Virtual Path Navigation using Two Rotary Encoders for Automated Guided Vehicle

Jia Sheng Lee, Lik Zen Lai, Kim Seng Chia*
Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Batu Pahat, Johor, Malaysia
*corresponding authors: kschia@uthm.edu.my

Abstract – *The aim of this project is to evaluate the feasibility of the use of two rotary encoders in navigating an automated guided vehicle (AGV) in more complex virtual paths that include 90 and 180 degree turning from one to another desired positions. The proposed AGV was remotely controlled using a Bluetooth in a smart phone. The proposed AGV consisted of a microcontroller, two rotary encoders, an ultrasonic sensor, and a Bluetooth. The overshoot and* systematic *error was minimized using experiment data when the AGV was turning. The readings of the rotary encoders were mapped to the actual travel distance via calibration experiments so that the AGV can be redesigned to travel in different desired path without repeating the calibration experiments. Findings indicate that the proposed navigation method can navigate the AGV from one location to another location without installing any physical path or track.*

Keywords: *Automated Guided Vehicle, Rotary encoder, Navigation, Virtual path*

Article History
Received 19 January 2019
Received in revised form 11 February 2019
Accepted 11 February 2019 2019

I. Introduction

The automated guided vehicle (AGV) also known as robot vehicle is a vehicle that automatically guided along a guide path defined by wires which emit electromagnetic signals in most cases [1]. The AGV is designed to travel and stop at intended point, perform specific tasks. They are especially useful in serving processes where change is constant, and barriers such as conveyors are undesirable. However, an AGV is still ineffective to totally replace human being in the unexpected and complex workplace environment nowadays [2].

Today's trend has brought a truly accelerated pace. We are certainly living longer, but our daily lives are overloaded. Automating tasks is crucial for everyday office errands, greeting guests, providing directions, and assisting staff with administrative tasks in the workplace that will consequently increase production in business performance [3]. Therefore, it is necessary to render the lives of busy human more efficient and stress-free in today's fast-moving society.

AGVs were built with safety in mind. All other forms of equipment which are controlled by human such as forklifts which lead to human error and possible accidents [4]. However, the AGV applications programmed with full arrays of sensors and cameras can be stopped immediately when a potential hazard of the accident is about to take place. AGVs are mainly used for industrial purpose where it has adequate space and obstacle free to travel from place to place. When an AGV was designed to suit an office environment where obstacles are unexpected, safety system such as automated obstacle avoidance instead of stopping the operation is worthy to be developed to ensure the safety in the office.

An AGV can be built to save the manpower and the cost of an office in terms of runners. AGV can be summoned anytime without any fixed cost such as healthcare coverage, salary, vacation and other investment that are normally provided by companies to their employees [5]. Unlike human, robots will not be affected by emotion and aging, or at least not in the same way we are [6]. They will never have to worry about eating and they will be able to put themselves into a hibernation mode for long periods of time if necessary without any complain. An AGV has high potential to save time, human power and money [7]. Hence, automating task in an office will consequently improve efficiency on everyday office errands and release extra time for employee to focus on more meaningful tasks.

Even though line tracking AGV that used optical sensors or magnetic sensors have been widely implemented in various applications [7]-[9], the implementation of a line tracking AGV is infeasible in terms of changing the track or path for the future needs. Previously, an alternative to control an AGV using two rotary encoders shows promising results in navigating the AGV the move straight line without any track or path [10]. Thus, this study evaluates the feasibility of the use of two rotary encoders in navigating the AGV in more complex virtual paths that include 90 and 180 degree turning during navigation from one point to another.

II. Methods and Materials

A. Engineering Design

The main components of the proposed AGV are an Arduino Mega microcontroller, two rotary encoders, an ultrasonic sensor, and a Bluetooth.

B. Encoder Sensor Design

Fig. 1 shows the rotary encoders are installed at both sides of an AGV. Initially, the AGV was manually moved to calculate the number of generated pulses and related to the actual travel distance. The diameter between the two encoders is 50 cm. The purpose of using two encoders is to ensure the AGV can move straight line and turn 90 degree and 180 degree accurately. When one side encoder reading is more than the other side, one of the DC motors will increase the speed to eliminate the difference so that the position of the AGV is controlled accordingly. The encoder is very sensitive and can easily measures a small distance travelled by the AGV. The encoder reading will be reset to zero in every check point at where the AGV turns and changes position.

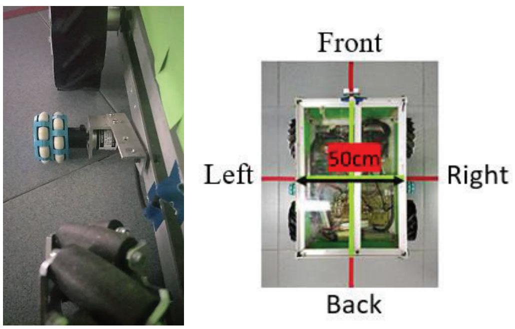


Fig. 1. Encoder sensor and top view of AGV

C. Ultrasonic Sensor

An ultrasonic sensor is placed in front of the AGV. This sensor is used as a warning switch to stop the AGV immediately when there is an emergency. When there is an obstacle in front of the AGV of about 15 cm, the AGV will immediately stop until the obstacle is moved or disappeared.

D. Pulse per Distance

Pulse per distance is experimentally estimated so that the AGV can navigate in the desired virtual path directly. In other words, the AGV can be programmed to move at any desired paths by mapping the distance of the desired path to the estimated pulse per distance in terms of an equation. The equation will be used to convert the measured distance to the number of pulses that the AGV needs to response. The pulse is generated by the encoders that are freely attached on the surface of the floor. The number of pulses is directly proportional to the travel distance. From the testing experiment, 12500 pulses were generated by each encoder when the AGV moves for a straight line of one meter. For the turning part, the degree of turning was calculated and compared with the one meter of 12500 pulses. The circle perimeter equation is used to calculate the degree of turning. For every step that is completed by the AGV, the encoder readings are set to zero to prevent accumulative errors.

E. Communication

The AGV is communicated by a Bluetooth system using a smartphone. The Bluetooth controller for the AGV is the existing software that is available and downloadable by any smart phone. This approach reduces the cost since the software is free and users can use their smartphone directly without additional device. The user only needs to press one button to command the AGV to move to the desired position. For example, if the user needs to move the AGV to position B as in Fig. 2, the user just needs to press the button B. Then, the AGV will automatically move to the desired position.

F. Error Computation

The testing criteria are based on the calculated rotary encoder values. The encoder difference after a test run is measured and displayed using the serial monitor of the Arduino IDE. The difference between the calculated value and the actual encoder values is the amount of the overshoot or error. The turning error is determined by capturing a picture on the top of the AGV and measured using the Onscreen Protractor java application. The encoder difference degree error is calculated by using equation (1).

$$\text{Error} = 2\pi r \times \frac{degree\ error}{360} \times 12500 \qquad (1)$$

The average error of encoder difference is calculated by using equation (2).

$$e_{ave} = \frac{\boldsymbol{overshoot + degree\ error\ (encoder\ difference)}}{\mathbf{2}} \qquad (2)$$

Encoder difference correction is calculated based on the difference between the calculated encoder difference and the average error. The corrected encoder difference is used in the programming to reduce the error. The readings of the encoders are calibrated to minimize the systematic errors along the path Point A to Point B.

G. AGV Performance Test

Fig. 2 shows the AGV that moves according to the desired virtual path (in red label) i.e. from Point A to Point B and back to Point A for 5 rounds on the track. The total distance from point A to point B is 205 cm (i.e. 50 cm + 130 cm + 25 cm) and the total travel path for the experiment is 410 cm.

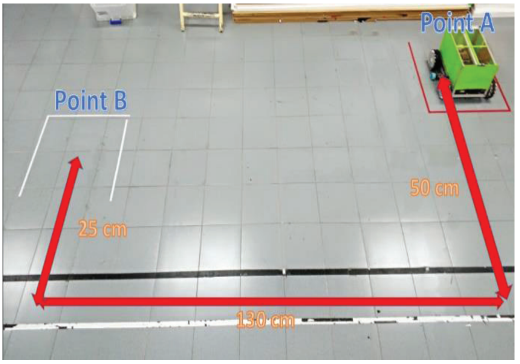


Fig. 2. The virtual path (labelled in red between point A and point B) that the AGV needs to travel

H. Turning Computation

Fig. 3 illustrates the desired position of the AGV when right and left turns are applied. The pulses for the encoder to turn 90 degree are calculated using the circle perimeter equation in which one-meter distance will produce 12500 pulses (from the experiment testing, 1 meter distance produce 12500 pulses). After the number of pulses was obtained for an encoder, the number of pulses different for two encoders which the AGV makes a right turn was calculated. The same concept was applied to the left turn of AGV.

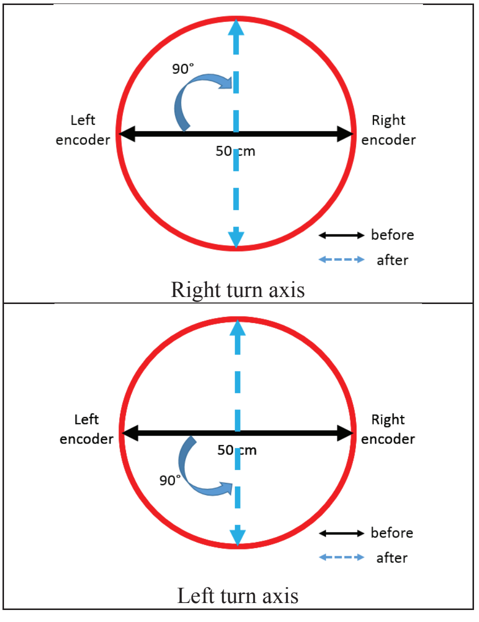


Fig. 3. The right turn and left turn axis for the AGV

The 90 degree right and left turns were estimated to generate 4908.74 pulses for each encoder, respectively. Thus, the difference between the two encoders to achieve 90 degree right or left turn was 9817 pulses.

A 180 degrees right turn was needed to turn the AGV to face to the back side. This can be achieved by multiplying the encoder difference for right turn by two, i.e. the encoder difference for the 180 degrees right turn was 19634 pulses.

III. Results and Discussion

A. Turning Test Discussion

During the turning test with the calculated encoder difference value, an obvious overshoot was observed. This could be due to the inertia of the AGV, i.e. the AGV was still in a moving motion although the motor was stopped. Besides, the friction force of the wheels was insufficient to stop the AGV immediately at the desired position without an overshoot or slip. This overshoot was reduced by deducting the calculated value with the average error that was estimated during the experiments.

B. AGV Performance Test Discussion

After the AGV was tested using the desired virtual path for five times, an average error of 12.6 cm was found. The cause of the error may come from the encoder readings or motor power failure. The encoder that was attached on the AGV touched on the surface of the floor freely without external force except the gravitational force. Consequently, the encoders might not firmly contact with the surface of the floor. This is because the encoders only depend on the gravitational force to pull the encoders to touch the surface of the floor. When the surface of the floor was not smooth or flat, the sensitivity of the encoders could be affected. Additionally, the encoders may be moving vertically sometimes and caused invalid readings.

Next, the motor torque of the left motor appeared greater than the right motor. This will cause the left motor to move slightly faster which caused the AGV to move to the right direction slightly if the control system was absent. Nevertheless, this concern could be eliminated by controlling the speed of the motors accordingly based on the external feedback readings of the encoders.

Fig. 4 (a) and (b) show the initial position when the AGV was positioned at the point A, and the first stop position when the AGV moved back from point B to the initial position of the point A (note: the picture was taken from the top view on position A), respectively. The error was calculated based on the difference between the center of point A and the center of AGV. Fig. 4 (c), (d), (e) and (f) illustrate the final position when the AGV was tested for the second, third, fourth and fifth rounds, respectively, in which, the AGV was from the point A moved to point B followed the desired virtual path, then returned and stopped at point A. The results indicated that the maximum error was 15 cm.

Fig. 5 summarized the test run performance that indicated that the maximum error was 15 cm while the minimum error was 7 cm when the AGV travelled from point A to point B, then returned to point A automatically. Visual inspection on Fig. 4 suggests that the systematic error was eliminated because the AGV stopped at different patterns.

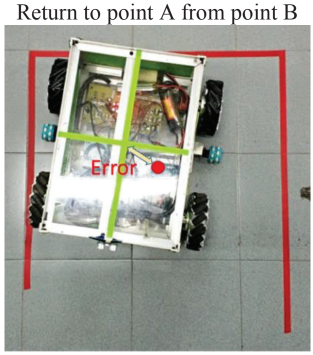
IV. Conclusion

The proposed automated guided vehicle (AGV) can receive user's command using a Bluetooth in a smart phone and detect an obstacle using an ultrasonic sensor successfully. The proposed AGV uses virtual path concept to avoid the need of a physical path so that no maintenance of the path is needed. The maximum error of the proposed AGV was 15 cm when it moved from the point A to point B, and then back to the point A by following the desired virtual path of total 410 cm travel distance. Nevertheless, the performance might be further improved by adding different types of sensors e.g. Rplidar and using control algorithms e.g. PID and Fuzzy controller. These potential enhancement initiatives shall be investigated in the future.

The AGV was initially placed in point A

(a) Initial point A

The first-round test run, Return to point A from point B

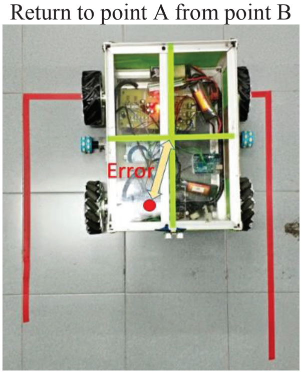


(b) Error after first round = 7cm

The second-round test run, Return to point A from point B

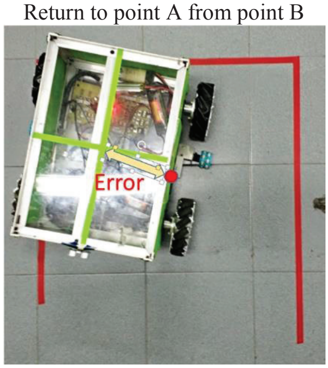
(c) Error after second round = 12cm

The third-round test run, Return to point A from point B



(d) Error after third round = 15cm

The fourth-round test run, Return to point A from point B



(e) Error after fourth round = 15cm

The fifth-round test run, Return to point A from point B

(f) Error after fifth round = 14cm

Fig. 4. The position of AGV – (a) the initial position at the point A, (b)-(f) the stop position after the AGV turned back from the point B to the point A.

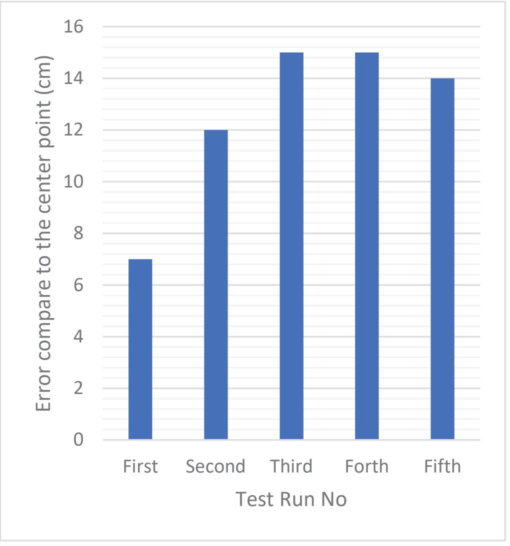


Fig. 5.Test Run error compare to center point when the AGV returned to point A from the point B

Acknowledgements

Authors would like to thank Xien Yin Yap and Jia Heng Ong who helped us to improve the AGV. Authors would like to acknowledge Faculty of Electrical and Electronic Engineering (FKEE), Universiti Tun Hussein Onn Malaysia (UTHM) for providing facilities and resources to this study.

References

[1] Allan S. MacKinnon, Donald J. Willemsen, David T. Hamilton, “Automated guided vehicle system.” United State patent US4530056A, Oct. 1982.

[2] C. Reads, “Current Automated Guided Vehicles (AGV) Are Ineffective,” no. January, 2018.

[3] R.R. Murphy, “Using Robot Competitions to Promote Intellectual Development,” AI Mag-azine, vol. 21, no. 1, 2000, pp. 77–90.

[4] T. B. Sheridan, “Human – Robot Interaction : Status and Challenges,” vol. 58, no. 4, pp. 525–532, 2016.

[5] M. C. Gombolay, C. Huang, and J. A. Shah, “Coordination of Human-Robot Teaming with Human Task Preferences,” 2015.

[6] Shukla Shubhendu and Jaiswal Vijay, “Applicability of Artificial Intelligence in Different Fields of Life,” International Journal of Scientific Engineering and Research, September 2013.

[7] S. K. Das and M. K. Pasan, “Design and Methodology of Automated Guided Vehicle-A Review,” no. June, 2016.

[8] K. S. Chia and Z. Nichols, “Ziegler-Nichols Based Proportional-Integral-Derivative Controller for a Line Tracking Robot,” vol. 9, no. 1, pp. 221–226, 2018.

[9] Suman Kumar Das, “Increase In Efficiency Using Pid Control Of An Automated Guided Vehicle For Product Ware House,” no. June, 2016.

[10] S. H. Lee and K. S. Chia, “Navigating an Auto Guided Vehicle using Rotary Encoders and Proportional Controller,” vol. 9, no. 2, pp. 71–77, 2017.