

DESIGN AND IMPLEMENTATION OF AN AUTONOMOUS CAR FOR NAVIGATION IN UNKNOWN ENVIRONMENT

Maaz Tahir, Fawad Wali

Department of Electrical Engineering

Muhammad Adnan, Nabeel Kamal

Department of Mechanical Engineering

maaztahir79@yahoo.com, adnan.chaudhry@outlook.com,

fawadwali67@ee.ceme.edu.pk, nabeel.kamal@seecs.edu.pk

NUST College of Electrical & Mechanical Engineering, Peshawar Road,

Rawalpindi, Pakistan

Imran Shafi

Abasyn University Islamabad campus, No. 210, Street# 6, I-9/2 Islamabad,

Pakistan

imran.shafi@abasyn.edu.pk

Muhammad Zeeshan, Muhammad Salman,

Department of Electrical Engineering, University of the Punjab, Lahore-54590,

Pakistan.

Email: zeshan_ceet@yahoo.com, msb709@hotmail.com,

Abstract—The paper describes the design and development of an intelligent autonomous vehicle that can navigate without human operator in unknown environments. We have interfaced to AT89C52 microcontroller with the ultrasonic sensors and GPS receiver for easy access to the destination by avoiding the obstacles in its way at low level. A camera is interfaced to pc for lanes' identification and road following task at higher level. The designed vehicle has modular structure and more modules can be added without much complexity. The experimental results demonstrate the effectiveness of the proposed design.

Keywords—autonomous vehicle; ultrasonic sensors; gps receiver; AT89C52 microcontroller ; MATLAB; fuzzy controller

I. INTRODUCTION

Autonomous vehicles are being deployed in environments which are considered hazardous for humans to work in. In this paper, design of an autonomous car is presented that can navigate in such environments without human intervention. The heart of the car is the Fuzzy Logic Controller that is used to handle the imprecise data from the sensors. The operation of the car is divided into two behaviors namely hurdle avoidance and lane following. The hurdle avoidance behavior is responsible to avoid obstacles in the path of the car and utilizes two ultrasonic sensors with Fuzzy Logic Controller to generate steering and speed commands for the motors. The lane following behavior is meant for driving the car on roads. This behavior also generates speed and steering commands for keeping the car within the lane. The car is tested in both indoor and outdoor environments and satisfactory results are obtained.

An autonomous vehicle is intelligent enough to reach desired destination safe and sound. So it can be used as a navigation aid for disabled which is one of its major contribution towards the welfare of humanity. This technique can be implemented in real vehicles also for assisting the drivers especially in foggy weather conditions.

A new kind of autonomous transport named bi-steerable car is described in [1]. Its primary focus is to increase the safety in dynamic environment zone. Bi-steerable car has two steering controls provided at the front and the rear axle respectively. The environment is observed closely to map the path and then the vehicle moves on the planned trajectory by avoiding obstacles. Probabilistic techniques are used for the synthesis of obstacle avoidance and the control commands for the motion generation of the car.

Navigation problem of computing the complete trajectory towards the destination has been consistent with the car like robots. A partial motion planning scheme (PMP) is presented in [2] as a solution towards this problem. The state-time space is iteratively generated by the PMP algorithm based on the incremental approach. The safety concerns are solved through the mechanism of inevitable collision states (ICS). Continuous curvature metric is used to enhance the quality of the trajectories generated. Simulation results prove the efficiency of the algorithm.

Odometry is one of the most common methods for determining the position of robots. There are two major kinds of errors associated with it namely systematic and non-systematic errors. A systematic error includes the variation in wheel diameters mostly. But in real time navigation the tire radii varies depending on the conditions of the path followed. It is considered as a non-systematic error. The technique presented in [3] improves the odometry calculations. Systematic errors are reduced by employing straight and circular path navigation. Infrared sensors measure the tire radii to overcome the non-systematic error.

A layered fuzzy controller is designed in [4] to overcome the navigation problems for a non-holonomic robot. Its idea is derived from the human behavior during walking, driving etc. It consists of two layers. First layer defines the motion planning which is based on Sugeno type fuzzy logic. Second layer contains a fuzzy controller for navigation. Both layers successfully made the robot to reach its desired destination safely by avoiding hurdles. The system is modeled and simulated in Simulink software tool box. Experimental results show that it is efficient in moderate environments. The major advantage it offers is that it can be implemented on any kind of robotic platform

A Neuro-Fuzzy controller is designed in [5] to solve the steering problems of robots. Geometric analysis provides information about the kinematic and dynamic constraints of the robot. As a result, the controller creates the trajectories that satisfy non-holonomic constraints. During the absence of obstacles trajectories of near-minimal lengths are created whereas minimum deviation occurs in the presence of obstacles. Controller's efficiency and easy understanding are admirable features. By using the CAD tools of the Xfuzzy environment controller has been designed and demonstrated.

II. PROPOSED DESIGN AND DEVELOPMENT

The experimental prototype is a rectangular shaped steered drive. There are two dc gear motors installed on the robot named as front motor (FM) and back motor (BM). FM moves the steering whereas BM generates the force required to drive the vehicle. Potentiometer is connected to the front motor which gives the steering angle (SA) in analog format. ADC0804 digitizes the values coming from the potentiometer. Two ultrasonic sensors (SRF04) named as left sensor (LS) and right sensor (RS) gives information about the environment which made hurdle detection possible. Camera takes image of the road which is processed in PC via MATLAB for providing road following behavior. MAX232 takes the processed information from PC and transfers it to the controller. AT89C52 microcontroller receives data from potentiometer, sensors and road information from PC. Two H-bridges are designed to drive the motors. H-bridges are activated and controlled through microcontroller. Now, controller generates motion commands on the basis of all the data received from various gadgets. The block diagram of experimental prototype is shown in Fig. 1.1. Fig 1.2 shows Robot Architecture.

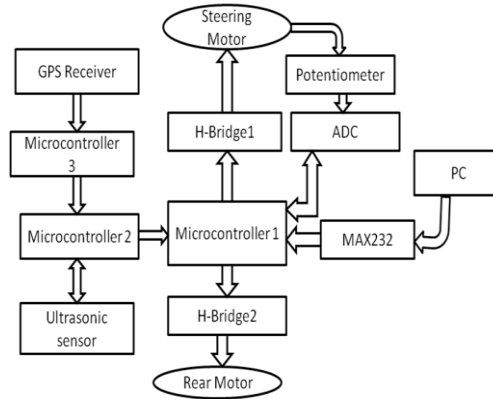


Fig 1.1 Block Diagram of the Entire System



Figure 1.2 Robot Architecture

III. VEHICLE FUNCTIONALITY

A. Hurdle Avoidance

There are two sensors namely LS and RS mounted on the vehicle. These sensors radiate in a directional pattern in the form of bursts through transmitter pin. After a short time delay they receive the echo at the receiver pin. This process goes on repeatedly. In case, if the signal is not received at the receiver pin, then the burst will be re-transmitted after 32 milliseconds delay. Afterwards, the data from the LS and RS is fed to the fuzzy controller. Moreover, potentiometer also provides the information about the SA. Now, the microcontroller puts this information in the fuzzy rules. Resultantly, the collision free trajectory is generated. Hence the hurdle avoidance behavior is accomplished.

(A1) Fuzzy Controller

The inputs to the fuzzy controller comprise the (LS) and (RS). They describe the information regarding hurdles. The distance from the ultrasonic sensor towards hurdles is divided into three regions. These regions are indicated by three linguistic variables namely near, medium and far. They range from [20, 80] on the universe of discourse. Membership functions are plotted. Then the rules are created. These rules define the correspondence between the membership functions and their related output. The first output of the fuzzy controller gives the speed of the back motor (BM) which is represented by the linguistic variables Slow, Medium and Fast. The second output of the fuzzy controller is the steering angle represented by the linguistic variables LR (large right), SR (small right), ST (straight), LL (large left), SL (small left). The block diagram of fuzzy controller is shown in Fig.2 while the membership functions are described in Figs. 3, 4 and 5. The triangular membership function is used to reduce the computational complexity of the system.

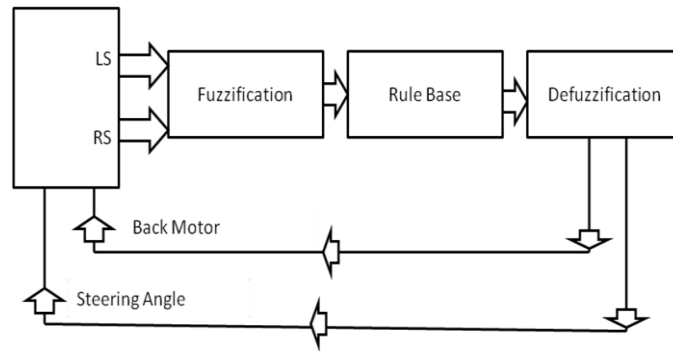


Figure 2 Block Diagram of Fuzzy Controller

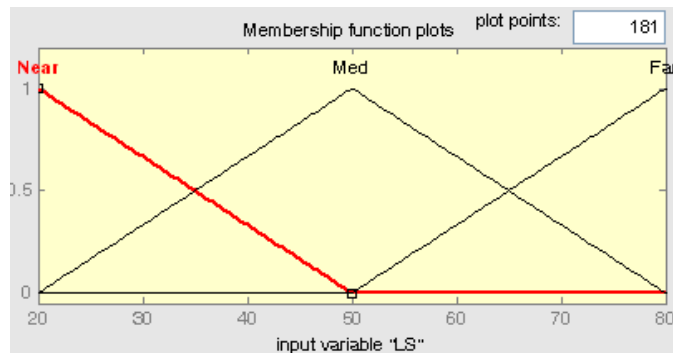


Figure 3 Membership function for right sensor (LS)

(A1.1) Fuzzification

In Fig. 3, three triangular membership functions for left sensor (LS) are shown. X-axis tells the range of the distance lying from the robot to the hurdle and y-axis describes the chance of occurrence. Near, medium and far region contains the value from 20-50, 20-80, and 50-80 respectively on the universe of discourse. All the three regions can contain any value lying between [0-1] on the y-axis. The right sensor (RS) membership functions are drawn in a similar fashion as for the left sensor (LS).

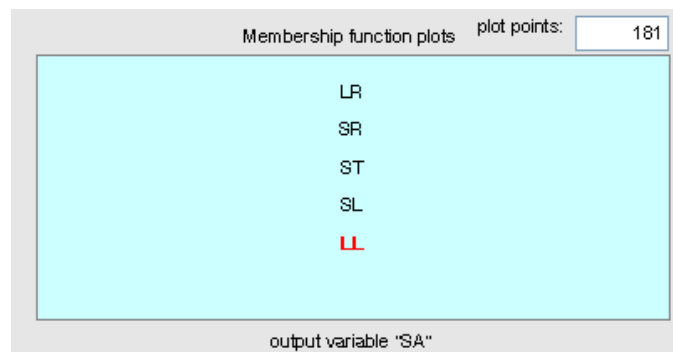


Figure 4.1 Membership function for steering angle (SA)

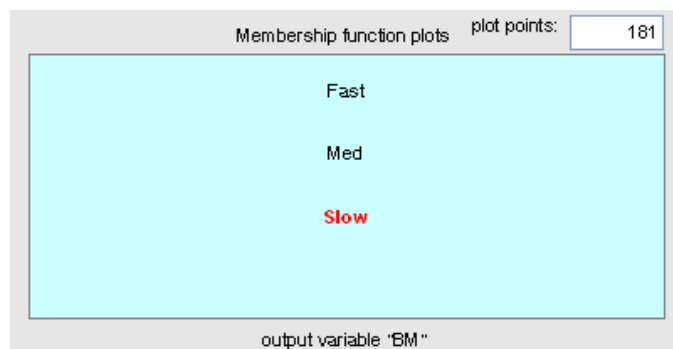


Figure 4.2 Membership function for back motor (BM)

Fig. 4.1 shows the membership functions for steering angle (SA). It contains five variables namely LR (large right), SR (small right), ST (straight), LL (large left), SL (small left). They describe the angle of rotation for steering. Steering motion depends on the hurdle information. This action has to be very fast and sharp in order to obtain a collision free trajectory.

Fig. 4.2 shows the membership functions for back motor (BM). It contains three variables namely fast, medium and slow. These variables describe the speed of the motor connected at the rear end of the vehicle. The speed also depends on the hurdle location. For example, if there is no hurdle lying in the path to be followed then fast speed could be maintained.

(A1.2) Fuzzy Rule Base

The nine rules for the fuzzy controller are :

[R1]: *If* LS is Near and RS is Near *Then* BM is slow and SA is LR (Fig.5.1)

In this case robot is at a place where both sensors measure in Near region, the robot has to take large right LR while back motor must be slow.

[R2]: *If* LS is Near and RS is Med *Then* BM is slow and SA is LR (Fig. 5.2)

In this case, robot is at a place where the LS and RS measure in near and med region respectively, according to fuzzy rule base robot will take large right while the BM speed will be slow.

[R3]: *If* LS is Near and RS is far *Then* BM is slow and SA is LR (Fig. 5.3)

In this case sensors are facing a hurdle that is in near region for the LS and in far region for RS, robot will respond by taking a sharp right turn while BM will go slow.

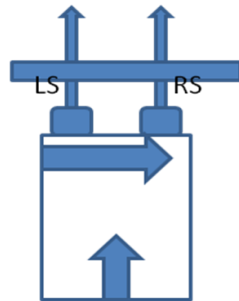


Figure 5.1 Fuzzy Rule 1

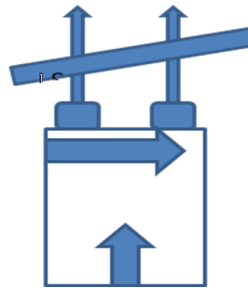


Figure 5.2 Fuzzy Rule 2

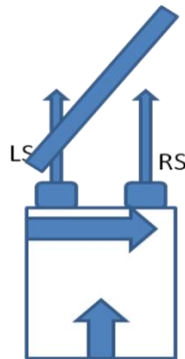


Figure 5.3 Fuzzy Rule 3

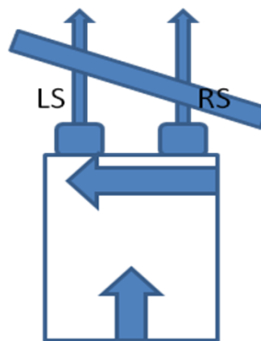


Figure 5.4 Fuzzy Rule 4

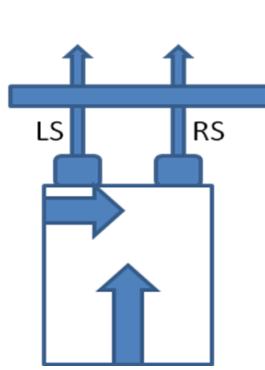


Figure 5.5 Fuzzy Rule 5

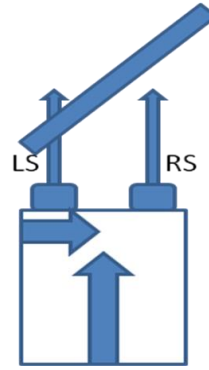


Figure 5.6 Fuzzy Rule 6

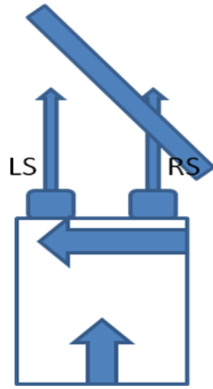


Figure 5.7 Fuzzy Rule 7

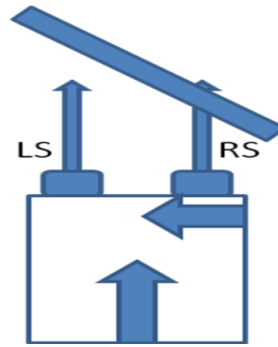


Figure 5.8 Fuzzy Rule 8

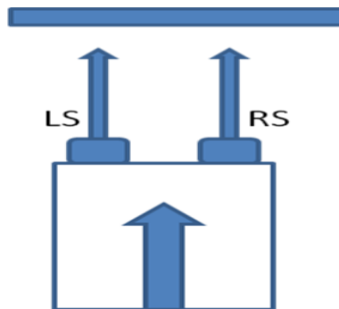


Figure 5.9 Fuzzy Rule 9

Figure 5 Fuzzy Control of Motor Speed and Steering Angle

[R4]: *If* LS is med and RS is near *Then* BM is slow and SA is LL (Fig. 5.4)

This rule describes a situation where LS measure in near region and RS in med region, robot will respond to this situation by taking a sharp left while keeping the BM slow.

[R5]: *If* LS is med and RS is med *Then* BM is med and SA is SR (Fig. 5.5)

The rule describes the response of the robot when both sensors measure in med region, robot will take a small right while keeping the BM at med speed.

[R6]: *If* LS is med and RS is far *Then* BM is fast and SA is SR (Fig. 5.6)

This rule provides the solution for the situation where LS and RS measure in med and far region respectively, in this situation robot must take SR to avoid collision while keeping BM fast.

[R7]: *If* LS is Far and RS is near *Then* BM is slow and SA is LL (Fig. 5.7)

Another possible condition that might occur is when LS measures in far region and RS measures in near region, to achieve collision free movement, robot will take a large left while keeping BM slow.

[R8]: *If* LS is Far and RS is med *Then* BM is med and SA is SL (Fig. 5.8)

If LS is measuring in far region and RS is measuring in med region then rule eight will be activated which will cause the robot to take a small left while moving at a medium speed.

[R9]: *If* LS is Far and RS is med *Then* BM is med and SA is SL (Fig. 5.9)

If both sensors are measuring in far region the robot can go straight at fast speed.

(A1.3) Defuzzification

To generate the crisp value as an output, defuzzification is applied to the singletons. Weighted average method is used for this purpose. The defuzzification process is shown in Fig. 6 where LS=30 and RS=80. Thus rule 3 and rule 6 are fired and resulting motion commands are SA=230 and BM=40.

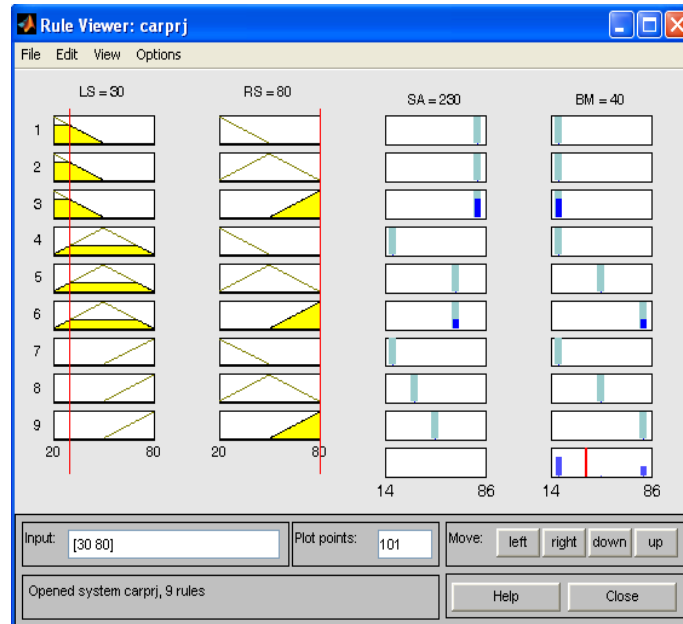


Figure 6 Defuzzification Process

B. Road Following Through Image Processing

For a mobile robot it is very important to have the road following behavior so that it can navigate successfully in real time unstructured environment. Road following behavior is achieved in this work by using the image processing technique. Two cameras mounted in front of the robot for taking the image of the road for extraction of information about the road. Both cameras takes the image at the same time which is combined to have a better view of the road. A sample image of the road is shown in Fig. 7.1. This image is then converted into gray scale [6]. The gray scale image matrix in which every element has a corresponding value that how dark or bright is the pixel at a particular point on the image. The gray scale image is converted into binary format [6] [7] by applying threshold. The binary image is shown in Fig. 7.2. The binary image is dilated [8] with a disk structuring element to make the boundary continuous. The resulting image is shown in Fig. 7.3. Hough transform [9] is applied at this dilated image to get the possible lines in the image. The lines with longer length are then selected which actually are the road boundaries.

The result of lines selection is shown in Fig. 7.4. A bounding box is generated using the lines which are slightly inward from the actual boundary lines to form a bonding box, known as ROI and shown in Fig. 7.5. The original binary image (OBI) is logically AND with ROI to extract the lane boundary which is then dilated to form a continuous line as shown in Fig. 7.6. Thus a line for lane boundary similar to road boundaries is obtained after the application of Hough transform on this dilated image as shown in Fig.7.7. After all the boundaries in the image have been detected, the robot stays within the lane by using slopes of lines. The final trajectory of the vehicle is shown in Fig. 7.8.



Figure 7.1 Original Road



Figure 7.2 Binary Image

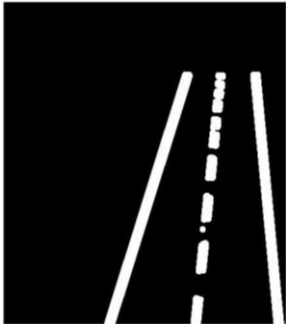


Figure 7.3 Dilated Image

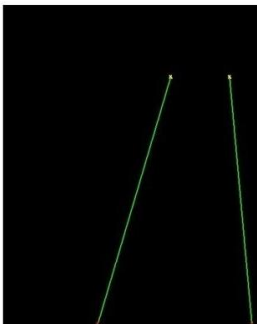


Figure 7.4 Result of Hough Transform

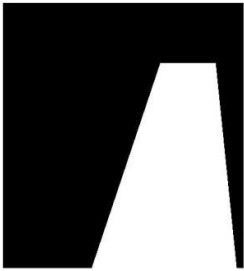


Figure7.5 ROI

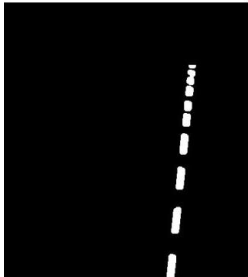


Figure 7.6 Logical AND of OBI and ROI

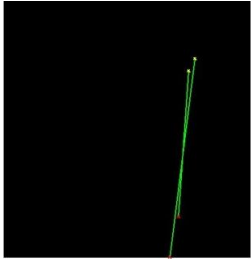


Figure 7.7 Result of Hough Transform

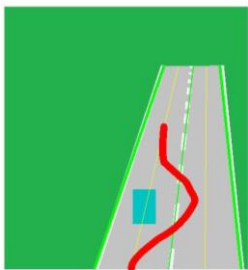


Figure 7.8 Final Trajectory Avoiding Hurdle

Figure 7 Image Processing of Road and Road following Behavior of Autonomous Car

IV. RESULTS AND CONCLUSIONS

A fuzzy controller is designed to drive the vehicle autonomously. The fuzzy controller contains nine rules. The sensor values are fuzzified using Sugeno inference system. Afterwards, rules are defuzzified using weighted average method. After receiving information from sensors and cameras, microcontroller decides the trajectories to follow in accordance to the rules. The fuzzy controller is simulated via MATLAB. The software results prove the efficiency of the controller. Moreover, the robot is tested in real time in the presence of static and dynamic obstacles. Experimental results have shown that it can operate equally well in complex and simple environment.

V. REFERENCES

- [1] Pradalier, C., Hermosillo, J., Koike, C., Braillon, C., Bessière, P., & Laugier, C. (2005). The cycab: a car-like robot navigating autonomously and safely among pedestrians. *Robotics and Autonomous Systems*, 50(1), 51-67.
- [2] Pradalier, C., Hermosillo, J., Koike, C., Braillon, C., Bessière, P., & Laugier, C. (2005). The cycab: a car-like robot navigating autonomously and safely among pedestrians. *Robotics and Autonomous Systems*, 50(1), 51-67.
- [3] Kim, C. S., Park, J. H., Hong, K. S., & Lee, M. H. (2004, November). Improving odometry accuracy for car-like vehicles by using tire radii measurements. In *Industrial Electronics Society, 2004. IECON 2004. 30th Annual Conference of IEEE* (Vol. 3, pp. 2546-2551). IEEE.
- [4] El-Khatib, M. M., & Hamilton, D. J. (2006, July). A layered fuzzy controller for nonholonomic car-like robot motion planning. In *Mechatronics, 2006 IEEE International Conference on* (pp. 194-198). IEEE.
- [5] Baturone, I., & Gersnoviez, A. A. (2007, July). A Simple Neuro-Fuzzy Controller for Car-Like Robot Navigation Avoiding Obstacles. In *Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International* (pp. 1-6). IEEE.
- [6] Baturone, I., & Gersnoviez, A. A. (2007, July). A Simple Neuro-Fuzzy Controller for Car-Like Robot Navigation Avoiding Obstacles. In *Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International* (pp. 1-6). IEEE.
- [7] Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2004). *Digital image processing using MATLAB*. Pearson Education India.
- [8] Jayaraman, S., Esakkirajan, S., & Veerakima, T. (2009). *Digital image processing*, ed iii.
- [9] Nixon, M. (2008). *Feature extraction & image processing*. Academic Press.