International Journal of Research in Engineering, Science and Management
Volume-2, Issue-10, October-2019
www.ijresm.com | ISSN (Online): 2581-5792

Gesture Control Car Controlling System Using PIR Sensor

D. R. P. Rajarathnam[1], S. T. Kumaravel[2], R. Praveen Kumar[3], N. Kavi Praveen Kumar[4],
P. Magendran[5]

[1]*Professor & HoD, Department of Mechatronics Engineering, Paavai Engineering College, Namakkal, India*
[2]*Associate Professor, Department of Mechatronics Engineering, Paavai Engineering College, Namakkal, India*
[3,4,5]*Student, Department of Mechatronics Engineering, Paavai Engineering College, Namakkal, India*

***Abstract*: The goal of gesture recognition in Computer Science field has always been the minimization of the distance between the physical world and the digital world. The way humans interact among themselves could be implemented in communication with the digital world by interpreting gestures via mathematical algorithm. Numerous ways and algorithms have been proposed and implemented to achieve the goal of gesture recognition and its use in communicating with the digital world. Gestures can be tracked using accelerometers. Since modern Smartphone are equipped with an in-built accelerometer, gesture control using Smartphone can be easy to implement, cheap to provide and the output will be more intuitive. This paper deals with the design and implementation of a wireless gesture controlled Robot using Arduino Lilypad processor and an Android operated application to control the gestures with minimal, and cheap hardware requirements. The system can be broadly classified into two components: The Hardware part consisting of Arduino Microcontroller, the Adafruit motor Shield, HC-05 Bluetooth module, and the Android Smartphone, and the software part consists of a Java based application run on android.**

***Keywords*: microcontroller, accelerometer, gesture recognition, embedded systems, android, module, motor driver, co-ordinates, robot.**

1. Introduction

Humans interact in the physical world by the means of the five senses. However, gestures have been an important means of communication in the physical world from ancient times, even before the invention of any language. In this era of machines taking control of every complex works, interactions with machines have become more important than ever. Robots are classified into two types: Autonomous robots like Line sensing or edge sensing robots, and Remote controlled robots like Gesture controlled Robots. Since this paper deals with gesture controlled robot, the primary focus will be on the remotely controlled robots only.

Undoubtedly, the output and the functioning of machines will be more intuitive if they are communicated using human gestures. A gesture is a form of communication in a non-verbal manner by using visible body movements or actions conveying messages.

There are several ways to capture a human gesture that a machine would be able to understand. The gesture can be captured using a camera, or a data glove. Gestures can also be in infrared waves, Acoustic, Tactile, optical or motion technological means [1]. The embedded systems designed for specific control functions can be optimized to reduce the size and cost of the device, and increase the reliability and performance. The Gesture are equipped with in-built accelerometer which may be used for gesture recognition and such other tasks. Moreover, Android platform is being used in the development of numerous applications for cellphones.

Researchers have shown interest in gesture recognitions and have built several robots and devices that are controlled by human gestures. There is a constant development in the field of gesture controlled devices. Apart from hand gesture recognition, emotional gesture recognition from face is also done in some cases. There are two types of gestures used in gesture recognition: Online gestures and Offline gestures. In Online gestures, direct manipulations like rotation and scaling are done. In Offline gestures, the processing is done only after the user interacts with the object. Gesture technologies are applied in several fields like in Augmented Reality, Socially assistive Robots, recognition of sign languages, emotion detection from facial expressions, Virtual mouse or keyboard, recognition of sign languages, remote control, etc.

There are various modes of communication between the microcontroller of the robot and the Smartphone. However, the popularly used means of communication is done via RF, Bluetooth or Wi-Fi. Using RF limits the distance from which the robot can be controlled. Using Wi-Fi increases the overall cost for setup. So, the robot has been built with Bluetooth which has intermediate range of distance covered and cost between RF and Wi-Fi. Arduino to detect the direction of movement of the user's hand and move the robot accordingly. The prime aim of the design is that as the user moves his hand in some direction, the robot moves in the same direction as well. In other words, the robot is solely controlled by the hand movements and gestures of the user. The goal of this paper is to develop a method to control and program a robot with gestures and assure high level of abstraction, cheap and minimal hardware and a simplified robot programming.

2. Working

The emergence of robots can be traced back to the 90‟s with Helpmate Robots and Robot-Caddy [1]. Since then, there is an exponential development in the field of robotics, and controlling robots through human gestures have been the topic of research for as long time. With the implementation of gestures to control robots, there have been several methodologies to perform the action. Some of the related works are being described in this section.

A. Light-based Gesture Recognition

Light or illumination tracking and controlling robots with light sensors are being done in a lot of cases. Such robots are autonomous in nature. Generally, there are some light sensors associated with the robot. The sensors send some rays of light and track them as they get absorbed in the surface or reflected back to it. According to this, the robot can be line-sensing robots where it is made to follow a black or a white path autonomously.

B. Vision-based Gesture Recognition [7]

Several robots are designed to be controlled by vision-based gestures. In such robots, there are, generally, some cameras as the sensor, which also acts as an interface to control the robot with some manipulators. The input gesture can be some patterns, movements of hands, color tracking, face recognition, finger tracking, or some templates. They are also used in ball tracking and Robo-football games where the robots play the traditional game of football by tracking the movement of the ball. Though it has paved a way for advanced robot operations, but it is affected by factors such as illumination, foggy weather, background lights, etc. [2].

C. Motion-based Gesture Recognition

The motions can be used to control a robot. This is generally done by incorporating an accelerometer to control the robot wirelessly. This can also be done using sensors. This method is beneficial over other methods in the sense that it can interact with machines naturally without being intruded by the factors that affects the mechanical devices [3]. One important development in this field is done by Sauvik Das et al in 2010, where he designed a spying device yielding location and activities of the user without his/her knowledge [4].

D. Sixth Sense Technology

The Sixth sense technology begins in 1990 by Steve Mann who implemented a wearable computing device via neck-projector or head-mounted projector coupled with a camera. Later, following his idea, Pranav Mistry, a young research scientist at MIT at that time came up with new applications of this technology. Pranav Mistry came up with the name, Sixth Sense Technology" and has since been named Wear Ur World (WUW). This technology applies all of the techniques mentioned above and designing applications that give an intuitive output with the connection of internet.

3. Technical requirements

In this work, there are six fundamental components are required as shown in the figure 1.

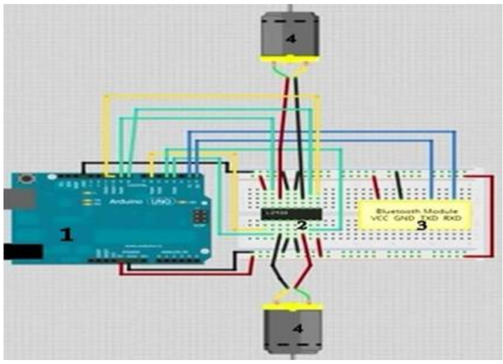

Fig. 1. Technical Components used in the work

A. Microcontroller

The first requirement for the design of the robot is the microcontroller. In this work, Arduino ATMEGA 328 has been used. It is an open-source electronics prototyping platform with 14 digital I/O pins, 6 analog inputs, 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header and a reset button.it is marked with 1.

B. Mechanical Components

Motor driver is an additional requirement in this work. But to make the output more intuitive, this component is necessary. DC motors are used in this work to rotate the four wheels due to its effectiveness while designing robot with a high gear ratio. The motors are shown in the figure as 4. The motor shield used here is Adafruit Motor Shield V2. The chip marked as 2 in the figure is motor shield, though a commercial manufactured motor driver is used to complete the work.

C. Arduino Lilypad

It is a designed for wearable application. It works on rechargeable batteries and allows easy connections with sensors and actuator's developed for an easy integration in clothes and fabrics. The Lilypad Arduino USB2 serial interface is the one we recommend, but any standard FTDI compatible interface suitable. The Arduino USB 2 serial interface behaves as an arduino UNO and shares the same driver.

D. Physical Characteristics

The Lilypad Arduino is a circle, approximately 50mm (2") in diameter. The board itself is 0.8mm (1/32") thick (approximately 3mm (1/8") where electronics are attached0). The operating voltage are 2.7-5.5v.

And the input voltage is 2.7-5.5v. The digital input pins 14 analog input pins 6 DC current per input pin 40mA flash memory 16KB SRAM 1KB.

4. System description

Gesture recognition is the main aim of this work. The user holds an Android operated Smartphone, and moves, or rotates his hand in any direction. The accelerometer within the phone is regulated to generate a maximum and minimum value for the movement of the hand in three dimensional co-ordinates depending upon the external environmental conditions. The android application does the work of sensing the accelerometer calibration and generating the maximum and minimum values from it. Depending upon the values obtained, it sends a determinant value to the microcontroller. The module receives data and transmits it to Arduino where it checks the determinant value and moves the robot accordingly. The whole process is under an infinite loop, so it runs as long as the power is supplied. The output depends on the accelerometer inputs directly that can be used to control the robot. The accelerometer input depends on the gestures of the user's hand. The steps stated above are broadly described in this section. The system consists of the following steps to work as mentioned.

A. Block diagram of the system

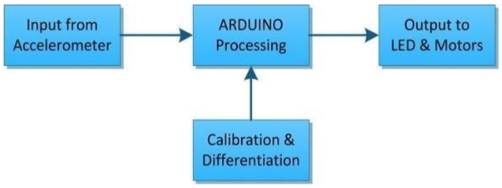

Fig. 2. Block diagram of the system

B. Transmission of data from application to Motor driver

The input to the application is the direction of movement of hand of the user given by the accelerometer. This is analog in nature. The signal goes to the digital pins of the Arduino board, which has an inbuilt AD/DA converter of 8 bit. The processer the received data. Based on the data received, appropriate signal is transmitted to the motor driver to rotate the motor in such a way that the robot moves in the direction of movement of the user's hand.

C. Gesture Recognition

The application designed in this work retrieves the value of the accelerometer and sends a determinant value to the microcontroller via transmitter. As the user moves his hand, the accelerometer reading changes. It is then retrieved by the application. There are two values: One is maximum value and the other is minimum value. The range is specified using these two values for each function of the robot. If the value retrieved by the application lies between these specified values, then the corresponding determinant is generated. This determinant is sent to the microcontroller, which then receives the determinant value, process it to recognize the corresponding gesture, and sends signals to move the robot accordingly. Gestures for movement of the robot Figure 5 shows the gestures to control the movement of the robot. When the user tilts his hand forward, the gesture is recognized as the forward movement, and the robot moves in the forward direction. The angle of the tilt or the difference between the angle of tilt of user's hand and the threshold value of forward movement gesture determines the speed of the robot. When the user tilts his hand on the right direction, the gesture is recognized as the right turn, and the robot moves in the right direction. When the user tilts his hand in the left direction, the gesture is recognized as the left turn, and the robot moves in the right direction. The angle of the tilt of user's hand determines whether the left or right turn is a normal turn or a sharp turn. A sharp turn is one in which a car changes direction without slowing down before turning. When the user tilts his hand backwards, the gesture is recognized as the move backward gesture, and the robot moves in the backward direction. If the user's hand is somewhere between the two gestures, i.e., the accelerometer value is somewhere between the threshold of two directions (forward and left turn, left turn and backwards, backwards and right turn, forward and right turn), then the robot moves in that diagonal direction.

D. Movement of motors and wheels

There are four DC motors used in the design of this robot: one motor for each wheel. The motors are controlled by the Adafruit motor shield. The shield is stacked on top of Arduino. Every shield stacked can run 4 DC motors. Installing the Adafruit Motor Shield library gives the flexibility of using the motors just by calling some pre-defined functions as motor1.setSpeed(value) that sets the speed of the motor to 250 rpm, or motor1.run(FORWARD) that makes the motor1 to rotate forward [10]. These functions are called from the program burnt in the Arduino microcontroller. The signal is sent to the motor shield that runs the motors.

The wheels are connected to the motors. 4 DC motors are used Two for left wheels, and two for right wheels. When the signal received in the motor shield is to move forward, all the four wheels of motors rotate forward, this turns all the four wheels in the forward direction. The robot moves in the forward direction. When the signal received in the motor shield is to turn the robot in the forward left direction, the left diagonal motors are rotated backwards while the right diagonal motors are made rotated forwards. This makes the robot turn in the forward left direction. When the signal received in the motor shield is to turn the robot in the forward right direction, the right diagonal motors are rotated. Backward while the left diagonal motors are rotated forwards. This makes the robot turn in the forward right direction. When the signal in the motor shield is to move backward, both the pairs of the motors are rotated backwards resulting the robot to move backwards. When the signal in the motor shield is to stop the robot, all the motors are made stationary resulting the robot to stop. Similarly, to rotate the robot in backward directions, similar methodology is used. To turn the robot in the backward left direction, the left diagonal motors are rotated forwards while the right diagonal motors are rotated backwards. This makes the robot turn in the backward left direction. To turn the robot in the backward right direction, the right diagonal motors are rotated forwards while the left diagonal motors are rotated backwards. This makes the robot turn in the backward right direction...Movement of the Motors and Wheels.

E. Designing the android application

The Android application is the key to control the robot using hand gestures. The application reads the accelerometer state and X, Y, and Z values are obtained in the application. There are two threshold values assigned for each movement: one is the MAX_THRESHOLD, and the other is the MIN_THRESHOLD. If the obtained value lies between these thresholds of a certain movement, then the character assigned to denote that movement, which is called the DET or determinant is sent to the robot via Bluetooth. The application continuously sense this until the application is ON. A graphical user interface has been designed for the comfort of the user. The application abstracts the calculations and accelerometer values, but the user interface shows the direction of movement of the hand so that the user is aware of wrong turns in the bot.

5. Algorithm for gesture controlled robot

A. Main Module

Step 1: Initialize the frequencies of the motors.
Step 2: Initialize SERIAL 9600
Step 3: Set the speed for the motors in rpm.
Step 4: While (1) do 1. DETcheck(); 2. While DET == F, move the robot forward Call check (); End While 3. While DET == B, move the robot backwards Call check (); End While 4. While DET == L, move the wheels left Call check (); End While 5. While DET == R, move the wheels right Call check (); End While 6. While DET == I, move the wheels right forward Call check (); End While 7. While DET == J, move the wheels right backward Call check(); End While 8. While DET == G, move the wheels left forward Call check (); End While 9. While DET == H, move the wheels left backward Call check (); End While End While.

B. Function check()

Step 1: Initialize DATAIN □□S
Step 2: Initialize VELOCITY □□0
*Step 3***:** If data on the serial lines > 0 then
1. DATAIN □□Character sent by the phone
2. If DATAIN == F, then DET □□F
3. If DATAIN == B, then DET □□B
4. If DATAIN == L, then DET □□L
5. If DATAIN == R, then DET □□R
6. If DATAIN == I, then DET □□I
7. If DATAIN == J, then DET □□J
8. If DATAIN == G, then DET □G
9. If DATAIN == H, then DET □H
10. If DATAIN == S, then DET □□S
End If *Step 3:* Set the velocity based on the data received in the multiples of 25. Set VELOCITY □□U, if no valid value for velocity is received. *Step 4:* Return DET.

6. Algorithm for the android application

Step 1: Connect to the Bluetooth Module
Step 2: Set THRESHOLD_MAX and THRESHOLD_MIN values for each direction forward, backward, Left and Right.
Step 3: Get the state of the accelerometer: I □□State of the accelerometer.
Step 4: VALUE □□I into X, Y, Z coordinate values **Step 5:** If VALUE is in between THRESHOLD_MAX and THRESHOLD_MIN for a direction, then set DATAOUT as the direction of VALUE.

7. Experimental results

The robot moves in the direction of the movement of the user’s hand: forward-left. The robot moves in the direction of the movement of the user’s hand: back. The robot moves in the direction of the movement of the user’s hand: forward –right.

8. Future scope and conclusion

The Gesture controlled robot designed in this work has many future scopes. The robot can be used for surveillance purpose. The robot can be applied in a wheelchair where the wheelchair can be driven by the movements of rider’s hand. Edge sensors can be incorporated to it to prevent the robot from falling from any surface. Some camera can be installed which can record and send data to the nearby computer. It can be implemented on a watch, or in any home appliances like Room heater. Modern ARDUINO chips support Intranet as well as Internet connections which can be utilized to a greater extent. This robotic car can be enhanced to work in the military surveillance where it can be sent to some enemy camps and track it’s activities via Internet. With a mind full of creation, the possibilities are endless. In this paper, the design and implementation of Gesture Controlled Robot is presented and developed using Arduino microcontroller. An algorithm has been provided and its working is detailed thoroughly. Since the updating possibilities are endless, updating the system has been kept as a future scope. The built device is cheap, and is easy to carry from one place to another. The addition of some additional sensors or camera will make it more productive. The limitation of the hardware being associated with a system has been reduced to a great extent. As an end thought, the system will allow the user to control it in a way that reduces the gap between the physical world and the digital world with an output more intuitive.

References

[1] Monika Jain, Aditi, Ashwani Lohiya, Mohammad Fahad Khan, and Abhishek Maurya, “Wireless Gesture Control Robot: An Analysis,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 1, no. 10, pp. 855-857, December 2012.

[2] S. Swapnali and P. G. Chilveri, "Hand Gesture Recognition using accelerometer sensor for traffic light control system," *2014 International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, 2014, pp. 1-5.

[3] Cannan, J. and Hu, H., “Human-Machine Interaction (HMI): A Survey”.

[4] Song, M., Kim, B., Ryu, Y., Kim, Y., and Kim, S., “A design of real time control robot system using android Smartphone” The 7th International

Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Busan- Korea, Nov. 2010.

[5] K. Brahmani, K. S. Roy, and Mahaboob Ali, “Arm 7 Based Robotic Arm Control by Electronic Gesture Recognition Unit Using Mems”, International Journal of Engineering Trends and Technology, Vol. 4, no. 4, April 2013.

[6] S. Waldherr, R. Romero and S. Thrun, “A gesture based interface for human-robot interaction”, In Autonomous Robots in Springer, vol. 9, no. 2, pp. 151-173, 2000.