JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Hand Gesture Recognition for Real-Time Game Control: Design and Development

Yash Markad[1], Suraj Kalwaghe[2], Ayush Kalantri[3], Swati Savkare[4]

Department of E&TC, SKNCOE, SPPU, Pune

[1]markadyash3@gmail.com, [2]surajkalwage8383@gmail.com, [3]ayushnkalantri@gmail.com,

[4]swati.savkare_sknccoe@sinhgad.edu

Abstract— **The way we interact with software applications has been changing, with a growing demand for more intuitive and innovative ways of control. This project proposes a wireless gesture controller using an MPU6050 gyro sensor, two Wi-Fi modules, and a Python script to capture hand motion and translate it into arrow key commands for software application control. The controller captures hand motion in real-time, sends the x and y-axis coordinates to a receiver using a MAC protocol, and the Python script translates the data into arrow key commands that control the software application. The system is tested for accuracy and precision, with results demonstrating high accuracy in detecting hand gestures. This system has the potential to revolutionize the way we interact with software applications and could be useful in various settings, such as gaming, virtual reality, and industrial automation. The wireless gesture controller has the potential to provide a more natural and intuitive way of controlling software applications, making it easier for users to perform complex actions with ease. Gesture controller provides an interactive and immersive experience to user.**

Keywords **Hand Gestures, Python, Virtual Reality, MAC Protocol, Real-time Detection, Human-Computer Interaction, ESP8266, MPU6050, Gaming, Accelerometer, Gyroscope, Wi-Fi Module, Gesture Control.**

I. INTRODUCTION

The way we interact with technology has evolved significantly in recent years, with new forms of input methods emerging to meet the demands of users. However, there is still a growing demand for more natural and intuitive ways to control software applications. This is where the concept of a wireless gesture controller comes into play. The wireless gesture controller is a system that allows users to control software applications using hand gestures. It consists of two Wi-Fi modules, an MPU6050 gyro sensor, and a Python script. The MPU6050 gyro sensor is used to detect hand motion, while the two Wi-Fi modules are used for communication between the transmitter and receiver modules. The transmitter module captures hand motion in real-time and sends x and y-axis coordinates to the receiver module using a MAC protocol. The Python script then translates this data into arrow key commands, which are used to control the software application. The primary benefit of the wireless gesture controller is its ability to provide a more natural and intuitive way of controlling software applications. Users can use hand gestures to control the application, making it easier to perform complex actions with ease. This is especially useful in situations where the use of a keyboard or mouse is not feasible or desirable, such as in gaming or virtual reality. To test the effectiveness of the wireless gesture controller, we conducted experiments to evaluate the accuracy and precision of the system in detecting and responding to hand gestures. The results showed that the system can detect hand gestures with high accuracy, making it a promising approach for interacting with software applications.

II. LITERATURE SURVEY

The concept of using hand gestures to control software applications is not new and has been a topic of interest in the field of human-computer interaction (HCI) for many years. In recent years, several studies have focused on developing new and innovative ways to use hand gestures as an input method.

Premangshu Chanda et al: [1]: conducted a study that deals with the design and implementation of a wireless gesture controlled robot using ATMEGA32 processor and an Android operated application to control the gestures via Bluetooth.

P.V.Patil et al: [2]: conducted study on represents a simple accelerometer controlled robotic arm using Atmega328. This interfacing is done using wireless communication through ZigBee. At the receiver Arm and bot position are controlled as per the user hand gesture. In this study accelerometer is used. The new approach is interfacing of LM 35 for temperature sensing. The robot moves in different directions. It is applicable for area where human is unable to reach.

Vivek Bhojak et al: [3]: presented a model to control robotic arm through human gestures using accelerometer. Accelerometer is mounted on human hand in order to perform the action of robotic arm according to the action of human hand. This control mechanism provides an easy movement& control. It provides a better way to control a robotic arm using accelerometer which is more intuitive and easier to work using, besides offering the possibility to control a robot by other wireless means.

Archika Setia et al: [4]: presented survey on design and implementation of an accelerometer based hand gesture controlled robot is controlled wirelessly using a small low cost, 3-axis accelerometer. A novel algorithm for gesture identification has been developed to replace the approach of conventional controlling mechanism by an innovative hand gesture based controlling.

Hari Prabhat Gupta et al: [5]: presents a continuous hand gestures recognition technique that is capable of continuous recognition of hand gestures using three-axis accelerometer and gyroscope sensors in a smart device. In this paper to reduce the influence of unstableness of a hand making the gesture and compress the data, a gesture coding algorithm is developed. It is observed that the proposed system predicts wrong gesture in case of partially performed gestures. Continuous Hand Gesture (CHG) technique is able to perform accurate gesture detection and is suitable for human machine interaction.

In conclusion, all the reviewed papers offer a wide range of approaches for wireless hand gesture recognition based game controller, including various methods such as Atmega328 controller based, Bluetooth based, accelerometer based, continuous hand gesture (CHG) technique. These approaches can be used to design and develop a wireless hand gesture recognition based game controller.

III. PROPOSED MODEL

The proposed model of the wireless gesture controller consists of three main blocks. MPU6050, ESP8266 and computer machine are the blocks of the wireless gesture controller. The first block MPU6050 sensor. This block takes input from the user and detects motion. Then output of the first block is given to second block. The second block is ESP8266. This block has two parts, transmitter and receiver. The output of the ESP8266 is given to the computer machine. Computer machine is a third block of system. The computer controls the python program. The software application runs on computer which will be controlled using controller.

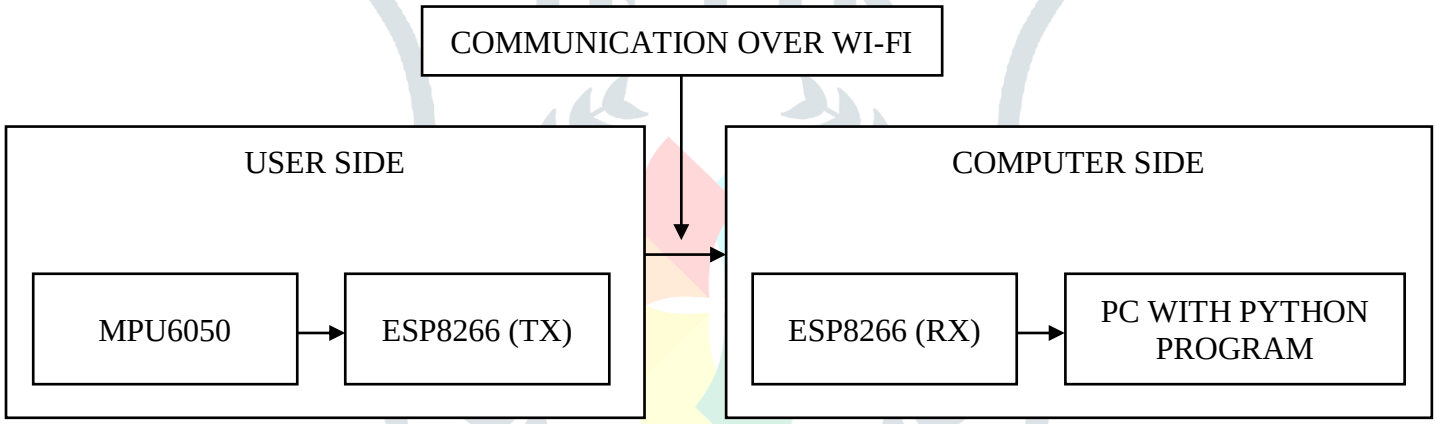

Figure no.1: Block Diagram

IV. IMPLEMENTATION

The project development involves various software and hardware elements such as different sensors like mpu6050, ESP8266 module, Programming control through PC, libraries such as pyserial, pykeyboard. The implementation of project is be divided into two parts: hardware system development and software system development. The complete system is implemented in two parts. This is an integration of software and hardware system.

Hardware System Implementation: The hardware system involves modules like ESP8266 wi-fi module, MPU6050 sensor and a computer. In the process of implementing, we connect the MPU6050 sensor and ESP8266 transmitter using jumper wires. This module is placed in an 3D printed box to make module easier to use. To make this cover box 3D printing technology is used. This module will be used by the user for controlling the software applications. The transmitter is connected to ESP8266 receiver over wi-fi. ESP8266 receiver is connected with computer machine using USB port. In this way the hardware system is implemented.

Software System Implementation: The software system of project is divided in three parts.

In first part of software system, code is written in C++ on Arduino IDE to program ESP8266 which will read data from controller.

In second part of software system, code is written in python on visual studio code platform to extract and process data. Libraries such as pyserial and pykeyboard are used in python program. Pyserial encapsulates the access for the serial port. It provides backends for python running on Windows, Linux operating system. Pykeyboard is a python module for providing keyboard control.

In third part of software system a software application is developed using python which display X and Y axis coordinates in real time. It is used to connect the module to the computer machine.

V. RESULT AND DISCUSSION

The results are analyzed, compared with expected results make the module highly precise. The outputs given by the module are very precise. The most important result parameters are accuracy and time delay. The accuracy of detecting hand gesture is a key factor for the precise gesture controller. For using gaming applications output with minimum delay is required. The module shows output with minimal delay of 20ms. This shows gesture controller is highly responsive. The project displays output via software applications running on computer machine. The wireless gesture controller gives output with high accuracy and minimum delay. When user gives input through hand gesture, the module receives input. The MPU6050 sensor tracks the motion using gyroscope and accelerometer. The accuracy of module in detecting hand gesture depend on MPU6050 sensor. The ESP8266 read data of motion from MPU6050. ESP8266 transmitter receives data about motion and then send that to ESP8266 receiver. The seamless communication between transmitter and receiver is ensured by ESP8266 wi-fi module. Hence to get real time output seamless communication and data transmission is required. The ESP8266 receiver is connected to the computer machine. The data received by the ESP8266 receiver is then extracted and processed. The data about the hand gesture is in the form of X, Y co-ordinates. The extracted data is then processed by the python program. The python program consists of pyserial, pykeyboard library. Then python program takes the input about the motion and then depending on the X, Y co-ordinates it receives from the input it manipulates hand gesture into keyboard commands. The final outcome shows that a user controls the software application running on a computer without using keyboard. This project result showcase uses of MPU6050 sensor, ESP8266 and python libraries such as pyserial, pykeyboard. The analysis of results helps in determining new use cases of module.

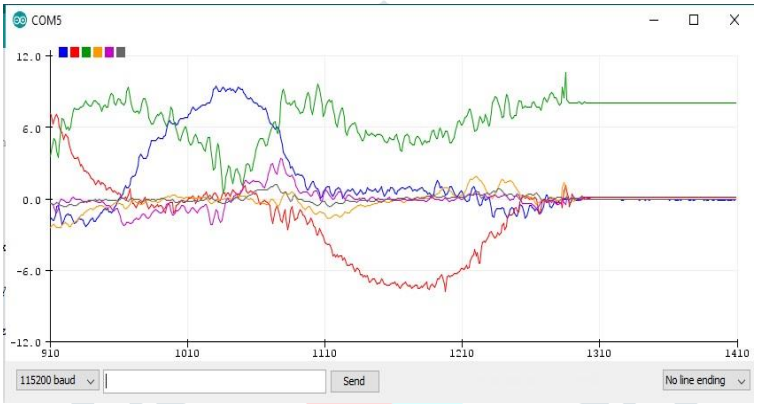

Figure 2: Reading along X, Y and Z axis

Figure 3: Wireless Hand Gesture Recognition based Game Controller

VI. CONCLUSION

The increase in convenience of using of software applications shows importance of wireless gesture controller. The wireless gesture controller can be used in technologies like Augmented Reality and Virtual Reality. This project displays how wireless gesture controller provide interactive and immersive experience to users in using different software, gaming applications. The use of various of sensors make the controller highly responsive and accurate. The wireless gesture controller makes it easier for the users as they can control their gaming applications and software applications from long distance. Users do not have to worry about connectivity as they are not tied to place to control and use software and gaming applications. This system has the potential to revolutionize the way we interact with software applications. The wireless gesture controller makes human-machine interaction simple, easy and interactive.

EFERENCES

[1] Premangshu Chanda, Pallab Kanti Mukherjee, Subrata Modak, Asoke Nath, "Gesture Controlled Robot using Arduino and Android", IJARCSSE, Volume 6, Issue 6, June 2016.
[2] P.V.Patil, M. B. Shete, T. M. Padalkar, "Wireless Hand Gesture Robot using Accelerometer", IRJET, Volume: 03 Issue: 04 , Apr-2016.
[3] Vivek Bhojak, Girish Kumar Solanki, Sonu Daultani "Gesture Controlled Mobile Robotic Arm Using Accelerometer" in IJIRSET Vol. 4, Issue 6, June 2015.
[4] Archika Setia, Surbhi Mittal, Padmini Nigam,Shalini Singh, Surendra Gangwar "Hand Gesture Recognition Based Robot Using Accelerometer Sensor" in IJAREEIE in Vol. 4, Issue 5, May 2015.
[5] Hari Prabhat Gupta, Haresh S Chudgar, Siddhartha Mukherjee, Tanima Dutta, and Kulwant Sharma, "A Continuous Hand Gestures Recognition Technique for Human-Machine Interaction using Accelerometer and Gyroscope sensors", IEEE Sensors Journal, Volume 16, Issue 16, August 2016.
[6] Khimraj, Praveen Kumar Shukla, Ankit Vijayvargiya, Rajesh Kumar, "Human Activity Recognition using Accelerometer and Gyroscope Data from Smartphones" in 2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3), February 2020.
[7] Y. Park, J. Lee, and J. Bae, "Development of a wearable sensing glove for measuring the motion of fingers using linear potentiometers and flexible wires," IEEE Transactions on Industrial Informatics, vol. 11, no. 1, pp. 198–206, February 2015.
[8] Z. Lu, X. Chen, Q. Li, X. Zhang, and P. Zhou, "A hand gesture recognition framework and wearable gesture-based interaction prototype for mobile devices," IEEE Transactions on Human-Machine Systems, vol. 44, no. 2, pp. 293–299, April 2014.
[9] Warren Triston D'souza, Kavitha R, "Human Activity Recognition Using Accelerometer and Gyroscope Sensors", International Journal of Engineering and Technology (IJET), vol. 9, No. 2, April 2017.
[10] R. Xu, S. Zhou, and W. Li, "Mems accelerometer based nonspecificuser hand gesture recognition," IEEE Sensors Journal, vol. 12, no. 5, pp. 1166–1173, May 2012.
[11] S. L. Lau and K. David, "Movement recognition using the accelerometer in smartphones," in 2010 Future Network & Mobile Summit, pp. 1–9, IEEE, July 2010.