Recording and Processing of Surface Electromyography (sEMG)

Alireza Salmanipour - 400101367

Sharif University of Technology

February 4, 2025

Contents

1 Technological and Experimental Background

1.1 Surface Electromyography (sEMG): Concept, Importance, and Applications

Surface Electromyography (sEMG) is a non-invasive method used to measure electrical activity of muscles. signals are recorded using surface electrodes placed on skin above target muscle.

Importance and Applications:

- **Biomedical Engineering:** sEMG is widely used in diagnosing neuromuscular disorders such as ALS and muscular dystrophy.
- **Rehabilitation and Prosthetics:** It is utilized to control robotic prostheses and assistive exoskeletons for individuals with motor impairments.
- **Human-Computer Interaction:** sEMG plays a crucial role in gesture-based control systems and brain-machine interfaces.
- **Sports Science:** Used in evaluating muscle fatigue and optimizing athletic performance.

1.2 Experimental Protocol

This experiment involves recording muscle activity during 13 distinct movements, each repeated 25 times.

Pipeline of Movements:

1. Resting hand (5 sec)
2. Gripping a water bottle (5 sec)
3. Resting hand (5 sec)
4. Moving grip upwards (5 sec)
5. Resting hand (5 sec)
6. Moving grip downwards (5 sec)
7. Resting hand (5 sec)
8. Moving grip to right (5 sec)
9. Resting hand (5 sec)
10. Moving grip to left (5 sec)
11. Resting hand (5 sec)
12. Rotating wrist outward (5 sec)
13. Resting hand (5 sec)

Each movement lasts 5 seconds and is repeated 25 times. Data is recorded using LabChart software, with each movement annotated using "Comment" feature.

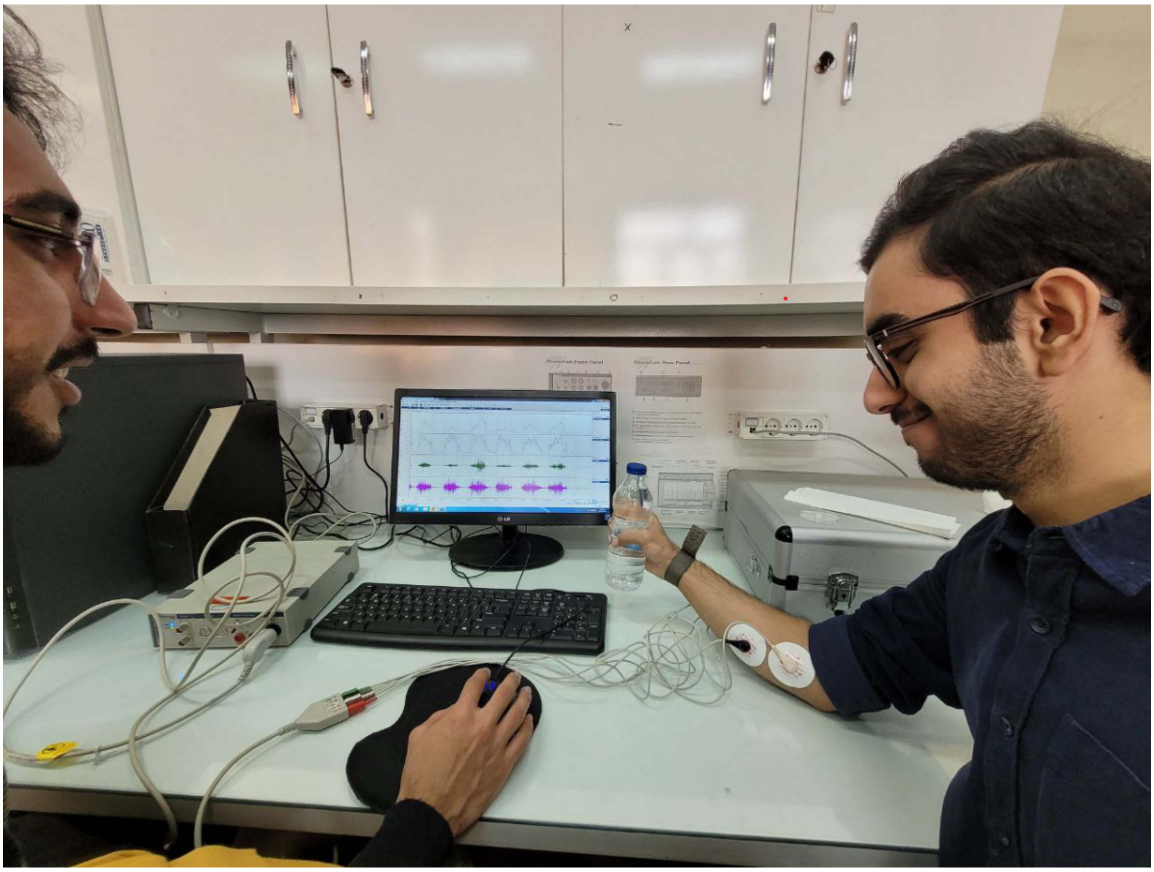

Figure 1: Pipeline of required hand movements for experiment

1.3 Objective of Experiment

The primary objective of this experiment is to process and classify muscle movements using sEMG signals. key goals include:

- **Preprocessing:** Filtering noise and normalizing recorded signals.
- **Segmentation:** Dividing signals into fixed-duration time windows.
- **Feature Extraction:** Calculating relevant statistical features such as RMS, MAV, ZC, SSC, WL.
- **Classification:** Training machine learning models (e.g., SVM, Random Forest) to recognize different hand movements.

1.4 Equipment Used in Experiment

1. Surface Electrodes

- **Type:** Disposable surface electrodes
- **Placement:** On Biceps and Triceps muscles
- **Output:** Raw EMG signal (microvolt level)

2. Signal Acquisition Device

- **Input:** Receives signals from electrodes
- **Processing:** Signal amplification and filtering
- **Output:** Digitized signal for further processing

3. LabChart Software

- **Purpose:** Real-time recording, visualization, and preprocessing of EMG signals
- **Features:** Live signal monitoring, annotation with "Comment"

Figure 2: Equipment used for recording sEMG signals

1.5 Features of Signal Acquisition System

The EMG recording system consists of three main components:

1.5.1 1. Features of EMG Signal Amplifier

- Enhances weak microvolt-level signals for accurate measurement
- Minimizes environmental noise and electrical interference
- Uses a differential amplifier to eliminate common-mode noise

1.5.2 2. Features of Analog-to-Digital Converter (ADC)

- **Sampling Rate:** 200 Hz
- **Bit Resolution:** 12-bit conversion for high accuracy
- **Digital Output:** Converts raw EMG signal into a format suitable for computer processing

1.5.3 3. Features of Interface Circuit (Between EMG System and Computer)

- **Data Transmission:** USB connection for real-time data streaming
- **Compatibility:** Communicates with LabChart software for data visualization and storage

1.6 Block Diagram of EMG Signal Acquisition System

The following diagram illustrates overall setup for EMG signal acquisition system.

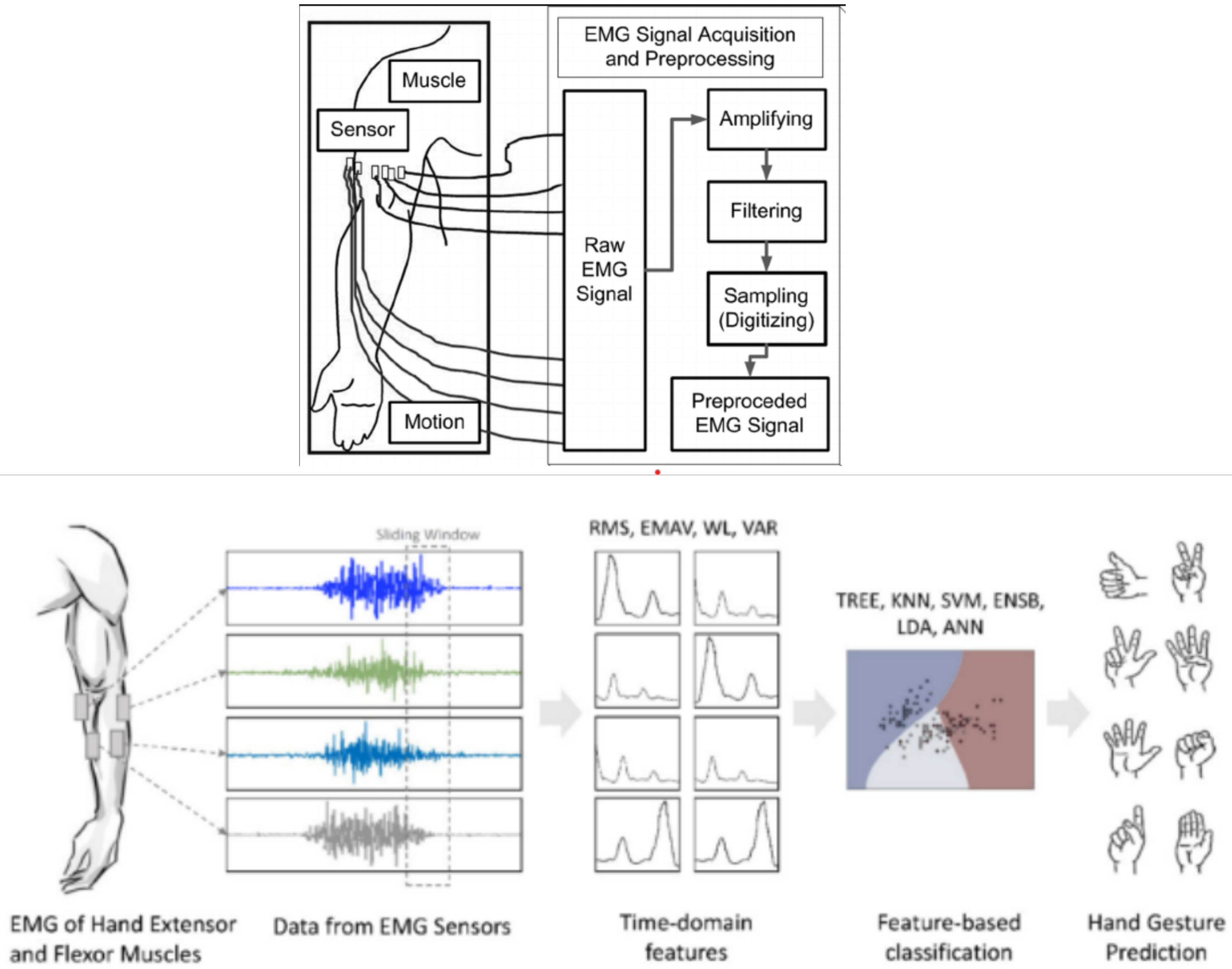

Figure 3: Block diagram of EMG signal acquisition system

2 Subject Information Table

Age (Years)	Gender (Male/Female)	Height (cm)	Weight (kg)	Laterality (Right/Left Handed)
21	male	170	60	Right Handed

Table 1: Group Member Information

3 Experiment Procedure Report

3.1 Electrode Placement

The electrodes were placed on surface of skin over specific muscles. placement was carefully chosen to capture electrical activity of target muscles.

Electrode Placement Details:

- **Muscles Monitored:**
  - Biceps Brachii: Captures signals for flexion movements.
  - Triceps Brachii: Captures signals for extension movements.
- **Electrode Type:** Disposable surface electrodes
- **Placement:** Electrodes were placed along muscle fibers for optimal signal capture.
- **Ground Electrode:** Positioned on wrist to minimize noise and interference.

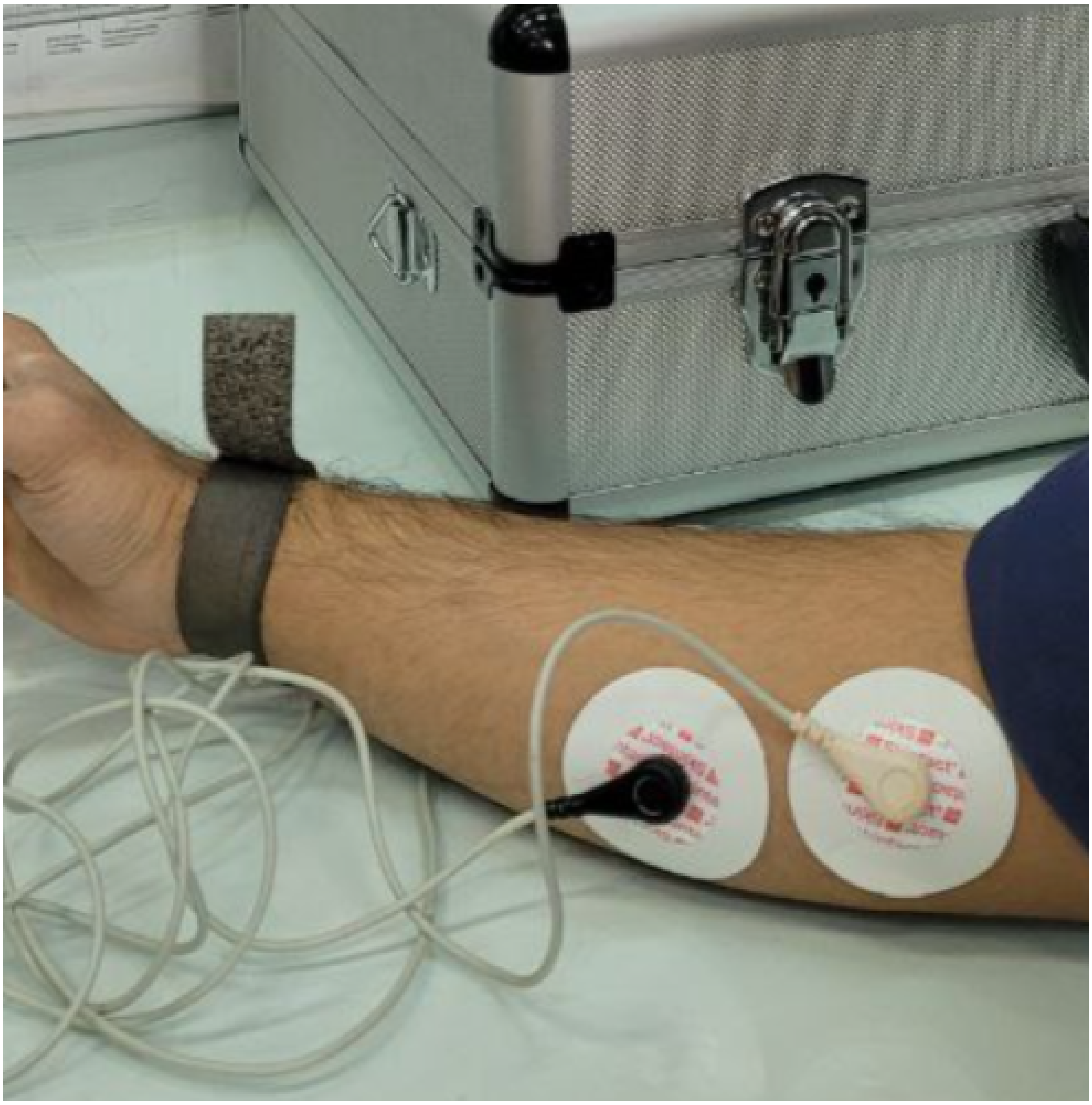

Figure 4: Electrode placement on biceps and triceps muscles

3.2 Recording Procedure

The recording followed a systematic protocol, with sequence of movements repeated 25 times.

Steps of Recording Process:

1. subject was seated comfortably, with their arm resting on a stable surface.
2. Surface electrodes were attached to skin over biceps and triceps muscles.
3. recording started with a 5-second rest period, followed by movements in following order:
   - Grip
   - Upward movement
   - Downward movement
   - Rightward movement
   - Leftward movement
   - Wrist rotation
4. Each movement lasted 5 seconds and was separated by 5 seconds of rest.
5. data was recorded using LabChart software, with each movement marked using "Comment" feature for segmentation.

3.3 Role of Muscles in Activities and Electrode Placement Justification

- **Biceps Brachii:**
  - Role: Responsible for flexion of elbow joint during gripping and upward movements.
  - Justification: Electrode placement along biceps captures electrical activity generated during muscle contraction.
- **Triceps Brachii:**
  - Role: Responsible for extension of elbow joint during downward and outward movements.
  - Justification: Electrode placement along triceps captures signals generated during muscle activation.

3.4 Proposed Methods for Preprocessing EMG Signal

To ensure quality of EMG signals, following preprocessing methods were applied:

- **Low-Pass Filtering:** A Butterworth filter with a cutoff frequency of 1 Hz was used to remove high-frequency noise.
- **Notch Filtering:** A 50 Hz notch filter was applied to eliminate powerline interference.
- **Normalization:** Min-Max normalization was performed to scale data between 0 and 1.

3.5 Features for Hand Movement Recognition

The following statistical features were extracted from each window of EMG signal:

- **Mean Absolute Value (MAV):** Represents average amplitude of signal.
- **Root Mean Square (RMS):** Provides an estimate of signal energy.
- **Zero Crossing (ZC):** Counts number of times signal crosses zero, indicating signal variability.
- **Waveform Length (WL):** Measures complexity of signal.
- **Slope Sign Change (SSC):** Captures changes in slope of signal, indicating muscle activity.

3.6 Proposed Methods for Processing EMG Signal

After preprocessing and feature extraction, following methods were used for processing EMG signal:

1. **Segmentation:**

   - signal was divided into 200 ms windows with a 190 ms overlap (10 ms step size).

2. **Classification Methods:**

3.6.1 K-Nearest Neighbors (KNN):

- **Description:** KNN is a simple and effective machine learning algorithm that classifies data points based on majority class of their nearest neighbors.
- **Working:** For each test sample, KNN finds k nearest points in training data and assigns most common class among them.
- **Strengths:** Easy to implement, non-parametric, and effective for small datasets.
- **Weaknesses:** Computationally expensive for large datasets and sensitive to irrelevant features.

3.6.2 Random Forest:

- **Description:** Random Forest is an ensemble learning method that builds multiple decision trees and combines their outputs to improve accuracy and reduce overfitting.
- **Working:** Each tree is trained on a random subset of data, and final prediction is based on majority vote from all trees.
- **Strengths:** Handles high-dimensional data well, robust to noise, and avoids overfitting due to averaging.
- **Weaknesses:** Can be slow for large datasets and less interpretable compared to single decision trees.

4 Experimental Results Report

4.1 Signal Registration with System

The signal registration process involved following steps using specialized software:

Software Used:

- **LabChart Software:** Used for real-time signal recording and annotation.
- **Python:** Utilized for advanced preprocessing and feature extraction.
- **MATLAB:** Applied for signal visualization and time-frequency analysis.

Steps in Registration Process:

1. Electrodes were placed on subject's arm, targeting biceps and triceps muscles.
2. "Comment" feature in LabChart was used to annotate each movement during recording process.
3. Data was exported from LabChart and processed further in Python and MATLAB for detailed analysis.

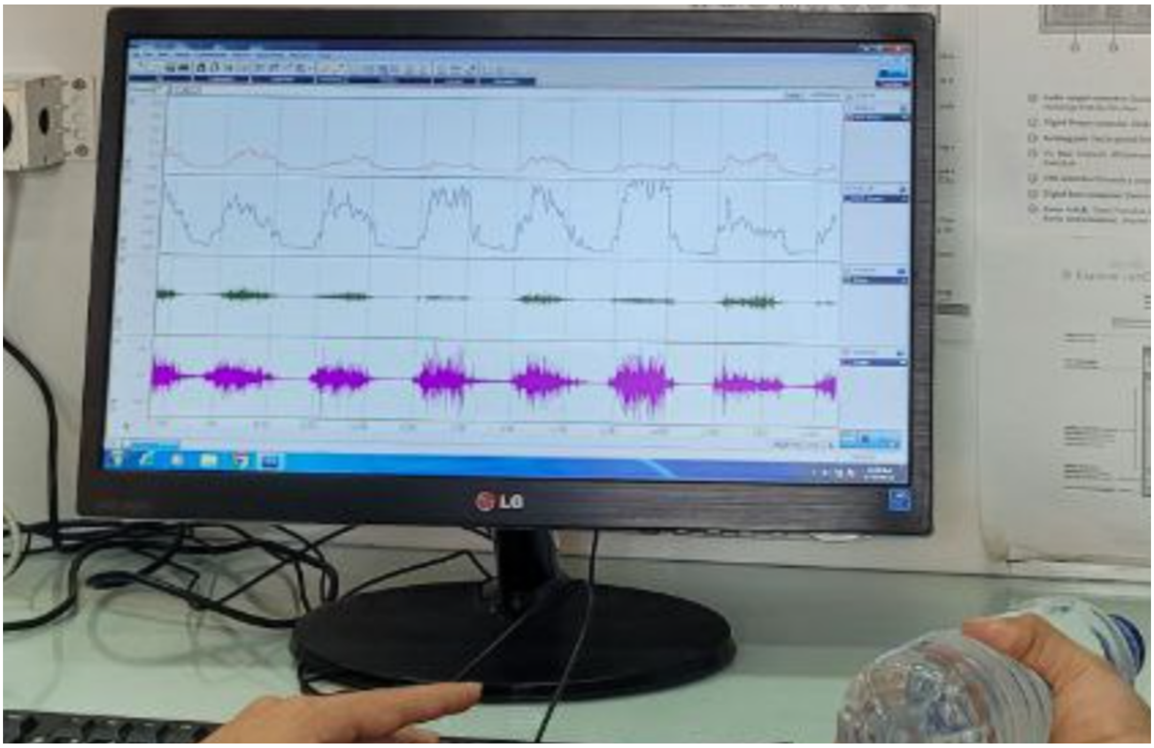

Figure 5: Signal registration process using LabChart software

4.2 Time-Domain Analysis of sEMG Signal

Time-domain changes in sEMG signal were observed during different hand movements:

- Resting periods showed minimal activity.
- Movements such as gripping and rotation resulted in noticeable peaks in signal amplitude.

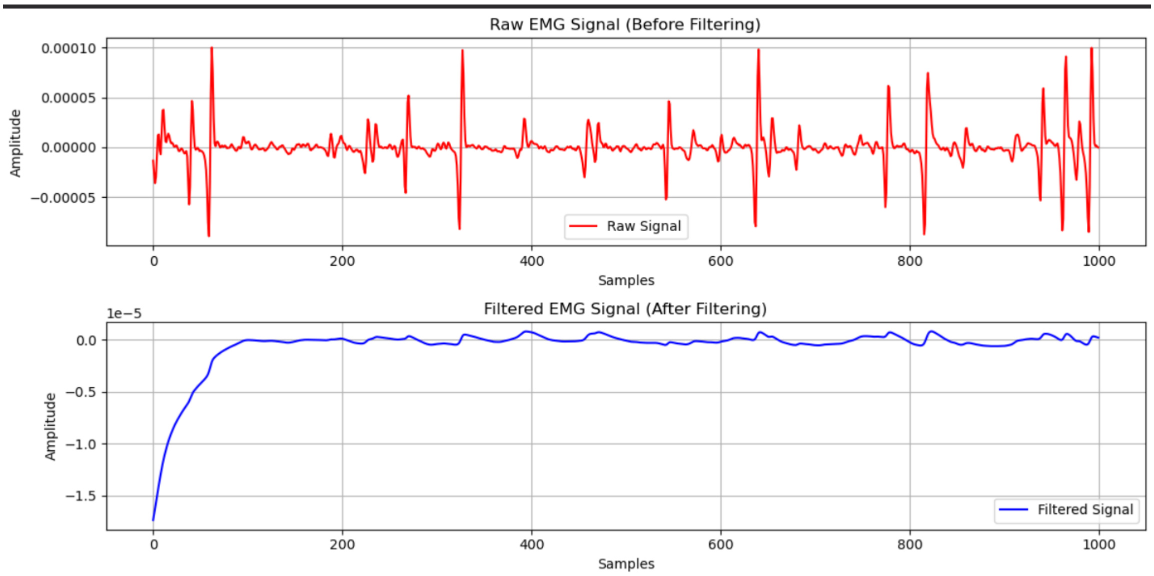

Figure 6: Time-domain signal changes for different hand movements

4.3 Frequency-Domain Analysis of sEMG Signal

Frequency-domain characteristics of sEMG signal were analyzed using FFT:

- Resting periods showed a frequency concentration below 20 Hz.
- Active movements showed significant power in 30-60 Hz range.

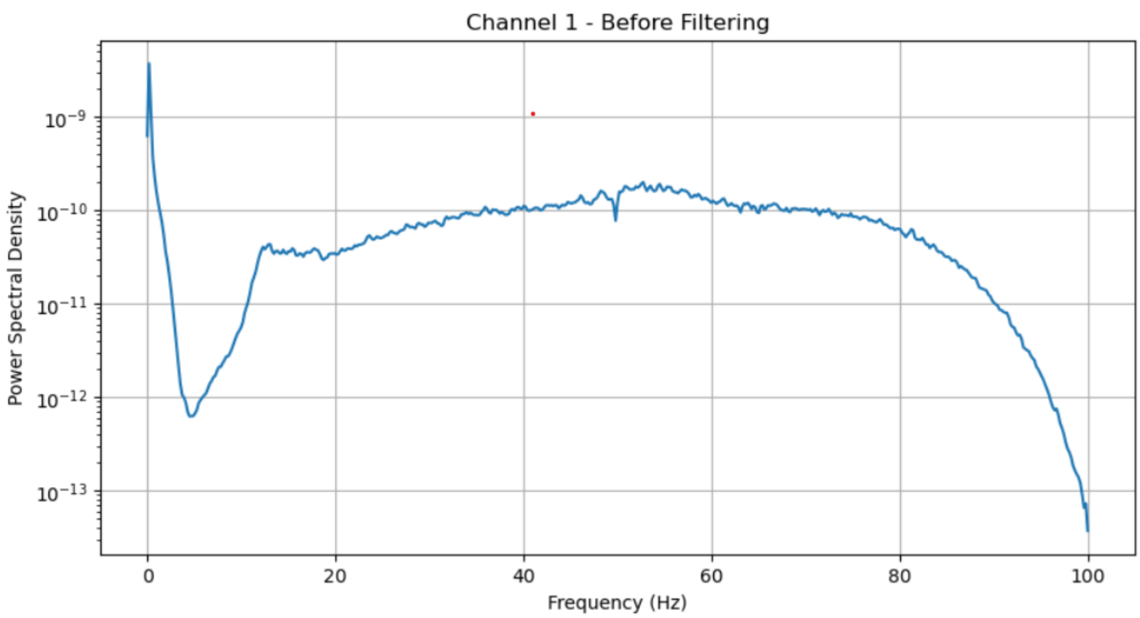

Figure 7: Frequency spectrum of sEMG signal for different movements

4.4 Signal Changes Based on Hand Movements

Key observations:

- Movements such as gripping and upward motion exhibited higher signal energy.
- Rotational movements showed distinct signal patterns in both time and frequency domains.
- Resting periods consistently showed minimal activity, validating preprocessing results.

4.5 Challenges in Signal Registration and Solutions

Challenges Encountered:

- Noise Interference: Powerline noise (50 Hz) and environmental interference affected signal quality.
- Electrode Placement Variability: Improper placement caused inconsistent signals.
- Motion Artifacts: Sudden hand movements introduced artifacts in signal.

Solutions Implemented:

- Applied a Notch Filter to eliminate 50 Hz noise.
- Ensured consistent electrode placement using anatomical landmarks.
- Used a Butterworth Low-Pass Filter to remove high-frequency artifacts.

4.6 Results of Proposed Preprocessing Method

Preprocessing Steps and Their Effects:

- Low-Pass Filtering: Successfully removed high-frequency noise, preserving muscle signal.
- Notch Filtering: Eliminated powerline interference, enhancing signal clarity.
- Normalization: Minimized variability between different signal ranges, enabling better feature extraction.

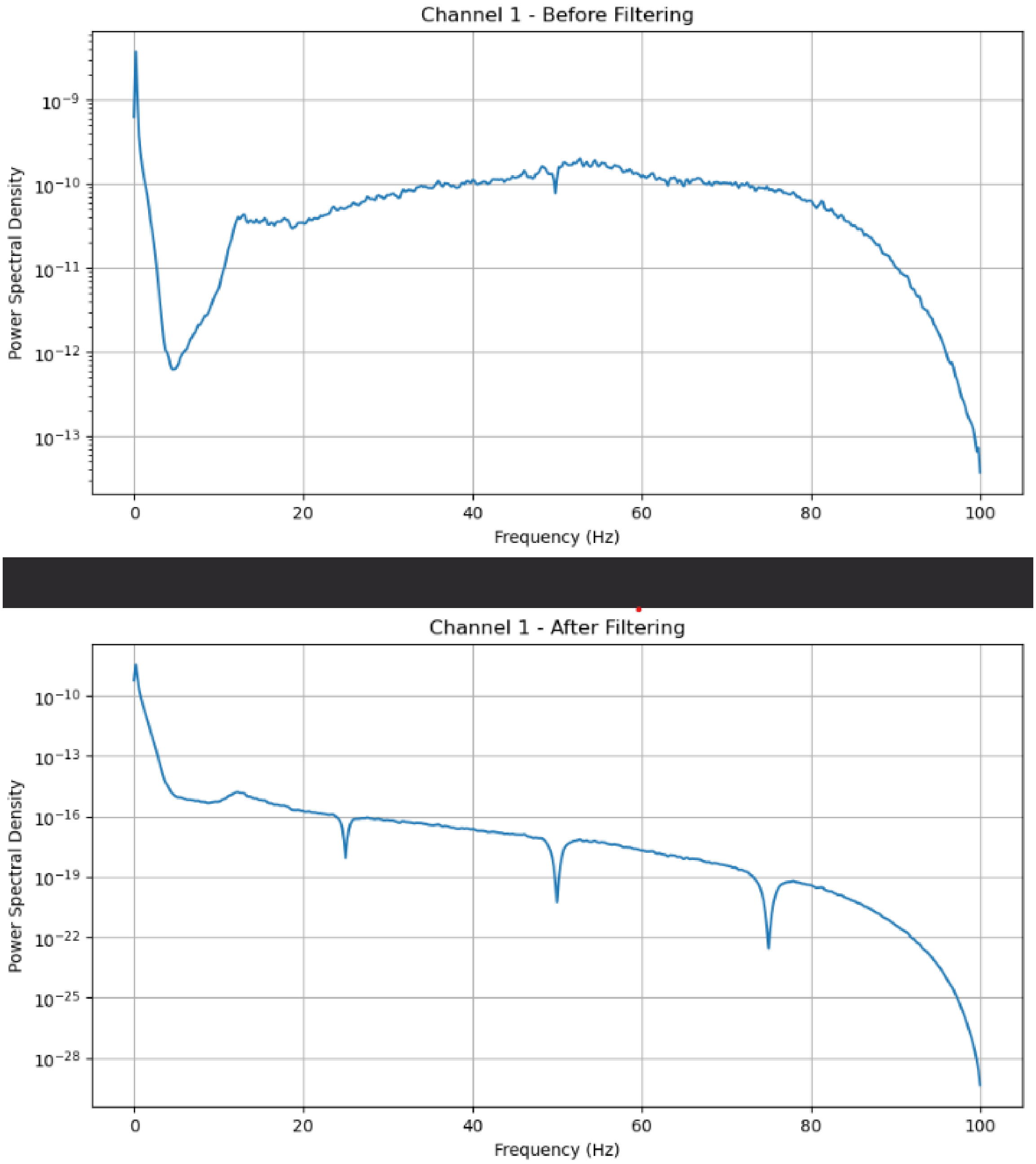

Figure 8: Comparison of raw and preprocessed sEMG signals

4.7 Results of Proposed Signal Processing Method

The processed features were used to train and evaluate two machine learning models: **K-Nearest Neighbors (KNN)** and **Random Forest**. following sections present classification performance of these models across different datasets.
We separate data into training and testing sets to evaluate model performance on unseen data and prevent overfitting. The training set is used to train the machine learning model by learning patterns in the data, while the test set is kept separate to assess how well the model generalizes to new, unseen samples. Without this separation, the model might memorize the training data instead of learning meaningful patterns, leading to poor real-world performance. A typical split ratio is 80% training and 20% testing, ensuring enough data for learning while maintaining a fair evaluation.

4.8 K-Nearest Neighbors (KNN) Results

The KNN model was evaluated based on its ability to classify different hand movements. classification reports and confusion matrices for multiple datasets are presented below.

4.8.1 KNN Results for Our Dataset

Class	**Precision**	**Recall**	**F1-Score**
Down	0.32	0.43	0.37
Grip	0.33	0.40	0.36
Left	0.34	0.38	0.36
Rest	0.25	0.16	0.19
Right	0.39	0.37	0.38
Rotate	0.39	0.35	0.37
Up	0.39	0.32	0.35
Overall Accuracy		0.34	

Table 2: KNN Classification Report - Our Dataset

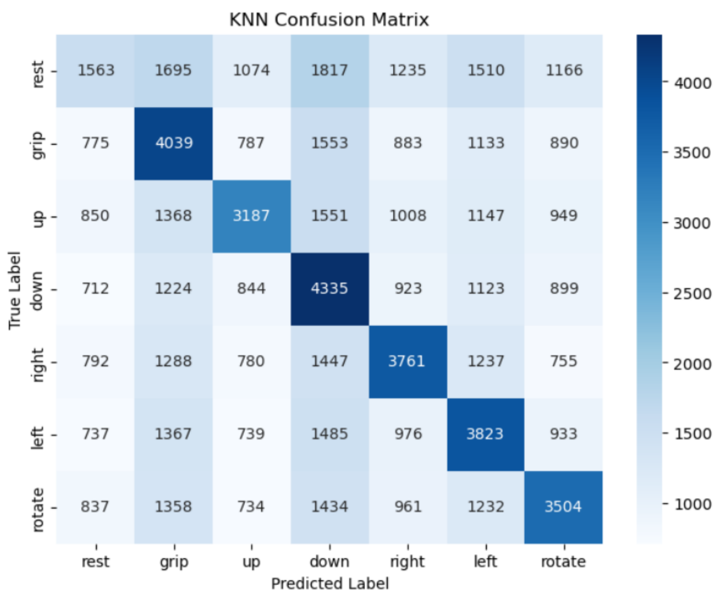

Figure 9: Confusion matrix for KNN model (Our Dataset)

4.8.2 KNN Results for DB2 Subject 1

Overall Accuracy	0.18

Table 3: KNN Classification Report - DB2 Subject 1

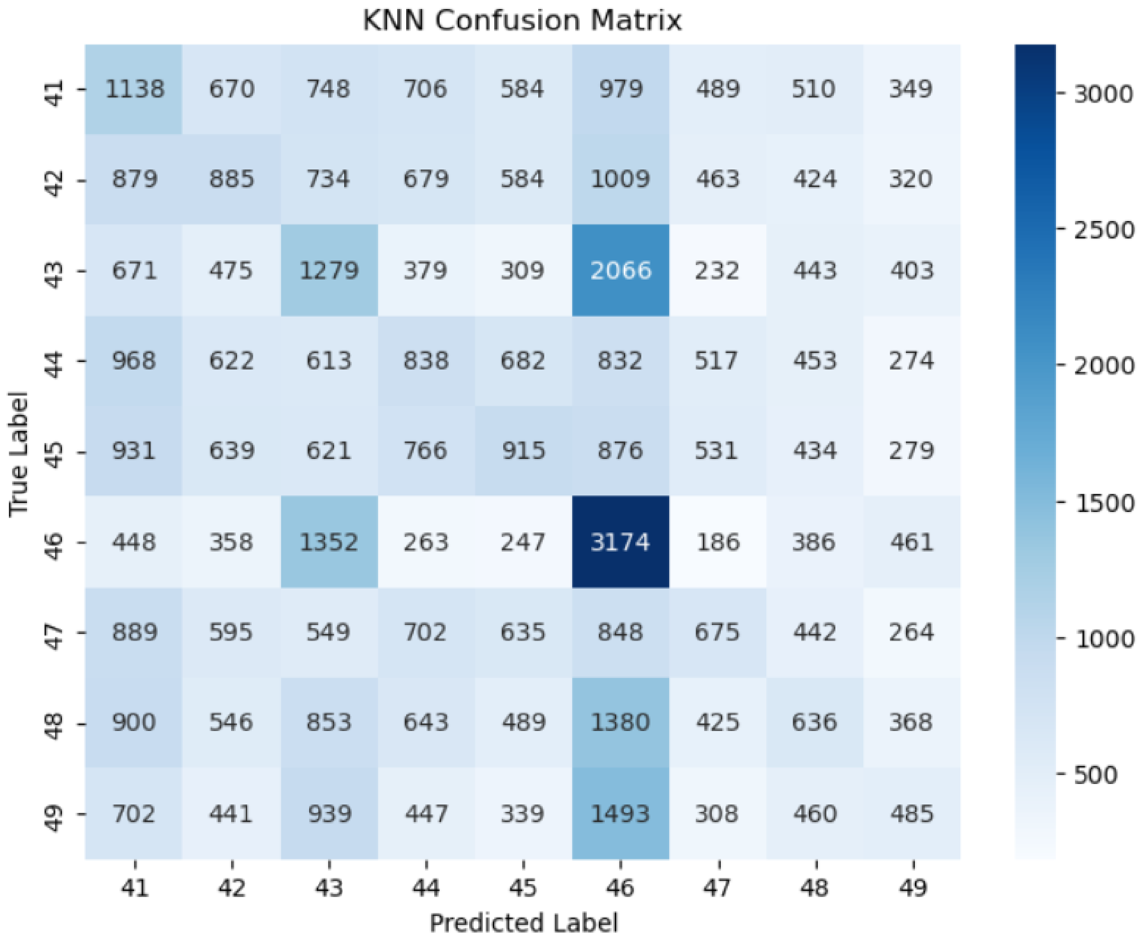

Figure 10: Confusion matrix for KNN model (DB2 Subject 1)

4.8.3 KNN Results for DB2 Subject 2

Overall Accuracy	0.21

Table 4: KNN Classification Report - DB2 Subject 2

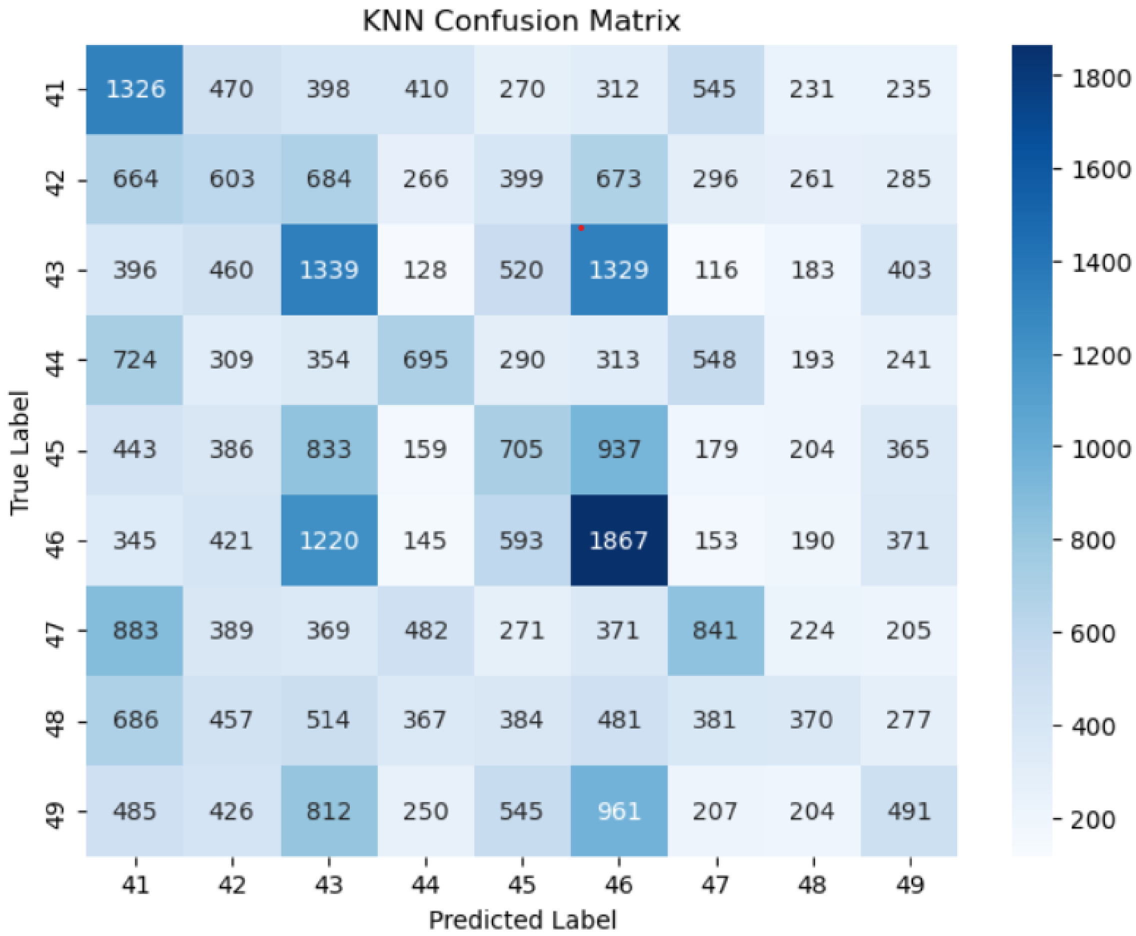

Figure 11: Confusion matrix for KNN model (DB2 Subject 2)

4.8.4 KNN Results for DB2 Subject 3

Overall Accuracy	0.22

Table 5: KNN Classification Report - DB2 Subject 3

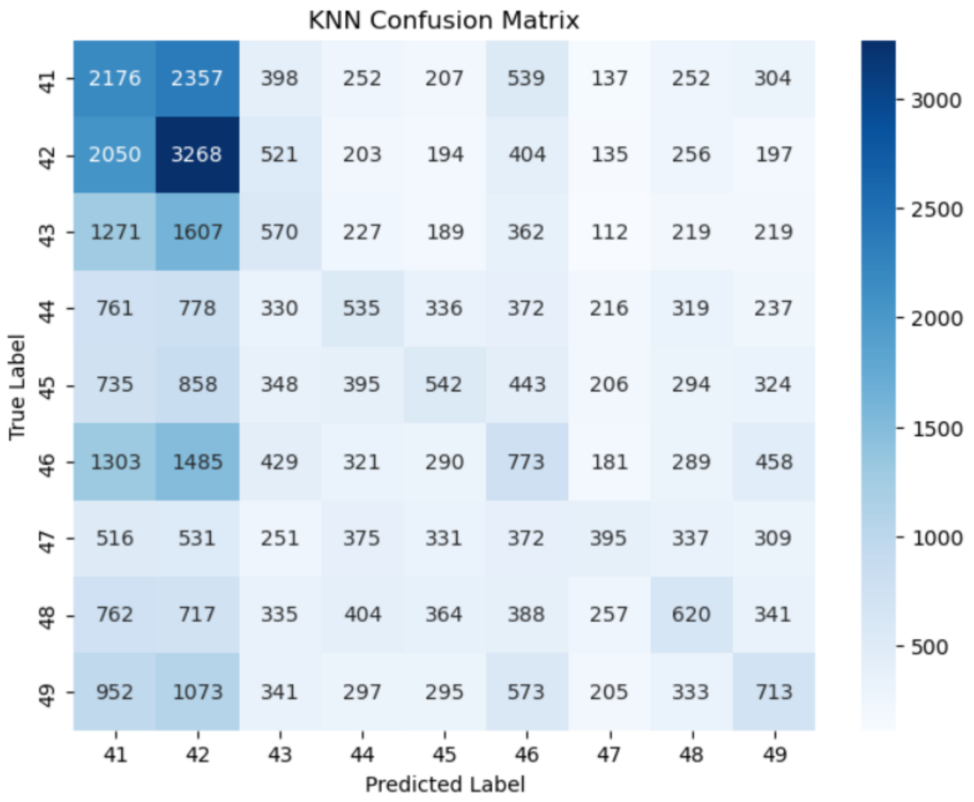

Figure 12: Confusion matrix for KNN model (DB2 Subject 3)

4.9 Random Forest Results

The Random Forest model was also trained and evaluated for classifying hand movements. classification reports and confusion matrices for multiple datasets are presented below.

4.9.1 Random Forest Results for Our Dataset

Class	**Precision**	**Recall**	**F1-Score**
Down	0.22	0.16	0.18
Grip	0.37	0.06	0.11
Left	0.17	0.44	0.24
Rest	0.35	0.03	0.06
Right	0.20	0.23	0.21
Rotate	0.19	0.32	0.24
Up	0.24	0.11	0.15
Overall Accuracy		0.19	

Overall Accuracy	0.19

Table 6: Random Forest Classification Report - Our Dataset

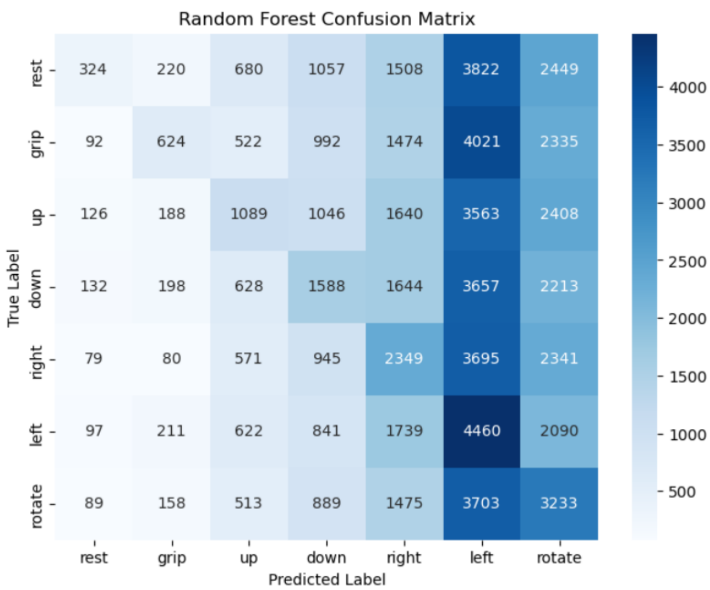

Figure 13: Confusion matrix for Random Forest model (Our Dataset)

4.9.2 Random Forest Results for DB2 Subject 1

Overall Accuracy	0.34

Table 7: Random Forest Classification Report - DB2 Subject 1

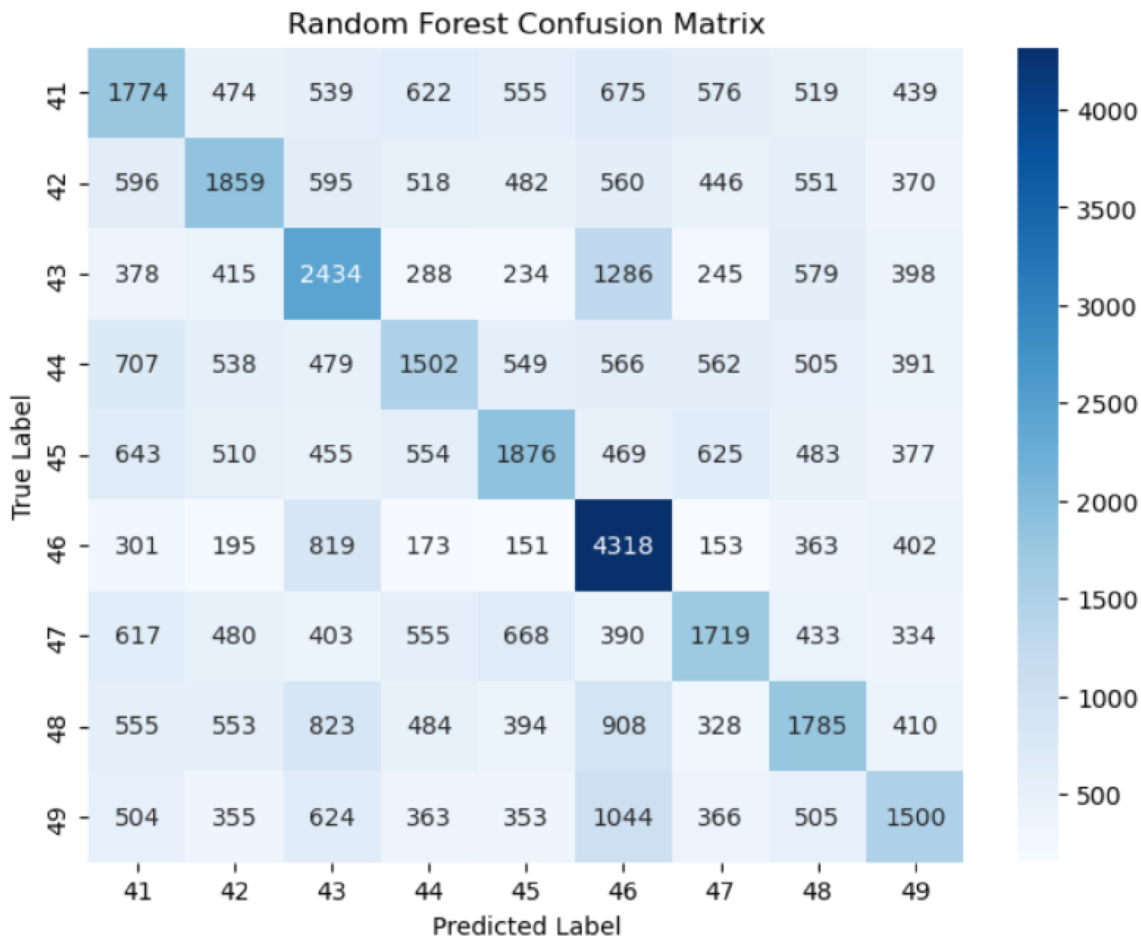

Figure 14: Confusion matrix for Random Forest model (DB2 Subject 1)

4.9.3 Random Forest Results for DB2 Subject 2

Overall Accuracy	0.40

Table 8: Random Forest Classification Report - DB2 Subject 2

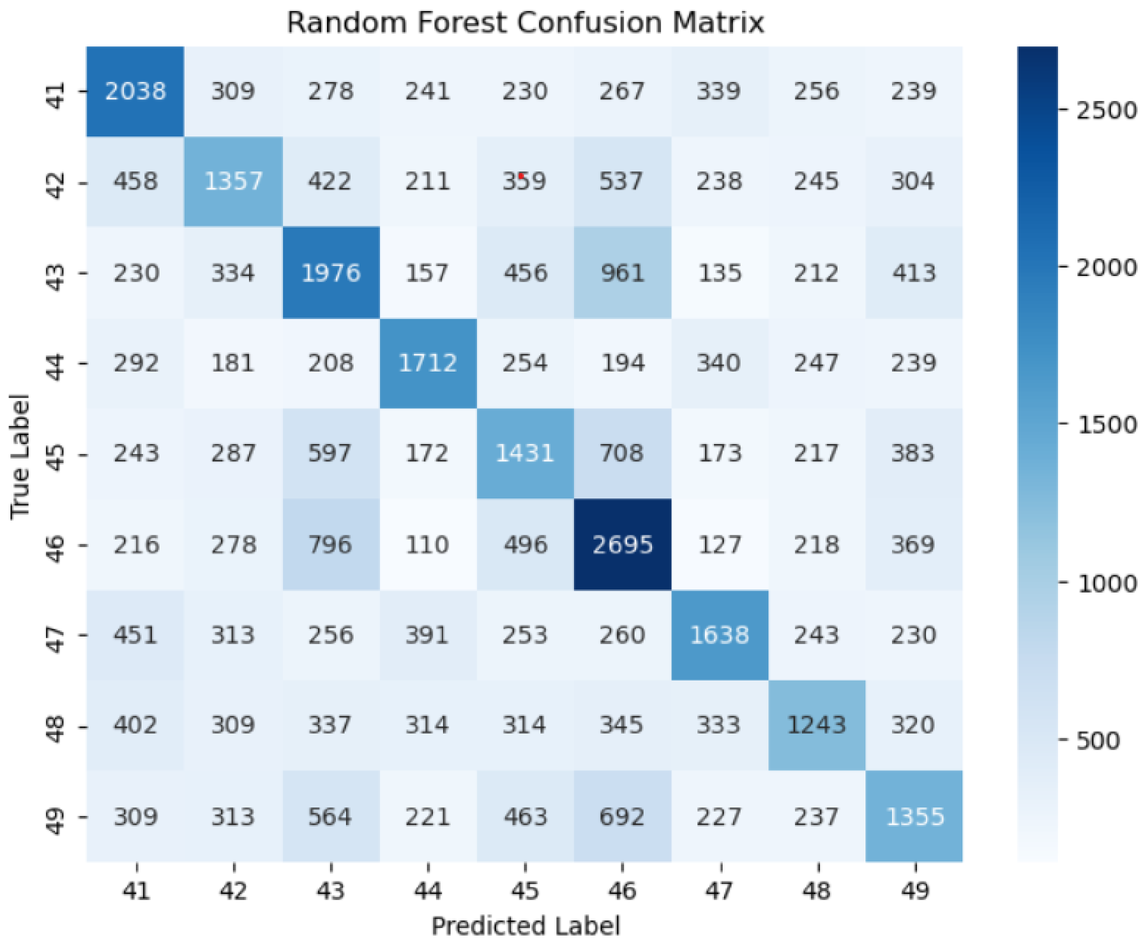

Figure 15: Confusion matrix for Random Forest model (DB2 Subject 2)

4.9.4 Random Forest Results for DB2 Subject 3

Overall Accuracy	0.33

Table 9: Random Forest Classification Report - DB2 Subject 3

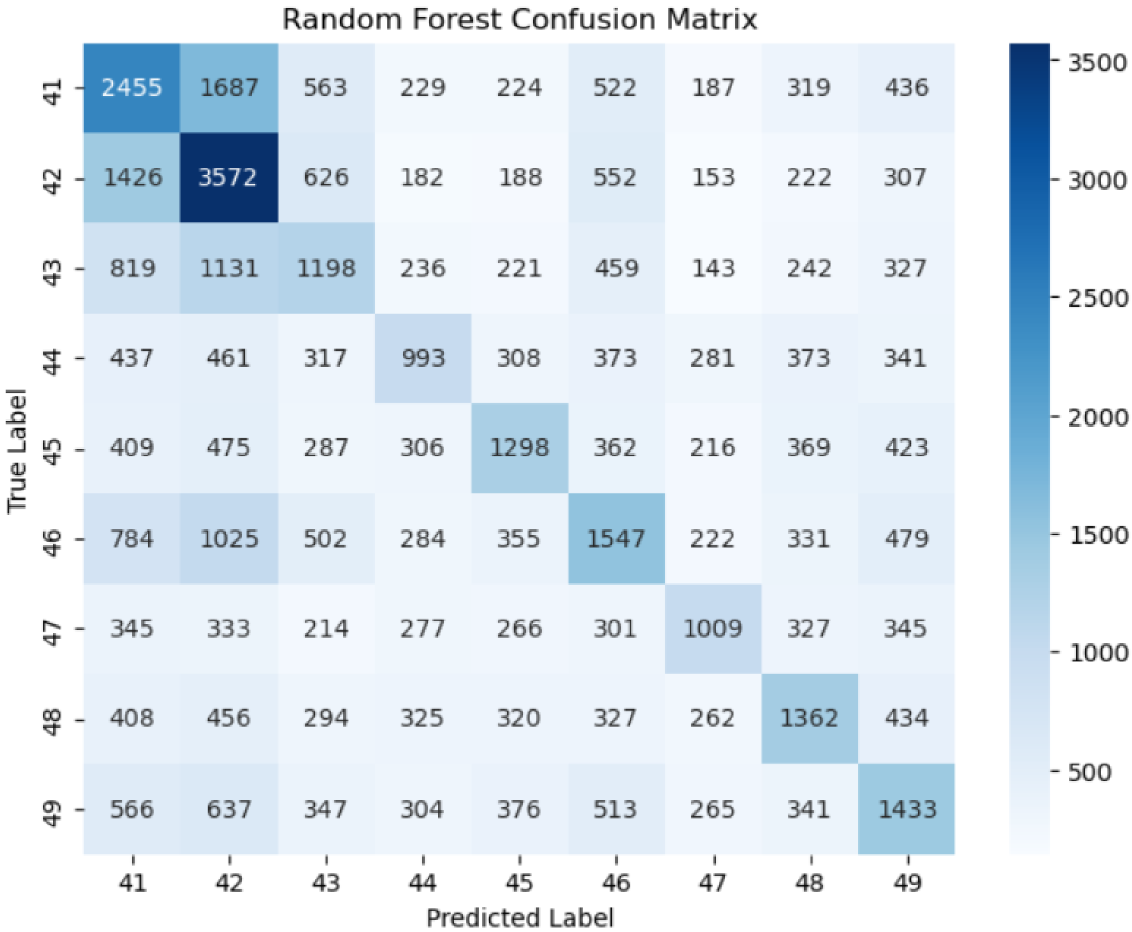

Figure 16: Confusion matrix for Random Forest model (DB2 Subject 3)

5 Analysis of Results

The classification results obtained from K-Nearest Neighbors (KNN) and Random Forest models were analyzed for multiple datasets, including our recorded dataset and three subjects from DB2 dataset. This section presents key observations, performance comparisons, and insights derived from results.

5.1 Overall Performance Comparison

The table below summarizes final accuracy scores obtained for each model across different datasets:

Dataset	KNN Accuracy	Random Forest Accuracy
Our Dataset	0.34	0.19
DB2 Subject 1	0.18	0.34
DB2 Subject 2	0.21	0.40
DB2 Subject 3	0.22	0.33

Table 10: Comparison of KNN and Random Forest Accuracy Across Datasets

5.2 Observations

1. KNN vs. Random Forest Performance:

- In our dataset, KNN outperformed Random Forest with accuracies of 0.34 and 0.19, respectively. This suggests that KNN may be more suitable for our specific data distribution.
- For the DB2 subjects, Random Forest consistently outperformed KNN, achieving significantly higher accuracy across all three subjects.

2. Dataset-Specific Performance:

- The lowest accuracy was observed for DB2 Subject 1 with KNN achieving 0.18 and Random Forest achieving 0.34, indicating possible variability or noise in the subject's EMG signals.
- The highest accuracy for Random Forest was recorded for DB2 Subject 2 at 0.40, suggesting more structured or predictable EMG data for this subject.

3. Class-Wise Performance Variability:

- For KNN, higher recall scores were observed for specific classes, indicating that some movements were detected more frequently.
- Random Forest exhibited better balance between precision and recall, leading to a more stable classification performance.
- Misclassification errors were prominent for similar movement classes, likely due to overlapping signal patterns.

5.3 Confusion Matrix Insights

The confusion matrices provide a visual representation of misclassification patterns.

1. KNN Confusion Matrices:

- High confusion between ”rest” and low-intensity movements (e.g., slight wrist rotation).
- Misclassification between similar hand movements (e.g., ”up” vs. ”grip”).

2. Random Forest Confusion Matrices:

- Improved classification for frequent movements but still struggled with minor class imbalances.
- Better differentiation between hand gestures compared to KNN.

5.4 Possible Reasons for Performance Differences

1. KNN’s Sensitivity to Noise:

- KNN is a distance-based classifier, making it highly sensitive to noisy data points.
- higher variability in EMG signals for DB2 subjects could explain why KNN performed poorly on them.

2. Random Forest’s Stability:

- ensemble nature of Random Forest makes it more robust against noise.
- Random Forest’s decision trees handle complex feature interactions better than KNN.

3. Feature Selection and Dataset Size:

- If features were not well-engineered, KNN may struggle to differentiate classes in a high-dimensional space.
- Random Forest benefits from a larger dataset size, making it a more scalable choice.

5.5 Recommendations for Improving Classification Performance

Based on analysis, following recommendations could help improve model accuracy:

1. Feature Engineering:

- Extract more informative features beyond MAV, RMS, and Zero Crossing.
- Utilize time-frequency features (e.g., wavelet transform) for better class differentiation.

2. Data Augmentation:

- Synthetic data generation through Gaussian noise addition could help train more robust models.
- Using data balancing techniques can mitigate class imbalance issues.

3. Model Optimization:

- Hyperparameter tuning for KNN (adjusting k-values) and Random Forest (increasing tree depth) could enhance performance.
- Implementing deep learning models (e.g., CNNs or LSTMs) could provide better generalization for EMG signal classification.

6 Challenges and Solutions for Improving Accuracy in This Experiment

The dataset has a class imbalance problem where ”rest” label dominates dataset. This section outlines key challenges and proposed solutions.

6.1 Challenge 1: Class Imbalance

Problem:

- ”rest” class has 352,197 samples, while other movement classes (e.g., ”grip”, ”up”, ”down”) only have 50,300 samples.
- Machine learning models tend to predict majority class (”rest”) more frequently, leading to poor classification for minority classes.
- This issue results in high overall accuracy but poor class-specific accuracy.

Solution:

- Downsample Majority Class (”rest”):
  - Reduce number of ”rest” samples to match other classes (50,300 samples).
  - This prevents model from becoming biased towards dominant class.
- Use Class Weights in Models:
  - Apply class_weight='balanced' in classifiers like Random Forest and SVM.

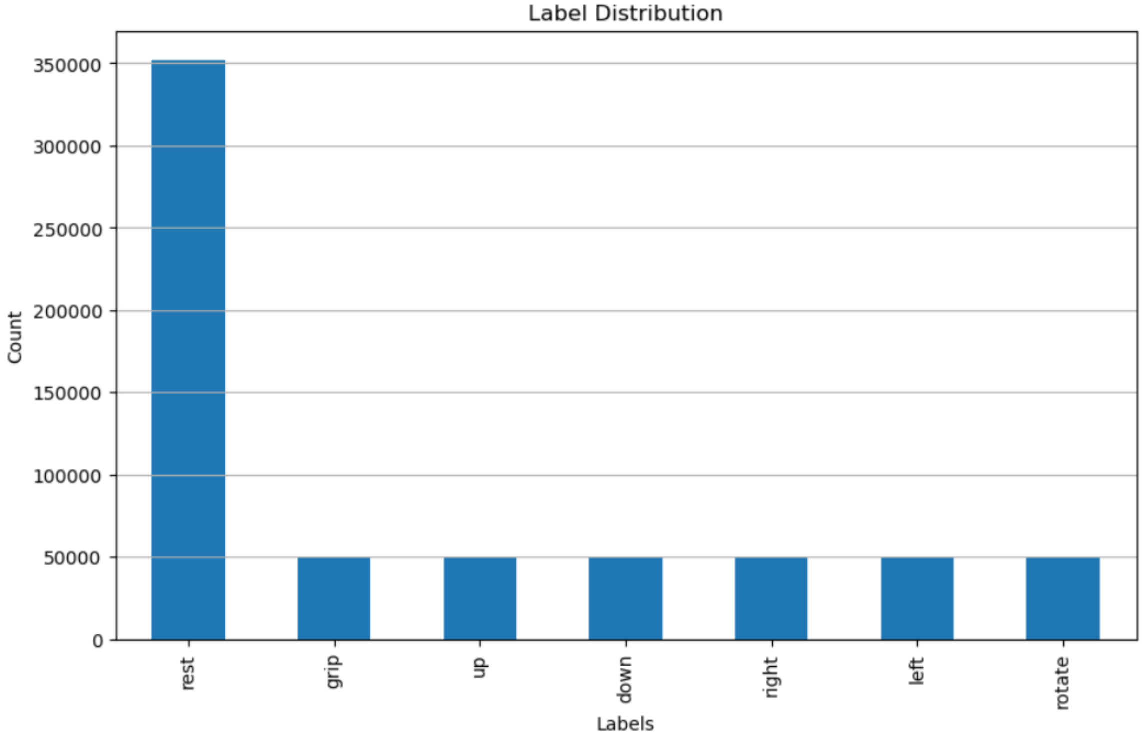

Figure 17: Label Distribution

6.2 Challenge 2: Overlapping Features

Problem:

- Some movements (e.g., "grip" vs. "down") have similar EMG patterns, leading to misclassification.
- Feature overlap reduces model effectiveness in differentiating between movements.

Solution:

- Extract More Discriminative Features:
  - Current features: MAV, STD, RMS, ZC, WL, SSC.
  - Add Frequency Domain Features: Power Spectrum, Mean Frequency.
- Use Feature Selection (PCA or Mutual Information):
  - Select only most important features to improve classification accuracy.

6.3 Challenge 3: Choosing Right Model

Problem:

- Some models overfit (Random Forest), while others struggle with non-linearity (LDA).
- SVM performs well but is computationally expensive for large datasets.

Solution:

- **Use XGBoost Instead of SVM:**
  - Faster training time and better generalization compared to SVM.
- **Deep Learning (If Dataset is Large Enough):**
  - Neural networks can learn complex EMG signal patterns.
  - Possible approaches: CNNs for spatial features, LSTMs for time-sequence analysis.

6.4 Summary of Challenges and Solutions

Problem	**Solution**
Class Imbalance	Downsample "rest", use `class_weight='balanced'`
Feature Overlap	Adding frequency domain features, use PCA
Slow Models	Use XGBoost instead of SVM
Limited Feature Set	Extract Power Spectrum, Spectral Entropy
Deep Learning Option	Use Neural Networks (if dataset is large enough)

Table 11: Summary of Challenges and Proposed Solutions

7 Summary

7.1 Accuracy and Precision of Experiment

The accuracy and precision of surface electromyography (sEMG) experiment and signal processing were evaluated as follows:

1. Accuracy of Preprocessing:

- Low-pass filtering and notch filtering effectively removed noise and interference.
- Normalization ensured consistent data ranges across all signals, enabling effective feature extraction.

2. Accuracy of Classification:

- K-Nearest Neighbors (KNN) achieved an accuracy of 34%, with misclassifications in some movements.
- Random Forest achieved an accuracy of 19%.

3. Precision in Signal Analysis:

- Time-domain features (e.g., RMS, MAV) were critical in identifying intensity of movements.
- Frequency-domain analysis confirmed dominant frequency ranges for different hand activities.

KNN and Random Forest are reliable classifier for sEMG signal processing but we need better methods like neural networks to achive better accuracy.

Preprocessing methods (e.g., filtering and normalization) significantly enhance signal quality for downstream processing.

8 References

The following sources were consulted and referenced throughout this report:

- ”Introduction to Electromyography” – Textbook on basics of sEMG.
- Research articles on preprocessing methods for sEMG signals:
  - Butterworth Filtering and Notch Filtering for sEMG signal enhancement.
  - Statistical feature extraction methods for biomedical signals.
- Documentation for Python libraries:
  - **SciPy**: For signal filtering and FFT analysis.
  - **scikit-learn**: For training KNN and Random Forest models.
- MATLAB documentation on FFT and signal visualization techniques.

9 Appendices

9.1 Files

- `raw_emg.mat`,`S1.mat`,`S2.mat`,`S3.mat`: Raw data file containing recorded EMG signals.
- `features_with_labels.csv`: Preprocessed and normalized signals used for feature extraction..
- `Code files`: Summary of classification results for all methods.

9.2 Other Pictures and FUN IN LAB :)