International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Robot Arm Control Using Hand Gesture

Rivin Jose Melvin[1], Sravya Prem Naik[2], Fathima Insha A[3], Harith H[4], Elda Maria Joy[5]

[1] *Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
jose.melvin004@gmail.com
[2] *Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
sravyapremnaik2003@gmail.com
[3] *Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
fathimainsha901@gmail.com
[4] *Student, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
harithhc007@gmail.com
[5]*Assistant Professor, Dept of CSE, Sree Narayana Gurukulam College of Engineering, Kochi, India*
mariajoy123.2008@gmail.com

---***---

Abstract - T Robotic arm control has traditionally relied on mechanical input methods such as joysticks or coding commands. However, gesture recognition provides an intuitive alternative, especially in environments where direct control might be challenging. This paper explores the use of Mediapipe for gesture recognition to control robotic arm movements in real-time. By integrating gesture recognition with robotic systems, we can enhance user interaction, making robotic arms more accessible for users without extensive training.

Key Words: Gesture Recognition, Robotic Arm Control, Mediapipe, ROS, MoveIt

1.INTRODUCTION

In recent years, gesture recognition has emerged as a promising human-computer interaction (HCI) technology, allowing users to interact with machines more naturally. Robotics, a field that has heavily relied on manual and pre-programmed control systems, stands to benefit significantly from this development. Gesture-based robotic control has applications in various fields, such as medical surgery, remote handling, and assistance for individuals with disabilities. By using Mediapipe—a widely-used framework for building perception pipelines—gesture recognition for robotic arm control can be efficiently implemented.

2. LITERATURE REVIEW

In recent years, robotics has undergone substantial advancements, expanding its impact across domains like industrial automation, healthcare, education, and human-machine interaction. The collection of literature reviewed in this document offers a broad perspective on key robotics innovations, from operating systems to gesture recognition technologies, while also examining the role of robotics in education. This essay synthesizes the major themes of each paper reviewed, highlighting the evolution of robotic technologies and their applications.

Robot Operating System (ROS) has transformed how robotic systems are programmed and controlled. In “Robot Operating System 2: Design, Architecture, and Uses In The Wild,” the focus is on ROS 2, which addresses ROS 1’s limitations in scalability, security, and adaptability. ROS 2, based on the Data Distribution Service (DDS), supports reliable communication and customization, making it ideal for varied environments from land to space. This evolution of ROS demonstrates the importance of modular and secure architectures for advancing robotics and facilitating its deployment in complex, real-world scenarios.

The literature on gesture recognition illustrates advancements in radar-based and sensor-based recognition systems. “A Dynamic Continuous Hand Gesture Detection and Recognition Method with FMCW Radar” highlights the benefits of radar-based systems, such as non-contact interaction, which enhance user experience. Unlike sensor-based methods, radar-based recognition does not require wearable devices, thus providing a more intuitive interaction. Such advancements in gesture recognition underscore its potential for various applications, including control of smart devices and robotics, emphasizing the need for methods that balance accuracy with user convenience.

Robotics also plays a significant role in education, particularly in developing computational thinking. “Teaching Computing on Basic Education with Robotics” discusses how robotics workshops introduce young students to programming, problem-solving, and algorithmic thinking through hands-on activities. Studies reveal that robotics encourages engagement and facilitates learning complex subjects, such as mathematics and computer science, by providing tangible, interactive learning experiences. However, there is a noted gap in research on robotics education for early childhood, which, if addressed, could lay a stronger foundation for computational thinking at an earlier age.

SLAM (Simultaneous Localization and Mapping) is a critical aspect of robotics, particularly for autonomous navigation. In “Implementation of SLAM and path planning for mobile robots under ROS framework,” the integration of FastSLAM for map construction and the improved artificial potential field method for obstacle avoidance are discussed. These developments are essential for enabling robots to navigate dynamic environments safely. The implementation under ROS and Gazebo simulation environments reflects the practical applications of SLAM, ensuring that robots can effectively map and navigate real-world spaces, from autonomous vehicles to delivery robots.

The neuro-robotics paradigm merges neuroscience and robotics, as exemplified in "The Neuro-Robotics Paradigm: NEURARM, NEUROExos, HANDEXOS." This paper focuses on wearable robotic systems designed for rehabilitation, where bio-inspired control strategies mimic human motion control. Systems like NEURARM, NEUROExos, and HANDEXOS support users in rehabilitation, offering functional assistance while adapting to the natural range of human motion. The integration of neuroscience with robotics not only advances assistive technologies but also provides insights into human motor control, paving the way for more responsive and adaptive robotic systems.

In "A Robotic Arm Simulator Software Tool for use in Introductory Robotics Courses," an accessible simulation tool is introduced for teaching the basics of robotic arm control and kinematics. Unlike costly physical hardware, this tool offers a cost-effective way for students to learn about robotic arms through interactive simulation. By allowing students to experiment with path planning, inverse kinematics, and control, such tools make robotics education accessible to institutions with limited resources, fostering interest and understanding in engineering and robotics.

3.METHODOLOGY

This project involves developing a gesture recognition system using Mediapipe, integrated with a robotic arm modeled in ROS. The following sections outline the major steps involved.

A. Robotic Arm Simulation in ROS

The robotic arm model was imported from SolidWorks to ROS using the URDF (Unified Robot Description Format) file. The robot is set up in a CATKIN workspace and launched in Gazebo for simulation. MoveIt was used for setting up motion planning, allowing for the execution of predefined or generated motion trajectories. The MoveIt Setup Assistant helped configure control parameters and generate the ROS package for simulation. This setup enables real-time motion planning, facilitating smooth transitions and rotation of robotic arm joints.

B. Gesture Recognition with Mediapipe

Mediapipe offers pre-trained models that can be used to identify specific hand gestures. The hand tracking model from Mediapipe was employed to detect finger positions and recognize specific hand poses that correlate with robotic arm commands. A custom mapping of gestures to movements was established. For instance, an open palm could represent an idle state, a closed fist could initiate movement, and pointing gestures could correspond to directional commands for the arm.

C.System Integration

ROS and Mediapipe were connected through Python scripting. Hand gestures detected by Mediapipe trigger motion commands, which are then translated into trajectory points for the MoveIt package. Joint positions are controlled by the ROS joint_trajectory_controller, enabling smooth and natural movement. The controller publishes joint states at a specified rate to keep the simulation and control aligned, ensuring accurate execution of gesture-based commands

D. Testing and Calibration

Testing involved calibrating gestures to ensure high recognition accuracy and smooth control. Variations in hand orientation, speed of gestures, and lighting conditions were accounted for by adjusting recognition thresholds in Mediapipe. The system was tested for latency and response accuracy in a variety of environments, ensuring that commands were reliably interpreted and executed within the robotic simulation.

4.RESULTS

Gesture-based control was successfully implemented, with Mediapipe's hand tracking offering high accuracy in detecting gestures that corresponded to robotic movements. The robotic arm responded to gestures with minimal latency, and the integration with ROS allowed for a smooth simulation. The setup enabled intuitive control of the robotic arm, making complex actions such as pick-and-place achievable through simple gestures.

5.DISCUSSION

The integration of Mediapipe and ROS presents a robust solution for gesture-based robotic control. While the system performed well in controlled environments, real-world applications may require further refinement to handle unpredictable lighting and background interference. Additionally, expanding the system to recognize a broader set of gestures could increase functionality, but may require more computational resources.

6.CONCLUSION

This project successfully demonstrates a proof-of-concept for controlling a robotic arm using gesture recognition, highlighting the potential of combining cutting-edge technologies to create an intuitive, hands-free interface for robotic systems. By leveraging **Mediapipe**, a powerful framework for real-time hand and gesture tracking, the system achieves high accuracy in detecting and interpreting a wide variety of hand movements, which are then used to control the robotic arm's motions. Mediapipe's robustness in diverse lighting conditions and its efficiency in processing gestures make it a promising tool for applications in robotics, human-computer interaction (HCI), and assistive technology.

On the backend, **Robot Operating System (ROS)** and **MoveIt!** serve as the infrastructure for simulating and executing the robotic arm's motions. ROS provides a flexible, scalable framework for managing the robotic arm's hardware and software interfaces, while MoveIt! integrates motion planning, control, and manipulation capabilities, enabling the robotic arm to respond accurately to the gestures detected by Mediapipe. The integration of these platforms ensures that the robotic arm can perform complex tasks such as reaching specific positions, grasping objects, and executing sequential movements based on real-time input from the user.

Looking toward future developments, several areas offer opportunities for enhancement. One key improvement would be to **optimize gesture recognition accuracy in more dynamic or challenging environments**, where factors such as background clutter, varying lighting conditions, or occlusions might interfere with accurate detection. Exploring more advanced machine learning techniques and training the system with diverse datasets could help improve robustness in real-world scenarios.

Furthermore, the system currently relies on a limited set of gestures, which could be expanded to allow for more **complex and nuanced control gestures**. For example, incorporating multi-finger gestures, finger tracking, or even hand pose recognition could enable more precise control over the arm's movements, such as adjusting the gripper's force or executing intricate tasks. Developing an adaptive control system that learns from user behavior over time could also contribute to a more personalized and intuitive interaction model.

Additionally, **integration with other sensory inputs**, such as voice commands or eye-tracking, could further enhance the multimodal control of the robotic arm, creating a more seamless and versatile user experience. Combining gesture recognition with force or touch sensors on the robotic arm could also allow the system to respond in a more tactile way, providing feedback to the user in a manner that feels natural and intuitive.

In conclusion, this project lays the groundwork for more sophisticated human-robot interaction systems by demonstrating the feasibility of controlling a robotic arm through gestures. With further optimization and expansion of its capabilities, this system could have applications in various fields, from assistive robotics for individuals with disabilities to manufacturing environments where hands-free control is essential. As the field of gesture recognition and robotic control continues to advance, the integration of more sophisticated algorithms and sensors will undoubtedly lead to even greater levels of precision, adaptability, and usability.

7.REFERENCES

1. The Neuro-Robotics Paradigm: NEURARM, NEUROExos, HANDEXOS Tommaso Lenzi, Stefano De Rossi, Nicola Vitiello, Azzurra Chiri, Stefano Roccella, Francesco Giovacchini, Fabrizio Vecchi, Member, IEEE, Maria Chiara Carrozza, Member, IEEE
2. An Approach for Human Machine Interaction using Dynamic Hand Gesture Recognition, Mohd. Aquib Ansari, Dushyant Kumar Singh
3. Hand Component Decomposition for the Hand Gesture Recognition Based on FingerPaint Dataset In Seop Na* , Soo Hyung Kim**, Chil Woo Lee**, Nguyen Hai DuongK. Elissa, "Title of paper if known," unpublished.
4. A Dynamic Continuous Hand Gesture Detection and Recognition Method with FMCW Radar, Aihu Ren, Yong Wang, Xiaobo Yang, Mu Zhou
5. Implementation of SLAM and path planning for mobile robots under ROS framework, Zixiang Liu
6. A Robotic Arm Simulator. Software Tool For Use In Introductory Robotics Courses, Fernando Gonzalez, Janusz Zalewski
7. Robotic operating system 2: design, Architeccture and Use in the Wild, Steve Macenski,Tully Foote,Brian Gerkey,Cris Lalancette,William Woodall