BILINGUAL PUBLISHING CO.
Pioneer of Global Academics Since 1984

Journal of Electronic & Information Systems
https://ojs.bilpublishing.com/index.php/jeis

ARTICLE

Development of IoT Based Mobile Robot for Automated Guided Vehicle Application

M. A. S. M. Alhaddad[1]   K. Kamarudin[1,2*]

1. School of Mechatronic Engineering, University Malaysia Perlis, Pauh Putra Campus, 02600, Arau, Perlis, Malaysia
2. Center of Excellence for Advanced Sensor Technology, University Malaysia Perlis, Lot 16-21, Jalan Jejawi Permatang, 02600, Arau, Perlis, Malaysia

ARTICLE INFO

Article history
Received: 22 July 2019
Accepted: 28 October 2019
Published Online: 31 October2019

Keywords:
Automated guided vehicle
Mobile robot
Internet of things
Mobile app
RFID

ABSTRACT

Mobile robot has been one of the researches focuses in this era due to the demands in automation. Many industry players have been using mobile robot in their industrial plant for the purpose of reducing manual labour as well as ensuring more efficient and systematic process. The mobile robot for industrial usage is typically called as Automated Guided Vehicle (AGV). The advances in the navigation technology allows the AGV to be used for many tasks such as for carrying load to pre-determined locations sent from mobile app, stock management and pallet handling. More recently, the concept of Industry 4.0 has been widely practiced in the industries, where important process data are exchange over the internet for an improved management. This paper will therefore discuss the development of Internet of Things (IoT) bases mobile robot for AGV application. In this project a mobile robot platform is designed and fabricated. The robot is controlled to navigate from one location to another using line following mechanism. Mobile App is designed to communicate with the robot through the Internet of Things (IoT). RFID tags are used to identify the locations predetermined by user. The results show that the prototype is able to follow line and go to any location that was preregistered from the App through the IoT. The mobile robot is also able to avoid collision and any obstacles that exist on its way to perform any task inside the workplace.

1. Introduction

The mobile robot is a robot that is capable of navigating around from one place to another in its environment without the need for physical or electro-mechanical guidance device. It has been used in many applications such as for remote sensing, transportation as well as to perform dangerous tasks like bomb defusing and minesweeper. Automated Guided Vehicle (AGV) is just the same as the mobile robot and it is a driverless car that can navigate to a predefined location by the user. Likewise, it is capable of following prescribed paths and it is usually used in the industries to transport raw materials to a preregistered location [1]. The

**Corresponding Author:*
K. Kamarudin,
School of Mechatronic Engineering, University Malaysia Perlis, Pauh Putra Campus, 02600, Arau, Perlis, Malaysia; Center of Excellence for Advanced Sensor Technology, University Malaysia Perlis, Lot 16-21, Jalan Jejawi Permatang, 02600, Arau, Perlis, Malaysia;
*Email: *kamarulzaman@unimap.edu.my*

 DOI: https://doi.org/10.30564/jeisr.v1i1.1061

AGV application is important in the workplac to reduce the labourers in the industries, ease the physical strain on them and to improve the operational performance [2].

In this project a mobile robot platform is designed using SOLIDWORKS for AGV application using the line following mechanism. The robot was controlled to navigate from one location to another through the Internet of Things (IoT) and RFID based tracking. The mobile robot should be able to avoid collision and any obstacles that could happen on its way to perform any task inside the workplace. The main idea of the robot's behaviour is to follow simple line algorithm and be able to perform the commands that will be sent from the mobile app. Therefore, this project involves designing, fabrication, integrating and interfacing robot's parts and sensors to IoT.

2. Related Works

IoT based automated guided vehicle deals with the material handling in the industry, which includes horizontal and vertical movements or combination of both. Material handling is an art of science which involves movement, packing, transporting. Material handling is an important activity in the production process. Out of total time spend on manufacturing 20% is for actual process & the remaining 80% is material handling moving the material from one location to another location. The above percentage should be altered according to the plant.

Yogesh Kakasaheb Shejwall and Sasi Kumar Gera [2] developed an IoT system for an automated guided vehicle that deals with the demands in the manufacturing industry. The AGV will be able to follow a given path allocated for it. This will happen with the aid of Arduino, an ethernet shield and RFID [3]. In this system, they can access and stored all the information in the device. this system will be sending the signals from the respective department using Bluetooth device and then the vehicle will receive these signals after that the vehicle will follow the path according to the given instruction. Delivering material to the respective location will be done after scanning the barcode of load material and after the vehicle accomplish that it will be back to the original position.

Nanda Kishor M and Prasad A P (2017) [4]. An IoT Based smart car parking using line following robot. This system proposed a vehicle used in a smart city that when it enters the parking slot a series of led will glow indicating the dedicated parking slot and then the car will follow the led pattern using the concept of the line follower robot. The vehicle will be controlled by the Arduino which is the main controller of the vehicle and this vehicle will consist of an IR sensor, RF module, LED stripes, line following robot. This system also constructs each car park as an IoT network, and the data that include the vehicle slot location and number of free slots in car park areas will be transferred to the data centre.

3. Methods

In this project Arduino will be the main controller to smoothly navigate the mobile robot. In this project there will be many procedures from designing the Three Dimensions (3D) model of the mobile robot using SOLIDWORKS, fabricating the designed parts, integrate the design parts and sensors to the mobile robot, implementation of the line following algorithm, development of IoT, interfacing IoT with the mobile App, interfacing the mobile app with the mobile robot through IoT and lastly testing the reliability of the line following mechanism in the mimicked industrial setup.

3.1 3D Design Progress

The mobile robot is designed (refer to Figure 4.1) in Computer Aided Design (CAD) programming 3D model using SOLIDWORKS where the robot will consist of two plates. Each plate got its own holes for wires to connect between the parts the lower plate will be attached with sensors, actuators and the Arduino. Nevertheless, the upper plate will be for the materials that will be transported from a place to another. There will be also six standoffs to hold and connect between the two plates. Two wheels will be in the front of the robot and a castor wheel will be in the back middle of the robot. two DC motors were also used to control the wheels. Also, there are the motor brackets to hold the DC motor and lastly there are the coupling to hold the DC motor with the wheel.

3.2 Line Following Mechanism

For accurate and smooth navigation for the mobile robot we have used a line following sensor contains of five IR sensor which are ML sensors, L sensor, middle sensor, R sensor and MR sensor to have smooth navigation through all the junctions in the way [5].There will be multiple junctions such as + junction, T junction and a wavy line. Hence, there will be many situations for the line following sensor. First situation is the ideal situation when the middle sensor is in the black line or the left sensor, middle sensor and right sensor is on the black line the mobile robot will move straight. Second situation is diagonally left when the right sensor or right sensor and most right sensor in on the black line the

DOI: https://doi.org/10.30564/jeisr.v1i1.1061

mobile robot will move diagonally to the left by slowing the RPM of the right DC motor until the ideal situation occur to continue moving straight. Third situation is diagonally right when the lift sensor is on the black line or the left sensor with the most lift sensor is on the black line. Consequently, in this case the mobile robot will turn diagonally to the right by slowing the RPM [6] of the left DC motor until the ideal situation occur to continue moving straight. Fourth situation is the situation where the Radio-frequency identification (RFID) will be used to control the right and left turn. If the RFID reader reads the RFID tag the mobile robot will turn left for one specific RFID tag by stopping the right DC motor. And will turn right for another specific RFID tag by stopping the lift DC motor.

3.3 Radio-Frequency Identification (RFID)

Radio-frequency identification uses electromagnetic fields to automatically identify and track tags attached to objects. To make our mobile robot smoother and to avoid any error that could happen we have used the RFID sensors to have a smoothly left and right turn. This will be working as there will be RFID reader stick on the bottom of the mobile robot and also there will be RFID cards on the floor near to the line that the robot will follow whenever the RFID reader reads the RFID tag it will perform a specific task such as turn right or turn left. RFID tag also used to define multiple locations which we will have four locations each location will have a specific RFID tag whenever the reader reads the RFID tag of the location that the user entered through the mobile APP the mobile robot will stop for a short time until the load has unloaded from the mobile robot after that the mobile robot will go back to the initial position.

3.4 Collision Avoidance

In order to avoid a collision that could obstruct the mobile robot on his way, an Ultrasonic sensor has been used. Ultrasonic sensor measures if there is any obstacle in front of the mobile robot and there will be two options to decide if the obstacle is far from the mobile robot then it will slow the mobile robot if its near it will stop the mobile robot from moving forward. By implementing this we will be able to make our mobile robot able to avoid any obstacles that may happen in the way of the mobile robot to its destination.

3.5 IoT Implementation to The Mobile App

To control our mobile robot remotely IoT will be used to interface a mobile App with the mobile robot. An open platform has been chosen to control the data from the App to the mobile robot we have choose Thingspeak as an IoT platform. The app will be designed through the MIT App inventor which is able to interface with hardware projects through internet [7]. this App will contain five buttons each button defines a specific location in the workspace and one button will be used to stop the mobile robot as there will be up to four locations. The app will also be able to show where the mobile robot is actually at. And it will send us a notification to update us whether the robot is already reaching the desired location or not. Moreover, there will be a label to communicate with the Thingspeak to show a notification from the Thingspeak platform whether the commands have been sent successfully or not. Therefore, if the commands sent to the Thingspeak from the App successfully it will give a notification stating that the command has been sent successfully. Nevertheless, if there was any error with the communication between the Thingspeak and the app we will receive a notification stating that there was an error with trying sending commands. There is another label will be used to show if the App is connected to the internet or not if it's connected it will show that its connected if it's not it will also show it's not connected in a label message somewhere inside the App.

4. Results and Discussion

This section discusses the results including the 3D design, fabrication of the parts, line following mechanism and interfacing the mobile robot with mobile App.

4.1 3D Design

The 3D design of the robot was successfully done and completed using the CAD programming software which is SOLIDWORKS as shown in Figure 4.1. The design is suitable for the mobile robot in the project and it consists of a double deck or two plates 300 mm length, 250 mm width and 10mm for the thickness of each plate, standoffs also were designed in 130mm as a length to hold and connect between the plates. Furthermore, two wheels were also designed to navigate the mobile robot from location to another. However, the designing also includes the roller-coaster, DC motor, a bracket to hold and support the DC motor to keep it stationary and a coupling to connect between the DC motor and the wheel. Figure 4.1 Shows the 3D design of the mobile robot. In addition, the mobile robot has been fabricated and print out into A 3D model as shown in Figure 4.2.

 DOI: https://doi.org/10.30564/jeisr.v1i1.1061

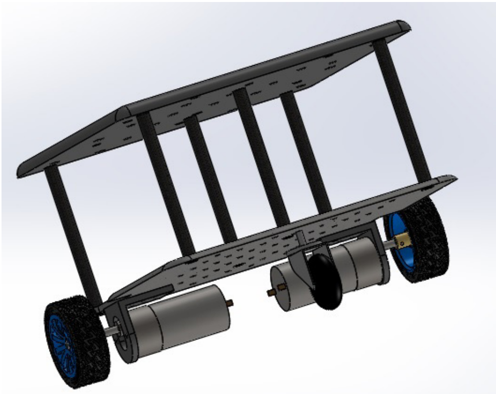


Figure 4.1. Designed Mobile robot

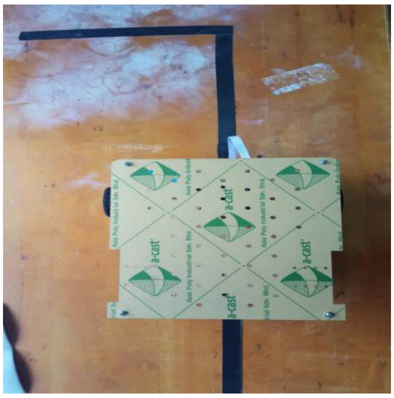


Figure 4.2. Fabricated mobile robot

4.2 Line Following Mechanism

Arduino UNO is used as the main controller to control the mobile robot in the workspace. The mobile robot is able to follow the line smoothly and pass all the junctions which are the + junction, T junction and wavy way using the line follower that contains of five IR sensors which are most right sensor, right sensor, middle sensor, right sensor, most right sensor. The line follower sensor will sense the black line in the floor if the middle sensor senses the black line the mobile robot will go forward by setting the DC motor's RPM to the same value. Nonetheless, if the left sensor senses the black line the mobile robot will move right by stopping the left motor and moving the right DC motor. For example, left DC motor RPM equals to the zero and right DC motor equals to two hundred by applying this the mobile robot will move right. Conversely, if the right sensor senses the black line the mobile robot will move left by stopping the right DC motor and moving the left DC motor. For example, we set the right DC motor equals to zero and the right DC motor equals to two hundred by applying this the robot will go left. There are other situations where two of the IR sensors are in the black line either the left sensor and the most left sensor with the middle sensor or the right sensor with the middle sensor are on the black line. In this situation the mobile robot will not be in the middle of the black line so this problem needs to be solved by playing with the RPM of the DC motors. For example, if the black line is not in the middle of the line we will turn diagonally the mobile robot either to the left or to the right by increasing the RPM of one DC motor and decreasing the RPM of the other DC motor by applying this the mobile robot will turn until it meets the ideal situation which is when the mobile robot is in the middle of the black line. In other words, until the middle IR sensor of the line follower sensor sense the black line. For the locations we have decided to use the RFID to control the decision either to turn completely right or left this has been done whenever the RFID reader reads the RFID card tag. The RFID card tags are located next to the line so whenever the RFID reader reads the RFID card tag it will decide which way will take based on the code. In our code location A, the mobile robot will follow the line until the RFID reads a specific RFID card tag and then it will turn right to location A once it reaches location A it will stop for ten seconds to unload the load then it will rotate and go back to the initial position. For location B the mobile robot will follow the line until the RFID reader reads the RFID card tag for location B then it will turn left ahead to location B once it reach location B the mobile robot will rotate and go back to the initial position and the same procedure for the other locations which are C and D. The mobile robot will keep going to any desired locations entered by the user through the app. Furthermore, it will stop if there is any obstacle in front of the mobile robot this has been done using ultrasonic sensor if the distance between the obstacle and the mobile robot is far the mobile robot will slowdown its speed by decreasing the DC motors RPM. Nevertheless, if the obstacle is near to the mobile robot the mobile robot will stop moving until this obstacle get removed. In Table 4.1 shows the result of the four trajectories of the line following mechanism

Table 4.1 Results of The Line Following Mechanism

No.	Trajectory	No. of trails succeed	Percentage	Figure
1.	L shape	10 out of 10	100%	

DOI: https://doi.org/10.30564/jeisr.v1i1.1061

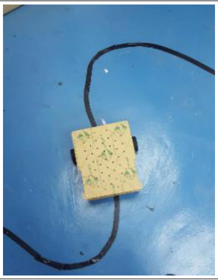
2.	S shape	9 out of 10	90%	
3.	Straight line	10 out of 10	100%	
4.	O shape	8 out of 10	80%	

Table 4.1 shows the result of the line following in four trajectories of an experiment of ten trails for each trajectory which are L, O, straight line and S shape. However, straight line and L shape got a 100 % accuracy. And S shape got 90% accuracy. Nevertheless, O shape got 80% accuracy. The mobile robot will be placed in initial position and if the mobile robot moves and came back to the initial position it will be success and miss of the line it will be fail even though if the mobile robot came back to catch up with the line.

4.3 Interfacing Mobile Robot with Mobile App

In order to navigate the mobile robot and smoothly let it follow the line to go to the desired locations. An IoT platform has been established which is the ThingSpeak to act as a data server in interfacing the mobile robot with the app. ThingSpeak will be receiving the data from the app and it will send it to the Arduino which is the main controller of the mobile robot. The mobile app has been also designed using MIT App inventor which will be sending commands to the mobile robot through the ThingSpeak. In the MIT app. Thus, there is five buttons first button define location A whenever this button pressed this button it will send a value of one to the ThingSpeak and then the Arduino will receive this coming value which is defines location A. However, the second button define location B once this button pressed it will send a value of two to the ThingSpeak and the Arduino will retrieve thins coming data which define location B. Moreover, the third button define location C once this button pressed a value of three will be sent to the ThingSpeak and Arduino will receive this data which is defines location C. Nonetheless, the fourth button define location D once the fourth button pressed a value of four will be sent to the ThingSpeak and then the Arduino will receive this data which is defines location D. Furthermore, the fifth button has been created to stop the mobile robot from moving once we press this button a value of zero will be sent to the ThingSpeak then the Arduino will receive this data which is defined to stop the mobile robot from moving whenever this button pressed. However, this app will also show us where the current location of the mobile robot is actually this location will be defined using the RFID. There will also be a label to show us the success of sending the commands if the commands have been sent successfully it will show us that the command has been send successfully is not it will show that there is an error with trying sending command. Another label has been created to show the status of the internet connection whether the app is connected to the internet or not if it is connected it will show that is connected if it is not connected it will show that it is not connected. Figure 4.4 Shows the features of the App. However, In Table 4.2 shows the results of the AGV interfaced with IoT.

Table 4.2 Results of the IoT Interfaced with the AGV

No.	LOCATION	Success of the move	Time travel per seconds
2.	LOCATION B	Success	4.6
3.	LOCATION C	Success	4.8
4.	LOCATION D	Fail	Fail
5.	LOCATION A	Success	5
6.	LOCATION B	Fail	Fail
7.	LOCATION C	Success	4.9
8.	LOCATION D	Success	4.8
9.	LOCATION A	Fail	Fail
10.	LOCATION B	Success	4.7

Table 4.2 shows the result of an experiment to send the mobile robot from the initial position to a predetermined location using the mobile App to send commands. However, ten commands have been sent from the mobile app as shown in Figure 4.4 to the mobile robot through IoT to navigate the mobile robot. Out of ten commands there are seven commands has been sent successfully and the mobile robot has gone to the desired location from its initial position and be back to its initial position in an average

 DOI: https://doi.org/10.30564/jeisr.v1i1.1061

time of 4.83 seconds. Furthermore, three out of the ten commands the command has not been sent successfully. Nevertheless, Figure 4.3 shows the Setup of AGV in order to navigate the mobile robot from one location to another. It shows the black line and the RFID cards along the black line.

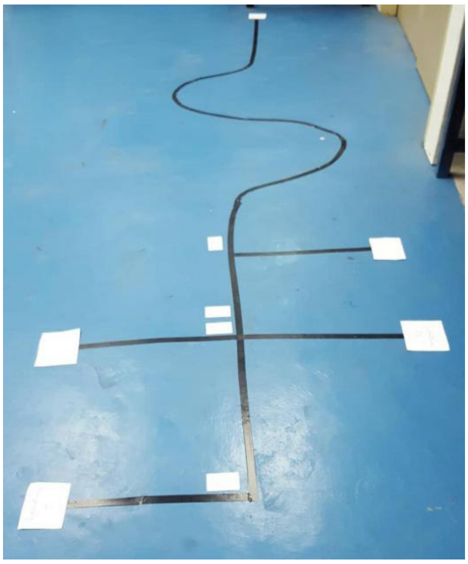


Figure 4.3 Setup for the mobile robot

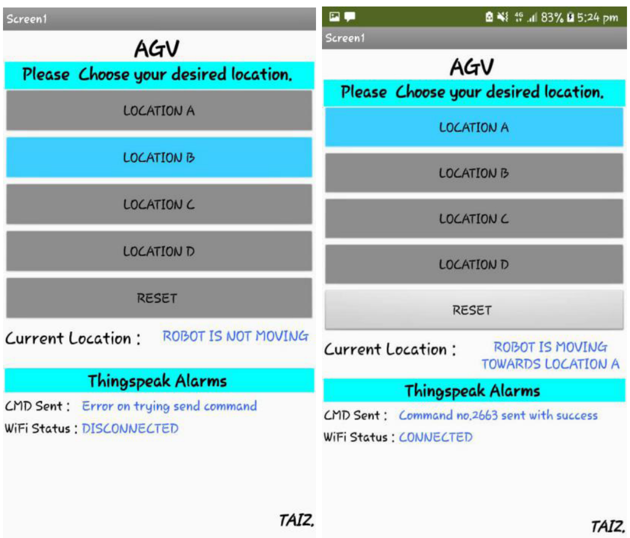


Figure 4.4 Mobile APP layout and Design

5. Conclusion

A mobile robot platform was designed and fabricated to replicate an AGV system. The robot was controlled to navigate from one location to another using line following mechanism with Internet of Things (IoT) based communication. RFID tags were used to identify the preregistered locations along the path. The results show that the robot was able to follow line and arrived at any location that was instructed from the Mobile App through the IoT communication. The mobile robot was also able to avoid collision and any obstacles that exist on its way during the operation. To improve the system, a laser scanner or Kinect camera can be added on top of the mobile robot to perform feature or facial recognition. This recommendation can help improve the obstacle avoidance and therefore provide a more reliable operation for AGV application.

References

[1] L. Tang and S. Yuta. Indoor navigation for mobile robots using memorized omni-directional images and robot's motion. Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst., no., 2002: 269–274.

[2] Y. K. Shejwal and S. K. Gera. Iot based automated guide vehicle. 2018: 3256–3259.

[3] R. Turbulence and D. N. Simulation. Introduction 1.1. no. Ddc, 1845: 1–12.

[4] I. Journal, O. Electrical, and D. Communication. Iot based smart car parking using line following robot 1, 2017.

[5] N. Hayat, D. Khushi, and M. Rashid. Algorithm for Line Follower Robots to Follow Critical Paths with Minimum Number of Sensors. Int. J. Comput., 2017, 24(1): 13–22.

[6] M. Z. Baharuddin, I. Z. Abidin, and S. S. K. Mohideen. Analysis of Line Sensor Configuration for the Advanced Line Follower Robot. Proc. Student Conf. Res. Dev. (SCOReD), Bangi, Selangor, Malaysia., 2006: 1–12.

[7] N. Lao. CloudDB: Components for exploring shared data with MIT app inventor. Proc. - 2017 IEEE Blocks Beyond Work. B B, 2017, 2017: 109–110.

DOI: https://doi.org/10.30564/jeisr.v1i1.1061