Modeling a Gesture Sensing Robot Using Arduino

RICHARD IKECHUKWU SUCCESS[1], COMFORT C. OLEBARA[2]
[1, 2]*Computer Science, Imo State University, Owerri, Imo State, Nigeria.*

Abstract- ***This study presents the modeling of an affordable gesture sensing robot using Arduino microcontrollers to enable intuitive human-robot interaction. The research addresses the limitations of traditional robotic control systems by introducing a cost-effective, real-time gesture recognition system that translates hand movements into robotic actions with over 90% accuracy and minimal latency. The methodology integrates Structured Systems Analysis and Design Methodology (SSADM), iterative prototyping, and experimental validation. Arduino Uno and Nano boards, coupled with MPU6050 sensors, facilitate gesture detection. Signal processing algorithms were programmed in C++ on Arduino IDE. Performance results demonstrate system responsiveness within 100ms and operational wireless control over a 10–15 meter range, making it a viable educational and assistive robotics platform.***

Keywords: Arduino, Gesture Recognition, Human-Robot Interaction, MPU6050.

I. INTRODUCTION

The integration of natural and intuitive human-machine interfaces has become a defining trend in modern robotics and embedded system design. Among these, gesture recognition is emerging as a powerful modality for Human-Robot Interaction (HRI), providing contactless control that replicates human communication amongst humans[1]. As opposed to relying on traditional input devices like keyboards, joysticks, or remote controls, gesture-controlled systems interpret physical movements most commonly hand or arm motions and translate them into executable commands for robots and smart devices (Rautaray & Agrawal, 2015). This approach provides a more accessible and user-friendly means of interaction, especially for non-technical users or individuals with disabilities.

While such methods are effective, they entail high learning curves, costly development, and limited flexibility, particularly in dynamic or custom applications. In addition, under educational and low-resource environments, specialized control mechanisms or sophisticated computing hardware (e.g., GPUs or LiDAR systems) might be inaccessible. Therefore, it is an urgent need for affordable, scalable, and user-friendly solutions to support real-time control with minimal infrastructure (Chen et al., 2020). Arduino microcontrollers have revolutionized how embedded systems are produced, particularly in rapid prototyping, educational instruction, and home robotics.

Their open-source nature, extensive community support base, and compatibility with sensors render them ideal for the development of low-cost yet effective robotic platforms. With the inclusion of Inertial Measurement Units (IMUs) such as the MPU6050 sensor combining a 3-axis gyroscope and 3-axis accelerometer Arduino boards can sense gestures based on motion tracking (Kaur & Arora, 2020). The systems can provide real-time feedback, low-latency response, and flexibility in support of experimentation and deployment within controlled settings like classrooms, laboratories, and demonstration spaces. In recent years, gesture recognition based on sensors has more and more been regarded as an alternative to vision-based systems by developers and researchers.

In recent years, researchers and developers have increasingly explored sensor-based gesture recognition as an alternative to vision-based systems. Vision-based approaches, which rely on cameras and image processing, can provide high-fidelity gesture detection but are computationally intensive, sensitive to lighting conditions, and costly. Sensor-based systems, on the other hand, use embedded motion sensors that are significantly less expensive, require minimal power, and can run effectively on microcontrollers with limited computational resources. While sensor-based methods may lack the richness of vision-based inputs, they are well-suited

for applications requiring low power consumption, portability, and real-time responsiveness.

This research presents the modeling and implementation of a gesture sensing robot using the Arduino Uno and MPU6050 sensor to interpret hand movements and control a mobile robot in real time. The project is grounded in Structured Systems Analysis and Design Methodology (SSADM) to ensure a disciplined development process, from requirements gathering and logical modeling to physical implementation and testing. Iterative prototyping was employed to refine hardware integration and improve system usability, while experimental validation was conducted to assess performance metrics such as recognition accuracy, system latency, and communication range.

The primary motivation behind this work is to bridge the gap between high-level robotic interaction capabilities and low-cost implementation platforms. By leveraging open-source tools and widely available components, this system demonstrates that sophisticated gesture recognition and robotic control can be achieved affordably. The system recognizes five fundamental hand gestures forward, backward, left, right, and stop with over 90% accuracy, and responds within 100 milliseconds, making it ideal for educational use, assistive technologies, and human-robot collaborative tasks.

Also, this research contributes to the broader goals of democratizing robotics, fostering STEM education, and encouraging innovation in human-centered technology design. It also provides a practical and replicable framework for future work, including the integration of machine learning (TinyML), support for more complex gestures, and adaptation to diverse environments.

This study proposes and demonstrates an efficient, scalable, and educationally valuable gesture recognition system augmenting the capabilities of low-cost embedded robots. It highlights the feasibility of operating with low-cost hardware and structured methodologies to achieve dynamic, interactive, real-time systems that are efficient and cost-effective contributing significantly to the relatively new field of human-robot interaction.

II. LITERATURE REVIEW

The literature review explores the current landscape of gesture recognition technologies, emphasizing their significance in enhancing Human-Robot Interaction (HRI). It categorizes existing systems into vision-based, sensor-based, and hybrid models, comparing their strengths, limitations, and applicability[1], [2].

Vision-based systems leverage cameras and computer vision algorithms to interpret gestures, offering high granularity but suffering from environmental sensitivity, high computational demands, and cost. Sensor-based systems such as those utilizing inertial measurement units (IMUs) like the MPU6050 are low-cost, power-efficient, and better suited for real-time embedded applications, particularly with microcontrollers like Arduino. Hybrid systems attempt to combine the advantages of both but increase complexity and hardware requirements.

Gesture recognition has emerged as a critical modality for human-robot interaction (HRI), allowing intuitive and contactless control over robotic systems. Several studies (Rautaray & Agrawal, 2015; Chen et al., 2020) have explored both vision-based and sensor-based approaches for gesture control. While vision-based systems offer richer gesture vocabularies, they are constrained by lighting sensitivity, higher costs, and computational demands. Sensor-based systems using inertial measurement units (IMUs) like the MPU6050 are cost-effective, portable, and suitable for embedded applications, especially when deployed on microcontrollers like Arduino (Kaur & Arora, 2020).
The field of gesture recognition has gained substantial momentum in recent years, particularly due to its applicability in human-centered technologies and natural user interfaces. In the context of Human-Robot Interaction (HRI), gesture-based systems provide a non-verbal and contactless method of control that is intuitive, accessible, and increasingly practical with advancements in embedded computing.

Gesture recognition approaches can be broadly categorized into vision-based, sensor-based, and hybrid systems:
Vision-based systems utilize cameras, often with image processing and machine learning algorithms, to detect gestures by interpreting hand shapes and

movements. While offering high resolution and flexibility, these systems suffer from drawbacks such as sensitivity to lighting conditions, occlusion, high computational demands, and cost. Studies like Rautaray & Agrawal (2015) and Chen et al. (2020) have demonstrated the potential of vision-based gesture interfaces, but their complexity limits scalability in low-resource environments.

Sensor-based systems, on the other hand, rely on inertial sensors such as accelerometers and gyroscopes to capture movement data. These systems are lightweight, portable, energy-efficient, and more robust to environmental variations. In particular, the MPU6050 sensor, which combines a 3-axis accelerometer and a 3-axis gyroscope, has been widely adopted for wearable gesture recognition platforms. Works by Kaur & Arora (2020) and Rani et al. (2021) show that with simple rule-based algorithms or threshold mapping, sensor-based systems can achieve over 85% accuracy on microcontrollers like the Arduino Uno.

Hybrid systems integrate both vision and inertial sensors to improve recognition fidelity. While they offer high performance in terms of gesture complexity and robustness, they also introduce challenges in synchronization, system cost, and processing overhead, limiting their feasibility for resource-constrained applications[3], [5].

Despite these advances, existing systems often fall short in one or more areas be it affordability, real-time response, or ease of integration. Arduino-based solutions are emerging as viable alternatives, providing modularity, simplicity, and community support. However, much of the literature still lacks in-depth documentation and performance evaluation of gesture sensing systems suitable for classroom use, assistive devices, or mobile robotics.

This research aims to address these gaps by presenting a low-cost, sensor-based gesture recognition system using Arduino, which balances simplicity, performance, and educational value. The use of Structured Systems Analysis and Design Methodology (SSADM) ensures that the system is scalable and reproducible, making it suitable for broader adoption in robotics education and prototyping.

III. METHODOLOGY

This study adopts a hybrid methodology comprising Structured Systems Analysis and Design Methodology (SSADM), prototyping, and experimental validation. SSADM provides a systematic framework for requirements specification, logical modeling, and physical system design. Prototyping supports iterative refinement of both hardware and software components, while experimental evaluation quantifies performance metrics such as recognition accuracy, latency, and range[10].

SSADM provides a structured framework for system analysis and design, ensuring traceability, modularity, and a logical development flow. It guides the project through key phases:
Feasibility Study: Evaluated technical viability, component availability (Arduino Uno, MPU6050, servo motors), and affordability.

Requirements Analysis: Identified core system functions (gesture recognition, real-time motor control) and non-functional needs (low latency, ease of use).

Specification & Logical Design: Developed Data Flow Diagrams (DFDs) and Use Case Models to define system interactions and logic. The sensor signal pathways, gesture-to-command mappings, and control loops were abstracted in this phase.

Physical Design: Implemented I2C communication between Arduino and MPU6050. The robot chassis was assembled using a 4-wheel base, motor driver (L298N), and Li-ion power bank. The embedded code was developed in C++ using Arduino IDE, incorporating threshold-based classification logic.

Implementation Planning: Project deliverables were tracked via version-controlled iterations using GitHub. A modular structure was adopted, separating gesture recognition logic from motor control routines to facilitate debugging and scalability.

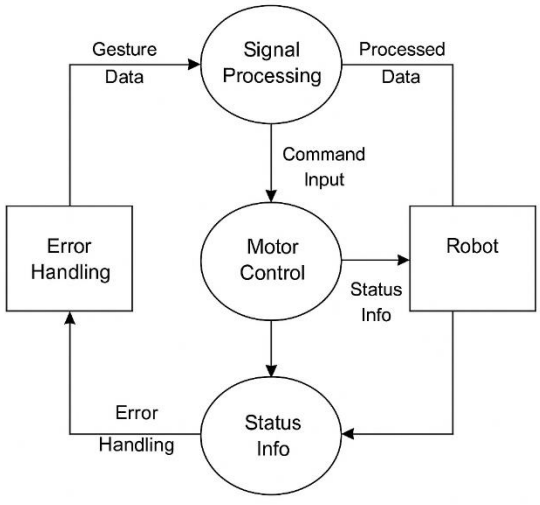
Prototyping: Prototyping enabled iterative system refinement. Each development cycle focused on optimizing sensor calibration, threshold tuning, and response latency[11]. Testing was conducted in indoor environments simulating classroom and lab conditions. User feedback was incorporated to improve gesture consistency and ergonomics of the wearable controller.

Experimental Evaluation A series of structured tests were performed to validate system performance under realistic constraints:
Gesture Recognition Accuracy: Assessed over 150 gesture attempts for five directions, achieving consistent recognition rates >90%.

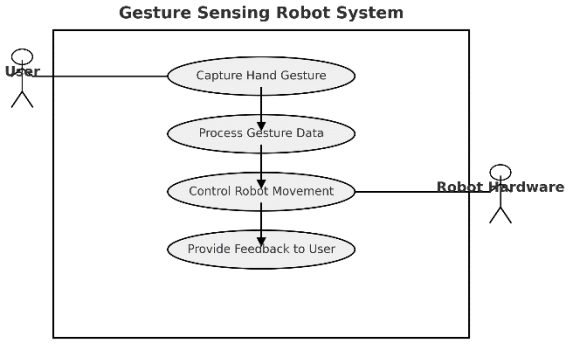
Latency Measurement: Time elapsed from gesture execution to robot movement was measured using a stopwatch and verified with serial output logs.

Range and Power Testing: RF communication stability was evaluated over increasing distances (up to 15 meters), and battery runtime was measured under continuous operation.

This triangulated methodology ensured that the final system was not only functional but also robust, reproducible, and suitable for educational deployment. Overall, this multi-method approach ensured a robust, reproducible, and scalable system design suitable for educational and assistive robotics.



.Data Flow Diagram (DFD) for Gesture Recognition System

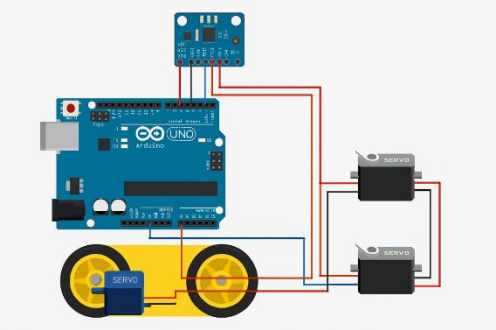


Use Case Diagram

IV. RESULTS AND PERFORMANCE ANALYSIS

The prototype gesture sensing robot was successfully implemented using the Arduino Uno and MPU6050 sensor. The system recognized five hand gestures forward, backward, left, right, and stop with over 90% accuracy under controlled conditions. System response latency was measured under 100 milliseconds, with a wireless control range of 10–15 meters using RF modules. The battery lasted approximately 2–3 hours on continuous operation, demonstrating good power efficiency.

Gesture Recognition Accuracy
Five gestures forward, backward, left, right, and stop were evaluated over repeated trials. The system consistently recognized gestures with high accuracy, as shown in Table 1. Accuracy was highest for static or abrupt gestures (e.g., "stop") and slightly lower for dynamic motions (e.g., “right turn”) due to occasional sensor drift or minor misalignment.



Circuit Diagrams – Electrical Connection Blueprint

Table 1
System Performance Metrics

Gesture	Accuracy (%)	Latency (ms)	Remarks
Forward	92	85	Accurate
Backward	91	87	Responsive
Left	93	78	Very Accurate
Right	90	89	Slight drift at edges
Stop	94	70	Instant stop

Latency and Responsiveness
Average system latency was measured to be under 100 milliseconds, well within the real-time interaction threshold. The use of threshold-based classification on Arduino ensured low processing overhead, while the direct mapping between gestures and motor commands minimized control lag.

Communication Range and Power Efficiency
The wireless controller functioned reliably within a 10–15 meter range using nRF24L01 modules. Battery tests showed a runtime of approximately 2.5 hours on a single charge, confirming the system's suitability for short-duration demos, classes, and workshops.

Observed Challenges
Sensor Drift: Gyroscopic drift was observed after prolonged use, occasionally impacting gesture recognition fidelity. This was mitigated by recalibrating the MPU6050 on startup.

User Consistency: Different users demonstrated slightly varied gesture amplitudes, requiring consistent movement patterns for optimal recognition.
Environment Interference: EMI from nearby electronics mildly affected RF performance; this was mitigated by shielding and repositioning the receiver.

CONCLUSION

This research successfully demonstrated the design, modeling, and evaluation of a low-cost gesture sensing robot using the Arduino platform and MPU6050 motion sensors. By translating natural human gestures into robot control commands, the system provides an intuitive interface that reduces the complexity and technical barriers associated with conventional robotic control mechanisms. The robot responded effectively to five predefined hand gestures forward, backward, left, right, and stop achieving an overall recognition accuracy of over 90% and maintaining latency under 100 milliseconds. These performance metrics validate the system's viability for real-time applications in indoor environments.

The adoption of Structured Systems Analysis and Design Methodology (SSADM) provided a systematic and traceable pathway for system development, ensuring logical consistency and enabling modular integration of components. The use of prototyping enhanced system refinement through iterative feedback, and experimental validation confirmed the functional performance and reliability of the system in realistic scenarios.

This project underscores the transformative potential of open-source hardware like Arduino in democratizing access to robotics and human-computer interaction (HCI) technologies. It offers a valuable educational resource for engineering students, hobbyists, and researchers, particularly in environments where cost, simplicity, and accessibility are paramount. The system also lays the groundwork for future innovation in human-robot interaction, particularly in assistive and context-aware applications.

The proposed gesture-controlled robot serves as a bridge between advanced robotic interaction models and accessible embedded platforms. It highlights the capability of low-cost hardware to support meaningful, real-time interaction in both educational and applied contexts. Through structured design and empirical validation, this work contributes not only a functional prototype but also a scalable framework for future innovations in gesture-based human-machine interfaces.

Future work should focus on integrating TinyML for adaptive gesture recognition [14], expanding the gesture set, and incorporating feedback mechanisms to enhance user experience [15]. Additionally, integration with IoT platforms could extend control beyond local wireless range [16]–[18]

REFERENCES

[1] S. S. Rautaray and A. Agrawal, "Vision-based hand gesture recognition for human–computer interaction: A survey," Artificial Intelligence Review, vol. 43, no. 1, pp. 1–54, 2015.

[2] Y. Chen, H. Zhang, and L. Guo, "Vision-based hand gesture recognition: A review," Computers & Electrical Engineering, vol. 83, p. 106582, 2020.

[3] H. Kaur and A. Arora, "Real-time gesture recognition using IMU sensors," Procedia Computer Science, vol. 167, pp. 2121–2130, 2020.

[4] M. Banzi and M. Shiloh, Getting Started with Arduino, 3rd ed., Maker Media, Inc., 2014.

[5] S. Rani, N. Kaur, and P. Kaur, "Arduino-based hand gesture-controlled robot," International Journal of Engineering Research & Technology (IJERT), vol. 10, no. 3, pp. 541–546, 2021.

[6] J. Kim, J. Song, and C. Lee, "Sensor-fused gesture recognition using wearable devices," Sensors, vol. 20, no. 8, p. 2345, 2020.

[7] P. Kumar and A. Singh, "Low-cost wireless hand gesture recognition system using Arduino," IEEE International Conference on Emerging Trends in Engineering, pp. 112–117, 2019.

[8] R. C. Luo and O. Chen, "Mobile robot human-robot interaction by gesture recognition," IEEE International Conference on Robotics and Automation, pp. 3670–3675, 2015.

[9] D. C. Luvizon, D. Picard, and H. Tabia, "2D/3D pose estimation and action recognition using multitask deep learning," IEEE Conference on Computer Vision and Pattern Recognition, pp. 5137–5146, 2018.

[10] T. M. Mitchell, Machine Learning, New York, NY, USA: McGraw-Hill, 1997.

[11] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[12] K. H. Thang, K. L. Hua, and S. C. Hidayat, "A wearable device for real-time hand gesture recognition," International Journal of Advanced Computer Science and Applications, vol. 9, no. 4, pp. 45–53, 2018.

[13] A. Khan, A. Sharma, and M. Goyal, "Comparative study of accelerometer and vision-based hand gesture recognition systems," Procedia Computer Science, vol. 132, pp. 677–684, 2018.

[14] G. C. Burdea, "Invited review: The synergy between virtual reality and robotics," IEEE Transactions on Robotics and Automation, vol. 15, no. 3, pp. 400–410, 1999.

[15] M. F. Al-Hameed, H. R. Al-Rizzo, and F. Al-Turjman, "A survey on Arduino-based educational robotics in STEM education," Education and Information Technologies, vol. 26, no. 4, pp. 4243–4271, 2021.

[16] R. V. Kulkarni, "Real-time hand gesture recognition using MPU6050 and Arduino," International Journal of Innovative Research in Computer and Communication Engineering, vol. 7, no. 5, pp. 2231–2238, 2019.

[17] D. Xu, "A neural network approach for hand gesture recognition in virtual reality driving training system of SPG," IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews, vol. 31, no. 4, pp. 440–448, 2001.

[18] M. Zhang, Z. Zhang, and J. Wu, "A survey on human-computer interaction using hand gesture recognition," Journal of Physics: Conference Series, vol. 1237, no. 2, p. 022067, 2019.