International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

Development of Line Follower Robot with Camera Surveillance System

W.H.M. Saad, H.R. Ramli, R. Marimuthu, S.A.A Karim, Z. Manap

Abstract: This study describes the development of a line follower robot for a surveillance camera monitoring system. An effective closed loop control fuzzy logic algorithm is used to constantly correct wrong movements of the mobile robot using a feedback mechanism. The robot senses a black line on a white surface and endeavors itself accordingly to follow the track. A manual navigation system has also been designed to overrule the automatic navigation control of the robot to reposition itself back on the track whenever it strays from the path unintentionally. The fuzzy controller algorithm is an advanced method to ensure the line follower robot moves accurately on the track. It is a replacement control technique of traditional switching method. To fuzzifying the digital input data of four infrared sensor that detecting the line, the data is converted into error and delta error that represent the current and previous position of the robot relative to the line that it follows. There are nine base rules that have been created with two inputs which are error and delta error to the robot direction whether to go to the right, move forward or to the left. Then, for defuzzification, center of sum and centroid of area method have been used to calculate the defuzzied value using trapezium area formulae. Based on the comparison between both control techniques, it is found that the line following surveillance robot with fuzzy logic controller works faster than conventional switching method to complete the same task with the average oscillation length using the fuzzy logic controller is reduced to half.

Keywords: Fuzzy logic controller, line following robot; oscillation length.

I. INTRODUCTION

In the current society, a surveillance camera was installed everywhere such as in the shopping malls, offices, hotels, parking lots, streets and so on. The main reason of installing a surveillance camera is to ensure the safety of the people surrounding the premise. Thus, a mobile surveillance camera is introduced with an intelligent line follower robot (LFR) which can follow the designated path while streaming live surrounding video to the user. The common problem that had been faced by the LFR is the smoothness of the maneuver while taking a sharp turn [1], [2]. For that reason, an effective algorithm must be implemented to ensure the robot moves on its path smoothly. In this study, Fuzzy logic controller (FLC) algorithm is chosen to replace the traditional switching method to overcome the problems. This switching method is also called Boolean logic control (BLC), on/off controller or binary logic controller [2]- [5].

The proposed LFR is using a webcam to stream the surveillance video on the PC using GUI in LabVIEW panel. The NI-myRIO is programmed using graphical programming method on LabVIEW software. Then, the Arduino UNO is used to design the FLC to process the signals from the IR sensor and give a feedback to the motor driver in order to constantly correct the wrong moves of the robot and to avoid getting out from the track. Fig. 1 illustrates the hardware block diagram of the LFR working system. Both myRIO and Arduino UNO microcontroller are used interchangeably due to some limitation of Arduino UNO microcontroller on processing the data from USB-webcam and the available motor driver of G15 is written for Arduino programming.

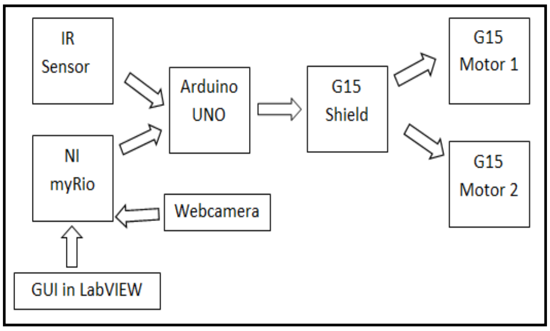

Fig. 1: Block diagram of LFR working principle.

A prototype is designed and tested to prove the validity of the FLC as shown in Fig. 2. The rest of the paper is organized as follows. Section II presents the methodology of the study on the robot development and FLC for LFR. In order to validate the effectiveness of the controller proposed, Section III provides the comparison of FLC and traditional method and finally, Section IV concludes the paper.

Manuscript published on November 30, 2019.
* Correspondence Author
W.H.M. Saad*, Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer, Centre for Telecommunication Research & Innovation Universiti Teknikal Malaysia Melaka, Durian Tunggal, Malaysia. Email: wira_yugi@utem.edu.my
H.R. Ramli, Faculty Engineering, University Putra Malaysia, Serdang, Selangor, Malaysia. Email: hrhr@upm.edu.my
R. Marimuthu, Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer, Universiti Teknikal Malaysia Melaka, Durian Tunggal, Malaysia. Email: ravin_ath@yahoo.com
S.A.A Karim, Department of Electrical Engineering, Politeknik Ibrahim Sultan, Pasir Gudang, Johor, Malaysia. Email: saifulanuar@pis.edu.my
Z. Manap, Fakulti Teknologi Kejuruteraan Elektri dan Elektronik, Universiti Teknikal Malaysia Melaka, Durian Tunggal, Malaysia. Email: zahariah@utem.edu.my

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

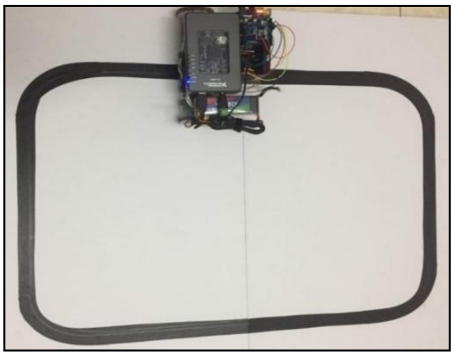

Fig. 2: Testing prototype on the track.

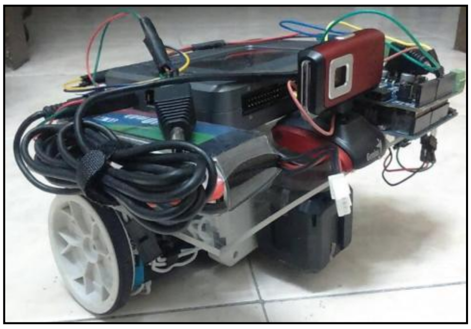

Fig. 3: The mobile robot prototype

II. MATERIAL AND METHOD

The major parts of the proposed robot monitoring system can be divided into four parts which are designing a basic structure of the robot, constructing the monitoring system, creating a manual navigation system and lastly is designing the FLC system for LFR mechanism.

A. Basic Structure of The Mobile Robot

Sturdiness of a robot structure is very important in order to ensure its good mobility. It can be a major factor that cause the robot to be shacking while following the line. The performance of the LFR is so much depending on how accurate and how fast the robot can move on its designated path [6]- [8]. Thus, sturdiness of the robot can be achieved by designing the structure with low center of gravity as can be seen in Fig. 3.

B. Camera Monitoring System

The LFR's movement can be remotely seen via the monitoring system on the PC. The webcam that had been placed at the top of the robot sent the visual view of the live video camera streaming to the user computer on LabVIEW GUI wirelessly through Wi-Fi. LabVIEW is known for graphical programming language as user need to create a function using block of function in virtual instruments (VIs) to make it work.

Fig. 4 shows the basic block diagram on VI function to capture and display image from the webcam and to broadcast it on a monitoring display. A webcam attached via a USB hub can also capture a still image and save it in a PC.

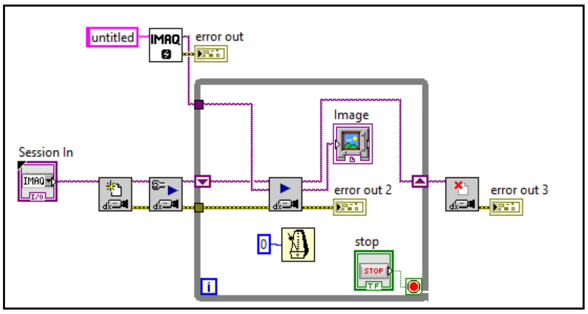

Fig. 4: VI arrangement to display the image from webcam

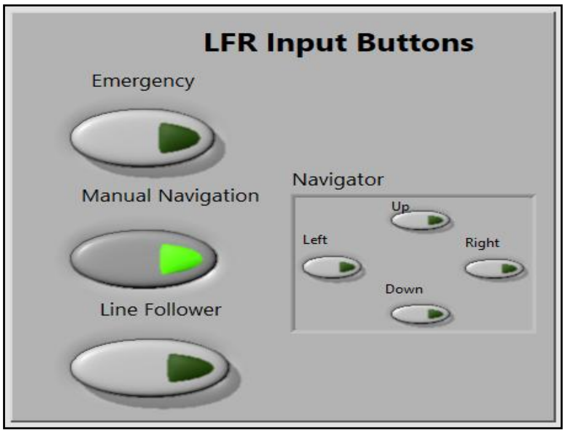

Fig. 5: LFR navigation selection mode

C. Manual navigation system

A manual navigation system has been developed as an emergency precaution for user to take over the automated navigation as user is able to control the robot manually whenever the LFR faces obstacles on its path or whenever the robot gets away from the track. Users can control the direction of the robot by pressing radio buttons on the PC. Four radio buttons like 'Left', 'Right', 'Forward' and 'Backward' have been created for the users so that they can control the LFR's directions as shown in Fig. 5. The manual navigation mode shall be turns on whenever the robot goes out of the track, thus, user can navigate the robot back to its original path. Then to continue its line navigation, user can push the 'Line Follower' button.

D. Manual navigation system

There are various algorithms that can be implemented for an LFR. The most popular ways are traditional binary logic or BLC, FLC and PID algorithms [9]. Each algorithm has its own advantages and disadvantages and it is totally depending on the application and situation of the control system.

Using the BLC causes a jerky movement even though the robot is on the straight path. This affects the speed of the robot. Therefore, in order to make a LFR has a smooth turn and accurate movement with a good velocity on the track, the BLC algorithm is replaced with FLC [10]. FLC is one of the modern feedback systems based on degrees of truth. This method works close enough with human's heuristic knowledge on how to control a system.

Fuzzifying the digital sensor data (0 and 1) is impossible.

Since the universe of discourse for FLC needs a range of values, a reference database is created which involves logical reflection of LFR movement at the time the infrared sensors sense the black line. All the possible detections of the sensors have been listed in Table-I and from that, a set point (SP) was decided. In this case, the total number of possible detections of four input sensors has eleven combinations and the SP are chosen to be at sensor position to be (0110). The value inside the table has been calculated using below formulas:

Table-I: Reference of IR sensor quantifying condition in term of value and error

LFR position	IR1	IR2	IR3	IR4	Value	Error	Last Error	ΔError
Slightly on the left side of the line	0	0	0	1	0.0909	-0.4545	-0.4545	0.0000 - 0.9090
	0	0	1	1	0.1818	-0.3636	-0.3636	-0.0909 - 0.8181
	0	0	1	0	0.2727	-0.2727	-0.2727	-0.1818 - 0.7272
	0	1	1	1	0.3636	-0.1818	-0.1818	-0.2727 - 0.6363
	0	1	0	1	0.4545	-0.0909	-0.0909	-0.3636 - 0.5454
Center set point	0	1	1	0	0.5455	0.0000	0.0000	-0.4545 - 0.4545
Slightly on the right side of the line	1	0	1	0	0.6364	0.0909	0.0909	-0.5454 - 0.3636
	1	1	1	0	0.7273	0.1818	0.1818	-0.6363 - 0.2727
	0	1	0	0	0.8182	0.2727	0.2727	-0.7272 - 0.1818
	1	1	0	0	0.9091	0.3636	0.3636	-0.8181 - 0.0909
	1	0	0	0	1.0000	0.4545	0.4545	-0.9090 - 0.0000

$$\text{Value} = \text{Current Detection}/\text{Number of Detection} \quad (1)$$
$$\text{Error} = \text{Current Value} - \text{SP} \quad (2)$$
$$\text{Last Error} = \text{Error} \quad (3)$$
$$\Delta\ \text{Minimum Error} = \text{Minimum Last Error} - \text{Error} \quad (4)$$
$$\Delta\ \text{Maximum Error} = \text{Maximum Last Error} - \text{Error} \quad (5)$$

There are four main components of FLC system as can be seen in Fig. 6. First is rule or knowledge base. The rule-base in Table-II gives the information to the FLC in the form of 'if-else' of rules set [4], [10]- [12]. This rule decides on how good FLC able to achieve the best output result.

A good rule should contain more than one input to give an effective output. For this line following system, there are nine rules that have been created and each rule has two inputs and one output. For example, "Error" and "Delta Error (ΔError)" are the two inputs direction represent the current and previous displacement of the robot position and the output is where the direction of the robot should go next [4].

Table-II: Reference database based on digital value

Rules	Input	Output
1	If Error is Negative AND ΔError is Negative	Right
2	If Error is Negative AND ΔError is Zero	Right
3	If Error is Negative AND ΔError is Positive	Right
4	If Error is Zero AND ΔError is Negative	Right
5	If Error is Zero AND ΔError is Zero	Forward
6	If Error is Zero AND ΔError is Positive	Left
7	If Error is Positive AND ΔError is Negative	Left
8	If Error is Positive AND ΔError is Zero	Left
9	If Error is Positive AND ΔError is Positive	Left

Next is a fuzzification process to modify the inputs as they can be interpreted and compared to the rules in the knowledge base. Usually, the inputs are in crisp values so that they will be converted into grades of membership for linguistic terms. As mentioned earlier, for both input to the FLC are described by three fuzzy sets such as Negative (N), Zero (Z) and Positive (P) and the output membership function is formulated as Left (L), Forward (F) and Right (R). For the output range, the universe of discourse angle is from -50^{o} to +50^{o}.

The "Error" and "ΔError" membership functions are defined as N, Z and P with triangle shape for zero and trapezoidal shape for negative and positive.

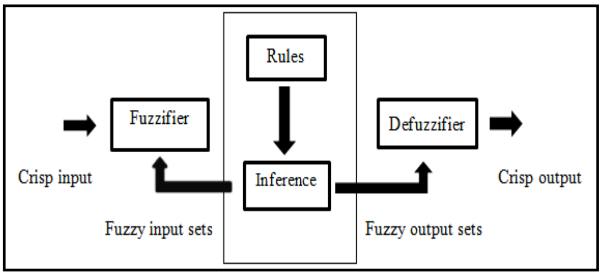

Fig. 6: Block diagram of FLC system

The range of inputs depends on the minimum and maximum values obtained from the Table-I. The range of input of "Error" is from -0.4545 to 0.4545 whereas the range of "ΔError" function is set from the value of -0.9090 to 0.9090. The output membership functions are designed for the motor direction with a range from -50 to 50. The membership functions of "Error", "ΔError" and "Motor Direction" are as shown in Fig. 7.

Then, the inference mechanism is used to evaluate the appropriate rules at relevant situation and decides what input should be added to the system.

Interference is becoming a medium between the knowledge base and the fuzzification. There are various implications method available to be used and among them, the "Mamdani" method has been chosen as it is the most reliable and easy to be implemented. The degree of fulfilment (DOF) of each rule is calculated using AND operator as shown in the Table-II.

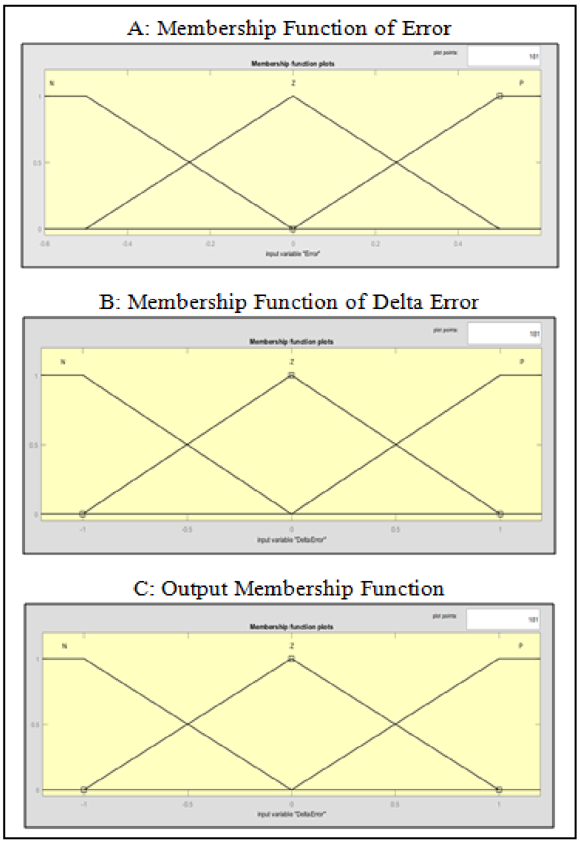

Fig. 7: Membership function of input and output variables

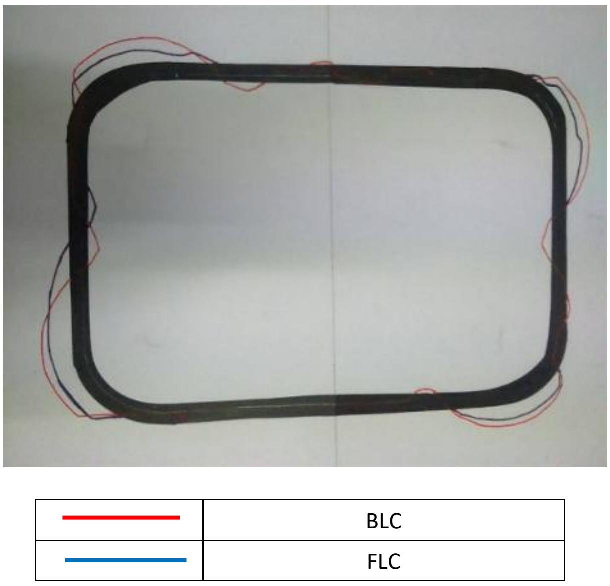

Fig. 8: Comparison between oscillation of FLC and BLC

Lastly, defuzzification process converts back the obtained result to the real-world inputs or values [11]. It transforms the fuzzified output into a crisp value with respect to the fuzzy sets that had been created. This value represents the action to be taken by the motors to ensure the robot is always on its path. There are many defuzzification methods that can be used such as Center of Sums (COS), Center of Gravity (COG) or Centroid of Area (COA), Bisector of Area Method (BOA), Weighted Average Method and Maxima Methods. In this study, COS and COA methods have been used to calculate the defuzzied value. The area of 'R', 'F' and 'L' region are calculated using trapezium formula and COA of them have been calculated in order to find the defuzzification value. All these calculations have been programmed into Arduino UNO.

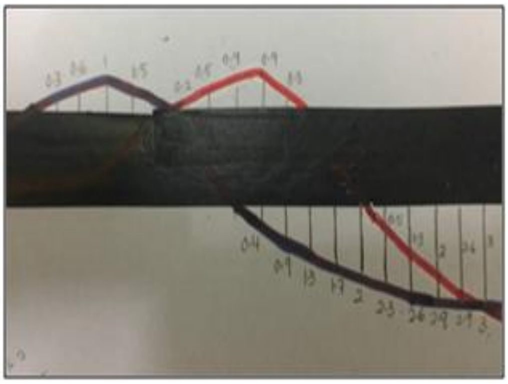

Fig. 9: Average length of discourse calculation from the oscillations of LFR

III. RESULTS AND DISCUSSION

The overall testing result of the fuzzy logic line follower robot is very promising. A simple track of the dark-line path surrounding with white background area was made for the testing purposes. The overall length of the track is around 2.45 m. The LFR moved very smoothly on the track without jerkiness whenever it passes curves and straight line. However, the limited membership functions make the robot struggles whenever facing sharp corners of 90 degree turns.

Table-III: Result of speed and oscillation comparison for both Fuzzy and binary methods

Parameters	**Type of line follower algorithm**	
	BLC	**FLC**
Time taken for one lapse	21.2 s	17.55 s
Average speed	$\frac{2.45\ \mathrm{m}}{21.20\ \mathrm{s}} = 0.1156\ \mathrm{ms}^{-1}$	$\frac{2.45\ \mathrm{m}}{17.55\ \mathrm{s}} = 0.1396\ \mathrm{ms}^{-1}$
Average length of the oscillation	0.0606 cm	0.0336 cm

The FLC is compared with the BLC on the same LFR. The output is as expected. Theoretically, fuzzy logic is an advanced closed-loop system than traditional on/off method [13]. Thus, it has more advantages than the binary logic controller. The comparison was made based on it speed to complete the track in one lap and oscillation of the LFR. The formula of speed calculation is as:

$$Average\ Speed\ (ms^{-1}) = \frac{\text{Distance (m)}}{\text{Time (s)}} \qquad (6.0)$$

First, the BLC algorithm has been applied to the LFR to obtain the output.

This method creates high jerkiness to the robot especially when following the curvy track. This situation affects the speed of the LFR to complete the lap. The BLC method of LFR finishes the circuit in 25.52s at best. In order to obtain the motion of the robot completing the track, a simple technique was implemented where a red color marker was set at the LFR caster wheel to mark the LFR movement accordingly.

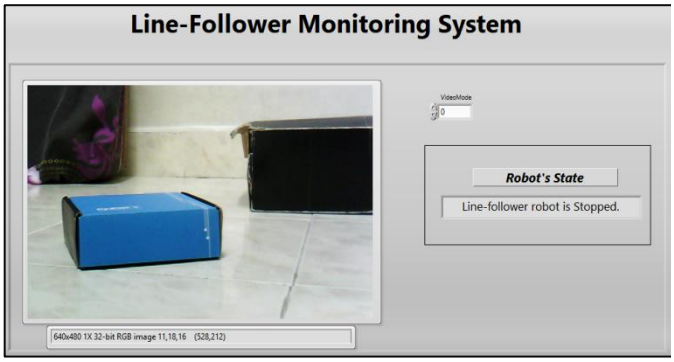

Fig. 10: Monitoring GUI in LabVIEW Front Panel on user PC

Then, the FLC algorithm has been applied to the same LFR. The time taken by the fuzzy logic LFR was around 19.85s. This means that FLC algorithm makes the robot to finish the lap by 5.67s faster than the BLC. For FLC of LFR, a blue color marker has been set at the caster wheel as the same position as red color maker previously. By comparing between FLC and BLC on the oscillation output of LFR as shown in Fig. 8, we can see that the number of oscillations of BLC is higher than the FLC algorithm. Even in straight path, there are some oscillations happen when the robot runs with BLC. This high number of oscillations can cause shaky monitoring on the surveillance video output where the users might possibly unable to monitor the camera view clearly. With FLC, the LFR only shows a slight oscillation when facing a curvy path. As for most of other situation, it moves smoothly without jerkiness especially on the straight line. Here, it clearly shows that FLC can produce less jerky motion, minimal left-right sway and faster speed than BLC method [14].

The comparison of oscillation for both techniques are measured by calculating the average discourse length of the LFR's oscillations from the desired track. Fig. 9 shows the measurement of oscillations distance in centimeters for each method. In order to calculate the average oscillation length of discourse, all the discourse lengths are measured in every centimeter of the track and the average value of it is obtained. The result shows that the average length of the oscillation of BLC was 0.0606 cm and in the meanwhile, the oscillation of FLC method was 0.0336 cm which is half of using BLC method. Thus, it is proven that the binary logic has higher oscillation's value compared to the advanced FLC method.

The speed of the LFR has been calculated for each fuzzy and binary logic approach. This actual speed is calculated based on the time recorded to complete the track in one loop. Table-III shows the result of the speed calculation. It conceivably implied that the performance of FLC is better than the BLC algorithm.

The example of monitoring system of the LFR can be seen in Fig. 10. This line follower monitoring system buffers the real-time images and videos through the Genius Slim 2020AF webcam which is connected to the NI-myRIO microcontroller USB port. This microcontroller is wirelessly connected to the PC which helps user to monitor the surrounding area from a distance.

IV. CONCLUSION

A fully functional prototype of a FLC-LFR surveillance monitoring system with additional manual navigation overtake is successfully designed and demonstrated. Mobile surveillance camera monitoring is very important where it gives higher coverage region than an ordinary surveillance camera. From the analysis of comparison between BLC method and FLC, it proves that, FLC is better than the traditional BLC for LFR with better speed and half of the oscillation on its navigation.

ACKNOWLEDGMENT

The authors would like to thank to the Machine Learning & Signal Processing (MLSP) research group under Center for Telecommunication Research and Innovation (CeTRI), Rehabilitation Engineering & Assistive Technology (REAT) research group under Center of Robotics & Industrial Automation (CeRIA) of Universiti Teknikal Malaysia Melaka (UTeM) and Faculty of Electronics and Computer Engineering (FKEKK), UTeM for the use of the existing facilities to complete this project.

REFERENCES

1. Ren, J., McIsaac, K. A., & Patel, R. V, "A fast algorithm for moving obstacle avoidance for vision-based mobile robots," in In Control Applications, 2005, CCA 2005, Proceedings of 2005 IEEE Conference on, Toronto, Ont., Canada., 2005.
2. Yang, X., M. Moallem, and R.V. Patel., "A Sensor-Based Navigation Algorithm for a Mobile Robot Using Fuzzy Logic," International Journal of Robotics and Automation, vol. 21, no. 2, p. 129–140, 2006.
3. Chang, M.-S., and J.-H. Chou., "A Novel Machine Vision-Based Mobile Robot Navigation System in an Unknown Environment," International Journal of Robotics and Automation, vol. 25, no. 4, p. 344–51, 4 2010.
4. Ismail, A. H., Zaman, A. A., & Terashima, K,, "Fuzzy logic approach for line following mobile robot using an array of digital sensors," International Journal of Application or Innovation in Engineering & Management, pp. pp. 108-115, 2016.
5. Engin M. &. E. D., "Path planning of line follower robot," in In Education and Research Conference (EDERC), 2012 5th European DSP, , Amsterdam, The Netherlands, 2012.
6. Bhagat, K., Deshmukh, S., Dhonde, S., & Ghag, S,, "Obstacle Avoidance Robot," International Journal of Science, Engineering and Technology Research (IJSETR), pp. pp. 439-442, 2016.
7. Cerqueira, T. A., Santos, T. L. M., & Conceição, A. G. S.,, "A New Approach Based in Potential Fields with Obstacles Avoidance for Mobile Robots," in 2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR), Recife, Brazil, 2016.
8. Pakdaman M. S. M. M. &. G. M. R., "A line follower robot from design to implementation: Technical issues and problems," in In Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on,, Singapore, 2010.

9. Sivaraj, D., Kandaswamy, A., Rajasekar, V., Sankarganesh, P. B., & Manikandan, G,, "Implementation of AVCS using Kalman Filter and PID Controller in Autonomous Self-Guided Vehicle," International Journal of Computer Applications, p. 2011, pp. 1-8.
10. Ibrahim, D., & Alshanableh, T,, "An undergraduate fuzzy logic control lab using a line following robot," Computer Applications in Engineering Education, pp. pp. 639-646, 2011.
11. Farooq, U., Amar, M., Asad, M. U., Abbas, G., & Hanif, A,, "Fuzzy logic reasoning system for line following robot," International Journal of Engineering and Technology, p. pp. 244, 2014.
12. Kuo C. H., "Development of a fuzzy logic wall following controller for steering mobile robots," in In Fuzzy Theory and Its Applications (iFUZZY), 2013 International Conference on, 2013, Taipei, Taiwan, 2013.
13. Saad, W. H. M., Karim, S. A. A., Azhar, N., Manap, Z., Yew, Y. S., & Ibrahim, M. I., "Line Follower Mobile Robot for Surveillance Camera Monitoring System," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), vol. 10, no. 2, pp. pp. 1-5–5, 4 July 2018.
14. Hasan, K. M., & Al Mamun, A,, "Implementation of autonomous line follower robot," in In Informatics, Electronics & Vision (ICIEV), 2012 International Conference on,, Dhaka, Bangladesh, 2012.
15. Ismail, Z. H., Naim, S., Ayob, A. F. and Amat, A. C,, "Mobile Robot Controller Based on Fuzzy Logic System in Uneven Terrain," Discovering Mathematics, vol. 39, no. 2, pp. 70 -82, 2017.

AUTHORS PROFILE

W. H M. Saad finished his degree in Bachelor of Electrical and Electronics Engineering from UPM, Malaysia in 2007 and he finished his PhD study in 2013 in the field of Multimedia Systems Engineering from the same university. Most of publication work done by him is in the field of embedded system, deep learning and image processing. He is now a senior lecturer at the Faculty of Electronics and Computer Engineering (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM).

H.R. Ramli works as a Senior Lecturer at Universiti Putra Malaysia, Department of Electrical and Electronic Engineering. His research interests are mainly in the area of biomedical engineering, image processing and robotics & Automation. He finishes his PhD from Imperial College London in the field of Biomedical/Medical Engineering on 2014

R. Marimuthu graduate in Bachelor of Electronics Engineering from Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka in 2017. His research interest is in automation and control system.

S.A.A Karim is working as a lecture in Department of Electric, Politeknik Ibrahim Sultan, Pasir Gudang, Johor. He previously graduate in Bachelor of Electric Power from Universiti Teknikal Malaysia Melaka in 2007. His research interest is in power management and control system.

Z. Manap is a senior lecturer in Universiti Teknikal Malaysia Melaka. Her interest is in routing, network simulation, network performance analysis and network performance evaluation.