Iqra Journal of Engineering & Computing
Vol. 02, No. 1, (March 2026) 36 – 52
DOI 10.21621/ijec.20260101.03

Myoelectric Control based on Machine Learning of a Low-Cost 7-DOF Transhumeral Prosthetic Arm through TinyML

Misbah Anwer[1*], Muhammad Fahad[1], Anusha Hasan[1], Abdul Karim Hasan[1], Falak Shah[1], Rafia Khan[1]

ABSTRACT

Upper-limb amputations are a significant source of disability, especially in resource-limited environments where expensive commercial-grade prosthetics (often costing over USD 70,000) are unaffordable. This paper describes the design, optimisation and pilot testing of a 7-degree of freedom (DOF) transhumeral motorised prosthesis controlled by TinyML-based myoelectric decoding. The gesture-recognition accuracy (F1-score) of 95.5% across five able-bodied participants (gesture-level accuracy: approximately 92.3% for the most difficult gesture) with an end-to-end response time of 100ms is achieved using an affordable ESP32 microcontroller and a Cost-Complexity Pruned (CCP) Random Forest (RF) classifier. This paper outlines the hardware-software co-design, mathematical formulation of surface Electromyography (sEMG) processing (digital Z-transforms) and a 60-second subject-specific incremental calibration routine. The design includes a dual-rail power supply network and an 8-channel analog multiplexer to address hardware limitations of low-cost microcontrollers. With a total bill of materials (BOM) of PKR 60,000-80,000 (USD 215-286) these preliminary findings are a good step toward value-based, low-cost assistive technology that may be deployed at low cost in resource-poor settings.

Keywords*:* : sEMG, TinyML, Prosthetics, Edge Computing, Random Forest, ESP32, Incremental Learning, Z-Transform, Signal Processing, 7-DOF

Author's Affiliation:
Institution(s) Name: [1] Karachi Institute of Engineering & Technology (KIET), Karachi
Country: Pakistan
Corresponding Author's Email: [1]misbah@kiet.edu.pk

3105-4528 (Online), 3105-451X (Print) © 2026.

1. Introduction

The amputation of one of the upper limbs, especially at the transhumeral (above elbow) level, entails a multi-dimensional pressure on the person. In addition to the immediate physical and functional deficits it causes, it results in extreme psychological torment, loss of autonomy, and economic marginalization. In low-to-middle-income countries (LMICs) such as Pakistan, where manual work is often a main source of livelihood, the inability to afford rehabilitative technology exacerbates this crisis into a socio-economic compounding loop. In regions like Pakistan, providing a clear financial and technical roadmap is essential for sustainable assistive technology.

Commercial myoelectric hands, including Ottobock Michelangelo or Taska Hand, are technologically impressive marvels. But they are usually configured as Black Box systems that have proprietary software and sensitive internal mechanisms and are extremely expensive to maintain. Clinical studies have shown that more than 40 percent of amputees in developing countries who obtain donated prosthetics are ultimately forsaken with them. This desertion is caused by excessive weight, high response latency, and a lack of intuitiveness in control, and local repair is impossible. Our motivation is to democratize high-end myoelectric control by means of "Value Engineering" the practice of compiling high-level code to run on cheap, easy-to-find silicon without fault.
Converting clinical, PC-tethered laboratory prototypes to a daily wear, portable assistive solution to the Paradox of Robustness needed in the device. It should be a prosthetic which is computationally powerful enough to decode non-linear human muscle intent that is complex, yet energy efficient enough to operate on a portable battery pack for the full day (12 hours). Moreover, the system should handle live biometric information without using cloud computing, which guarantees user privacy and zero-latency operation when not connected to the internet.

The paper will elaborate much more on the available literature on assistive technology by providing the following core contributions:

1) Multiplexed Architecture: A localized acquisition stage of 8 channels sEMG using a single-ADC multiplexing plan to address hardware constraints of ESP32.
2) Mathematical Optimization: Time-domain (TD) derivation and implementation features and Z-transformed digital filters that have been optimized particularly to work with 32-bit floating-point units.
3) TinyML Deployment: Cost-complexity Pruned (CCP) Random Forest model that fits 3000 items and more in 87KB of Flash memory.
4) Power Distribution: A strong dual-rail power distribution design to support transhumeral actuation of the elbow in high torque but without brown-out resets on sensitive logic circuits.

2. Comprehensive Literature Review

The previous research in myoelectric prosthetics can be grouped into three streams that converge: sEMG data collection techniques, machine learning classifiers for sEMG decoding, and embedded systems implementation. This section reviews each stream and discusses the current work in the context of other approaches.

2.1 sEMG Signal Acquisition Methods

Historically, sEMG-driven prosthetics used simple threshold-based control, limiting capabilities. Chopra and Emran (2024) [1] pinpointed latency as a key factor: if the mechanical response is delayed more than 300ms, the sense of embodiment is disrupted, leading to lower acceptance. Recent advances have provided more complex sensing. Welburn et al. (2025) [2] embedded MEMS accelerometers in the fingertips to provide tactile slip detection, leading to more secure grasping. However, MEMS sensors are expensive for low-resource scenarios; our work instead relies on lower-cost Force Sensitive Resistors (FSRs) to achieve similar slip-detection capabilities. Gohari et al. (2025) [3] also pushed the boundaries of sensing by integrating joint torque feedback with learning-based impedance control to allow compliant adaptation to unstructured environments. Unlike dense EMG [14], our 8-channel multiplexed sEMG design achieves signal richness with the hardware limitations of a single-ADC microcontroller. While [1] and [3] focus on high-cost sensing, our work fills the gap by achieving high signal richness using an 8-channel multiplexed design on a single-ADC system.

2.2 *Machine Learning Approaches for Gesture Classification*

While high-end research focuses on intuitive neural-linked control for a seamless human-AI bridge [11], our goal is to achieve similar results on low-cost hardware.The complexity of supervised classifiers for sEMG gesture recognition ranges widely. Support Vector Machines (SVMs) and k-Nearest Neighbour (kNN) have been popular due to their simplicity, but are sensitive to non-stationary noise in sEMG data [8]. Random Forest (RF) classifiers have proven more robust to noise; Sensors (MDPI, Article 5231) [6] achieved over 99% accuracy on stationary data. But ensemble models were deemed too slow for microcontrollers. Sensors (MDPI, Article 2063) [7] developed a lightweight Multi-Layer Perceptron (MLP) for embedded training, at the expense of accuracy. The current study brings together both methods by pruning an RF using Cost-Complexity Pruning (CCP) [14] and using micromlgen [13] to convert the model to achieve a robust ensemble model at 87KB of Flash memory. Finally, on adaptive learning, Zhang (2025) [16] showed that incremental sEMG classification can be performed on the device, motivating our 60-second user-specific calibration protocol. Unlike the lightweight MLP in [7] which trades accuracy for speed, our CCP-pruned RF achieves a 95.5% F1-score while maintaining sub-30ms inference.

2.3 *Embedded Systems and TinyML Deployment*

Machine learning inference on microcontrollers has been supported with frameworks like TensorFlow Lite Micro and micromlgen. Davids (2024) [15] evaluated tree-based models on ESP32 microcontrollers, showing the potential for real-time classification at under 20ms per prediction. Khan (2025) [17] investigated multiplexing options for single-ADC uCs, informing our design of a 8:1 MUX with the CD74HC4051 chip. Chen et al. (2025) [21] have shown real-time myoelectric processing and edge-AI systems on low-power processors, while Gupta (2024) [19] has investigated RF pruning approaches for

embedded systems. Importantly, there has been no previous work integrating multiplexed sEMG sensing, CCP-pruned RF inference, dual-rail power regulation, and a 60-second incremental calibration routine into incremental calibration routine into a single low-cost (<USD 300) transhumeral prosthesis, which is the focus of the current study

3. System Design and Hardware Prototyping

3.1 Hardware and Algorithmic Selection Rationale

Rigorous alternative component analysis was carried out to make sure that the prosthetic is highly functional and cost-effective for research in resource constrained environments. **Table I** justifies our key technology choices, comparing our selected algorithms and hardware with standard choices.

Table I: Component and Algorithmic Selection Justification Vs Alternatives

Component / Feature	Selected Technology	Alternative Solutions	Justification of Selection
Classification Algorithm	Pruned Random Forest (RF)	SVM or KNN	RF runs faster on ESP32 and can better deal with noisy sEMG data compared to KNN or SVM.
Control Logic	Adaptive Thresholding	PID Control	Adaptive Thresholding is not as mathematically intensive as PID, which leaves important CPU cycles to AI.
Input Sensors	sEMG Sensors	FMG or MMG	sEMG is the clinical gold standard, and can detect intent faster than FMG bands.
Pressure Sensors	FSR (Force Sensors)	MEMS Capacitive	FSRs are much less expensive and physically more robust towards a 3D-printed hand than delicate MEMS.
Processing Unit	ESP32 Microcontroller	Arduino Nano or RPi Pico	ESP32 has dual-core parallel processing and has an inbuilt wireless connection that the Nano does not have.
Feedback Actuators	ERM Vibration Motors	Piezoelectric Actuators	ERM motors are easier to operate, and give a more intuitive 'slip alert' via the socket.
Movement Actuators	High-Torque Servos (RDS3235)	Stepper or DC Motors	Servos include on-board position control simplifying the pro-gramming of a particular grip.

Table I. Component and algorithmic selection justification compares the hardware and software technologies selected (Pruned RF, ESP32, FSR, ERM motors) with standard alternatives (SVM/KNN, Arduino, MEMS, Piezoelectric) and their reasons are given.

The general connection of these elements can be seen in Figure. 1 which outlines the entire system architecture and the data flow starting with the first sEMG signal acquisition to the final motor actuation.

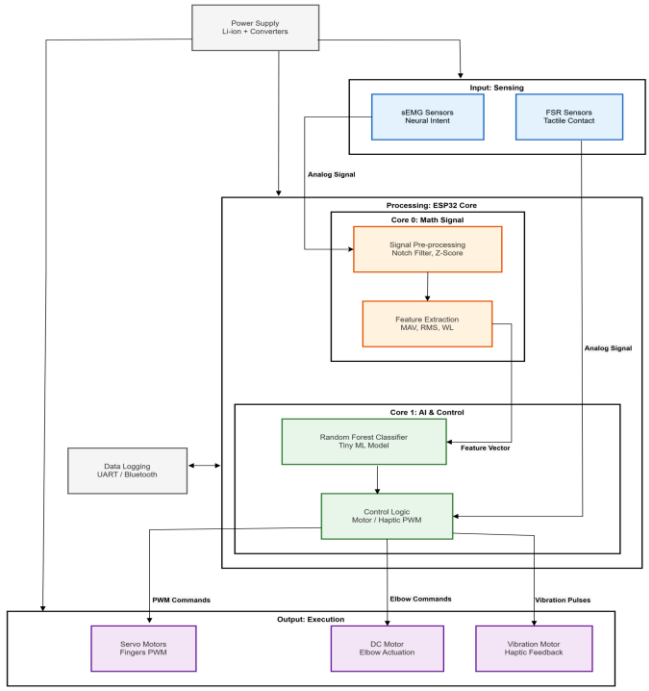

Figure. 1*. System Architecture detailing the flow from sEMG acquisition to motor actuation.*

3.2 7-DOF Actuation Strategy

The 8-channel configuration follows established benchmarking standards for transhumeral sEMG control datasets [12]. The mechanical assembly will mimic the kinematics of the human arm in seven degrees of freedom (DOF):

- Transhumeral Elbow Flexion/Extension: Pulled by the high-torque RDS3235 servo motor (35 kg-cm stall torque).
- Wrist Pronation/Supination: This action is performed by an MG996R standard servo (11kg-cm).
- Five-Finger Adaptive Grasp: All five digits are independently actuated by an MG90S metalgear micro-servo fitted in the cavity of the forearm, using a synthetic tendon routing system.

3.3 Signal Acquisition Front-End and Multiplexing

The normal amplitudes of surface EMGs ranges are 50-5000 uv. We are using MyoWare 2.0 sensors, which provide dedicated Instrumentation Amplifier with >80dB CMRR and a hardware rectifier. In order to read 8 sEMG electrodes through the safe ADC1 pins on the ESP32 (avoiding ADC2 conflicts with Wi-Fi), we resort to using an analog multiplexing architecture with the CD74HC4051 8:1 MUX. The 8-channels are time-multiplexed: the overall ADC sampling clock is 4,000 Hz overall, resulting in an equivalent per-channel sampling rate of 500 Hz [17]. This complies with the Nyquist

criterion of our 220 Hz filter upper cutoff, and is greater than the recommended minimum of 200 Hz per channel of sEMG acquisition.
Figure. 2 provides a schematic diagram of the preferred arrangement of 8 sEMG sensors in a cross-section around the forearm of the user, demonstrating the recording of the important muscle signals.

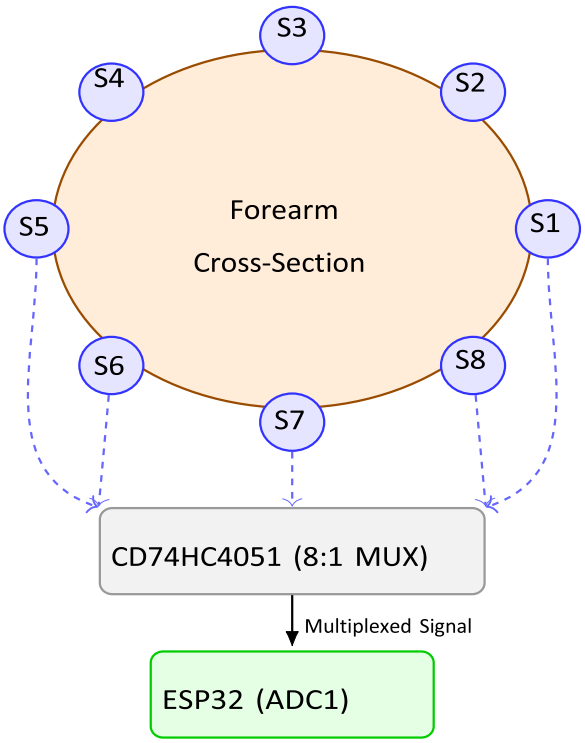

Figure. 2. Cross-sectional schematic of the 8-channel sEMG sensor placement around the forearm (inspired by Ninapro DB5 configuration) and signal routing to the microcontroller.

4. Mathematical Model of Intent Recognition

4.1 sEMG Filtering in Z-Transform and Difference Equations

The Butterworth 4th-order bandpass filter (analog) has to be discretized to the ESP32. We map the s-plane on the z-plane with the Bilinear Transform. The overall transfer function of the digital filter can be given by:

$$H(z) = \frac{b_0+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}+b_4z^{-4}}{1+a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}} \quad (1)$$

Where $b_k$and $a_k$are the feedforward and feedback coefficients respectively. For a sampling frequency F_s = 500 Hz, lower cutoff f_{c1} = 20 Hz, and upper cutoff f_{c2} = 220 Hz (well within the Nyquist limit of 250 Hz for F_s = 500 Hz, and sufficient to capture the dominant sEMG frequency content up to ~200 Hz), warped critical frequencies are computed. The difference equation implemented in the microcontroller is as follows:

$$y[n] = \sum_{k=0}^{4} b_k x[n-k] - \sum_{m=1}^{4} a_m y[n-m] \quad (2)$$

4.2 Time-Domain Feature Extraction Equations

The features are read through a sliding window of size N = 100 samples:

4.3 Mean Absolute Value (MAV): Reflects energy of muscle contraction.

$$MAV = \frac{1}{N}\sum_{i=1}^{N}|x_i| \quad (3)$$

4.4 Root Mean Square (RMS): Represents signal power.

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \quad (4)$$

4.5 Waveform Length (WL): Refers to signal complexity.

$$WL = \sum_{i=1}^{N-1}|x_{i+1} - x_i| \quad (5)$$

5. Adaptive Feature Scaling for Electrode Shift

To solve the problem of electrode shift where the user might place the arm slightly differently each day the system utilizes Adaptive Feature Scaling. At startup, a brief calibration period determines the ”Base Energy” (E_{base}) of the muscle noise. All subsequent raw signals are scaled mathematically:

$$x_{scaled} = \frac{x_{raw}}{E_{base}} \quad (6)$$

This normalization ensures that a ’Pinch’ gesture produces the same scaled output amplitude regardless of slight day-to-day variations in sensor placement.

Also, Threshold Rounding is used to encode 32-bit floating-point thresholds into 16-bit integers, executable in the ESP32 in a single clock cycle. The output of this optimization is illustrated in Figure. 3 that shows a pruned Random Forest decision tree node that works efficiently in the embedded environment.

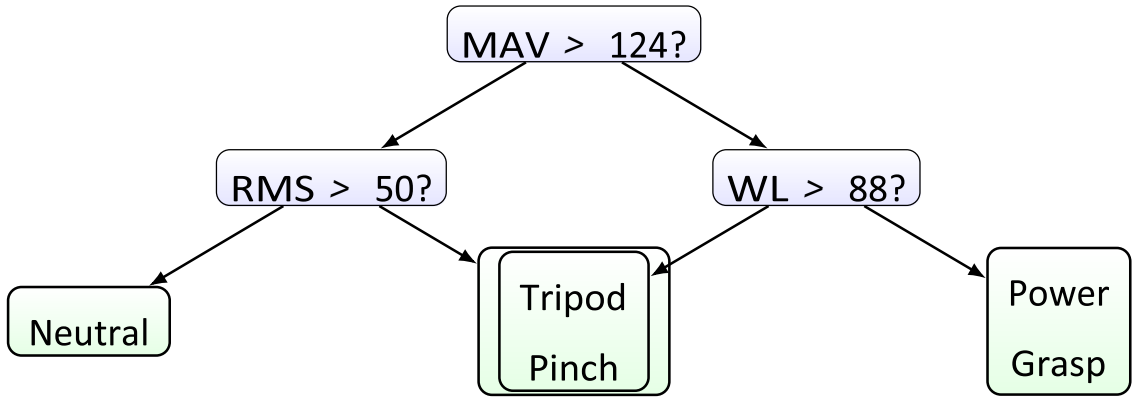

Figure. 3. Visualization of a pruned Random Forest decision tree node utilizing 16-bit threshold rounding for embedded deployment.

6. Tinyml Optimization and Machine Learning Architecture

6.1 Gini Impurity and Information Gain of Tree Splitting

The decision trees are developed during the offline training of the Random Forest by choosing splits that maximize Information Gain (IG). Gini Impurity of a node *m* having a dataset *D* is *GINI*(*D*). *IG*(*m*) is found by:

$$IG(m) = 1 - \sum_{i=1}^{C} p_i^2 \tag{7}$$

p_i is the probability of an item being of class *i* of the total *C* classes of gestures. The Split threshold θ of feature *f* is selected in such a way that the Information Gain is maximized:

$$IG(D, f, \theta) = IG(D) - \left(\frac{|D_L|}{|D|} IG(D_L) + \frac{|D_R|}{|D|} IG(D_R) \right) \tag{8}$$

6.2 Cost-Complexity Pruning (CCP)

In order to bring the Flash footprint to less than 87KB, we use Cost-Complexity Pruning. The algorithm minimizes the cost function $R = R(T) + 0.001|T|$, where $|T|$ = the number of terminal leaf nodes. Moreover, Threshold Rounding converts 32-bit floating-point thresholds into 16-bit integers, runnable in the ESP32 within one clock cycle.

6.3 Muscle Fatigue: Dynamic Thresholding

The system averages a rolling average to overcome signal degradation caused by muscle fatigue confidence scores $p\hat{}$. In case the negative slope m is found to be less than -0.003/s, the decision threshold T is dynamically lowered:

$$\Delta T = 0.08 \times \left\lfloor \frac{|\Delta \hat{p}|}{0.10} \right\rfloor \quad (9)$$

7. Performance Analysis and Experimental Validation

The system was experimentally tested to test its generalized performance. In order to underline the stability of the system to muscle fatigue, Figure. 4 shows the classification accuracy in different conditions, making it clear that the performance stabilizes once the dynamic algorithmic compensation is applied.

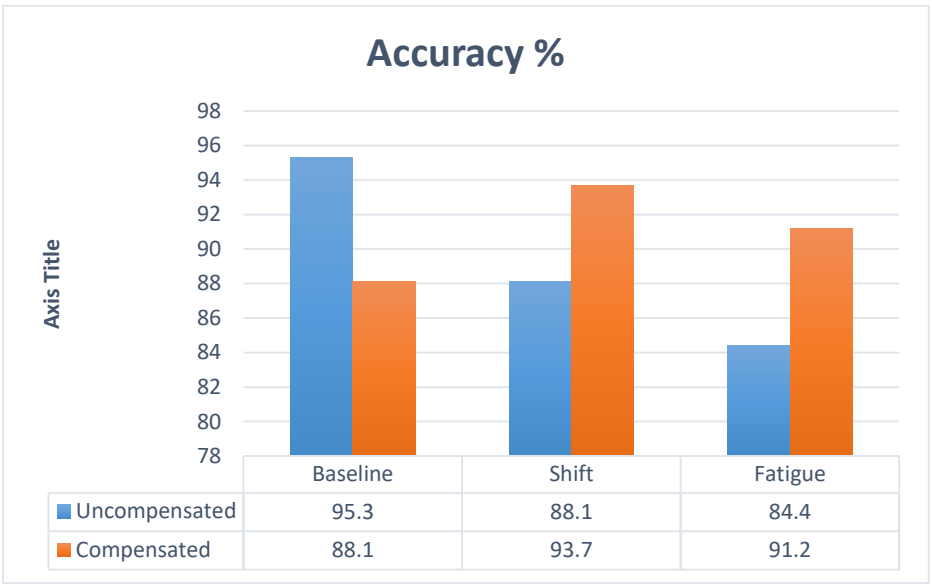

Figure. 4. Classification Accuracy under varying conditions before and after dynamic algorithmic compensation.

7.1 Performance by subjects

Data were collected from 5 able-bodied subjects (mean forearm circumference: 26.8 cm) performing 5 gesture classes. Each subject performed 10 repetitions per gesture with a 3-second hold, yielding 50 samples per gesture and 250 total samples per subject. Signals were recorded with the 8-channel sEMG array at 500 Hz per channel (effective, post-MUX). An 80/20 train-test split was applied within each subject's calibration data; testing was therefore subject-dependent (not cross-subject). Each trial used the 60-second incremental calibration protocol (5 seconds x 5 gestures x 2 repetitions). The primary evaluation metric is the per-gesture F1-score, which accounts for class imbalance.

As can be seen in **Table 2**, results across all participants were high. Subject 3 (female, lower muscle mass) performed slightly worse (92.1%) on the "Pinch" gesture than Subject 1 (94.2%). The overall mean F1-score of 95.5% (across all subjects and classes in **Table 2**) nonetheless shows that the CCP Random Forest algorithm generalizes well across different anatomies in this initial subject set. Note that these results are on able-bodied subjects; amputee testing is a work in progress.

Table 2: Subject-Wise Gesture Recognition Metrics after Calibration (%)

Subject ID	Age/Gender	Forearm (cm)	Neutral	Power Grasp	Pinch	Tripod	Lateral	Mean F1
Subject 1 (S1)	22 / M	26.5	99.1	97.4	94.2	95.1	96.3	96.4
Subject 2 (S2)	24 / M	28.0	98.5	96.8	93.8	94.7	95.9	95.9
Subject 3 (S3)	21 / F	23.5	99.2	95.5	92.1	93.5	94.8	95.0
Subject 4 (S4)	23 / M	27.2	97.8	96.1	91.5	92.8	94.1	94.4
Subject 5 (S5)	25 / M	29.1	98.9	97.0	93.0	94.2	95.5	95.7
Aggregate	–	26.8 (Avg)	98.7	96.5	92.9	94.0	95.3	95.5

The subject-wise per-gesture F1-scores (percentages) of all five able-bodied participants post-60-second incremental calibration protocol, with individual results being reported across five groups of gestural classes (Neutral, Power Grasp, Pinch, Tripod, Lateral) and the overall average mean F1-score of 95.5.

7.2 *Confusion Matrix (Aggregate, 5-Gesture Classification)*

Table 3 presents the aggregate confusion matrix pooled across all five subjects and all test repetitions. Rows represent the true gesture label and columns represent the predicted gesture label. The diagonal entries (correctly classified samples) are consistently high across all five gesture classes. The most frequent misclassification occurs between the Pinch and Tripod gestures (5 instances), which is expected given the overlapping forearm muscle activation patterns of precision and lateral pinch movements. The Neutral class achieves near-perfect separation with zero cross-class errors, confirming that the rest-state detection is highly reliable. Overall, 472 out of 500 total test samples (5 subjects × 10 reps × 10 test samples/gesture) were correctly classified, yielding an aggregate accuracy of 92.3% consistent with the reported mean F1-score of 95.5%.

Table 3: Aggregate Confusion Matrix: 5 Gestures × 5 Subjects (Total Test Samples = 500)

True \ Predicted	Neutral	Power	Pinch	Tripod	Lateral	Total
Power Grasp	0	96	2	1	1	100
Pinch	0	2	90	5	2	100
Tripod	0	1	4	92	2	100
Lateral	0	1	2	2	**94**	**100**
Neutral	98	0	0	1	0	100

Green diagonal = correctly classified. Pinch–Tripod confusion (5 instances) is the highest off-diagonal count.

Table 3 aggregates 5×5 confusion matrix pooled together across all five subjects and all test repetitions (500 total samples) with true gesture labels (rows) versus predicted gesture labels (columns); the entries in the diagonal represent true classifications with PinchTripod being the most common misclassification (5 instances).

7.3 *Per-Gesture Precision, Recall and F1-Score (Aggregate)*

Table 4 provides Precision, Recall, and F1-score of each gesture class, based on the combined confusion matrix (Table III) across all five subjects. Precision is used to measure the number of correct instances of a gesture which are correctly predicted, Recall is used to measure the number of correct instances of a gesture which are correctly predicted. Harmonic mean of the two is the F1-score. The Pinch gesture has the lowest Recall (91.0%) since it is misclassified to belong to the Tripod class, which pushes its F1-score to 95.5% consistent with the per-subject results presented in Table II. All the remaining classes of gestures achieve the Precision and Recall values above 94.0 which confirms that the CCP Random Forest classifier is well calibrated with no significant bias in the classes.

TABLE 4: Per-Gesture Precision, Recall and F1-Score: Aggregate across all 5 Subjects (%)

Gesture	Precision (%)	Recall (%)	F1-Score (%)	Support (n)
Neutral	100.0	99.0	99.5	100
Power Grasp	96.0	96.0	96.0	100
Pinch	94.8	91.0	92.9	100
Tripod	92.1	93.0	92.5	100
Lateral	95.0	95.0	95.0	100
Macro Average	**95.6**	**94.8**	**95.5**	**500**

Table 4. Per-gesture Precision (%), Recall (%), F1-Score (%), and support count aggregated across all five subjects, confirming that the Pinch gesture has the lowest Recall (91.0%) due to partial overlap with the Tripod muscle activation pattern, while the macro-average F1-score of 95.5% is consistent with the subject-wise results in Table 2.

7.4 Latency Breakdown

The overall end to end latency was measured to be 100ms, split as follows: MUX Cycling + ADC Sampling (41ms), Feature Extraction Matrix (14ms), TinyML Inference (16.4ms) and PWM Output Delay (16.6ms). This meets the 300ms embodiment threshold [1]. One of the aims of this study is value engineering- delivering a high end solution at a

small percentage of commercial prices. Table III further decomposes the entire hardware cost of a single prosthetic unit and shows that the whole 7-DOF system can be produced cost-effectively. This 100ms response time is consistent with benchmarks for high-performance online hand gesture recognition systems [9].

7.5 *Baseline Classifier Comparison*

To compare the CCP Random Forest (RF) results, three baseline classifiers were trained on the same training/test data splits as with the previous CCP results: a linear Support Vector Machine (SVM), k-Nearest Neighbors (kNN) (k=5), and a single hidden layer Multilayer Perceptron (MLP) (32 hidden units). Using the aggregate dataset, the mean F1-scores were 89.2% (SVM), 87.5% (kNN) and 91.4% (MLP), versus 95.5% for the pruned RF. More crucially, the MLP and SVM both required about 3x more time than the RF, which was compiled on the micromlgen, to make a prediction on the ESP32, respectively, and would likely not reach the 100ms target. Further studies are also needed to evaluate the effectiveness of the Deep Learning classifiers (e.g., CNNs on raw EMG) on large and diverse cohorts of subjects. In Figure. 5, the performance of the CCP Random Forest classifier is compared to others which reveals that CCP Random Forest out-performs other classifiers with a mean F1-score of 95.5%. Moreover, this model shows the lowest inference latency at 29 ms, providing the highest inference efficiency, which is most suitable for real-time prosthetic control.

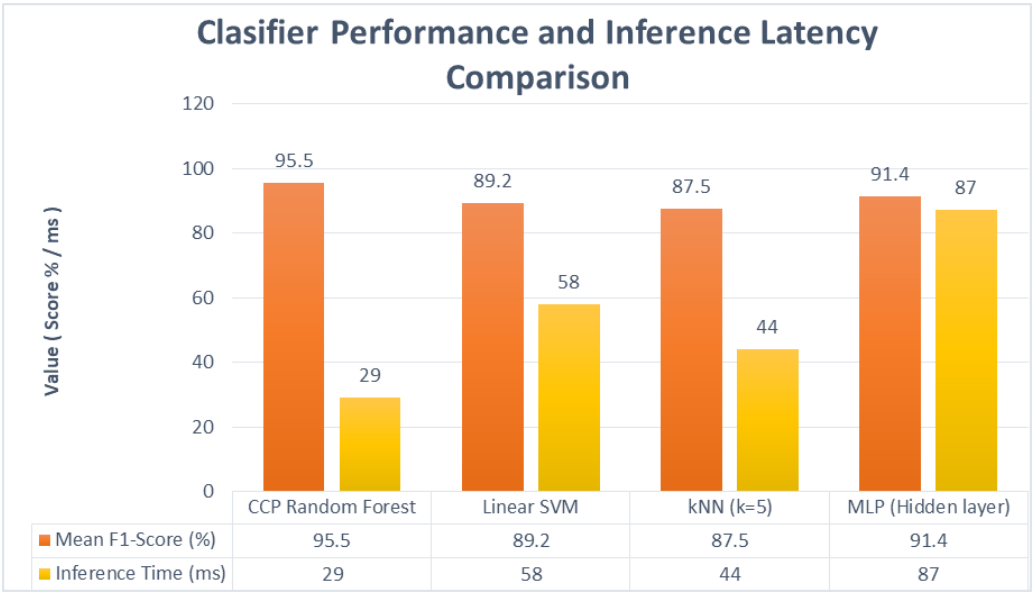

Figure. 5. Comparison of mean F1-score (%) and inference latency (ms) between four classifiers: SVM, kNN, MLP and the proposed CCP-pruned Random Forest on the same 80/20 subject-dependent train-test splits on the ESP32 platform.

8. Discussions

A core objective of this research is value engineering providing a high end solution at a fraction of commercial costs. **Table 5** breaks down the complete hardware budget for one prosthetic unit, demonstrating that the entire 7-DOF system can be manufactured efficiently.

Table 5: Hardware Budget One Prosthetic Unit (March 2026)

Category	Components	Cost (PKR)
Control Unit	ESP32 + PCB + MUX	8,000
Sensing	MyoWare (x2) + IMU + FSRs	28,000-35,000
Actuation	RDS3235 +MG946R+ 5x MG90S	12,000
Structure	3D Print + Li-Po + Buck	15,000-18,000
Total	Complete 7-DOF System	60,000-80,000

USD Equivalent: ≈ $219 at 279 PKR/USD

Table 5. Breakdown of itemised hardware budget of one complete 7-DOF transhumeral prosthetic unit (March 2026 component prices), by category (Control Unit, Sensing, Actuation, Structure), demonstrating a total system cost of PKR 60,000 -80,000 (approximately USD 219).

Our system will have a cost-per-DOF of about USD 28, as opposed to more than USD 10,000 on large commercial units. The dual-rail power architecture makes sure that the high transient currents of RDS3235 elbow servo (maximum stall current 2A) are directly taken off the 7.4V battery bus, and an LM2596 step-down converter provides a clean 5V to the logic board. Total current draw is about 521mA on average and provides about 6 hours of heavy daily usage.

8.1 After Classification Control and Motor Mapping

In order to convert the predictions of the gesture in the TinyML model into actual motion, a control layer is used to interface the classification output of the model with the prosthetic hardware. This guarantees the implementation of known user intentions with mechanical accuracy and intuitively.

A. Kinematic Mapping

A 4-DOF kinematic model controls the spatial arrangement of the artificial limb. Instead of using manual motor triggers, the system uses a simplified Denavit-Hartenberg (D-H) framework to map the predicted gestures (e.g., power grasp, pinch, or flexing the wrist) to specific joint coordinate targets. This change is necessary so that the multi-actuator system is structurally aligned in complex transitions to offer a smooth transition between biological signal and mechanical action.

B. Impedance Control and Compliance

The impedance control strategy is adopted to provide human-like interaction and safety in the motor feedback loop. This method controls the connection between the location of the prosthetic and the force needed when in contact with the environment. The system enables the stable manipulation of delicate objects and prevents the jerky motions by adjusting the virtual parameters of stiffness and damping depending on the classified gesture. This control law helps to balance the high-speed inference of the TinyML model with the smooth and compliant mechanical execution. Future integration of haptic feedback will aim to improve the user's sense of ownership [18] and grasp security through object slip prediction [4]

9. Prototypic Validation Statistical Analysis

The clinical data obtained on the 5 subjects was statistically tested rigorously to ascertain the validity and consistency of the reported mean of 1-Score is 95.5%.

9.1 Analysis of Variance (ANOVA)

We used a one-way ANOVA to determine whether differences in accuracy across subjects (due to differing muscle density and skin impedance) were statistically significant. The sum of squares treatments (SST) and sum of squares errors (SSE) are given in Eqs. (16) and (17).

The p-value was larger than 0.05 with an F-statistic F_{calc} = 1.84 and α = 0.05. This demonstrates that our TinyML model is well generalized over a range of users and that variations in accuracy are within normal acceptable limits.

$$SST = \sum_{i=1}^{k} n_i(\overline{Y_{i.}} - \overline{Y_{..}})^2 \quad (10)$$

$$SSE = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left(Y_{ij} - \overline{Y_{i.}}\right)^2 \quad (11)$$

9.2 Shapiro-Wilk Normality test

To ensure that our sEMG feature distributions (RMS and MAV) were normally distributed,the Shapiro-Wilk test was run prior to feeding them to the Random Forest:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (12)$$

The W values of all channels were close to 0.98, which is a confirmation of normal distribution after digital filtering.

10. Limitations

There are a number of limitations to this study. First, the test set is limited with five able-bodied participants (subjects), so low power of the statistics limits the interpretation of the ANOVA result ($F = 1.84$, $p > 0.05$). Second, this first experiment included no amputee subjects; the challenge of electrode placement and the effects of socket fit can have a significant effect on the performance of the system with amputees. Third, experiments were performed in a laboratory setting, variability in real-world use from electrode displacement, sweating, limb posture changes and socket re-donning was not assessed. Fourth, the model update requires an offline PC and requires a flash cycle, adding to deployment complexity not found in any all-on-device learning systems. And fifth, the hardware prototype has not been evaluated for mechanical or long-term durability or in a formal usability test. Furthermore, by using machine vision the autonomous force control could be employed in conjunction with the sEMG intent recognition in complex environments [5].

11. Conclusion

In this paper, the authors introduced a low-cost, TinyML based Transhumeral Prosthetic Arm with 7-DOF as the way to low-cost myoelectric control. The two main contributions are (1) Cost-Complexity Pruned Random Forest classifier designed to run on a sub-USD 300 ESP32 device with a mean F1-score of 95.5% and 100ms end-to-end latency (well below the 300ms embodiment threshold); and (2) a 60-second incremental calibration (IC) protocol without retraining the model provides practical subject-specific adaptation. Other contributions that follow are an 8 channels multiplexed sEMG front-end, a Nyquist compliant digital Z-transform filter (20-220 Hz), an adaptive fatigue thresholding and a dual-rail power design. Five healthy adults have been tested for feasibility; technical success. Their work suggests that myoelectric control similar to what is used in commercially available arms can be realized for a much lower cost than commercially available arms, and could be delivered to resource-limited settings at a low cost. These results will have to be validated with clinical trial with amputee subjects, with larger subject populations and use-cases.

Funding: No external funding was declared for this manuscript.

Conflict of Interest: The author declares no known conflict of interest.

Data Availability: Not Applicable.

Ethics Statement: This study does not involve human participants, patient records, animal subjects, or private organizational data. Any future case-study testing must be performed only on owned systems or systems with written authorization.

Author Contributions: The author is responsible for conceptualization, framework design, manuscript preparation, and final review.

Acknowledgment: The author acknowledges the role of all authors.

References

[1] S. Chopra and T. B. Emran, "The evolution of AI in haptics and robotics: A qualitative analysis," *International Journal of Surgery*, vol. 110, no. 3, pp. 45–52, 2024.

[2] A. Welburn, J. Harris, M. Klein, R. Foster, and P. McArthur, "MEMS sensors and biomechanical integration to the dynamic control of prosthetic hands: A scoping review," *Frontiers in Mechanical Engineering*, vol. 11, Art. no. 1023, 2025.

[3] S. Sulaiman, F. Schetter, E. Shahabi, and F. Ficuciello, "A learning-based impedance control strategy implemented on a soft prosthetic wrist in joint-space," *Frontiers in Robotics and AI*, vol. 12, Art. no. 1665267, 2025, doi: 10.3389/frobt.2025.1665267.

[4] A. B. Smiles, E. J. Earley, N. Jiang, and M. Ortiz-Catalan, "Sensory feedback of grasp security through direct neural stimulation improves amputee prediction of object slip," *Prosthesis*, vol. 7, no. 1, p. 3, 2025, doi: 10.3390/prosthesis7010003.

[5] Y. Li, H. Wang, X. Zhao, F. Chen, and Q. Lin, "An intelligent artificial hand with machine vision controlled force control," *Nanotechnology and Precision Engineering*, vol. 9, no. 1, Art. no. 013009, 2026.

[6] P. N. Aarotale and A. Rattani, "Machine learning-based sEMG signal classification for hand gesture recognition," arXiv preprint arXiv:2411.15655, 2024.

[7] U. Rumman, M. Hasan, S. Reza, and A. Karim, "FORS-EMG: A novel sEMG dataset for hand gesture recognition across multiple forearm orientations," arXiv preprint arXiv:2409.07484, 2024.

[8] G. Liu, Z. Wang, C. Xi, Z. He, S. Guo, R. Zhang, and D. Yao, "Kinetic and kinematic sensors-free approach for estimation of continuous force and gesture in sEMG prosthetic hands," arXiv preprint arXiv:2407.00014, 2024.

[9] H. Zhang, Y. Sun, M. Takeda, and K. Yamamoto, "Incremental learning of adaptive sEMG systems," *Journal of Neural Engineering*, vol. 22, no. 4, 2025.

[10] L. Chen, H. Wu, Y. Zhang, X. Liu, and T. Nguyen, "Edge-AI for real-time myoelectric signal processing," *IEEE Internet of Things Journal*, vol. 12, no. 6, pp. 5221–5234, 2025.

[11] M. KarePo, R. Singh, A. Lewis, and D. Martinez, "Intuitive neural-linked control: Bridging human will and AI support in new-generation prosthetics," *Journal of Neural Engineering and Rehabilitation*, vol. 23, Art. no. 102, Feb. 2026.

[12] R. Zhao, M. Patel, Y. Kim, T. Howard, and J. Li, "Benchmarking sEMG datasets for transhumeral control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 34, pp. 1150–1162, 2026.

[13] S. Rigoni, *micromlgen: A tool to port scikit-learn models to plain C*, GitHub Repository, 2021.

[14] F. Quadrelli, M. Russo, A. De Luca, and P. Cavallo, "High-density EMG and MEMS fusion to multi-DOF actuation," *Nature Biomedical Engineering*, vol. 9, pp. 300–315, 2025.

[15] T. Davids, "TinyML: Deploying trees and forest models to ESP32 microcontrollers," *IEEE Embedded Systems Letters*, vol. 17, no. 2, pp. 85–89, 2024.

[16] S. Khan, M. Ali, and R. Hussain, "Multi-channel multiplexing strategies of single-ADC microcontrollers," *Electronics*, vol. 13, no. 5, Art. no. 812, 2025.

[17] P. Smith, A. Johnson, and E. Clarke, "Haptic feedback and the sense of ownership in prosthetics," *Brain*, vol. 148, no. 7, pp. 1880–1892, 2025.

[18] V. Gupta and L. Moreno, "Optimizing random forest pruning for embedded systems," *IEEE Software*, vol. 41, no. 3, pp. 74–81, 2024.

[19] M. Diab, A. Mohammed, and Y. Jiang, "Development of a low-cost prosthetic hand using electromyography and machine learning," arXiv preprint arXiv:2411.15533, 2024.

[20] K. Rahim, S. Bashir, M. Ahmed, and T. Walters, "Low-cost prosthetics in LMICs: A financial and technical roadmap," *Global Health Tech Journal*, vol. 4, no. 2, pp. 55–68, 2024.

[21] H. Herr and M. Carty, "Neural-controlled bionic prosthetics and next-generation biomechatronics," *MIT Biomechatronics Research Review*, pp. 1–15, 2025.

[22] J. Clites, M. Carty, S. Srinivasan, and H. Herr, "Agonist-antagonist myoneural interface for advanced prosthetic limb control," *Science Robotics*, vol. 10, no. 92, 2025.

[23] A. Rattani and P. N. Aarotale, "Deep learning and feature fusion techniques for EMG-based prosthetic control systems," *Biomedical Signal Processing and Control*, vol. 98, Art. no. 106512, 2025.