Implementation of Wireless Human Hand Gesture Controlled Robotic Arm

Mrs.P.Sasitha [1], Y.Keerthi Sri [2], P.Sai Lakshmi [3] ,T.Narendra [4] ,G.Siva Sankar [5]

[1]Associate Professor,Dept.Of ECE,PBR VITS,Kavali,Nellore District,Andhra Pradesh,India
[2345]UG Students, Dept.Of ECE,PBR VITS,Kavali,Nellore District,Andhra Pradesh,India

---***---

Abstract - Implementation of Wireless Human Hand Gesture Controlled Robotic Arm presents an innovative and user-friendly approach for controlling robotic systems using natural human hand gestures. The proposed system eliminates the need for traditional input devices such as joysticks and wired controllers, enabling intuitive and contactless human–machine interaction . The system integrates a vision-based gesture recognition module using a Web camera interfaced with a Raspberry Pi, which processes real-time hand gestures using computer vision techniques implemented in Python. The recognized gestures are transmitted wirelessly to an Arduino Nano using Zigbee communication modules. The Arduino Nano then controls a multi-degree-of-freedom (6DOF) robotic arm by driving servo motors based on the received commands .This approach ensures real-time responsiveness, improved flexibility, and ease of operation.

Index Terms: Hand gesture recognition, wireless control system, robotic arm, Raspberry Pi, Arduino Nano, Zigbee communication, computer vision, human–machine interaction (HMI), embedded systems, real-time control, servo motor control, automation, Python, Embedded C.

1.INTRODUCTION

This project presents the design and implementation of a hand gesture-controlled robotic arm using a Raspberry Pi as a gesture processing unit and an Arduino Nano as the robotic arm controller [4]. The system utilizes a USB Webcamera connected to the Raspberry Pi to detect and interpret hand gestures in real-time. Zigbee modules with TTL interfaces enable wireless communication between the gesture recognition unit and the robotic arm.[8] The Raspberry Pi processes hand gestures using computer vision techniques and transmits the corresponding gesture data to the Arduino Nano via Zigbee [10]. On the receiving end, the Arduino Nano controls the movement of a 6DOF robotic arm based on the received gesture commands, allowing for intuitive and precise manipulation of the robotic arm [5]. This system offers a practical solution for contactless control of robotic systems, with potential applications in remote operation, automation, and assistive technology.[6]

2. LITERATURE SURVEY

The increase in human-machine interactions in our daily lives has made user interface technology progressively more important. Physical gestures as intuitive expressions will greatly ease the interaction process and enable humans to more naturally command the machines. Here the movement of hand controls the movement of the robotic arm (left, right, up, down). The hand gesture is captured by two techniques, vision based and accelerometer based. The accelerometer used is a three axes accelerometer which captures the hand gestures. Both gestures will be used to control a wireless robotic arm [1].

Robotics is an integral to present day engineering revolution. In this new industrial revolution, robotics has become a major influence in our industrial as well as day to day life. What has made it more important is the fact that integrating it with control electronics could be result in various interesting results. Some of these are: IOT based, Wi-Fi based, Bluetooth based, Gesture based, etc. In this paper an attempt has been made to make a study in gesture control. In the presented paper, a robot is designed whose motion would be controlled by hand control [2].

This paper presents the development and implementation of a Arduino Robot Arm controlled by human gestures. The project, named , explores the integration of hand gestures to control a six-axis robotic arm. The system utilizes flex sensors and accelerometers in a robotic glove to capture human gestures, enabling intuitive control of the robotic arm. The paper discusses the hardware components, assembly process, and the software implementation for gesture-based control. The presented solution offers a cost-effective alternative to commercial robotic arms, providing a platform for experimentation and customization [3].

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

3. EXISTING METHOD

Traditional methods of controlling robotic arms typically involve manual interfaces such as joysticks, buttons, or computer-based input devices like keyboards or mice. These methods require direct physical interaction with the control system, which can be cumbersome and limit the user's ability to operate the robotic arm freely [6]. In more advanced systems, robotic arms are often controlled by programming interfaces or sensors that rely on pre-defined commands or motion-capture technology. However, these systems often require a high degree of technical knowledge or complex setup and may not offer intuitive control.

Furthermore, existing systems that use motion sensors or touch-based interfaces tend to have limitations in terms of user flexibility and accessibility [7]. While some approaches explore voice commands or brain-computer interfaces, they still lack the simplicity and natural feel that hand gestures provide, limiting their practical applications. These conventional methods highlight the need for an alternative, intuitive, and accessible control mechanism like hand gesture recognition, which could greatly improve user interaction and efficiency in controlling robotic systems [8].

4. PROPOSED METHOD

The design and implementation is a wireless, vision-based hand gesture-controlled robotic arm designed to achieve intuitive and real-time human–machine interaction . The system consists of two primary units: a gesture recognition (transmitter) unit and a robotic control (receiver) unit. A USB camera connected to a Raspberry Pi captures real-time hand gestures, which are processed using vision algorithms implemented in Python to detect and classify human hand gestures . The Raspberry Pi acts as the central processing unit, converting the recognized gestures into corresponding digital control commands . These commands are transmitted wirelessly through Zigbee modules using serial communication protocols .

On the receiver side, an Arduino Nano receives the transmitted data via a Zigbee module and interprets the incoming commands . Based on the decoded signals, the Arduino generates Pulse Width Modulation (PWM) signals to control multiple servo motors of a 6DOF robotic arm. The robotic arm executes movements such as gripping, lifting, and directional positioning according to the input gestures. A switched-mode power supply (SMPS) is used to provide stable and regulated power to both the control circuitry and servo motors, ensuring reliable operation.

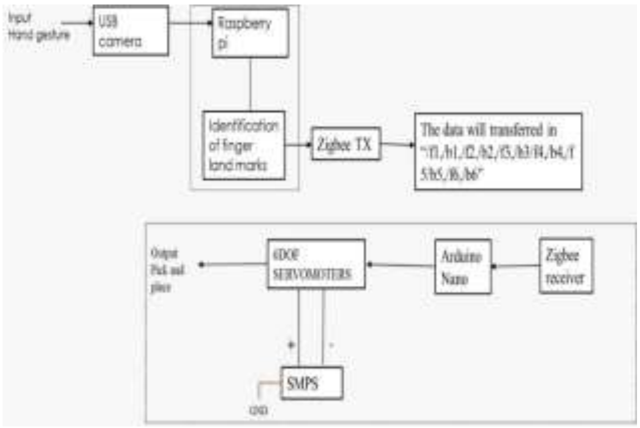

Fig 1: Flow Chart of Proposed System

Fig:1 shows the flow chart of the proposed system consists of the following sequential modules: image acquisition (USB camera), gesture processing (Raspberry Pi), wireless transmission (Zigbee transmitter), wireless reception (Zigbee receiver), control unit (Arduino Nano), and actuation unit (servo-driven robotic arm). Each block performs a specific function, enabling seamless data flow from gesture input to mechanical output. This modular architecture ensures low latency, efficient processing, and ease of scalability.

ADVANTAGES

- Reduces human effort in repetitive and complex tasks.
- Useful for differently-abled individuals to control robotic systems.
- Zigbee-based system allows remote operation without wired constraints.
- Enables safe without physical contact.
- Hand gesture recognition provides natural and intuitive control.

APPLICATIONS

- Used for controlling robotic arms in manufacturing and assembly lines.
- Enables touchless operation in surgeries and sterile environments.
- Helps physically challenged individuals control devices easily.
- Gesture-based control of household appliances.
- Demonstrates concepts of robotics, embedded systems, and AI.
- Applied in human–machine interaction and computer vision research.

5. RESULTS AND DISCUSSION

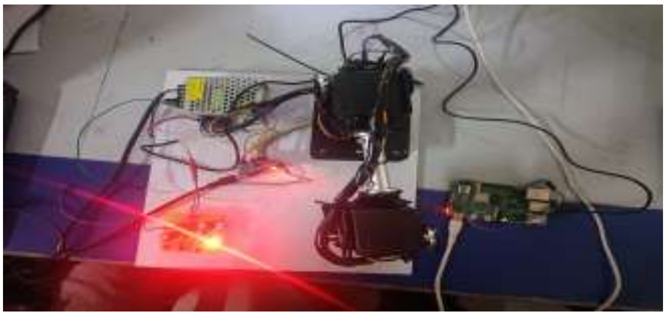

Fig 2: Hardware Setup of the Proposed System

Fig:2 describes The hardware setup these commands are encoded in a predefined format and sent to the receiver side. On the output side, a Zigbee receiver module receives the transmitted data and forwards it to an Arduino Nano microcontroller. The Arduino Nano processes the received commands and controls multiple servo motors arranged in a 6 degrees-of-freedom (6DOF) robotic arm. These servo motors enable precise movements such as rotation, lifting, and gripping for pick-and-place operations. A Switched Mode Power Supply (SMPS) is used to provide a stable power source to the servo motors and control circuitry, ensuring reliable performance. All components are interconnected using appropriate wiring and grounding to maintain system stability.

Fig 3: Software Setup for the proposed system

Fig:3 describes the software setup for the Implementation of Wireless Human Hand Gesture Controlled Robotic Arm is developed to enable real-time gesture recognition, data processing, and robotic control. The system is programmed using embedded C/C++ in the Arduino IDE for the Arduino Nano, which controls the servo motors based on received commands. On the processing side, the Raspberry Pi is configured with Python and relevant libraries such as OpenCV and Media Pipe to capture live video from the USB camera and detect hand gestures by identifying finger landmarks. The processed gesture data is converted into specific control commands and transmitted wirelessly via the Zigbee module.

The Arduino Nano receives these commands through serial communication and translates them into corresponding motor actions to control the 6DOF robotic arm. The laptop, as shown in the setup, is used for coding, debugging, and serial monitoring of the system to ensure proper communication between modules. Proper integration of software tools, libraries, and communication protocols ensures accurate gesture recognition and smooth operation of the robotic arm in real time.

Fig 4: Representation of hand gesture (10111)

Fig:4 shows the real-time implementation of a wireless hand gesture recognition system used to control a robotic arm. A laptop is running a Python-based interface where a webcam feed is processed to detect and track hand movements. The system uses computer vision libraries such as OpenCV along with Media Pipe to identify finger landmarks and gestures.

In the displayed window, the user's hand is captured, and multiple key points (landmarks) on the fingers are marked and connected, indicating successful hand tracking. The text Binary represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. This binary output is generated based on the position of detected landmarks and is used as a command signal.

The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motors move according to instructions. These instructions are transmitted wirelessly using a Zigbee module to the receiver side. At the receiving end, a microcontroller like Arduino Nano decodes the signal and actuates the robotic arm using servo motors. This setup demonstrates an efficient human-machine interaction system where hand gestures are converted into digital commands for controlling robotic movements in Real time.

Fig 5: Representation of hand gesture (01000)

Fig:5 The text "Binary: 01000" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. This binary output is generated based on the position of detected landmarks and is used as a command signal.The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor moves backward.

Fig 6:Representation of hand gesture (11011)

Fig:6 The text "Binary: 11011" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. This binary output is generated based on the position of detected landmarks and is used as a command signal.The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor moves down.

Fig 7:Representation of hand gesture (00011)

Fig:7 The text "Binary: 00011" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. This binary output is generated based on the position of detected landmarks and is used as a command signal.The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor moves backward of right.

Fig 8:Representation of hand gesture(00001)

Fig:8 The text "Binary: 00001" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. This binary output is generated based on the position of detected landmarks and is used as a command signal.The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor rotates right.

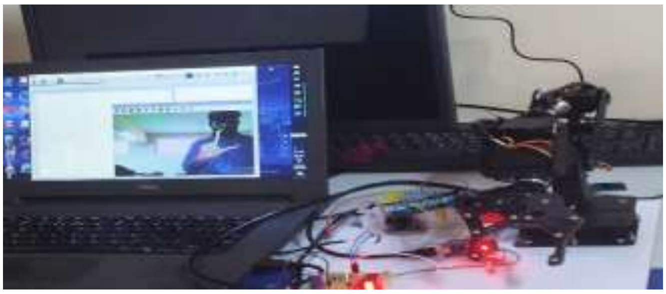

Fig 9:Representation of hand gesture (11000)

Fig 9 The text "Binary: 11000" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor arm gets open and it is used to pick a object

Fig 10: Representation of hand gesture(00000)

Fig:10 The text "Binary: 00000" represents the gesture encoding, where each bit corresponds to the state (open/closed) of a finger. The recognized gesture is then processed by a controller (such as a Raspberry Pi), which interprets the gesture and servo motor arm gets close and it is used to place a object.

6.CONCLUSION

Implementation of Wireless Human Hand Gesture Controlled Robotic Arm has been successfully developed as a complete system that integrates computer vision, embedded systems, and wireless communication to achieve intuitive robotic control. The system effectively captures real-time hand gestures using a camera and processes them through advanced vision frameworks such as OpenCV and Media Pipe, which accurately detect and track finger landmarks.

The processed gesture data is converted into a binary format representing different finger positions, enabling efficient interpretation of commands. These commands are handled by a Raspberry Pi, which acts as the central processing unit . The use of Zigbee ensures low-power, reliable, and real-time wireless communication between the transmitter and receiver sections . On the receiver side, the Arduino Nano decodes the received data and controls the servo motors of the robotic arm with precision.

The robotic arm, designed with multiple degrees of freedom (6-DOF), successfully performs operations such as object picking and placement based on user gestures. The system demonstrates good responsiveness, acceptable accuracy, and stability under controlled conditions. It also eliminates the need for physical controllers, making the interaction more natural and user-friendly.

REFERENCES

[1] S. Hameed, M. A. Khan, B. Kumar, Z. Arain, and M. Hasan,"Gesture Controlled Robotic Arm using Leap Motion," *Indian Journal of Science and Technology,* vol. 10, no. 45, pp. 1–7, 2017.

[2] T. Goswami, D. Sharma, R. Pratyush, and A. Kumar,"Gesture Controlled Robotic Arm," *International Journal of Engineering Research & Technology (IJERT),* vol. 8, no. 10, 2020.

[3] P. Neto, N. Pires, and P. Moreira,"Accelerometer-Based Control of an Industrial Robotic Arm," 2013.

[4] S. Zhou, Q. Shan, F. Fei, W. J. Li, C. P. Kwong, and C. K. Wu et al., Gesture recognition for interactive controllers using MEMS motion sensors, in Proc. IEEE Int. Conf. Nano/Micro gineered and Molecular Systems, Jan. 2009, pp. 935940.

[5] N. C. Krishnan, C. Juillard, D. Colbry, and S. Panchanathan, Recognition of hand movements using wearable accelerometers, J. Ambient Intell. Smart Environ., vol. 1, no. 2, pp. 143155, Apr. 2009.

[6] Q. Chen, N. Georganas, and E. Petriu, Real-time vision- based hand gesture recognition using Haar-like features, in Proc. IEEE IMTC, 2007, pp. 16.

[7] L. Yun and Z. Peng, An automatic hand gesture recognition system based on Viola-Jones method and SVMs, in Proc. 2nd Int. Workshop Comput. Sci. Eng., 2009, pp. 7276.

[8] L. Shi, Y. Wang, and J. Li, A real time vision-based hand gestures recognition system, in Proc. 5th ISICA, 2010, pp. 349358.

[9] Xu, Ruize,Zhou, Shengli,Li, Wenjung Jung,MEMS Accelerometer Based Nonspecific-User Hand Gesture Recognitionin Sensors Journal, IEEE, Volume: 12 , Issue: 5 Page(s): 1166 1173, May 2012

[10] https://www.electronicshub.org/gesture-controlled-robotic-arm/

https://www.engineersgarage.com/wireless-robotic-arm-using-zigbee/