of the human patrol which is highly mobility must implant into the CCTV system. A robot with a camera which can move inside the path of patrol can carry out this task, but the storage of the data obtained will be an issue. This issue can be solved through the IoT solution in fourth industry revolution. This method will record the situation on the blind sport of CCTV system.

II. RELATED WORKS

A. *Type of Robot*

Robot is an importance platform to test and stimulate an algorithm of navigation system in corridor. There are 3 type of robot used in previous work to perform this operation which are mobile robot [2–8], humanoid [9] and micro air vehicle (MAV) [10]. The mobile robot is used to perform this task due to the control of the mobile robot is easy. The combination of rotation of each motor in either clockwise (cw) or counterclockwise (ccw) will result in different motion. The Table I below will show the motion of mobile robot and its corresponding motor rotation combination.

Table I
Relation of direction of motor and the motion of mobile robot

Motion of mobile robot	Rotation of left motor	Rotation of right motor
Go forward	cw	cw
Go reverse	ccw	ccw
Go left	ccw	cw
Go right	cw	ccw

However, the mobile robot only can move in flat surface. Humanoid robot can be overcome this kind of problem. Humanoid robot can move in the rough surface of corridor such as stair but hard to control it motion. The leg of the humanoid robot, NAO having 6 degree of freedom (DoF) [11]. In order to move one step, the control algorithm of humanoid robot required to calculate all angle for each of the motor. This will increase the complexity of the system. MAV is better that humanoid robot in overcome surface problem because it did not move at surface. At the same time, the factor influence it control also increase. The environment factor such as wind and the mass of sensors at MAV will affect it performance. To run a new algorithm, the robot with easy control is better because this is easy to troubleshoot the problem in the algorithm. Hence, the mobile robot is the better choice.

B. Type of Sensors

Sensor is the key to let a system or robot to become intelligence. The sensors to perform navigation task having many types. First type of sensor is combination of Global Positioning System (GPS) and Inertia Measurement Unit (IMU) [7,12]. GPS is used to pinpoint location for a device, the accuracy of GPS is in range of few meter and ineffective in indoor environments. Function of IMU used to getting the direction of a device moving but it is having a cumulate error when it uses for a long time period. The combination of GPS and IMU can overcome the weakness of each other by increase the accuracy of the GPS and eliminate the cumulate error of IMU.

Next, the laser sensor such as lidar or laser scanner is another type sensor used in navigation system and obstacle avoiding system [4-7]. Laser sensor is used to calculate the distance between object ahead and sensor. Laser sensor is very fast respond, easy to use and use less processing power. However, many laser sensors required to get enough information of surrounding because laser sensor only can detect the distance of a surface which is perpendicular to laser sensor. The angle of each sensors is an importance issue in order to create a 3D map.

Besides that, camera also can be use as vision sensor [2,9,13]. Camera is easy to setup by fixed it position on a robot. Vision sensor is can get large information about environment by real time analysis of image frame by frame. For example, in [10] was used stereo camera to build a 3D image, this can get the information in term of range data, 3D virtual scan and 3D occupancy map. This will be causing this sensor required a hardware with high processing power to analysis it, so the response time of this sensor depend on the specification of the hardware.

Last but not least, Kinect is used in [3] to perform the same operation. Kinect is a combination of vision sensor and laser sensor. From the combination of both sensors, more information can be obtained. From the library of Kinect, user can direct get the information needed with need to know or develop an algorithm. Other than that, the price of Kinect also higher than other sensors.

C. Corridor Scanning Method

For a robot to move automatically in corridor environment, the method to analysis the information collected by sensors is importance. There are 2 method to analysis the data which are 3D map and image processing. For 3D map, laser sensors are required to obtain data at difference angle [3-5]. By getting the data of distance from each sensor at difference angles and this data is used to do alignment, filtering and bounding. Through this process, the distance between laser sensors at two adjacent angles is estimate, so if the number of sensors is less, this will be causing the 3D map to have many inaccuracies. Hence, large number of sensors required to create a detailed 3D map for better navigation system.

Another method is through image processing method to process the data of a camera [2,3,9,13,14]. Segmented Hough Transform and Canny's algorithm is the most common method used in the analysis corridor environment. Line segment detector (LSD) is an upgrade version of previous method by refer to methods proposed by Burn et al. method and Desolneux et al.. Both methods are using edge detection and line extraction to get the vanish point of the corridor. However, LSD is faster than pervious method in obtaining the line from an image, but LSD can't get short line from the image [8]. Therefore, it will be loss some of the data in the image. Besides that, block-based image processing method also used to obtain data about the environment [6]. This method using the pixel of the image to get the information. Grouping of the same pixel into a large pixel will increase the respond time but this method only effective in large color different such as road. Hence this method more convenience using in analysis outdoor environment.

D. Autonomous Navigation

Autonomous navigation is required for a robot to travel across a distance without the supervise by a human. Autonomous navigation in indoor and outdoor is difference. In outdoor environment, vehicle is control by a hardware which processing the data from the sensor. There are two way to carry this outdoor task which are combination of vision and sensors [7,12] or vision and artificial neuron network (ANN) [13]. From the combination of vision and sensors such as GPS and IMU, the system can understand the item at surrounding of vehicle, the position of vehicle, and the direction of the vehicle move. For the vision and ANN method, the system is fully depending on the information collected by the camera only. Therefor the ANN will analysis and getting the data as much as possible to make sure the vehicle can move safely on the road.

In indoor corridors, the navigation methods are easier since the speed of mobile robot is not as fast as outdoor vehicle and did not endanger human life, so lesser safety issues need to consider. Few methods can carry out this task which are vision only or with sensor fusion, ANN and finite state machine (FSM). By using vision, the system can move I at center of the corridor by adjust it direction from the feedback of the vanish point [9]. Moreover, the vision also can help the robot to follow the path from the analysis of the image. For example, the path given is turn left at junction, then the robot will be move straight before it reach the junction until the image processing will a feedback of junction ahead and it will turn left. There is not distance measurement required in the navigation. By adding the sensors fusion, the system more correctly with error modal from the sensor feedback [2]. This will increase the accuracy of the system because compare data from two sources. Next, ANN method is used the analysis the data from image [3] or from sensor such as laser sensor [5-6] and give the correspond command to control the robot. The few layers ANN system will analyses the data received and give the output to the motor to move from time to time until reach the destination. The last method is FSM, which is generate a route from a topological map in sequence of steps. This method is effective in navigation and did not need to know the actual position of the robot during navigation. On the other hand, this method required a topological map insert to the system and known initial position before any navigation start.

III. Methodology

In methodology flowchart, it will show the overall method use to complete this research. This flowchart will discuss the method use for hardware and software developing. The Figure 1 show the flowchart of the navigation system that developed in this project.

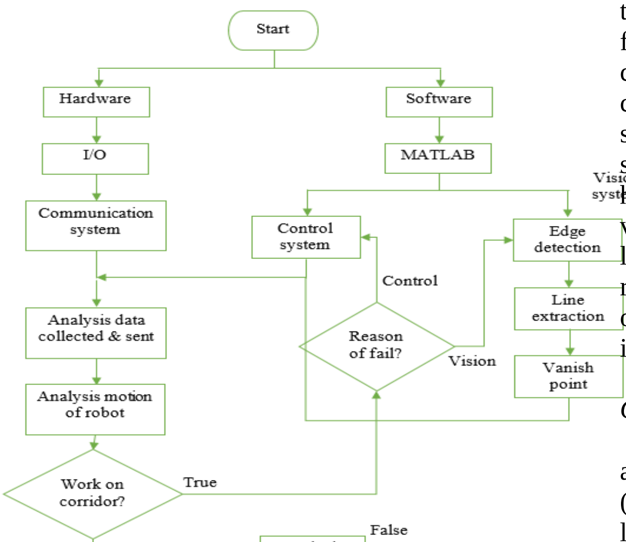


Fig. 1. Flowchart of navigation system

In hardware part, the input output (I/O) configuration is required to developing communication system while in the software part, there are two systems to be developed which are vision and control system. The vision system uses to collect the information and guide the system through it feedback. On the other hand, the control system needs the feedback of vision system to interface with communication system to navigate the system.

A. *Edge detection*

This vision system is collecting the information by capture the image in corridor and process it into useful data. Edge detection is the importance process to convert RGB image capture by the robot to an image of the edge in image. Edge detection is an image processing method used to find the boundaries of object in an image. This method works by detecting the discontinuities of the brightness within the image. It is used to extract the data in the image as an edge image. This edge image is a binary image. There a many type of edge detection methods. The different type of edge detection method is used to obtain the edge of the same image and the result of the different method is compared to choosing the suitable method for the vision system

B. *Line extraction*

The Hough transformation is covert the edge image into data form. The data obtained from the calculation is the first point and end point of a line, the shortest distance of the line with the origin point, rho and the angle of the rho from the origin. From the first point and end point of the line, the gradients of the lines are obtained. These gradients are used to form linear line equations. These line equations are extended the line to the border of the image to making the intersect of line easier. The angle obtained from the Hough transformation is the importance characteristic in this system. These angles are used to classify the lines into different sections. There are 4 sections, which are vertical, horizontal, left and right section. The vertical section is from -30° to 30° while the horizontal is from 70° to 90° and -70° to -90°. The vertical line will be ignored in system while the horizontal line is used to determine the stop condition. The left and right section angle is the same but different in the sign only and these lines is the importance line which are used in vanish point calculation.

C. *Vanish point*

From the lines plot from the data, the line in positive angle (left section) is intersect with the negative angle (right section). Each of the lines is intersect will all the line individually. Two lines are used to calculated the intersect point at one time. The intersection point is obtain from the gradients and y-intersects of the line equations of both of the lines. The intersect points are recorded. The vanish point is calculate using the median of the intersect points. The horizontal distance between vanish point and the center of image is the feedback of vision system. The horizontal distance is used due to the vanish point is not at a constant y position (vertical line) and the robot only can adjust its position in the vertical direction. Hence the vertical distance can be ignored in the system.

D. Development of vanish point algorithm

After the image was processed using Sobel method, the image is converted to a binary image with only the edge information only [16-18]. This image was undergone Hough Transformation to transform the information into data that can be used. Hough Transformation formula is shown in Eq. 1.

$$r = x\cos\theta + y\sin\theta \quad (1)$$

Where r is the distance from the origin to the closest point on the straight line, and theta θ is the angle between the x axis and the line connecting the origin with that closest point.

From the data calculated through Hough Transformation, the lines which fulfil the Hough parameter will be stored as it first and last point as a variable. The points were used to obtain it line equation in Eq. 2.

$$y = mx + c \quad (2)$$

Where x and y are the coordinate of the point, m is the gradient of the line and c is the y-intersect of the line. With the line equation, the lines were extended from one end to another end of image. This will allow the intersection of line to occur. The point of intersect of the lines is calculate by using these formulas in Eq. 3 and Eq. 4.

$$xintersect = \frac{c_2 - c_1}{m_1 - m_2} \quad (3)$$

$$yintersect = m_1 \times xintercept + c_1 \quad (4)$$

Where xintersect and yintersect in Eq. 3 and Eq. 4 are the coordinate for the intersection, m_1 is the gradient and c_1 is the y-intersect of line 1 while m_2 is the gradient and c_2 is the y-intersect of line 2.

After all intersection of points was calculated, the median of all the points is obtained as the vanish point [19-20]. The mobile robot will be react based on the distance of the vanish point and the center of camera in x-axis. All the lines and points were plotted in the image with different color for easy to recognize.

E. Test run in corridor with different environment.

To analyze the performance of the system, 3 corridors were chosen based on its environment. The first corridor is one of the corridors in Sport Complex of UTeM as shown in Figure 2. This corridor was chosen due to it was clear of obstacle and the end of the corridor is a wall without anything at the end. Hence, the vanish point calculated is not influence by another factor.

Fig. 2. Image of first test run location

The second corridor that test run occurred was in front lift in third floor of FKE as shown in Figure 3. The location chosen because of it don't have any obstacle in its runway but the end of corridor is not flat. This will affect it calculation of vanish point.


Fig. 3. Image of second test run location
(III.4)

The last location was in front of FKE lecture's office at first floor as shown in Figure 4. This location has a flat end, but it has many obstacles at side of the corridor. These obstacles will be count into the calculation of the vanish point and will affect the decision made by the mobile robot.

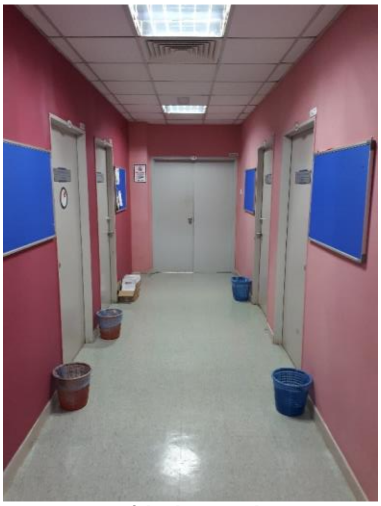


Fig. 4. Image of third test run location

The mobile robot was run in with the navigation system developed in every corridor above. In each corridor, the mobile robot is run 15 times and the result of each run will be recorded. The accuracy of the system in each corridor was calculated using formula of Eq. 5.

$$accuracy = \frac{number\ of\ success\ runs\ or\ image\ detected}{total\ number\ of\ runs\ or\ image\ tested} \quad (5)$$

During each run, the image captured along the path was stored in laptop. The vanish point of the image were obtained by the same system and all related line and point were shown in the image. These processed images were observed, and the type of errors occurred was recorded in a table.

IV. Results

A. *Comparison between edge detection method*

To identify the different of Canny and Sobel edge detection, two photos is taken in different corridor. these photos are captured using the camera of robot. The Figure 4.1shown the image used to compare the result of these edge detection method.

(a)

(b)

Fig. 5. Image of (a) corridor left and (b) wall left

From the Table 2, we can observe that the corridor is restore almost all the detail using the Canny edge detection method, but edge detected from the wall is too much noise. This noise will affect the calculation for the vanish point and lead the mobile robot to a wrong direction. The edge image in the Table 2 are used in the vanish point algorithm and the result shown in Table 3 with all the importance data is plotted.

Table II
Comparison of Canny and Sobel edge detection method (III.5)

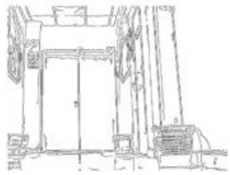
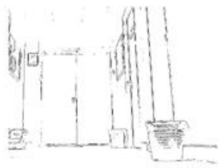
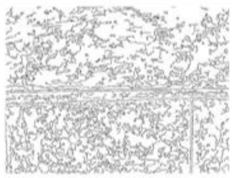
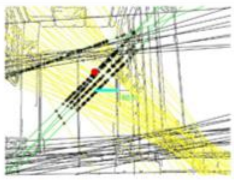
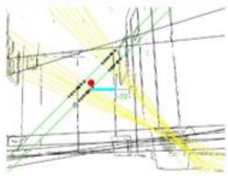
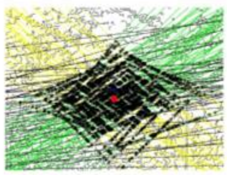
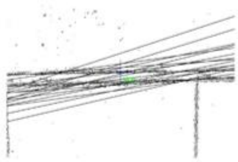
	Canny	Sobel
Corridor		
Wall		

Table III
Comparison of both method after processed

	Canny	Sobel
Corridor		
Wall		

From the Table 3, Canny edge detection method is obtaining more line than Sobel edge detection method. The higher the number of lines, the more accurate the vanish point calculated. However, in the image of the wall, the noise detected by the Canny edge detection method is obtained the unwanted lines, this will lead the mobile robot to make wrong decision. Through this experiment, Sobel edge detection method is chosen to use in our system.

B. Development of vanish point algorithm

A photo of corridor as shown in Figure 6 is selected to processed using the vanish point algorithm. This photo is chosen due to the line of corridor is obvious and no obstacle in corridor.

Fig. 6. Sample image for vanish point algorithm

This photo will undergo the edge detection and converted to binary image shown in Figure 7. This photo is shown the edge contain in the sample image.

Fig. 7. Edge of sample image

From the edge image, Hough Transformation was converting all the edge into a data that can used for the algorithm to calculate.

The rho and theta are representing the r and θ in Eq. 1. While the point 1 and point 2 were used in the Eq. 2 to obtain the line equation and extended to the frame of image as shown in Figure 8. The different color of lines used in the Figure 8 are used to differentiate the lines according to it θ value. Yellow lines are representing lines with θ values from -30 to -70, which also same angle with the line separate the floor and the right wall. The green lines are same as the yellow lines but is in positive angle, so it is shown the angle of the line separate the floor and the left wall. The blue lines are representing the horizontal line which is the angle between 70 to 90. The vertical line was not used in the algorithm, so it was not plotted in the image.

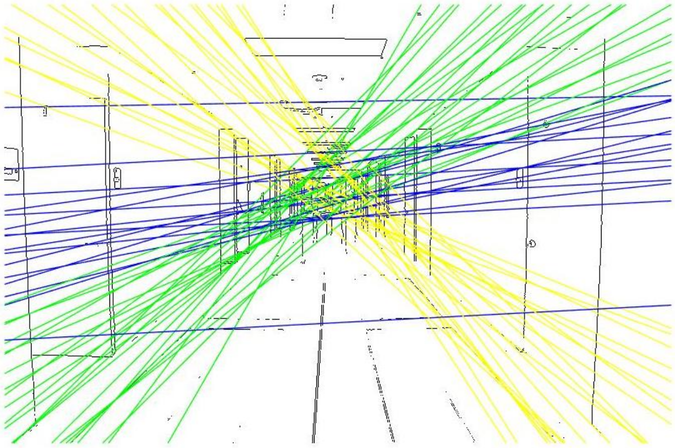


Fig. 8. Lines from Hough Transformation ploted in edge image

The intersection point was only calculated from the intersect of yellow and green lines only. All points were plotted in the Figure 9 with black color. The intersects are between yellow and green lines only to reduce the error .

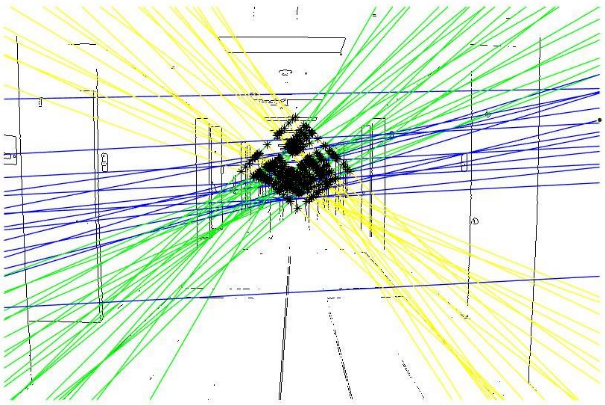


Fig. 9. Intersection point of yellow and green lines

From the point of intersection, the vanish point was obtained by calculated it median. Through this median calculation, the exclude of the intersect with the vertical line, let the deviation of the vanish point to be more consistent. The vanish point was represented as a red dot on the Figure 10.

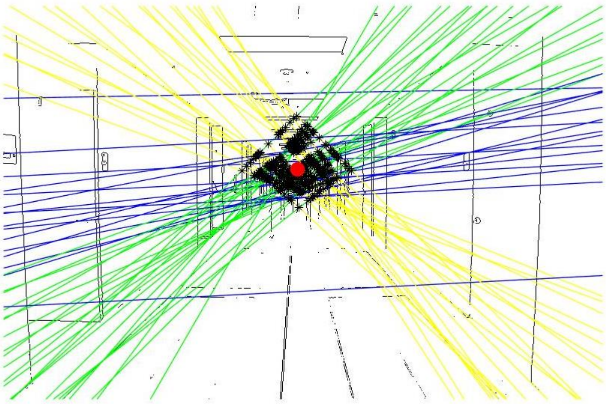


Fig. 10. Vanish point (red dot)

This vanish point is used to compare with the center of the image. The distance of these 2 points in x-axis is the guideline for the mobile robot to move in center of corridor. In Figure 11, the lines are removed, and all the importance information were plot in the original photo. The distance of vanish point and the center of image (blue cross) in y-axis is shown in cyan color line and its value is shown with green color. The negative value means the vanish point is on the left-hand side while positive is in right-hand side.

Fig. 11. The output of vanish point algorithm

The sample image is a photo of corridor with vanish point inside the image, but in some condition, the algorithm can't detect the vanish point. In this kind of situation, this algorithm needs an alternative command for the mobile robot to move until find the vanish point.

For example, in the situation of Figure 12, the system cannot find vanish point inside the image, so the system will give an output of no left line. In this output, the robot control system will turn left in a small degree and capture a new image to calculate. This process will be continuing until it finds the vanish point. This process also can apply to situation of no right line.

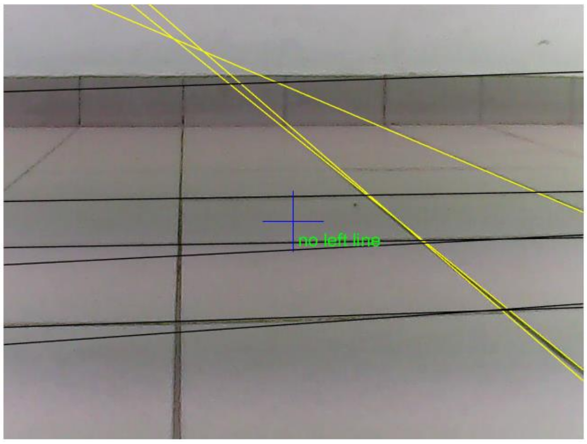


Fig. 12. Image of corridor without the left line

C. Test run in corridor with different environment.

In this experiment, the mobile robot with the system is tested in 3 location. These location are Sport Complex of UTeM, in front lift in third floor of FKE and in front of FKE lecture's office at first floor.

1) Test run in location 1

This experiment is repeated 15 times. All the data and observations are recorded in the Table 4.

Table IV
Number of error detected and success run in location 1

Number of trials	Frame capture per trial	Number of detection errors				Success in completing the task
		Vanish point	Left line	Right line	Unwanted line	
1	21	1	3		1	Fail
2	14		2			Fail
3	5			1		Fail
4	5			1		Fail
5	20		3	1		Fail
6	19		1			Success
7	16		2			Success
8	11					Success
9	17				1	Success
10	19					Success
11	16		1			Success
12	22					Success
13	9			1		Success
14	13		1			Success
15	15				1	Success
Total	222	1	13	4	3	10 success
		21				

$$Accuracy\ of\ navigation = \frac{10}{15} = 66.67\% \qquad Accuracy\ of\ image\ detection = \frac{222-21}{222} = 90.54\%$$

From the Table 4.4 and calculation, the accuracy of the mobile robot move in this location is 67%. Although this success rate is low, but the accuracy of the image detection is high. This situation is occurring due to the system making the wrong calculation in crucial moment.

2) Test run in location 2

This experiment is repeated 15 times. All the data and observations are recorded in the Table 5.

Table V
Number of error detected and success run in location 2

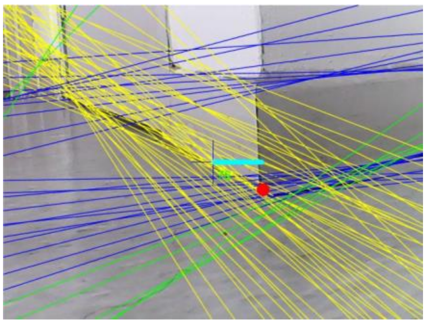
Number of trials	Frame capture per trial	Number of detection errors				Success in completing the task
		Vanish point	Left line	Right line	Unwanted line	
1	6					success
2	5					success
3	10			1		success
4	6				1	fail
5	17			1	2	fail
6	9					success
7	15				2	success
8	26		6	2		success
9	16				2	fail
10	9					fail
11	9				1	fail
12	17				1	success
13	14				4	fail
14	9					success
15	5					fail
Total	173	0	6	4	13	8 success
		23				

$$Accuracy\ of\ navigation = \frac{8}{15} = 53.33\%$$

$$Accuracy\ of\ image\ detection = \frac{173 -}{17} = 86.71$$

In this location, the success rate is low because of the surface of corridor end is not flat and causing the robot to recognize the wrong edge. In Figure 13, the green lines below the vanish point which from the corner will confuse the robot.

(a)



(b)

Fig. 13. Vanish point which from the corner (a) before and (b) after

Other than that, the robot also fails to enter the turn before it is making left turn. This cause the camera is obtain the image too far, and the corridor side too wide, so it can't detect the corridor line.

3) Test run in location 3

This experiment is repeated 15 times. All the data and observations are recorded in the Table 6.

Table 6: Number of error detected and success run in location 3

Number of trials	Frame capture per trial	Number of detection errors				Success in completing the task
		Vanish point	Left line	Right line	Unwanted line	
1	4					Success
2	3					Success
3	6			2		fail
4	6				1	fail
5	7					Success
6	4					Success
7	6		1	1		fail
8	6				1	fail
9	11				3	fail
10	6			1	1	fail
11	1					fail
12	7		1	1		fail
13	4					Success
14	9			2	2	fail
15	9				1	Success
Total	89	0	2	7	9	6 success
		18				

$$Accuracy\ of\ navigation = \frac{6}{15} = 40\%$$

$$Accuracy\ of\ image\ detection = \frac{89 - 18}{89} = 79.78\%$$

The accuracy of navigation in this location is too low due to the obstacles (dustbin) in the side of corridor is blocking the corridor lines and create unnecessary lines. This line will affect the whole vanish point calculation.

4) Overall performance of the algorithm

To obtain the overall performance of the algorithm, the total errors and total frame capture will be obtain. This data is shown in the Table 7.

Table 7: Total frames and errors

Location	Total frames capture	Total errors
1	222	21
2	173	23
3	89	18
Total	484	62

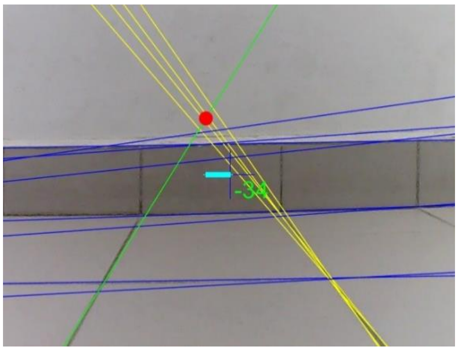
$$Accuracy\ of\ system = \frac{484 - 62}{484} = 87.19\%$$

The overall accuracy of system is acceptable, however the success rate in different location is vary depend on the environment. Besides that, the frame rate processed by the mobile robot is one frame per second. This will lead the collided or missed the importance data. Therefore, to overcome this problem, the processing speed need to be increased by run this algorithm in more powerful laptop.

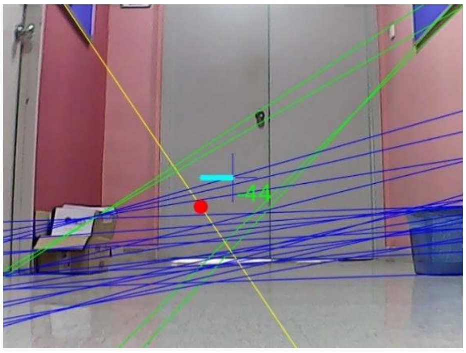
5) Effect of environment

As the result shown the lesser the obstacles in the corridor, the lesser the noise created in system. Hence the better the performance of the system. Besides that, the clearer the line of corridor, the better the performance.

In Figure 14, the image on the left which have the clearer line than image on right, so the system gets the vanish point more accurate. Other than that, the smaller the wide of the line detected, the closer the robot reach the end.



(a) ending (wall) location



(b) current location

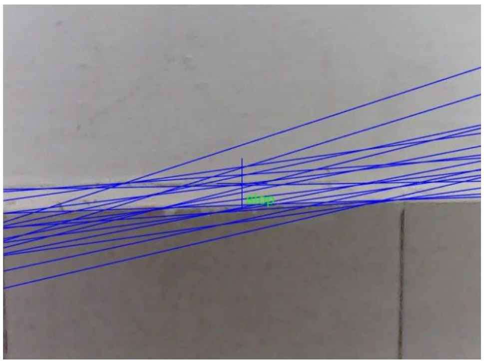
Fig. 14. Image captured before reach end

This situation also depends on the angle of view of the camera. The angle of the camera used in this robot is not wide, so it only can get part of information in front it. In Figure 15, the distance of the robot and the end of the corridor (door) is longer, but the view of the robot is too far in front of its actual position. Figure 16 shown the robot view in situation in Figure 15. Both view in the robot also detect the condition of the end of corridor but the distance of the robot stop is varied.

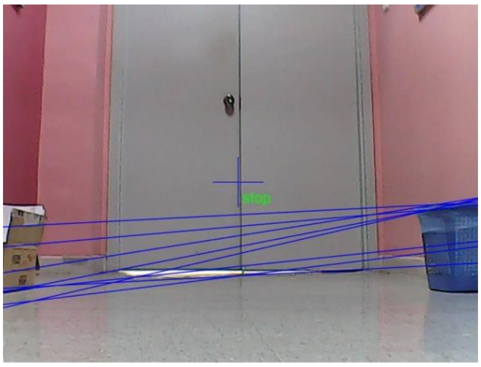
(a) ending (wall) location

(b) current location

Fig. 15. Image of the distance robot detect an end position



(a) ending (wall) location



(b) current location

Fig. 16. View of robot

V. Conclusion

In conclusion, the performance of the algorithm is acceptable, but the overall performance of the robot is limited by the hardware use. By improve the hardware used, the robot performance can be increased. The highest success rate in navigation is at location 1, however rate is 66.67% only. Overall accuracy of the system is 87.19% which is varied too much with the success rate. This situation is due to the frame rate processed by the system. The frame rate is 1 fps, so the robot having time gap of 1 second before getting next command. In this time gap, the robot might be missed an importance information or collide with the wall before the robot is able to respond. Other than that, the angle of view also affects the performance of the robot. The camera used in this robot is limited the information that can be obtain. In future, the overall performance of this project can be improved through the upgrade of the hardware such as used higher processing power make more frame rate can be process and the location of the camera mounted into the robot.

Acknowledgment

The authors are grateful for the support granted by by Center for Robotics and Industrial Automation, Universiti Teknikal Malaysia Melaka (UTeM) in conducting this research through grant JURNAL/2018/FKE/Q00007 and Ministry of Higher Education.

References

[1] “PM’s Office guards stole Umno election funds: Najib, SE Asia News & Top Stories - The Straits Times.” [Online]. Available: https://www.straitstimes.com/asia/se-asia/pms-office-guards-stole-umno-election-funds-najib. [Accessed: 12-Nov-2018].

[2] E. Bayramoglu, N. A. Andersen, N. K. Poulsen, J. C. Andersen, and O. Ravn, “Mobile robot navigation in a corridor using visual odometry,” *Adv. Robot. 2009. ICAR 2009. Int. Conf.*, pp. 1–6, 2009.

[3] D. S. O. Correa, D. F. Sciotti, M. G. Prado, D. O. Sales, D. F. Wolf, and F. S. Osório, “Mobile robots navigation in indoor environments using Kinect sensor,” *Proc. - 2012 2nd Brazilian Conf. Crit. Embed. Syst. CBSEC 2012*, pp. 36–41, 2012.

[4] A. Adán, B. Quintana, A. S. Vázquez, A. Olivares, E. Parra, and S. Prieto, “Towards the automatic scanning of indoors with robots,” *Sensors (Switzerland)*, vol. 15, no. 5, pp. 11551–11574, 2015.

[5] D. O. Sales, F. S. Osório, and D. F. Wolf, “Topological Autonomous Navigation for Mobile Robots in Indoor Environments using ANN and FSM,” *1ª Conferência Bras. Sist. Embarcados Críticos*, 2011.

[6] I. Susnea, V. Minzu, and G. Vasiliu, “Simple, real-time obstacle avoidance algorithm for mobile robots,” *Recent Adv. Comput. Intell. Man-Machine Syst. Cybern.*, no. figure 2, pp. 24–29, 2009.

[7] V. Sezer and M. Gokasan, “A novel obstacle avoidance algorithm: ‘Follow the gap method,’” *Rob. Auton. Syst.*, vol. 60, no. 9, pp. 1123–1134, 2012.

[8] X. Li and B. J. Choi, “Design of obstacle avoidance system for mobile robot using fuzzy logic systems,” *Int. J. Smart Home*, vol. 7, no. 3, pp. 321–328, 2013.

[9] A. Paolillo, A. Faragasso, G. Oriolo, and M. Vendittelli, “Vision-based maze navigation for humanoid robots,” *Auton. Robots*, vol. 41, no. 2, pp. 293–309, 2017.

[10] L. Heng, L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, “Autonomous Obstacle Avoidance and Maneuvering on a Vision-Guided MAV Using On-Board Processing,” pp. 2472–2477, 2011.

[11] “Robot App Store | Apps For Every Robot!” [Online]. Available: http://www.robotappstore.com/Robopedia/Degrees-of-Freedom. [Accessed: 11-Nov-2018].

[12] M. Goebl *et al.*, “Design and Capabilities of the Munich Cognitive Automobile,” *Proc. IEEE Intell. Veh. Symp.*, pp. 1101–1107, 2008.

[13] P. Y. Shinzato and D. F. Wolf, *Features image analysis for road following algorithm using neural networks*, vol. 7, no. PART 1. IFAC, 2010.

[14] R. Grompone Von Gioi, J. Jakubowicz, J. M. Morel, and G. Randall, “LSD: A fast line segment detector with a false detection control,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 4, pp. 722–732, 2010.

[15] “小R讲堂-Arduino DS视频智能小车.” [Online]. Available: http://www.xiao-r.com/index.php/Study/catalog/cid/1. [Accessed: 11-Nov-2018].

[16] Hairol Nizam Mohd Shah, Marizan Sulaiman, Ahmad Zaki Shukor, "Autonomous detection and identification of weld seam path shape position", The International Journal of Advanced Manufacturing Technology, vol. 92(12), pp. 3739–3747, 2017.

[17] HNM Shah, M Sulaiman, AZ Shukor, MZA Rashid, "Vision based Identification and Classification of Weld Defects in Welding Environments: A Review", Indian Journals of Science and Technology, vol. 9(20), pp. 1-15, 2016.

[18] Hairol Nizam Mohd Shah, Marizan Sulaiman, Ahmad Zaki Shukor, Zalina Kamis, "An experiment of detection and localization in tooth saw shape for butt joint using KUKA welding robot", The International Journal of Advanced Manufacturing Technology, vol. 97(5), pp. 3153-3162, 2018.

[19] Hairol Nizam Mohd Shah, Marizan Sulaiman, Ahmad Zaki Shukor, Mohd Zamzuri Ab Rashid, "Recognition of butt welding joints using background subtraction seam path approach for welding robot", International Journal of Mechanical & Mechatronics Engineering, vol. 17(1), pp. 57-62, 2017.

[20] Hairol Nizam Mohd Shah, Marizan Sulaiman, Ahmad Zaki Shukor, Zalina Kamis, Azhan Ab Rahman, "Butt welding joints recognition and location identification by using local thresholding", Robotics and Computer-Integrated Manufacturing, vol. 51, pp. 181-188, 2018.