❑   160

Design and implementation of line follower and obstacle detection robot

Ahmed Bendimrad, Ayoub El Amrani, Bouchta El Amrani
Laboratory of thin films and surface treatment by Plasma, Sidi Mohamed Ben Abbellah University, Morocco

Article Info	ABSTRACT
  	In this paper, we propose a method for a line follower robot based on the instantaneous computation of the radius of curvature of this line, using infrared line sensors. The number and layout of its sensors, as well as the method chosen, play an important role in the robot's response to the line, with the desired accuracy and speed. In addition, the robot must be equipped with an anti-collision system, using an ultrasonic distance sensor, to detect and avoid obstacles in several situations, especially at line crossovers, when other robots share a common complex line.

Corresponding Author:

Ahmed Bendimrad,
Laboratory of thin films and surface treatment by Plasma,
Sidi Mohamed Ben Abbellah University,
Fes 30050, Maroko.
Email: ahmed.bendimrad@usmba.ac.ma

1.   INTRODUCTION

A line follower robot is a robot designed to follow a line or a path already predetermined by the user. This type of robot is used mainly in semi to totally autonomous factories. In this environment, these robots function as carriers of materials to deliver products from one point of manufacture to another [1], but its use can be generalized in any field in our daily lives. In addition to the simple line tracking ability, this robot must also have the aptitude to navigate in a line crossovers and take the necessary steps to find the right path to follow after a crossover when the other paths are occupied by one or more obstacles [2, 3].

There are several techniques for line tracking robots", such as using a camera that takes instant images of the line [4], or using a light source with an LDR [5], or using IR sensors that are more accurate than LDRs and cheaper than a camera. We have therefore chosen to use digital infrared sensors for line detection.

Many systems based on IR sensors use discrete methods to follow a line, which can often generate a discontinuity or oscillation in the behavior of the line follower robot. Our proposed solution is therefore the use of a method based on the instant calculation of the radius of curvature of the line taking into account several geometric parameters as well as a time counter from the control system based on a microcontroller. For obstacle detection, an ultrasonic sensor mounted on a servomotor was used to scan an angle of 180 degrees and to be able to detect several obstacles, instead of using several ultrasonic sensors [6]. The robot can follow a black line on a white background. The robot is driven by two differential wheels driven by two geared motors [7-9]. These geared motors are controlled by the microcontroller after processing the information acquired by the various sensors.

2. PROTOTYPE TEST AND ESSENTIAL COMPONENTS

A line following robot usually consists of the following elements: Sensors (usually infrared and ultrasonic); Microcontroller or Microprocessor; Actuators (motors); Wheels; Chassis and Batteries [10].

Our study was tested on a test prototype designed for this research. This prototype is a mobile robot with two independent drive wheels, containing all the components necessary for its operation. The whole is protected by a hand-cut plexiglass frame, as shown in Figure 1

We used the following essential components:

Figure 1. Prototype test

2.1. Microcontroller

We chose as control system the Pic 16f877A microcontroller to acquire the information of the different sensors via their digital or analog inputs [11], to do the processing and the programmed calculations, and to deliver via their digital outputs the commands to be executed by the motors in Pulse Width Modulation (PWM) [12, 13]. We used the timer2 of this microcontroller for the time calculation by generating an interrupt at every 80 μs.

2.2. Sensors
2.2.1. Infrared ray sensors

The distinction between the black and white tracks was based on reflection infrared sensors, these sensors consist of an infrared emitting LED that acts as an emitter and an infrared sensitive phototransistor that acts as a receiver, and these two components are placed side by side in a holder. When this sensor is in front of a white track, the infrared beam is strongly reflected and the phototransistor is then satured. When this sensor is in front of a black track, the infrared beam is practically not reflected and the phototransistor is then blocked [14]. There is a wide variety of IR sensors, but the CNY70 was chosen directly because of its small size (7mm x 7mm x 6mm) and the accuracy required by the proposed principle of following a line [15].

2.2.2. Ultrasonic sensor

This type of sensor is based on the use of ultrasound to measure the distance to an obstacle without contact with the latter. In an ultrasonic sensor, the transmitter and the receiver are located in the same housing. The transmitter sends a train of waves that will reflect on the object to be detected and then return to the source. The time taken to go back and forth makes it possible to determine the distance of the object from the source. The ultrasound sensor we chose is the HC-SR04 [16, 17], which has the advantage of being simple to use and to have an excellent non-contact detection range (from 2 cm to 4m), with high measurements precision and stable. Its operation is not influenced by the light or the color of the obstacles to be detected.

2.2.3. Speed sensor

This sensor consists of an IR transmitter, and an IR receiver, between which is placed a transmissive rotary Encoder Disk coupled to the axis of rotation of the wheel [18]. When the wheel rotates with an angular velocity ω, the infrared beam is interrupted by the Disk, with a frequency that depends on the number of slit of the disk and the angular velocity of the wheel. Thus, an electrical pulse of the same frequency is interpreted by the microcontroller which calculates the angular velocity ω. The speed sensor we have chosen is based on the comparator LM393 which delivers a digital signal which can be processed more easily by the microcontroller.

2.3. Motor driver and motors

The power circuits are generally formed by transistors of high current gain or a group of a number of these transistors in integrated circuits (transistor bridge). For this robot, we will use two geared motors for the motorization of the wheels, and a servomotor for the radar obstacle detector. The power circuit adapted to the two gear motors is the L298N [19], it contains two transistor bridges to control two motors in both directions with a maximum current of 1A per motor. As for the servomotor, it must likewise have a separate power circuit. The circuit chosen for this is the L293D [20, 21], it also contains two bridges of transistors and works with the same principle, but only delivering 0.6A per channel which is sufficient for the used servomotor which consumes 0.16A empty and hardly undergoes load [22].

2.4. Power supply and voltage regulators

For the power supply, we chose to use a single source of energy for all the assembly of the robot. It consists of three series-connected 18650 7800 mAh 3.7V rechargeable Li-Ion Batteries, which forms a single source of power with a capacity of 7800 mAh and a voltage of 11.1 V. The electronic robot assembly is divided into two parts, a part containing the control board, the sensors and the LEDs and operates with a voltage of 5V, and a part containing the motors with their drivers that operate with a voltage of 6V. We will then need to set the value of the voltage at 5V on the one hand, and 6V on the other hand during the operation of the robot from the voltage source (11.1 V). For this, the positive voltage regulators 7805 and 7806 are used, the last two digits represent the output voltage.

2.5. Programming language

There are several languages of programming with which one can program a microcontroller (Assembler, C, Pascal …), one chose the language C because it is simpler, more comprehensible, it can be easily modifiable by using functions, and in plus there are several software microcontrollers programming using this language as MPLAB and MIKROC PRO FOR PIC.

2.6. Programmation software

The programming software chosen is MIKROC PRO FOR PIC. It is easy to use with a clear interface and containing several predefined tools and functions, which facilitates programming without going into the complications of the manual settings of some microcontroller registers, such as serial communication functions, pulse width modulation, the ASCII code conversion tool and the hexadecimal and decimal binary numbers.

3.   LINE FOLLOWER

3.1. Calculation of radius of curvature of the line

For the calculation of the radius of curvature of the line to be followed, a method based on the geometry of the line, the distribution of three infrared sensors CNY70, and the position of each of these sensors with respect to the line as indicated in the Figure 2.

We chose a black line on a white ground, so the infrared sensor returns "0" when it is on the black line, and "1" when it is on the white field.

The following parameters were used (Figure 2):

a.   l: Half-width of the line to follow.
b.   k: Half of the distance between the IR sensor number 1, and the IR sensor number 2.
c.   h: Distance between the IR sensor number 3, and the line passing through the IR sensors numbers 1 and 2.
d.   d: Distance traveled linearly by the robot between the instant of change of state of the IR sensor number 3, and the moment of change of state of the IR sensor number 1 or 2 (or vice versa).
e.   x: Distance between point A where the IR number 3 sensor leaves the line, and the line through the IR number 1 and 2 sensors.
f.   e: Distance between the point M where the change of curvature begins, and the line passing through the IR sensors numbers 1 and 2.
g.   s: Distance between the axis of rotation of the two driving wheels, and the line passing through the IR sensors numbers 1 and 2.
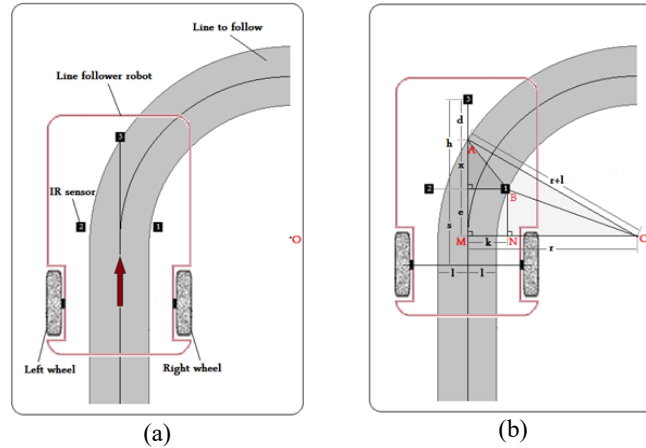
Figure 2. Sensors layout and method of calculating radius of curvature. a: Step 1, b: Step 2

The idea we have had is to express the radius of curvature r as a function of x, taking l and k as parameters.

We consider the triangle (OAM) rectangle in M, and we apply the theorem of Pythagoras, where:

$$e^2 = l^2 - x^2 + 2rl - 2ex$$

and we consider the triangle (OBN) rectangle in N, and we apply the theorem of Pythagoras, where:

$$e^2 = l^2 - k^2 - 2rl + 2rk$$

So :

$$e = \frac{2r(2l - k) + (k^2 - x^2)}{2x}$$

we conclude:

$$4(2l - k)^2 r^2 + (4(2l - k)(k^2 - x^2) + 8x^2(l - k))r + (k^2 - x^2)2 - 4x^2(l^2 - k^2) = 0$$

it is a second order equation whose solution accepted by our method for $k \neq 2l$ is:

$$r(x) = \frac{-AB + (A - 2C)x^2 + 2x\sqrt{C}\sqrt{(C - A)x^2 + AB + DA^2}}{2A^2}$$

with:

$$A = 2l - k \; ; B = k^2 \; ; C = l - k \; ; D = l + k$$

Since the parameters l and k are fixed during the same experiment, the radius of curvature r then depends on a single variable x.

If we take the 12.5 mm (half-width of black tape), and if we take each time a value of k such that $k > l$, we can see the effect of the parameter k on the behavior of (x) for $x > 0$ as shown by its graphical representation in the Figure 3.
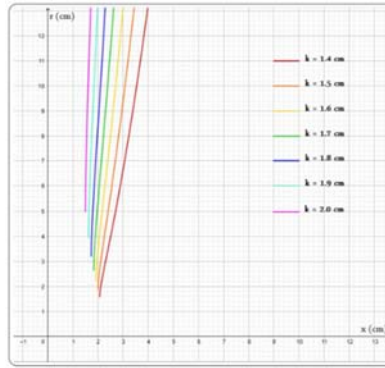
Figure 3. Graphical representation of r (x) for different values of the parameter k

From Figure 2, the measurement of the radius of curvature becomes less precise when the parameter k increases. For example, r(x) varies between 6.5 cm and 13 cm for a variation of x equal to 1 cm if k = 1.4cm, and 0.2 cm if k = 2 cm. Then we will choose the value 1.4 cm for the parameter k, and the expression of r (x) is simplified.

$$r(x) = \frac{0.7x^2 - 1.08 + x\sqrt{0.19x^2 - 0.8}}{1.21}$$

The value of x is determined from the IR sensors (Figure 1) and the microcontroller timer according to three cases:

a.   If the number 3 IR sensor leaves the line before the IR sensor number 1 or 2 detects it, then: $x = h - d$
b.   If the IR sensor number 1 or 2 detects the line before IR sensor number 3 leaves it, then: $x = h + d$
c.   If the number 3 IR sensor leaves the line at the same time as the number 1 and 2 IR sensors detect it, then: $x = h$.

The value of d is determined from the infrared encoder speed sensors installed on the two driving wheels, and from the microcontroller timer according to the relation: $d = v.t$

a. v: The speed of the robot.
b. t: the time between the moment the IR sensor number 3 changes from "0" to "1", and the moment the IR sensor number 1 or 2 goes from "1" to "0".

3.2. Variation of the direction according to the radius of curvature

As the model of the robot used is unicycle, the variation of orientation is done by the difference of angular speed between its two drive wheels. This difference must then take into account the radius of curvature (x) of the line and the robot architecture as shown in Figure 4.
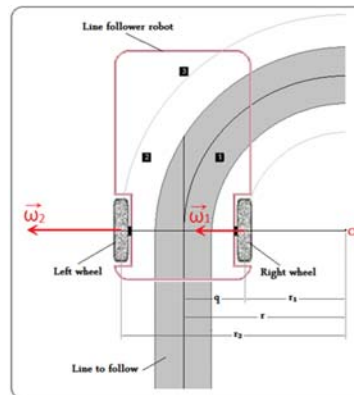

Figure 4. Geometric parameters of robot and line to follow

We will need the following parameters:
a. q: Half of the distance between the two drive wheels.
b. r: Curvature radius from the middle of the line to follow.
c. r_1: Curvature radius concerning the right wheel.
d. r_2: Radius of curvature concerning the left wheel.
e. ω_1: Angular velocity of the right wheel.
f. ω_2: Angular velocity of the left wheel.

If the robot follows the line, then each wheel draw on the plane a circle of radius that depends on r (x), and q.

$$r_1 = r - q$$
$$r_2 = r + q$$

and since both driving wheels have the same radius, we can demonstrate that:

$$\frac{\omega_1}{\omega_2} = \frac{r_1}{r_2}$$

so :

$$\frac{\omega_1}{\omega_2} = \frac{r - q}{r + q}$$

3.3. PID regulation of angular velocity

The method proposed to follow a line is theoretically effective if we suppose that the response of the engines to the commands sent by the microcontroller is instantaneous. But this is not the case in reality, because the robot has an inertia that delays it to take the angular velocity calculated and sent by the microcontroller. Hence the need to introduce a PID regulator to compensate for speed errors as quickly as possible [23, 24]. The Figure 5 shows the block diagram of this regulator [25].
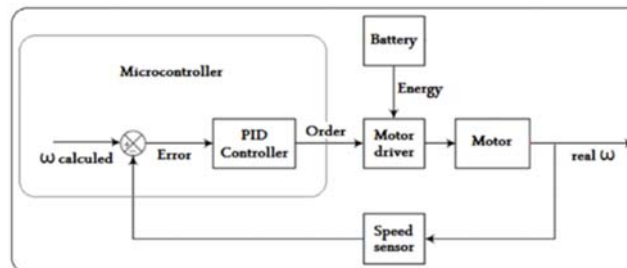

Figure 5. Functional diagram of the operating principle of the PID regulation of ω

A parallel PID controller was used in the form of a C language program that was introduced into the microcontroller.

4.   AVOIDING OBSTACLES
4.1. Distance determination with ultrasound

To measure a distance with ultrasound, one must first know their speed of movement. It is of the order of 340 meters per second, so it is easy to determine the distance traveled by the ultrasonic wave if we know the time it took to travel the same distance.

The ultrasonic distance sensor uses the echo principle to determine how far an object is:
a.   A short ultrasound signal of 40 kHz frequency is sent by the transmitter.
b.   This signal is reflected by the surface of an obstacle at the distance do of the sensor.
c.   The reflected signal is detected by the receiver after traveling the same distance do.

The wave must travel twice the distance between the ultrasonic sensor and the obstacle, hence the distance:

$$d_0 = \frac{v_s \cdot t_0}{2}$$

a. t_0: Time between transmission and reception.
b. d_0: Distance between source and obstacle.
c. v_s: Velocity of ultrasound movement in the area (340 m / s).

4.2. Behavior with respect to line crossover in the presence of obstacles

To detect line crossover, three IR sensors were added as in Figure 6.

An ultrasonic sensor mounted on a servomotor was used to scan an angle of 180 degrees and to be able to detect several obstacles.

When an obstacle is detected just after a crossover, the robot stops on it and the ultrasonic sensor is directed by the servo motor in search of a free path to follow it. If he doesn't find any more, the robot stays arrested.
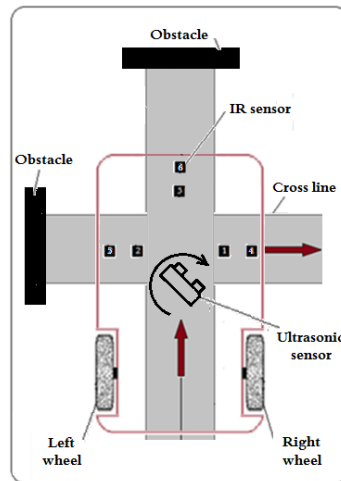
Figure 6. Choice of the free line on a line crossover

5. RESULTS AND ANALYSIS

The experimentation of the proposed method on the test prototype gave satisfactory results after several tests and adjustments of some parameters, such as the distance h (Figure 2.b) and the parameters of the PID controller. The positioning accuracy of IR sensors plays an important role in optimizing robot performance. To do this, the IR sensors were mounted in the robot chassis in such a way that they could slide linearly, so that they could change their location for calibration as presented in Figure 6.
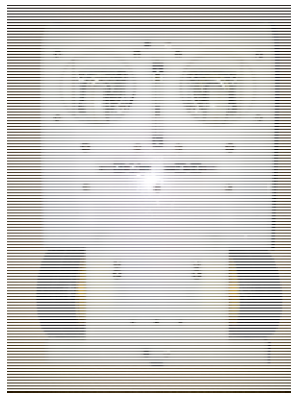
Figure 6. test prototype seen from above

The stabilization of the robot on the line is improved at low speed, but at high speed the stabilization is limited by the response time of the microcontroller used, so there is a speed threshold not to be exceeded to adhere to the line. This speed threshold is all the higher as the microcontroller is powerful.

6. CONCLUSIONS

Our objective was to design a line follower and obstacle detector robot designed for autonomous navigation along a black line on white ground, using a technique based on the calculation of the curvature radius of the line, avoiding any obstacles that may exist on this line. in the event of a line crossover, the robot is able to choose the free line among others that are occupied by obstacles. the behaviour of the test prototype was satisfactory at low speed. The addition of a distance sensor at the rear to avoid obstacles in the opposite direction would be an important addition to this robot, in addition to other improvements in general.

REFERENCES

[1] M. Zafri Baharuddin, Izham Z. Abidin, S. Sulaiman Kaja Mohideen, Yap Keem Siah, and Jeffrey Tan Too Chuan, *Analysis of Line Sensor Configuration for the Advanced Line Follower Robot*, Universiti Tenaga Nasional, 2006.

[2] Pushkar Goel, Geetika Arora, and V.K. Panchal, "Incorporating Perception Radius to the Line Follower Robot", *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*, pp. 1-5, 2014.

[3] Surya prakash.M, Ajay Vignesh.K, Shyamsunthar.J, Raman.K, Senthil Raju.J, and Raju.N, "Computer Vision Assisted Line Following Robot", *International Conference on Modeling, Optimization and Computing*, 2012.

[4] Larry Andrés Pérez Sánchez, Cesar Augusto Aceros Moreno, and Holguer Andrés Becerra Daza, "Design and construction of a line follower robot guided by pixels values of a camera connected to an FPGA", *2015 20th Symposium on Signal Processing, Images and Computer Vision (STSIVA)*, pp. 1-5, 2015.

[5] Kazi Mahmud Hasan, Abdullah-Al-Nahid, and Abdullah Al Mamun, "Implementation of autonomous line follower robot," *IEEE/OSA/IAPR International Conference on Informatics, Electronics & Vision*, pp. 865-869, 2012.

[6] Hasan U. Zaman, Md. Majidul Haque Bhuiyan, Montashir Ahmed, and S.M Tarek Aziz, "A Novel Design of Line Following Robot With Multifarious Function ability," *IEEE International Conference on Microelectronics, Computing and Communications (MicroCom)*, 2016.

[7] Fatima R.Ali and AbdulmuttalibT. Rashid, "Design and implementation of static and dynamic objects store systems using line follower robots," *International Conference on Advances in Sustainable Engineering and its Applications*, pp. 37-42, 2018.

[8] Michal Kelemen, Daniel John Colville, Tatiana Kelemenová, Ivan Virgala, and Ľubica Miková, "*Educational Model of Line Follower Robot LINA 2010*," *Solid State Phenomena Vols 220-221*, pp. 989-994, 2015.

[9] Siti Nur Hanisah Umar, Elmi Abu Bakar, Mohd Shukri Soaid, and Zahurin Samad, "Study on multi tasks of line following differential wheeled mobile robot in-class project," *Int. J. Modelling, Identification and Control*, vol. 21(1), 2014.

[10] Vítor Gabriel Reis Caitité, Diego Maradona Gonçalves dos Santos, Iago Conceição Gregório, Willian Braga da Silva, and Victor Flores Mendes,"Diffusion of Robotics through Line Follower Robots," *2018 Latin American Robotic Symposium, Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, pp. 604-609, 2018.

[11] Md. Badruduja Bhuiya, "Controlling Line Follower Robot with the Remote Web Server," *Proceedings of SAI Intelligent Systems Conference (IntelliSys)*, pp 559-583, 2016.

[12] F Kaiser, S Islam, W Imran, K H Khan, and K M A Islam, "Line follower robot: Fabrication and accuracy measurement by data acquisition," *International Conference on Electrical Engineering and Information & Communication Technology*, pp. 1-6, 2014.

[13] Seyed Ehsan Marjani Bajestani and Arsham Vosoughinia, "Technical Report of Building a Line Follower Robot," *IEEE International Conference on Electronics and Information Engineering*, 2010.

[14] Snehal Kokare, Rajveer Shastri, and Shrikrishna Kolhar, "Line Follower with Obstacle Information System Using ZigBee," *Fourth International Conference on Computing Communication Control and Automation*, pp. 1-5, 2018.

[15] Mehran Pakdaman and M. Mehdi Sanaatiyan, "Design and Implementation of Line Follower Robot," *2009 Second International Conference on Computer and Electrical Engineering*, vol. 2, pp. 585-590, 2009.

[16] Fazal Noor, Mohammed Swaied, Moath AlMesned, and Nasser AlMuzini, "A Method to Detect Object's Width with Ultrasonic Sensor," *IEEE International Conference on Computing, Electronics & Communications Engineering*, 2018.

[17] Sun Zhi, Yang Yi, and Wang Zhi-sheng, "Design of distance measuring and reversing system," *IEEE 3rd Advanced Information Technology, Electric and Automation Control Conference*, 2018.

[18] Nurul Hasanah, Ali Husein Alasiry, and Bambang Sumantri, "Two Wheels Line Following Balancing Robot Control using Fuzzy Logic and PID on Sloping Surface," *International Electronics Symposium on Engineering Technology and Applications*, 2018.

[19] Mohd Zikrul Hakim bin Noor, Mohd Syafiq bin Mohd Zainuddin, Muhammad Farid bin Saaid, and Megat Syahirul Amin bin Megat Ali, "Design and Conceptual Development of a Sunbathe Laundry Robot," *IEEE 8th International Colloquium on Signal Processing and its Applications*, 2012.

[20] Kunal Khade, Revati Naik, and Amey Patil, "Design of all color line follower sensor with auto calibration ability," *IEEE 7th International Symposium on Embedded Computing and System Design*, 2017.

[21] F Kaiser, S Islam, W Imran, K H Khan, and K M A Islam, "Line Follower Robot: Fabrication and accuracy measurement by data acquisition," *IEEE International Conference on Electrical Engineering and Information & Communication Technology*, 2014.

[22] Mustafa Engin and Dilsad Engin, "Path planning of line follower robot," *5th European DSP Education and Research Conference (EDERC)*, pp. 1-5, 2012.

[23] Z. Ul Abideen, M. Bilal Anwar, and Hassan Tariq, "Dual Purpose Cartesian Infrared Sensor Array Based PID Controlled Line Follower Robot for Medical Applications," *IEEE International Conference on Electrical Engineering*, 2018.

[24] Ishraq Ul Haque, Abul Al Arabi, Shadat Hossain, Tanjina Proma, Nafi Uzzaman, and M Ashraful Amin, "Vision Based Trajectory Following Robot and Swarm," *IEEE 2nd International Conference on Control and Robotics Engineering*, 2017.

[25] Alexia Toumpa, Alexandros Kouris, Fotios Dimeas, and Nikos Aspragathos, "Control of a line following robot based on FSM estimation," *University of Patras*, vol. 51, pp. 542-547, Greece, 2018.