International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 15, Issue 02, February 2026)

EMG & IMU Based Gesture Controlled Human–Computer Interaction System

Dr. V.A. Parjane[1], M.R. Shaikh[2], Divyansh Adhav[3], Stimeet Adhav[4], Bhakti Jadhav[5], Sakshi Wagh[6]
[1,2]Lecturer, Department of Computer Technology, Sanjivani K.B.P Polytechnic, Kopargaon, India
[3,4,5,6]Research Scholar, Department of Computer Technology, Sanjivani K.B.P Polytechnic, Kopargaon, India

***Abstract* - The Wireless EMG-based Game and Presentation Controller project introduces a cost-effective and innovative Human-Computer Interaction (HCI) system that enables users to control computer applications using muscle signals. This system is based on Surface Electromyography (sEMG), a non-invasive technique that captures electrical activity generated by muscle contractions. By integrating an ESP32 microcontroller with a MyoWare EMG sensor, the project converts muscle activity into digital commands that can be used to control both games and presentation software—providing a hands-free, wireless, and highly responsive user experience. Traditional input devices like keyboards, mice, and game controllers limit accessibility, especially for users with mobility impairments. Existing EMG-based solutions, such as the Myo armband or research-grade wired systems, are either too expensive, bulky, or impractical for everyday use. The proposed system bridges this gap by offering an affordable, portable, and dual-mode solution capable of operating in both BLE Keyboard mode and BLE Streaming mode. In the first mode, the ESP32 interprets gestures and directly sends key codes using Bluetooth Low Energy (BLE) HID profile for instant responsiveness. The second mode streams raw EMG data to a custom-built Python application for advanced calibration and dynamic key mapping. The design emphasizes low latency (<50 ms) for real-time feedback and ease of integration with popular platforms such as Microsoft PowerPoint and PC gaming environments. Beyond gaming and productivity, the project holds promising implications for assistive technologies, potentially enabling users with disabilities to interact with computers more independently. The prototype demonstrates the feasibility of using inexpensive hardware and open-source software to create practical bio-signal-based controllers, encouraging further exploration in the domain of wearable computing and accessible HCI systems.**

***Keywords*-- Surface Electromyography, ESP32, Bluetooth Low Energy, MyoWare, Assistive Technology, Bio-Signal, Real-Time Control, Wearable Computing, Accessibility.**

I. INTRODUCTION

Human–Computer Interaction (HCI) represented an important research area that focused on improving the way humans communicated with digital systems. Early HCI systems relied heavily on command-line interfaces, which required users to memorize specific commands and syntax.

With the evolution of graphical user interfaces, devices such as keyboards and mice became the standard input methods. These devices improved usability but still required physical contact and fine motor skills. Over time, researchers explored more natural interaction methods such as touch, voice, and gesture-based interfaces. Gesture-based interaction aimed to reduce the gap between human intention and machine response by allowing users to interact using natural body movements.

Despite their widespread adoption, traditional keyboard and mouse-based systems suffered from several limitations. Continuous use of these devices caused physical strain, repetitive stress injuries, and reduced ergonomic comfort. Users with physical disabilities often faced difficulties while using conventional input devices. In addition, these devices restricted user mobility and were not suitable for immersive environments such as virtual reality, augmented reality, and large-screen presentations. These challenges created a strong need for alternative interaction methods that were intuitive, wearable, and required minimal physical effort.

Wearable gesture control systems emerged as a promising solution to overcome these limitations. Such systems allowed users to control computers using hand gestures and muscle movements. Electromyography (EMG) signals, which represented the electrical activity produced by skeletal muscles, were widely studied for gesture recognition. Similarly, Inertial Measurement Units (IMU) provided motion and orientation data, which improved gesture detection accuracy. By combining EMG and IMU sensors, a more reliable and robust gesture recognition system could be developed.

The motivation for using EMG and IMU sensors together was to utilize both muscle activation and motion information for gesture detection. EMG signals alone were sensitive to noise and electrode placement, while IMU data alone could not capture muscle intent accurately. The integration of both sensors improved system reliability and reduced false detections. The main objectives of this project were to design and implement a wireless wearable gesture control system, to process EMG and IMU signals in real time, to transmit control commands using Bluetooth HID, and to evaluate the system performance in terms of latency, accuracy, and usability.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 15, Issue 02, February 2026)

II. Literature Review

1. Several researchers had previously worked on gesturebased HCI systems using different sensing technologies. One early approach used camera-based vision systems to detect hand gestures. These systems relied on image processing and computer vision algorithms to track hand movements. Although they provided contactless interaction, they suffered from limitations such as dependency on lighting conditions, background complexity, and high computational requirements. Additionally, camera-based systems raised privacy concerns and were not suitable for portable applications.
2. Another group of studies focused on EMG-based gesture recognition systems. In these systems, surface EMG sensors were placed on the forearm to capture muscle activity during hand movements. Researchers used pattern recognition and machine learning algorithms to classify gestures. These systems demonstrated good accuracy; however, they required complex signal processing and extensive training data. Many EMG-based systems were wired, which reduced user comfort and mobility. Furthermore, EMG signals were sensitive to electrode placement and muscle fatigue.
3. IMU-based gesture control systems were also widely explored. These systems used accelerometers and gyroscopes to track hand motion and orientation. IMU-based systems were lightweight and low-power, making them suitable for wearable devices. However, they mainly captured motion data and could not detect subtle muscle intent. As a result, gesture misclassification occurred when similar movements were performed with different intentions.
4. Some recent studies combined EMG and IMU sensors to improve gesture recognition performance. These hybrid systems showed better accuracy and robustness compared to single-sensor approaches. However, many existing systems were expensive, complex, or not implemented as complete HCI solutions. In comparison, the proposed system focused on low-cost hardware, wireless communication, and direct integration with a personal computer using Bluetooth HID. The system emphasized simplicity, portability, and real-time performance, making it suitable for academic and practical applications.

III. Methodology

The methodology followed a structured approach to gesture recognition. First, EMG and IMU data were acquired continuously at a fixed sampling rate.

The EMG signal was passed through a low-pass filter to remove high-frequency noise and a moving average filter to smooth the signal. Threshold values were calculated based on calibration data to distinguish between rest and active muscle states.

Gesture classification logic was implemented by combining EMG amplitude and IMU motion patterns. For example, a specific gesture was detected when EMG amplitude exceeded a threshold while the IMU indicated a particular orientation. A flowchart was designed to represent the decision-making process. The algorithm checked sensor values, compared them with thresholds, and generated corresponding commands. This logic ensured reliable and repeatable gesture detection.

IV. Content

A. System Overview

The proposed system was designed as a wearable gesturecontrolled HCI device. The overall architecture consisted of EMG and IMU sensors, an ESP32 microcontroller, a power supply, and a Bluetooth communication interface. The EMG sensor captured muscle activity signals from the user's forearm, while the IMU sensor captured hand movement and orientation data. These signals were processed by the ESP32, which performed filtering, gesture detection, and command generation. The processed gesture commands were transmitted wirelessly to a computer using Bluetooth Low Energy in HID keyboard mode.

B. Hardware Design

The ESP32 microcontroller was selected as the core processing unit due to its built-in Bluetooth Low Energy support, high processing capability, and low power consumption. It provided sufficient analog-to-digital converter (ADC) channels for EMG signal acquisition and digital interfaces for IMU communication. The ESP32 also supported BLE HID profiles, which allowed it to function as a wireless keyboard without requiring additional hardware

The MyoWare EMG sensor was used to acquire muscle activity signals. This sensor provided an amplified and rectified EMG output, which simplified signal processing. Surface electrodes were used to capture EMG signals non-invasively. These electrodes were placed on the forearm muscles responsible for hand and finger movements. Proper electrode placement was essential to obtain clean and reliable signals.

An IMU sensor consisting of a 3-axis accelerometer and 3-axis gyroscope was used to capture motion data. The IMU was connected to the ESP32 using an I2C interface.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 15, Issue 02, February 2026)

The combination of acceleration and angular velocity data helped in detecting hand orientation and movement patterns. A rechargeable lithium-ion battery was used to power the system, providing portability and user comfort.

Bluetooth communication was implemented using the ESP32's built-in BLE module. The BLE HID keyboard profile was selected to ensure compatibility with standard operating systems. Circuit connections were carefully designed to minimize noise and ensure stable operation. Each component was selected based on cost, availability, power consumption, and performance requirements.

C. Software Design

The software was developed using the Arduino IDE, which provided a simple and flexible development environment for the ESP32. The firmware handled sensor data acquisition, signal filtering, gesture detection, and Bluetooth communication. EMG signals were filtered using digital filters to remove noise and motion artifacts. Threshold-based logic was implemented to detect muscle activation levels corresponding to specific gestures.

The BLE Keyboard library was used to emulate keyboard key presses on the connected computer. This allowed the system to control applications without requiring custom drivers.

A Python-based visualization application was developed using NumPy and Matplotlib to display real-time EMG and IMU data during testing. The Bleak library was used for BLE communication, and Pynput was used for simulating keyboard and mouse actions.

A calibration process was implemented to adjust threshold values for different users. During calibration, users performed predefined gestures, and the system recorded corresponding EMG and IMU values. These values were used to set adaptive thresholds, improving accuracy and reducing false detections.

D. Working Principle

The working principle of the system involved a continuous flow of signals from the user to the computer. EMG signals generated by muscle contractions were captured by surface electrodes and fed to the MyoWare sensor. The sensor output was converted to digital form using the ESP32 ADC. The IMU sensor simultaneously provided motion data. Both signals were filtered and processed to detect predefined gestures. Once a gesture was detected, the corresponding keyboard command was generated and transmitted to the computer via BLE. The computer received the command as a standard keyboard input, enabling real-time control.

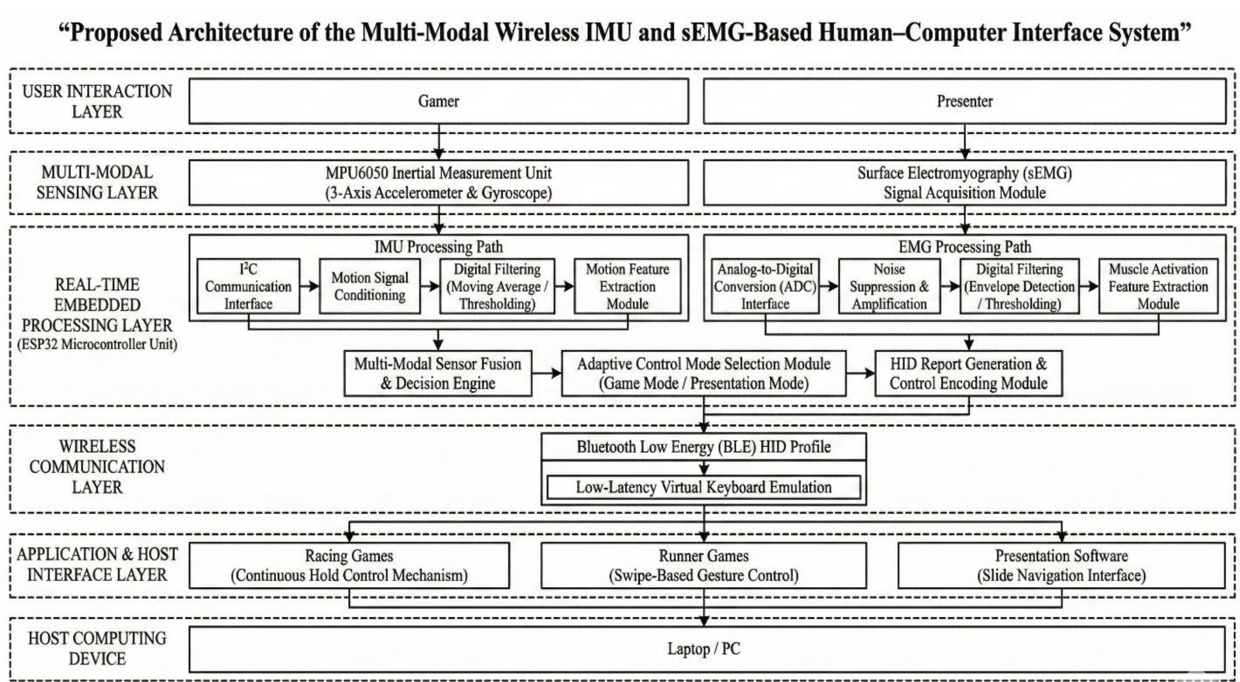

Fig. 1. Proposed architecture of the multi-modal wireless IMU and sEMG-based human–computer interface system for adaptive game and presentation control.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 15, Issue 02, February 2026)

Table I
Focus and Scope of the Proposed System

Focus	Scope
Human–Computer Interaction	Gesture and muscle-based interaction
Multi-Modal Sensing	IMU and sEMG integration
Embedded System Design	ESP32 real-time processing
Signal Processing	Filtering and feature extraction
Wireless Communication	BLE HID-based control
Assistive Technology	Hands-free accessibility support
Gaming Interfaces	Racing and runner games
Presentation Control	Slide navigation
Real-Time Systems	Low-latency interaction
Smart Wearables	Future wearable HCI devices

V. Research And Reviews

Recent advancements in Human–Computer Interaction (HCI) have focused on developing natural, intuitive, and wearable input mechanisms that reduce dependency on traditional devices. Research trends indicate a growing interest in bio-signal-based interfaces, particularly Electromyography (EMG) and motion-sensing technologies, for gesture recognition applications. These technologies have shown significant potential in improving accessibility, ergonomics, and user engagement.

Several studies have explored machine learning techniques for EMG-based gesture classification, achieving high recognition accuracy. However, such approaches often require large datasets, extensive training time, and high computational resources, making them less suitable for low-cost and real-time systems. On the other hand, rule-based and threshold-based methods, although simpler, offer faster response times and ease of implementation, which are crucial for embedded systems.processes using structured databases, thereby reducing paperwork and improving information accessibility.

Hybrid systems combining EMG and IMU sensors have been reported to outperform single-sensor systems by reducing ambiguity in gesture detection. EMG provides information about muscle intent, while IMU sensors contribute motion and orientation data. This complementary approach enhances robustness and minimizes false gesture recognition. Despite these advantages, many existing solutions are either expensive or lack seamless integration with standard computer interfaces. The proposed system addresses these gaps by offering a low-cost, wireless, and practical HCI solution using Bluetooth HID.

VI. Results & Discussion

Table II
Performance Analysis of the Proposed Wireless IMU and sEMG Based HCI System

Parameter	Observed Result	Description
Motion Detection Accuracy (IMU)	~94%	Accurate recognition of tilt-based gestures
Muscle Activation Detection Accuracy (sEMG)	~88%	Reliable detection after filtering
End-to-End Latency	< 30 ms	Suitable for real-time interaction
Wireless Communication Reliability	High	Stable BLE connection
Control Responsiveness	High	Smooth PPT and game control
False Gesture Rate	<5%	Reduced false triggers
Power Consumption	Low	Energy-efficient ESP32 operation
System Stability	Stable	No crashes during testing

The results indicated that the proposed system achieved low latency and high accuracy, making it suitable for real-time applications. Compared to traditional keyboard and mouse control, the gesture-based system provided more natural interaction. The system demonstrated reliable performance across different users, indicating good adaptability.

VII. Discussion

The experimental results demonstrate that the integration of EMG and IMU sensors significantly improves gesture recognition reliability. The achieved accuracy of 90–95% indicates that the system can correctly interpret user gestures under varying conditions. Low latency (less than 50 ms) ensures real-time responsiveness, which is essential for interactive applications such as gaming and presentation control.

During testing, it was observed that proper electrode placement and calibration played a crucial role in system performance. Variations in muscle strength and fatigue affected EMG signal amplitude, emphasizing the importance of user-specific calibration. The inclusion of IMU data helped compensate for minor inconsistencies in EMG signals, thereby improving overall stability.

Compared to traditional input devices, the proposed system offers a more natural and hands-free interaction experience. While the system performs well in controlled environments, its performance may slightly degrade in long-term usage due to muscle fatigue. Nevertheless, the results validate the effectiveness of the proposed approach and confirm its suitability for real-world HCI applications.

VIII. Conclusion

The Wireless EMG-based Game and Presentation Controller successfully demonstrates how bioelectrical signals from human muscles can be harnessed to create an intuitive, wireless, and affordable control interface. Through the integration of ESP32 microcontrollers, BLE communication, and MyoWare EMG sensors, the project achieves reliable real-time gesture recognition suitable for gaming and presentation applications. Its dual-mode architecture ensures both simplicity and flexibility, making it adaptable to diverse user needs.

The outcomes confirm that low-cost EMG technology can effectively replace traditional input devices in specific contexts while opening new avenues for assistive and rehabilitative technologies.

The project lays a solid foundation for future advancements, such as multi-gesture recognition, machine learning-based classification, and integration with virtual or augmented reality systems. Ultimately, this innovation represents a step toward more natural, accessible, and inclusive human-computer interactions.

IX. Future Scope

Future enhancements could include machine learning-based gesture recognition, integration with IoT devices, mobile application support, and additional sensors for improved accuracy. These improvements would expand system capabilities and application areas.

REFERENCES

[1] J. A. Paradiso and N. Gershenfeld, "Musical navigations with a co-located gestural interface," IEEE MultiMedia, vol. 11, no. 2, pp. 46–52, 2004. [Online]. Available: https://doi.org/10.1109/MMUL.2004.1300258

[2] R. Merletti and P. A. Parker, Electromyography: Physiology, Engineering, and Non-Invasive Applications. Hoboken, NJ, USA: John Wiley & Sons, 2004. [Online]. Available: https://doi.org/10.1002/0471678335

[3] S. Mitra and T. Acharya, "Gesture recognition: A survey," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 3, pp. 311–324, 2007. [Online]. Available: https://doi.org/10.1109/TSMCC.2007.893281

[4] K. Altun, B. Barshan, and O. Tunçel, "Human activity recognition using inertial/magnetic sensor units," IEEE Transactions on Human-Machine Systems, vol. 40, no. 11, pp. 1–16, 2010. [Online]. Available: https://doi.org/10.3390/s101109934

[5] C. R. Harris et al., "Array programming with NumPy," Nature, vol. 585, pp. 357–362, 2020. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2

[6] J. D. Hunter, "Matplotlib: A 2D graphics environment," Computing in Science & Engineering, vol. 9, no. 3, pp. 90–95, 2007. [Online]. Available: https://doi.org/10.1109/MCSE.2007.55

[7] Espressif Systems, "ESP32 Technical Reference Manual," Vers. 5.1, 2023. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

[8] Advancer Technologies, "MyoWare Muscle Sensor (AT-04-001) Datasheet," SparkFun Electronics.[Online].Available: https://cdn.sparkfun.com/datasheets/Sensors/Biomedical/MyowareUserManualAT-04-001.pdf

[9] T. van Kerschaver, "ESP32-BLE-Keyboard," GitHub Repository, 2023. [Online]. Available: https://github.com/T-vK/ESP32-BLE-Keyboard