Exploring User Accessibility and Human-Machine Interaction Using EMG - Phase 2 Report

Isha Arora, Sai Rahul Ponnana, Tanmay Gadgil & Tarandeep Singh

December 9, 2023

GitHub: GitHub Repo

1 Summary

Biometrics plays a crucial role in contemporary authentication systems, finding applications in consumer electronics, public and private security. Biological and behavioral traits are leveraged to identify or verify an individual's identity. Traditional biometrics like fingerprints and facial scans, widely employed in smartphones and laptops, face increasing risks of data leakage and spoofing with technological advancements[3]. Novel biometric traits based on bio-signals, such as electrocardiogram (ECG), electroencephalogram (EEG), and electromyogram (EMG), are proving more resilient to spoofing than conventional methods.

For instance, facial recognition systems can be deceived using photos and videos, while obtaining bio-signals like EMG requires more coordination and specialized sensors, serving as an effective liveness detection method[3]. EMG, traditionally used in gesture recognition for prosthetic control, presents a unique dual property: gesture recognition and biometrics. Despite challenges, recent studies confirm EMG as an accurate biometric trait[9][5]. Unlike traditional biometrics, EMG is more covert, less likely to be compromised, and enables users to set customized gestures as passcodes for enhanced security, a feature not possible with EEG and ECG[7].

This report builds upon the work done for phase 1 of this project. In this phase we i) focus on improving the gesture recognition performance on unseen users by fine-tuning gesture data ii) build a suite of models to identify and classify users and iii) build a system that aims to extract a unique fingerprint for each user that can then be used as part of an authentication system.

2 Methods

In this phase of the project, we would need to revisit our feature extraction process along with coming up with new methods for user identification and authentication.

2.1 Data Collection and Parsing

The dataset includes 17 gestures, 16 of which are as listed in 1. The 17th gesture is the resting data collected after each trial. These gestures are collected over three sessions with total of 7 trials per session. Further details can be found here:

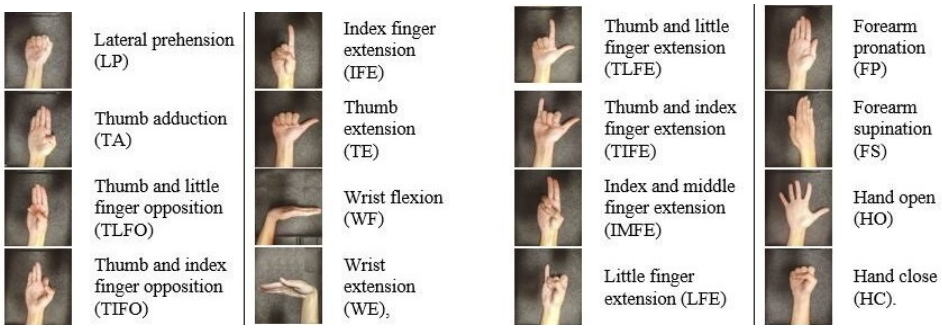

Figure 1: Gestures in the Data Set

More details can be found in appendix 7.1.

2.2 Feature Extraction - Frequency Division Technique

The initial step involved re-referencing the monopolar EMG signals from 28 channels using a common average procedure. This procedure subtracted the mean value of the 28 channels from each channel for every sample. Following this, the processed signals underwent segmentation into 200 ms width windows with a 150 ms overlap. Subsequently, each window underwent feature extraction using the frequency division technique (FDT)[4]. The FDT feature extraction method calculates the magnitude of L frequency bands. Denoting the frequency values of the two endpoints of the i^{th} band as $f_{i,1}$ and $f_{i,E}$, the i^{th} feature for each window is computed as follows:

$$FDT_i = F[\sum_{j=1}^{n_i} X(f_{i,j}), i = 1, 2, \cdots, L$$

For each five-second trial recording, the extracted features result in a P $\times$ D matrix, where P is 33 and D is 168. Multiple trials from different days were utilized for the training of biometric models.

P is the number of time windows (epochs) for every trial given a participant and gesture. The number of epochs can be calculated as:

$$epochs = \frac{TotalSamples - WindowSamples}{OverlappingSamples} + 1$$

For each trial, for a given participant and gesture,

TotalSamples = 5*2048 = 10240
WindowSamples = 0.2*2048 $\approx$ 409
OverlappingSamples = 0.15*2048 $\approx$ 307

$$epochs = \frac{10240 - 409}{307} + 1 \approx 33$$

2.3 Model Fine-tuning

Our goal for this task was to see if we could improve gesture recognition performance for new users unseen by the training process. We aim to achieve this by fine-tuning the existing model to samples extracted from new users during the calibration process. To investigate if such a process was possible, we ran several experiments where we held out users. The unturned gesture recognition performance was then recorded for each user in the held-out set. We then individually tune the model on either one, two, or three trials to see if we are getting a boost in performance. One trial here would mean one example of each gesture from a given user. We ran 50 experiments with a randomly selected holdout set for each trial size (150 experiments in total).

2.4 Feature Extraction using an Autoencoder

This task involves investigating whether we can embed gesture data across gestures in a latent space where it is possible to cluster users efficiently. This task involves training an autoencoder model that will embed our extracted features into a latent space that can be used for user authentication. Figure 2 describes the training process for such an autoencoder and its use during inference time.

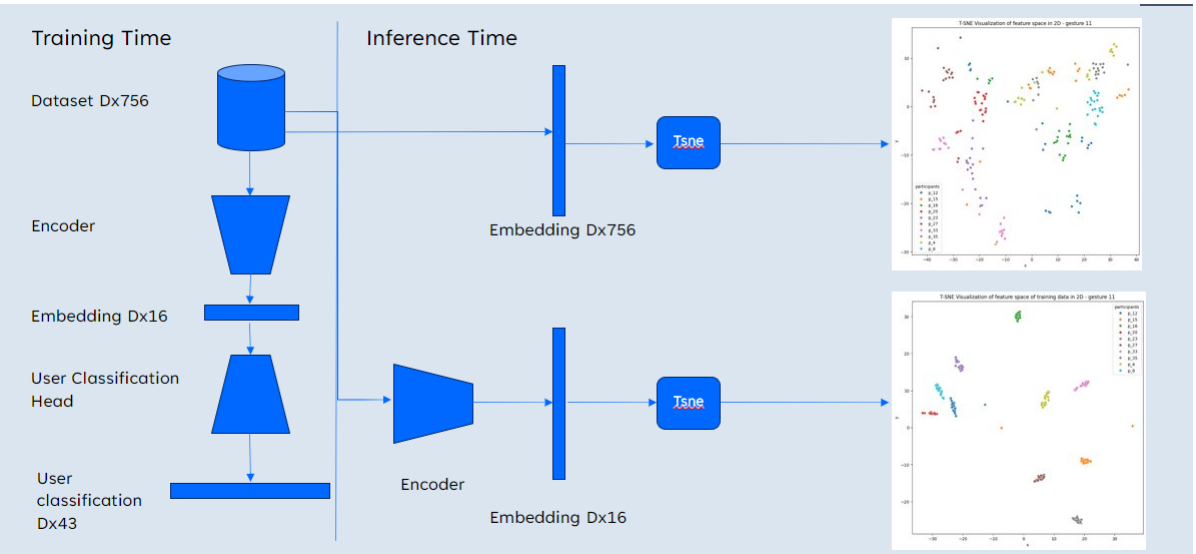

Figure 2: This process describes the training regime used for our embedding autoencoder and its use during inference time.

The training process involves the creation of a model that has the following two components i) an autoencoder that embeds the gesture features into a latent space and ii) a classification head that uses the features extracted from the latent space to classify and identify different users. The training process will be a standard classification task with a cross-entropy loss where the labels are a set of 38 users. There is also a holdout set of 5 users which the encoder model shall not interact with. We hypothesize that an encoder trained using such a training paradigm can learn a latent representation that clusters gestures from the same user together and maximizes the distance to gestures from other users.

During inference, we use the embeddings generated by the encoder to cluster users and obtain user centroids. We use these centroids to either identify or authenticate a user.

2.5 User Classification using an LSTM

In this study, we developed an LSTM-based model for user classification. The dataset, containing features from the fourth column onwards, underwent standard scaling to normalize the features. The data was then segmented into sequences of length 33, with batches of 33 utilized for training.

The target variable, representing user identities, was encoded numerically, and one-hot encoding was applied to facilitate categorical classification. The dataset was split into training and testing sets, with a 70-30 ratio.

The LSTM model architecture consisted of two layers – a 128-unit LSTM layer with return sequences set to true, followed by a 64-unit LSTM layer. Dropout layers were incorporated to mitigate overfitting, and a dense layer with a softmax activation function produced a 43-class output corresponding to the number of users.

The model was trained using categorical cross-entropy loss and the Adam optimizer for 100 epochs. Training progress was monitored for accuracy and loss, and the model was evaluated on the test set.

The classification report below provides detailed metrics, offering insights into the model's performance across different user classes. This LSTM model demonstrates promising results for user classification tasks.

2.6 User Binary Classification

We also performed a model to classify whether a participant is a legit user or an imposter by manually performing this model on 39 participants and 4 participants to be considered imposters. We performed this model before authentication just to confirm that we have the features that are just enough for any model to distinguish between authorized users on the database and any other users considered imposters.

In this study, an LSTM-based classification model is proposed to address user binary authentication challenges. The data undergoes initial preprocessing, where feature scaling is applied to ensure uniformity and improve convergence during model training. Additionally, class balancing is performed using the Synthetic Minority Over-sampling Technique (SMOTE) to mitigate potential biases stemming from imbalanced class distributions. The prepared data is then organized into sequences of length 33 and batched for training an LSTM architecture. The model comprises two LSTM layers with 64 and 32 units, respectively, augmented by dropout layers for regularization. The final layer employs a sigmoid activation function, enabling binary classification, crucial for the authentication task. The model is trained over 10 epochs, and its performance is assessed

on a separate test set. The evaluation includes accuracy metrics, a confusion matrix, and a detailed classification report, providing insights into the model's effectiveness across different authentication scenarios. The obtained results suggest promising capabilities for user authentication, opening avenues for further refinement and exploration in future studies.

2.7 User Authentication

Once the encoder model for user embeddings is trained, we use a clustering-based system to authenticate users.

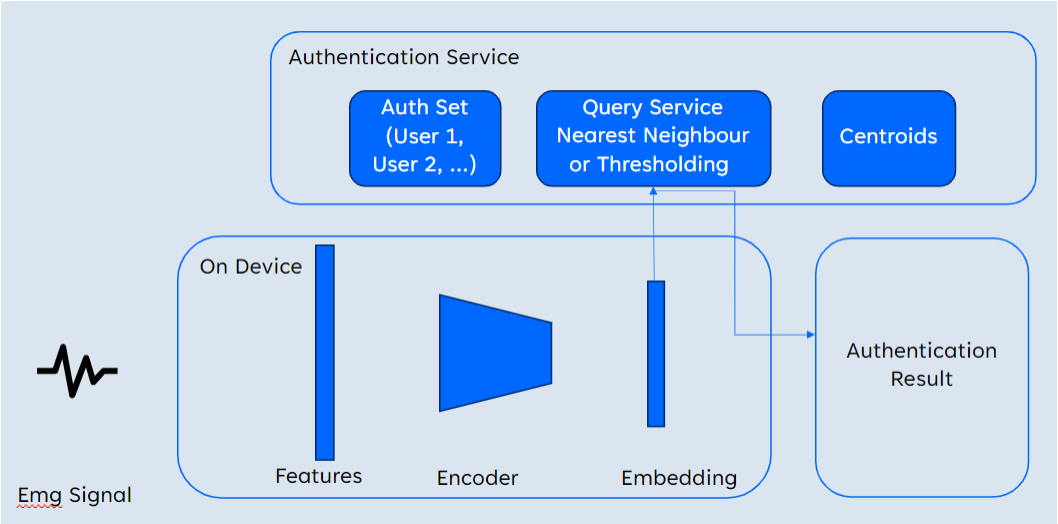

Figure 3: This process is our workflow for user authentication models.

The first step is to identify user centroids in the embedding space (16 dimensions) that can be used to identify a user for each gesture. We can see in section 3.2, that once the data is passed through the autoencoder, the clusters generated are tighter and well separated from other clusters. We see the existence of outliers in this data. We filter out the outliers using median filtering. Once we create a filtered set of clusters, we find the centroid for a given participant and a given gesture and use that as a reference to identify or authenticate users. Once the centroids have been identified, we store them on an authentication server. Figure 3 gives a brief overview of the authentication process.

The authentication process starts with a user performing a gesture. Features from the gestures are extracted and embedded using the autoencoder. The user embeddings are sent to the authentication service which can either search for the nearest user or check if it matches another user within the authentication set based on a threshold (better security). The service then returns whether a user has been authenticated or not.

To obtain performance metrics for our process we conduct experiments that randomly select users to be a part of the authentication set. We then check the False Acceptance Rate and the False Rejection Rate for each gesture. We vary the size of the authentication set (1,5,10 and 20), and the sampling population (just the training set, just the testing set, and both the training and testing set) to obtain metrics for all these experimental parameters. Results for the same can be seen in section 3.4

2.8 User Identification using a learned latent space

The above process can also be used to identify users. Once the service receives the embedding, it searches the centroid database to find the nearest or most likely user. New user centroids can be learned by conducting a calibration process. Users can then be identified by finding the closest user centroid to their gesture.

3 Results

3.1 Model Finetuning

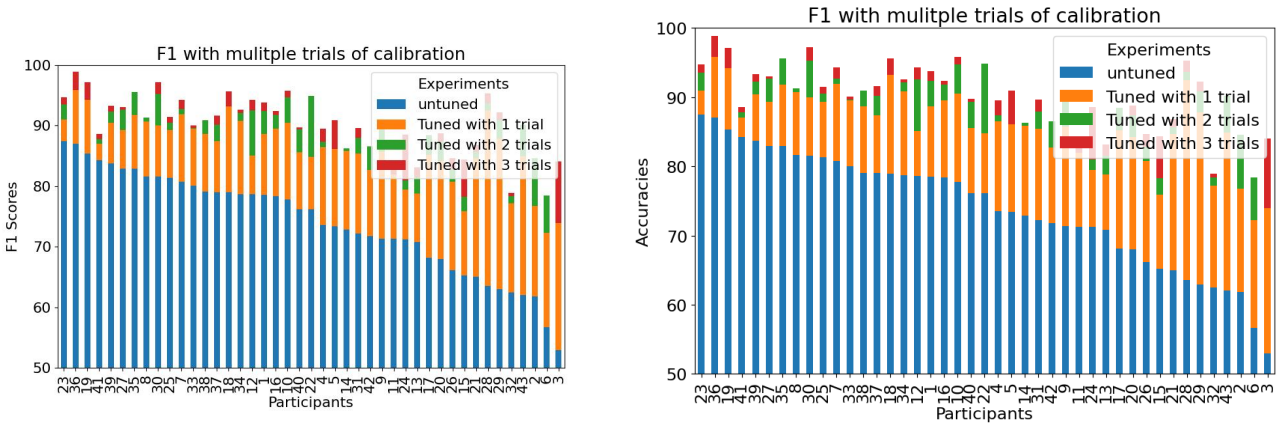

Figure 4: Tuning experiment for improving gesture recognition performance on hold-out data

We conducted 150 experiments where we aggregated hold-out performance for each participant. The blue bar represents aggregated performance for a participant for each holdout set. Next, we tune the model with one, two, and three trials represented by the yellow, green, and red bars. The figure 4 represents accuracy and F1 scores for these experiments.

3.2 User Embeddings using the Autoencoder Model

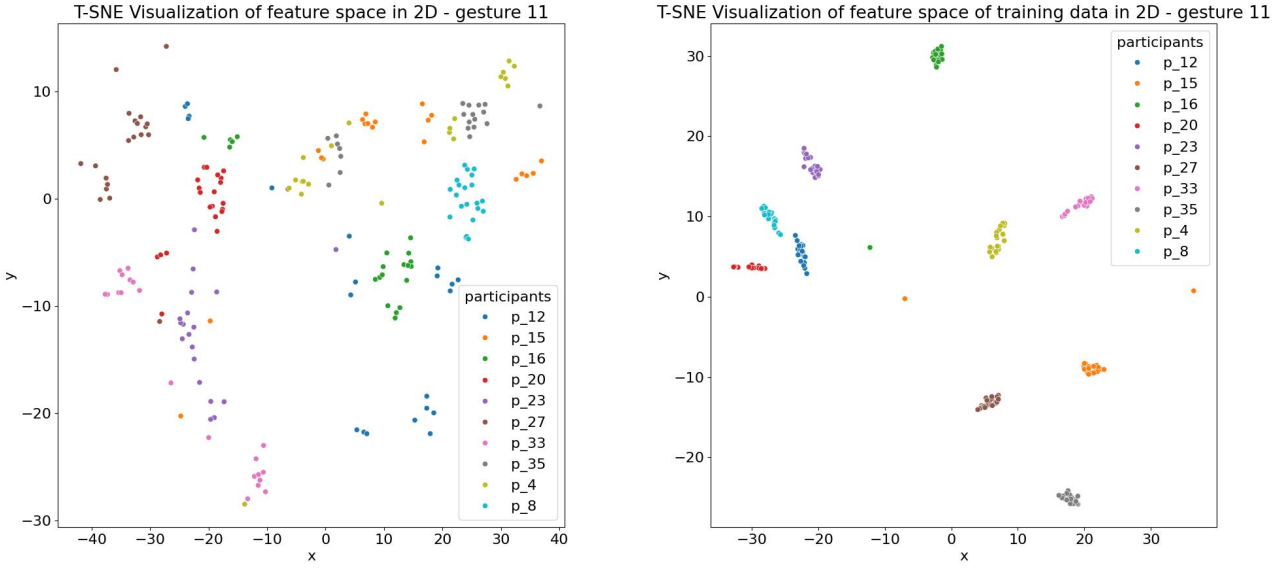

Figure 5: **Left**: T-SNE visualization before embedding using the encoder model. **Right**: T-SNE visualization after embedding using the encoder model.

Best-perf classes	Precision	Worst-perf classes	Precision
1	0.99	43	0.90
27	0.99	20	0.90
17	0.99	25	0.88
41	0.98	16	0.87
26	0.98	14	0.84

Metric	Value
Accuracy	0.94
Precision	0.94
Recall	0.94

Table 1: *Best Performing Users; Worst Performing Users; Performance Metrics Overview: Achieving High Accuracy, Precision, and Recall.* This table presents a comprehensive evaluation of a model's performance, specifically in a classification task. The metrics provide insights into the overall effectiveness of the model and highlight its proficiency in correctly classifying instances.

Gestures	TD–1 Auth		TD–10 Auth		TestD–2 Auth		O–1 Auth		O-10 Auth	
	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR
2	0.0	2.4	0.4	2.2	6.3	9.5	0.6	4.4	1.4	3.8
14	0	3.4	0.7	3.2	9.8	14.7	1.1	8.3	2.2	7.8
3	0.1	2.9	0.1	2.9	5.1	9.5	0.9	5.9	1.9	5.7
5	0	4.7	1.1	2.6	6.1	9.5	1.1	6.8	2.3	6.5

Table 2: Best Gestures - Performance in Percentage. TD - users in the training dataset, TestD - Users in the hold-out set, O - A dataset with all users both in the training and holdout set. 1 Auth indicates that there is one user in the auth set, 10 auth indicates 10 users in the authentication set.

3.3 User Classification using LSTM

Users 1, 27, 17, 41, and 26: These classes exhibit outstanding precision values, ranging from 98.10% to 99.09%. The model excels in accurately identifying instances belonging to these classes. *Users 43, 20, 25, 16, and 14:* Precision in these classes ranges from 84.21% to 90.74%. These classes represent the model's struggle in avoiding false positives for certain categories. The Overall performance of the model is mostly consistent with an overall accuracy of 94% and F1 score 0.94 as shown in Table 1.

3.4 Authentication Results

Table 2 highlights the FAR and FRR rates for our authentication experiments. FAR stands for False Acceptance Rate which indicates the number of intruders that were authorized and FRR stands for False Rejection Rate which represents the number of authorized users that are rejected. Certain gestures are particularly good for user identification as they amplify the differences between users. Gestures 2, 14, 3, and 5 have the best performance. We have also performed user identification results for which can be found here in Appendix 7.3.

3.5 User Identification in a Cross-Day Analysis setting

Since our data covers 3 sessions over 3 different days, we wanted to see if the data was homogeneous enough to use a single session or two sessions to help identify users.

Figure 6 shows some disparity between sessions for some participants like Participant 43 as seen in the color lavender. The circles, crosses, and squares are quite visibly separated. On the other hand, Participant 41 in green does not seem to have any visible separation.

This brought up the need to analyze sessions separately and focus on a Cross-Day Analysis. Using the LSTM model as explained earlier, we performed two types of Cross-Day Analysis:

Single Cross-Day (SCD) Analysis focused on using a single session for our training data and considered the remaining two sessions for testing. Six trials from the training session were considered to be training data for the model and one trial each from the testing sessions was the testing data. This step was repeated seven times by varying the train and test trials, resulting in a seven-fold cross-validation for each training session. The cross-validation was further repeated for each session to be considered as the training session.

Cumulative Cross-Day (CCD) Analysis focused on using two sessions for training and the last session was used to test the model. Similar to Single Cross Day Analysis, six trials from each of the training sessions were considered to be training data for the model and one trial from the testing session was the testing data. This step was repeated seven times

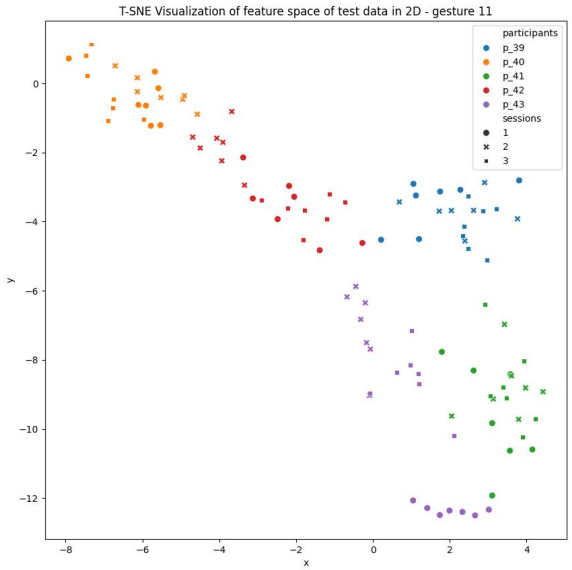

Figure 6: Session-wise difference in test embeddings. The cross, dot, and square represent different sessions and this chart shows variability between sessions.

by varying the train and test trials, resulting in a seven-fold cross-validation for each combination of training sessions. The cross-validation was further repeated for all combinations of the sessions.

These analyses are further visualized in Figure 9 in the appendix.

The biometric performance is reported as an average across these repetitions. Averaging the results helps smooth out anomalies and provides a clearer picture of the system's performance over time.

Overall Mean Accuracy	**Single Cross Day Analysis**	**Cumulative Cross Day Analysis**
Rank-1	35.87	56.27
Rank-3	55.38	74.29
Rank-5	65.08	80.47

Table 3: Cross-Day Analysis: Performance in Percentage

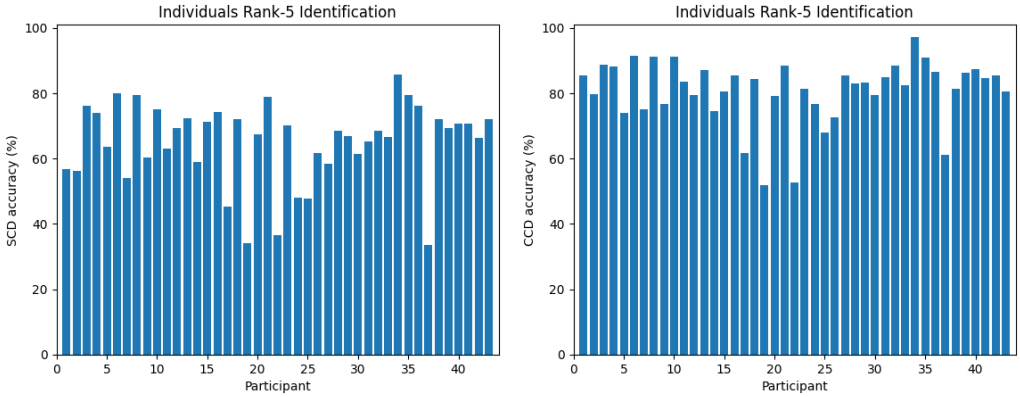

Figure 7: Individual Identification Rank-5 performance. **Left:** Single Cross-Day Analysis **Right:** Cumulative Cross-Day Analysis

As seen in Figure 7 some participants were particularly hard to identify when using a single session for training as compared to when using two sessions which could be attributed to intra-participant variability (due to mood, physical condition, fatigue) or even differences in consistency over sessions (some individuals may naturally be more consistent in their gestures than others). Training over more sessions probably helps better capture this variability and lack of consistency allowing for better generalization.

4 Discussion

Traditional biometric methods, like fingerprint and facial recognition, have seen widespread integration into everyday consumer applications, benefitting from decades of continuous research and the availability of extensive datasets encompassing thousands of individuals. However, the escalating threats of data breaches and spoofing in our technologically advanced society necessitate exploring alternative biometric traits such as biosignals, gait, and keystroke dynamics. Prior studies indicate that these unconventional traits exhibit biometric authentication performance ranging from extremely low equal error rates. The present investigation, focusing on wrist electromyography (EMG)-based biometrics within a multi-code framework for hand gesture combinations, reveals a noteworthy authentication FAR of 0.1%-0.5% on training data and 6%-8% on held-out participant data and an identification accuracy of 94.37%. Notably, this approach allows for code customization based on knowledge, providing enhanced security compared to other biosignals like ECG and EEG, which lack the same level of customizability. Despite the challenges posed by non-stationary factors affecting EMG signals, cross-day analysis underscores sufficient commonalities for robust multi-day EMG-based authentication. Furthermore, biosignals, including EMG, offer heightened resistance to spoofing and serve as crucial indicators for liveness detection, ensuring the user's physical presence during authentication attempts. This study contributes valuable insights into the efficacy and security of EMG-based biometrics, emphasizing its potential for practical and reliable authentication systems.

5 Future Scope

The exploration of wrist electromyography (EMG)-based biometrics within a multi-code framework for hand gesture combinations has yielded promising results. The present investigation not only establishes the potential of EMG-based authentication but also opens avenues for future research and enhancements in the field of biometric security. The following aspects offer compelling directions for future exploration:

- Retraining and Recalculation Strategies: Finding the optimal balance between retraining the model and recalculating centroids is crucial. Future research could delve into advanced training regimes and model adaptation strategies to improve the system's adaptability to evolving user characteristics.

- Encoder Model Enhancement: Exploring alternative approaches to create the encoder model may contribute to improved feature extraction and classification. Innovative training regimes and novel architectures for the encoder model can be investigated to enhance the robustness and efficiency of the authentication system.

- Cross-Session EMG Data Mitigation: Addressing the challenges posed by cross-session EMG data is essential for real-world applicability. Future work should focus on developing methods to mitigate the effects of variations in EMG signals across different sessions, ensuring consistent and reliable authentication performance.

- Synthetic Data Generation: Investigating the generation of synthetic data or the collection of new data can be instrumental in expanding the dataset and improving the generalization capabilities of the model. This approach can contribute to handling diverse user scenarios and increasing the system's overall effectiveness.

- Time-Frequency Representation with Continuous Wavelet Transform: Assessing the performance of a time-frequency representation of signals, particularly through techniques like Continuous Wavelet Transform, presents an exciting avenue for exploration. This could provide valuable insights into whether such representations lead to improved authentication accuracy.

- Narrowing Feature Scope for Wrist Devices: Tailoring the authentication system to focus specifically on wrist electrodes opens up possibilities for integration with wrist-based devices, such as smartwatches. Future research can explore the optimization of features and algorithms for this specific application, enhancing user experience and device security.

- Liveness Detection and Spoofing Resistance: Further investigations into the integration of biosignals, including EMG, for liveness detection could bolster the system's resistance to spoofing attempts. Understanding the nuances of biosignal responses in real-time scenarios can enhance the security of the authentication process.

6 Statement of Contribution

This project has enabled us to explore many paths to solve the gesture recognition and user authentication and identification problems.

Tanmay conducted experiments and performed analysis to fine-tune the models for gesture recognition performance. He also built out the user authentication architecture and conducted experiments to verify user authentication results. He has also worked on building several base models including for user identification and classification.

Isha conducted experiments and performed the analysis in regards to User Identification Cross-Day Analysis for Single Cross-Day Analysis and Cumulative Cross-Day Analysis. She also performed the CWT transformation Feature Extraction using Fast Fourier Transform in MATLAB.

Rahul dealt with Feature extraction, performed Discrete Fourier Transform, Feature Division Technique and few other Signal Processing Techniques to extract features from the raw EMG signals. He has build the User Classification model using LSTM and also worked on building Binary User Classification, an initial baseline for User Authentication model, by conducting experiments on a part of participants.

7 Appendix

7.1 Data Collection details

There were 43 participants (23 males, 20 females) in our study. The participants, with an average age of 26.4±2.89, had an average forearm length of 25.2±1.74 cm, and an average wrist circumference of 16.2±1.21 cm. The experimental setup included a PC and a monitor on a desk, positioned 0.75 m in front of a height-adjustable chair. EMG signals were acquired using the EMGUSB2+ (OT Bioelettronica, Italy), a commercial bio-signal amplifier, filtered between 10 Hz and 500 Hz with a gain of 500, and sampled at 2048 Hz. The electrodes (AM-N00S/E, Ambu, Denmark) were placed in a ring around the wrist after preparing the skin[5][7]. Electrode positions were standardized across participants. Participants, seated comfortably, received visual instructions on the computer screen for 16 hand and wrist gestures, including gestures like lateral prehension, thumb adduction, and wrist flexion. The order of these gestures was randomized, followed by a resting trial. A ten-second rest period separated each trial. One run comprised continuous data acquisition of 17 gestures, and each subject performed seven runs, totaling 119 trials. Participants could request additional rest as needed. The detailed pictorial representation of gestures is provided in the Appendix.

Data was collected from three distinct days/sessions, labeled Session 1, Session 2, and Session 3. Each session was stored in separate directories. Within each session, information about each participant was parsed for subsequent analysis [6]. A comprehensive set of features was extracted from the parsed data. These features encompassed attributes from both the time and frequency domains [1]. The features were calculated for all of the 28 sensors, providing a comprehensive representation of the signal characteristics. The parsed and feature-extracted data from Session 1, Session 2, and Session 3 were integrated into a single cohesive dataset for further analysis.

7.2 Continuous Wavelet Transform

Continuous Wavelet Transform (CWT) is frequently used in signal processing to analyze signals that are continuous in time (such as EMG signals). It is a mathematical tool used to analyze signals using wavelets of different scales. It is, in essence, a way of decomposing a signal into its constituent wavelets and analyzing each wavelet at different scales. A time-frequency analysis method, it presents the input data sequence as a 2D graphic of the time-frequency representation.

7.2.1 Feature Extraction

Out of the 28 forearm and wrist channels, the 12 wrist monopolar EMG signals were considered. The wrist presents a more feasible and attractive position for biometrics applications because wrist-worn devices are well-established[2] hence becoming the focus of our analysis.

$$CWT_w(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\psi\left(\frac{t-b}{a}\right) dt \tag{1}$$

where:
$x(t)$ = input signal
w = angular frequency
a = scale parameter
b = translation parameter
ψ = mother wavelet

Although our EMG data was filtered between 10 Hz and 500 Hz, the relevant information for the EMG signal band is within 20-450Hz and hence the CWT was applied to this frequency range. Generally, the mother wavelet used to generate CWT in EMG signals is the Morlet wavelet

Each participant performed each gesture in 7 trials over 3 sessions. Choosing a single gesture, in our case we chose "Thumb and Index Finger Opposition", CWT was performed for each monopolar signal resulting in a total number of samples per participant:

Samples per participant = number_trials * number_channels * number_sessions = 7 * 12 * 3 = 252
Total samples = Samples per participant * number_participants = 252 * 43 = 10836

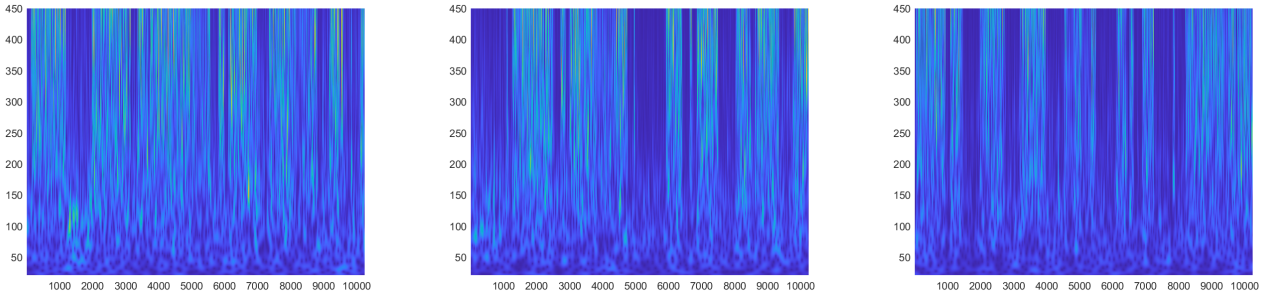

Figure 8: Time-frequency graphs using CWT of one wrist channel EMG signal for three different subjects

It is to be noted that each gesture is represented by 12 monopolar channels.

7.2.2 Modeling

The modeling phase for predicting users involved the implementation of a Convolutional Neural Network (CNN) to analyze the Continuous Wavelet Transform (CWT) of EMG signals. The architecture of the CNN was designed to extract meaningful features from the CWT representations and classify the data into different classes. However, during the training process, several challenges emerged, primarily related to computational resources by image CWT dataset, which hindered the completion of the model training within the specified timeframe.

The CNN2D architecture consisted of three convolutional layers followed by max-pooling layers, aiming to capture hierarchical patterns in the CWT data.

Despite the well-defined architecture, training the model became a challenging task due to limited computational resources. The extensive computations involved in processing the CWT data and training a deep neural network led to significant delays, preventing the completion of the training process within the project deadline.

Moreover, to facilitate the training of the CNN2D model, a dataset was for each participant. This dataset preparation step involved careful preprocessing of the CWT data, ensuring that it was compatible with the CNN architecture. The intricacies of creating participant-specific datasets added an additional layer of complexity to the overall modeling process.

7.3 User Identification Results

	Gestures	**Accuracy as Percentages**
Best Gestures	Gesture 2	97.6
	Gesture 7	96.5
	Gesture 3	96.1
	Gesture 14	96.1
Worst Gestures	Gesture 1	89.9
	Gesture 17	72.6

Table 4: Accuracy on the training dataset

	Gestures	Accuracy as Percentages
Best Gestures	Gesture 6	92.4
	Gesture 10	90.4
	Gesture 3	89.5
	Gesture 11	89.5
Worst Gestures	Gesture 12	69.5
	Gesture 17	58.1

Table 5: Accuracy on the test dataset

7.4 Cross-Day Analysis

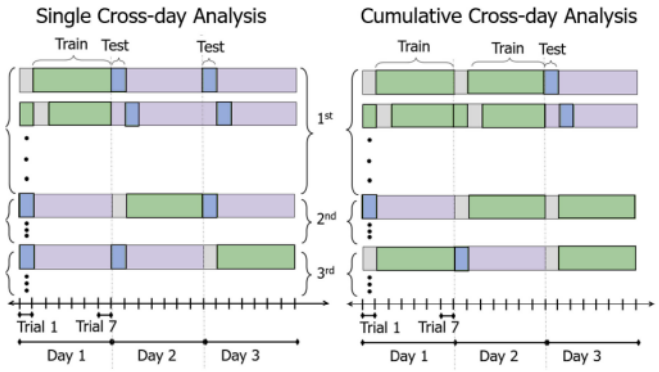

Figure 9: Single cross-day and Cumulative cross-day analysis. The corresponding training data for each analysis are represented in green; the testing data are represented in blue. The x-axis represents the timeline of the study consisting of multiple days; the x-axis tick marks represent the different trials performed in one day.[8]

References

[1] Reza Bagherian Azhiri, Mohammad Esmaeili, and Mehrdad Nourani. Emg-based feature extraction and classification for prosthetic hand control. *arXiv preprint arXiv:2107.00733*, 2021.

[2] Fady S. Botros, Angkoon Phinyomark, and Erik J. Scheme. Electromyography-based gesture recognition: Is it time to change focus from the forearm to the wrist? *IEEE Transactions on Industrial Informatics*, 18(1):174–184, 2022.

[3] Gabriel Dahia, Leone Jesus, and Mauricio Pamplona Segundo. Continuous authentication using biometrics: An advanced review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10(4):e1365, 2020.

[4] Boyu Fan, Xiang Su, Jianwei Niu, and Pan Hui. Emgauth: Unlocking smartphones with emg signals. *IEEE Transactions on Mobile Computing*, 2022.

[5] Xinyu Jiang, Ke Xu, Xiangyu Liu, Chenyun Dai, David A Clifton, Edward A Clancy, Metin Akay, and Wei Chen. Cancelable hd-semg-based biometrics for cross-application discrepant personal identification. *IEEE Journal of Biomedical and Health Informatics*, 25(4):1070–1079, 2020.

[6] Kyung Hyun Lee, Ji Young Min, and Sangwon Byun. Electromyogram-based classification of hand and finger gestures using artificial neural networks. *Sensors*, 22(1):225, 2021.

[7] Ashirbad Pradhan, Jiayuan He, and Ning Jiang. Score, rank, and decision-level fusion strategies of multicode electromyogram-based verification and identification biometrics. *IEEE Journal of Biomedical and Health Informatics*, 26(3):1068–1079, 2021.

[8] Ashirbad Pradhan, Jiayuan He, Hyowon Lee, and Ning Jiang. Multi-day analysis of wrist electromyogram-based biometrics for authentication and personal identification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 5(4):553–565, 2023.

[9] Sherif Said, Abdullah S Karar, Taha Beyrouthy, Samer Alkork, and Amine Nait-ali. Biometrics verification modality using multi-channel semg wearable bracelet. *Applied Sciences*, 10(19):6960, 2020.