Data Augmentation for EMG-Based Finger Movement Detection

by

Egor Maksimenka

Supervised by Juhua Hu

A senior thesis submitted in partial fulfillment of the departmental honors requirements for the degree of

Bachelor of Science
Computer Science & Systems
University of Washington Tacoma

June 2021

Presentation of work given on June 2nd, 2021

The student has satisfactorily completed the Senior Thesis, presentation and senior elective course requirements for CSS Departmental Honors.

Faculty advisor: ______________________ Date 06/12/2021

CSS Program Chair: Donald Ch ______________________ Date 06/21/2021

UNIVERSITY OF WASHINGTON TACOMA

UNDERGRADUATE THESIS

Data Augmentation for EMG-Based Finger Movement Detection

EGOR MAKSIMENKA
egorm@uw.edu

Thesis advisor : Prof. Juhua Hu

Major : Computer Science and Systems
Department : School of Engineering and Technology

Thesis advisor : Prof. Juhua Hu

Author : Egor Maksimenka

Data Augmentation for EMG-Based Finger Movement Detection

Abstract

Electromyography (EMG) is an electrodiagnostic medical technique that allows for the recording and analysis of electrical activity within a person's muscular system. Through pattern recognition of such electrical signals during motion, experts can isolate medical abnormalities or motion patterns. A consequence of this is that, if isolating the muscles associated with finger motion, EMG signals can be used to predict the motion of specific fingers.

Identification of finger motion is an important classification problem that falls under the umbrella of human activity recognition. Such practices have many applications and will be thoroughly investigated in the upcoming decades. For example, if an individual loses an extremity such as a hand, EMG classification can be implemented in the replacement prosthetic. Additionally, cybersecurity protocols benefit from such classification technologies as well. Utilizing individualized hand gesture signals could potentially see use-cases in the future of cybersecurity. The applications are limitless and it is important to streamline and improve the EMG classification process. However, EMG classification does have issues that make such tasks difficult. The data is often noisy and potentially limited in size. Previous research has found that computing time-series features from each sequence is sufficient to create input for a machine learning model. While this helps deal with the noisy input, it does raise the time overhead in model fitting.

This thesis will tackle this problem by directly utilizing the time-series themselves as an input, as opposed to computed feature vectors. Additionally, this thesis will utilize synthetic data as a data augmentation solution to augment limited training data and balance any potential imbalances in the data. Together, these strategies allow for direct usage of raw time-series sequences as input, and yield high performance in finger movement detection, which demonstrates the effectiveness of the proposed data augmentation strategy.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

When engaging a muscle or undergoing motion, the corresponding muscle stores a certain amount of electrical potential energy. This energy changes throughout the range of the motion. An electromyograph (EMG) records this data, and stores it in the format of a time-series sequence of data [6]. Each muscle has a distinct structure, thus for different motions utilizing potentially different muscles, the resulting EMG time-series will have different underlying patterns. This fact can be taken advantage of and utilized in classification problems [6].

This thesis tackles the problem of classifying finger motion given EMG data recordings. Each finger has differing EMG signal patterns, since the forearm has to contract in different ways to engage different fingers. Given this, it is possible to identify what fingers are being engaged at any moment based on a signal as the example illustrated in Fig. 1.1. This problem, which falls under the umbrella term of human activity recognition, has a myriad of potential applications. Prosthetics would benefit from the capability of predicting motion from electrical signals [7]. If one lost their finger in a work-place accident or some other context, an EMG-driven replacement prosthetic would potentially be more accurate and respond faster than other prosthetics, with greater functionality. Additionally, cybersecurity as a field would potentially benefit from EMG classification of finger movement detection [4]. Hand-based gestures have been something of consideration lately with security protocols.

However, classification of EMG signals comes with various challenges. First, EMG data often consists of multiple time-series sequences that store electrical potential energy recordings. This data is very noisy, and picking up on underlying patterns of such a sequence is difficult. Prior research focused on feature computation, and strategies involved in computing such features [1]. Absolute mean value, Wavelength, and others are examples of such features, and are

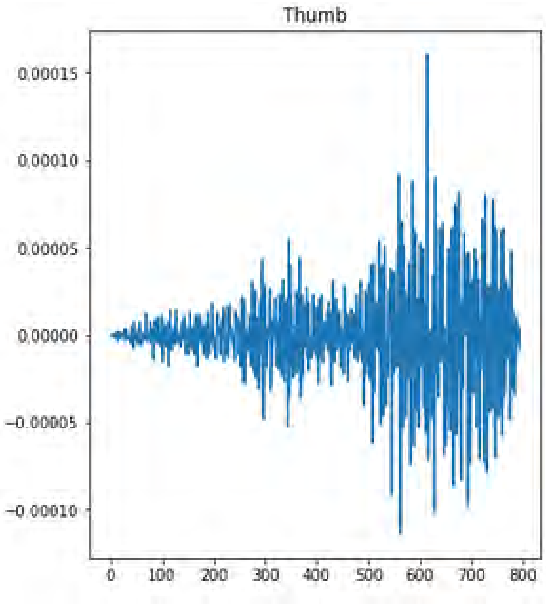


Figure 1.1: An example EMG recording of thumb engagement.

typically inserted into a vector for classification. This strategy is successful in fact, with respectable performance. However, this approach comes with a substantial time overhead. Feature computation is a time intensive task, which can be a bottleneck for our real-time environment. Second, it is often expensive to collect large amount of EMG recordings to sufficiently learn the underlying patterns associated with each finger. Moreover, some fingers can be used more than others that may make them over-represented, which leads to an issue with unbalanced data. Finally, given the nature of time-series recordings, it is often the case that sequences are of different lengths [8]. Specifically, EMG recordings might encompass different time intervals, resulting in time-series input vectors of varying lengths. However, many machine learning models often accept only input of the same length. Therefore, it is important to standardize sequence lengths.

To address these challenges, we propose to directly use raw time-series data instead of computed features for classification to save time in our real-time application. A data augmentation strategy is proposed to tackle the problem of limited and unbalanced training data. This empirical study using time series collected from one human subject demonstrates the effectiveness of the proposed data augmentation strategy.

Chapter 2

Related Work

EMG classification as a machine learning task is a problem that has been investigated extensively [6]. Previous research provides a good understanding of how to select a model for the aforementioned task, as well as how to compute features for classification purposes.

In *Classification of EMG Signals*, Abu et al. attempt to classify five different hand positions: cylindrical grasping, supination, pronation, resting, and open hand [1]. Although this work focuses on different movements, the fundamental idea of classifying specific motions is paralleled. They utilized the feature vector computation strategy by means of features such as absolute mean value, root mean square, median, and waveform length. Subsequently, they applied an artificial neural network for the classification problem. However, feature computation in real-time is not feasible due to its time overhead. We, in this thesis, aim to do EMG classification based on raw EMG signals only to avoid the time overhead of feature computation.

Varying Length Time-series Classification focuses on working with uneven-length time-series data, rather than EMG data specifically. Tan et al. [8] attempt to determine optimal strategies in regards to working with time-series sequences of varying lengths. Two potential solutions were discussed: utilizing distance based machine learning models, and augmenting the data by means of length standardization. Considering that distance based methods are time consuming, we consider length standardization in this paper. More specifically, expanding shorter sequences is often better than shrinking longer sequences due to the potential loss of information. We will employ their strategy of low-amplitude noise padding at the suffix of the time-series.

Chapter 3

Methodology

In this chapter, the strategies used to address those three main challenges (i.e., varying length signal of each movement, limited training data, and unbalanced data) are described.

3.1 Length Augmentation

As mentioned, due to the variate recording time involved in receiving the EMG signal data, the length of each movement time series is variable. As illustrated in Fig. 3.1, the time lengths of movements from one human subject can vary a lot. Given that each sequence of time-steps can be interpreted as a vector of features, each vector in the data-set is often expected to have the same length in machine learning models. Therefore, the lengths of the sequences should be standardized, in which they are all brought to the same length.

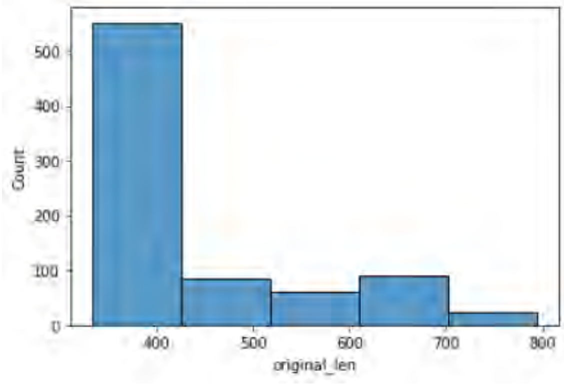


Figure 3.1: The distribution of movement time series lengths of one human subject.

Concretely, we borrow the padding idea in [8], in that some neutral data is appended to sequences. It is important for the chosen padding value to be neutral, so that the padding values will not be used for discrimination between different finger movements. A value of zero is optimal for EMG signal neutrality. However, there can be multiple locations where padding data is inserted. Padding values can be inserted at either the front, the rear of the sequence, or equally split between them. Practically, we found no performance difference between these approaches. Thus, for the sake of simplicity, the padded values are inserted in the front of the sequences in this thesis to enlarge each sequence to be the maximum length. It should be noted that it is possible that some recorded EMG time series contain zeros at the end of the recording attributed to the sensor itself that can be treated as noise and stripped before length standardization. Therefore, the length standardization procedure can be summarized as follows and Fig. 3.2 shows an example:

1. For every sequence, strip the ending zeros and derive the longest sequence length

2. For every sequence:

   (a) Determine the difference d between the current sequence length and the longest sequence length

   (b) Append d zeros to the front of the current sequence

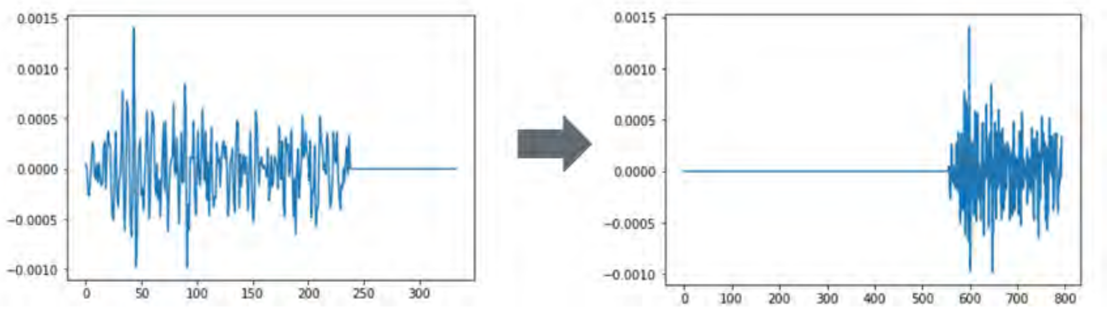


Figure 3.2: A sample EMG sequence (before and after padding).

3.2 Instance Augmentation

In this section, we aim to address the following two concerns.

- The size of training data is very limited due to expensive data collection procedure.

- Some fingers are used more often than others that the availability of training data for different fingers varies.

Both of these have an impact on machine learning performance. A small data-set might not have enough data for optimal machine learning fitting. In order to pick up on underlying patterns necessary for classification, a larger data-set is more beneficial, especially on noisy data such as EMG sequences. Stemming off of that, unbalanced target attributes create a bias in prediction, leaning towards the dominant data. Properly balanced data (approximately equal distribution of target labels) ensure a fair and unbiased model fit. Therefore, we propose to augment the number of training instances by generating *synthetic data*. Synthetic data is data that is not obtained by direct measurement. Such data is *artificially generated* and is not something observed in the real world. Such an approach is useful when dealing with limited data availability or if data is expensive to obtain. This thesis is a prime example of the target use-case of synthetic data.

We utilize a specific approach to generate synthetic data, namely the usage of the *Gaussian distribution*. Given the noisy nature of the data, minor variations can be made on the noise of each sequence while preserving the nature of the pattern over time. Thus, a strategy arises in which data can be duplicated, with duplicates exhibiting minor variations in their noise. These duplicates are associated with the same target label belonging to the original, and would be added to the training data. The strategy involves generating random noise from the Gaussian distribution, centered at a mean of zero and an unknown standard deviation parameter. This random noise would then be added to a given sequence. This new sequence then can be added to the training data and treated as a new data-point. An initial intuition may be to duplicate the data-points, without adding noise variation to each duplicate. However, a machine learning model will not learn from redundant information. The more diverse the provided training data, the greater the classification capabilities after learning from the data. In order to ensure optimal learning, minor variations on original data-points allows for greater training diversity and thus, more optimal learning.

This approach has two parameters that can be tweaked in order to maximize classification performance and run-time: the number of duplicates per original data-point, k, and the standard deviation of the Gaussian, σ. Increasing k subsequently increases the size of the newly augmented data-set. More specifically, the size of the new data-set becomes $N \times (k + 1)$, where N is the original size of the data-set. For example, choosing $k = 1$, or adding an extra data-point for every original data-point, results in a new data-set twice as large. Additionally, σ is also a parameter to consider. The lower the value of σ, the smaller the deviation from the original data-point's sequence. σ should be minimized to the point where there is enough distinction between the original data-points and their duplicates, but not a large enough distinction that it alters the underlying patterns.

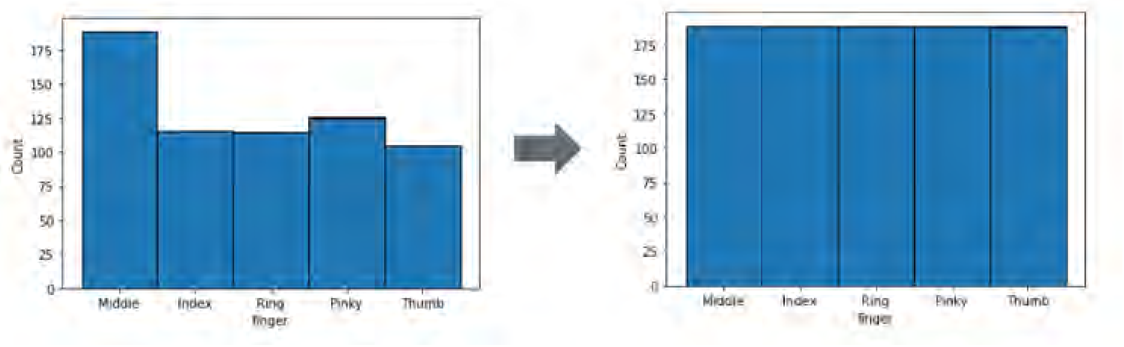


Figure 3.3: The distribution of target labels before and after balancing.

The two areas of concern raised earlier can directly be addressed utilizing synthetic data. We can generate more synthetic data for the minor classes so that each class will have the same number of instances as shown in Fig. 3.3, where the algorithm to achieve this can be summarized as follows. Consequently, we can also randomly sample a number of original sequences of each class to generate more instances for each class similarly to further augment the training data.

1. Determine most frequently occurring target label, with the count N
2. For every other target label
    (a) Determine the difference m between the occurrences of the finger and N
    (b) Randomly sample m sequences corresponding to the current finger, generate synthetic data from it, and insert into the data-set.

In summary, our proposed method can be visualized as the data pipeline illustrated in Fig. 3.4. The pipeline consists of 4 different stages. The first stage is padding the data to equal length, followed by a standard data normalization, in which each sequence will be normalized to the range of $[0, 1]$. After normalization, the data will be fed into a machine learning model forming a *baseline* model. In our proposal, the data will be further balanced and expanded via the data synthesis algorithm described above to augment the training data as the new feed of the machine learning approach.

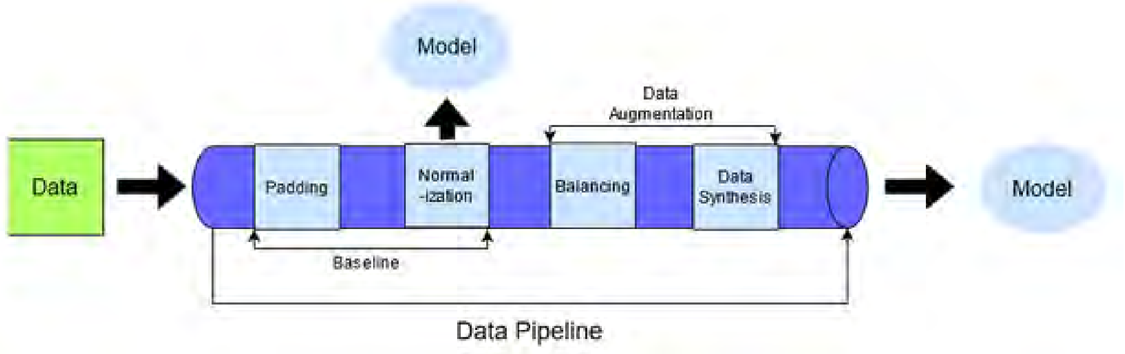


Figure 3.4: A visualization of the different data processing stages in the pipeline.

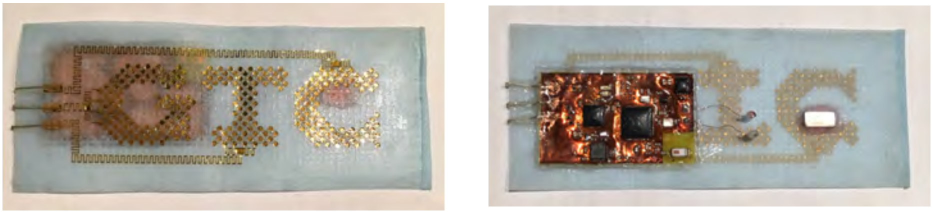
Chapter 4

Experiment

In this chapter, we aim to evaluate the proposed method using EMG signals collected from one human subject.

4.1 Data Description

Surface EMG is used for finger movement classification in this work. We used thin variants of EES proposed in [9]. Fig. 4.1 shows the FS mesh that lays on the skin, as well as the circuit that records the EMG signals and connects to the Android platform via Bluetooth.

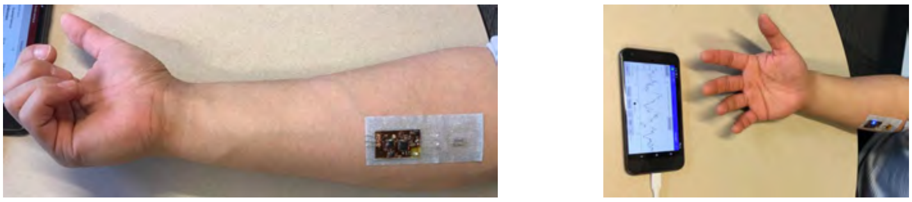


(a) Bottom view of the sensor

(b) Top view of the sensor

Figure 4.1: (a) Three electrodes of the Epidermal sensor, each in the form of an FS mesh with exposed metal (Au) that contacts the skin directly (b) Electrical circuits for Bluetooth connection with the android device and EMG recording.

Our test subject was a 27-year-old female and the sensor was positioned on their right forearm. Fig. 4.2 shows where we attached the sensor to record the EMG signals generated as a result of finger movements of our test subject as well as the real-time interface of the sensor with the Android mobile app. We recorded over 100 samples for each of the five finger movements which gave us 500 samples per experiment. Since our goal is to be able to detect movements with different length of out of each 100 samples, we recorded 50 samples with the length of 1 second closed finger, and 50 samples with the length of half a second closed finger. That means the total duration of the close and open finger are 2 second and 1 second respectively.



(a) Attached sensor

(b) Sensor interface with an Android device

Figure 4.2: (a) Sensor attachment on the forearm of the right hand of the subject (b) Real-time interface of the sensor and the Android mobile device

EMG signals that originate in the muscle are inevitably contaminated by various noise components and artifacts that originate at the skin-electrode interface, in the electronics that amplify the signals, and in external sources [2, 5]. For this reason, it is necessary to filter these signals after the acquisition. Because this noise are often at low frequencies, we can use a high pass filter to remove the noise. Suggested by the sensor provider, we used Butterworth high pass filter with a corner frequency of 20Hz [3].

With the filtered time series containing all movements, we need to extract each movement. Considering that each movement can cause a spike in the signal, we simply use a threshold value to extract each movement. The threshold value is heuristically set by the average value of the whole filtered EMG signal sequence. Thereafter, each movement would contain different length of signals with all absolute values larger than the threshold. In summary, we obtained 812 movements across 3 experiments with the expectation of reading 1500 movements. This demonstrates the difficulty of collecting large amount of EMG signals and the need of data augmentation. Table 4.1 summarizes the total number of movements obtained for each finger, where the imbalance between different classes can be observed.

	Thumb	Index	Middle	Ring	Pinky
#Movements	131	145	236	143	157

Table 4.1: Obtained movement statistics

4.2 Experiment Setup

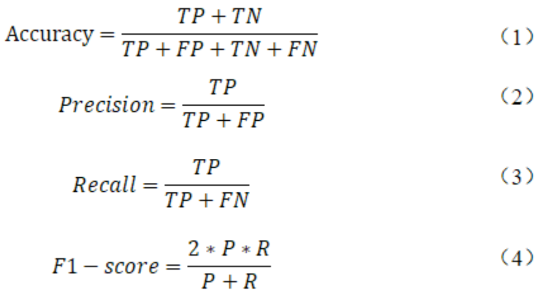
To demonstrate the effectiveness of the proposed data augmentation method, we use 5-fold stratified cross-validation to compare the performance between the baseline model and the mode trained on augmented data. In terms of the machine learning model, we chose the Random Forest algorithm due to its ease of use as well as its powerful classification capabilities. Both the baseline model without data augmentation and the one using data augmentation are trained using the same default parameter setting as shown in Table 4.2.

Hyperparameter	Value
n estimators	100
criterion	gini
max depth	None
min samples split	2
min samples leaf	1
min weight fraction leaf	0.0
max features	auto
max leaf nodes	None
min impurity decrease	0.0
min impurity split	None
bootstrap	True
oob score	False
n jobs	None

Table 4.2: Random Forest Hyperparameters

Specifically, we split the data in Table 4.1 into 5 folds using a stratified sampling strategy that keeps distribution of classes. For the baseline random forest model, we conduct five training trials. In each trial, one of those 5 folds is treated as a test fold, while the remaining 4 folds are used for training. Each training slice subsequently undergoes the data processing pipeline while leaving the testing slice intact for testing. We then report the confusion matrices corresponding to all 5 folds.

A confusion matrix is a visual method of interpreting the performance of a machine learning classification model. Every prediction has an associated cell in the matrix governed by its row and column. A prediction's row is what it is predicted to be, and its column is what it actually is. Thus, a well performing machine learning classifier will have most of its guesses grouped along a one-to-one diagonal, with each prediction corresponding correctly to its actual state. A poorly performing model will have random values throughout the matrix, indicating no correlation between the prediction and reality. Various classification metrics like Accuracy, Precision, Recall, and the F1 score are derived from the number of correct (True) and incorrect (False) predictions in the confusion matrix as seen in Figure 4.3, and are further used to analyze classification performance. As a side note, in Figure 4.3 the values TP, TN, FP, FN refer to true and false classifications in a positive/negative binary, in that positive and negative are the only two possible values. In this case, instead of positive and negative, there would be 5 different true and false values corresponding to each finger.



$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - score = \frac{2 * P * R}{P + R} \quad (4)$$

Figure 4.3: Formulas on Calculating Performance metrics.

The accuracy of the model, otherwise known as classification accuracy, is the ratio of the number of correct predictions to the number of total predictions. Considering imbalanced distribution between different classes, the accuracy itself may hide detailed performance for each finger type that can be misleading. Thus, we have chosen to include the other mentioned metrics as well. Precision can be interpreted as what proportion of identifications are actually correct when predicted to be so. Recall is more complicated, but is understood to be the percentage of predicted results that are correctly classified by the model. Recall and precision go hand-in-hand, and optimizing both is important. The F1-score conveys the balance between the precision and recall. It is the harmonic mean of the two metrics. These 4 performance metrics will give a comprehensive estimate on how well-performing a machine learning model is.

For our proposed model using data augmentation, in each 4 folds of training data, we generate $k = 1$ synthetic data-points for every original training sequence, where the standard deviation parameter of the noise added for augmentation is set to $\sigma = 0.01$. It should be noted that k can be increased to generate more training data. We find that $k = 1$ is sufficient for our application. $\sigma = 0.01$ is found to be sufficient as well. Thereafter, to increase the number of sequences in minor classes like "Thumb" in Table 4.1, we further randomly sample a number of original sequences in that class to generate more synthetic sequences to make all classes have the same number of training data. The test data is not augmented as the baseline model. Therefore, we use the confusion matrix and average prediction metrics to compare between the baseline and the proposal in the following section.

4.3 Performance Comparison

Tables 4.3 and 4.4 provide the prediction confusion matrices of the baseline and the proposal, respectively. Table 4.5 compares the average prediction metrics.

	Finger	Index	Middle	Pinky	Ring	Thumb
Ground truth	Index	18	84	17	14	12
	Middle	27	138	29	23	19
	Pinky	9	97	24	22	5
	Ring	4	74	27	26	12
	Thumb	7	72	21	17	14
	Predictions					

Table 4.3: Baseline Confusion Matrix

	Finger	Index	Middle	Pinky	Ring	Thumb
Ground truth	Index	139	6	0	0	0
	Middle	0	236	0	0	0
	Pinky	1	1	155	0	0
	Ring	1	0	0	142	0
	Thumb	0	3	0	0	128
	Predictions					

Table 4.4: Augmented Confusion Matrix

Methods	Accuracy	Precision	Recall	F1
Baseline	0.271	0.250	0.230	0.217
Data Augmentation	0.982	0.987	0.979	0.983
Improvement	+0.711	+0.737	+0.749	+0.766

Table 4.5: Prediction Performance Comparison

Looking at both the performance metrics and the confusion matrix of the baseline model, an observation can be made that many predictions are biased to the "Middle" finger. It corresponds to an issue raised earlier, that is, the data imbalance problem. Moreover, without sufficient training data, the model is too weak to make good predictions as shown in Table 4.3 and the performance metrics are pretty low as shown in Table 4.5 for the baseline.

By utilizing data augmentation, the performance can be dramatically improved as shown in the confusion matrix of Table 4.4 and 4.5. It is evident that data augmentation using synthetic data for EMG signals can solve both the data imbalance and limited training data problems.

More importantly, the high performance of the proposed method indicates that directly conducting classification using the raw filtered signals are feasible to save the overhead time of feature computation. As a reference, in *Classification of EMG Signals*, Abu et al. yields an accuracy of approximately 80% [1]. In comparison, our data pipeline yields an accuracy of 98.2%. Utilizing our pipeline resulted in performance increases by over 10%, demonstrating its capabilities relative to other research.

Chapter 5

Conclusion and Future Work

5.1 Conclusions

In this thesis, four main challenges were presented: the time overhead of feature computation, different sequences have different lengths, machine learning tasks require a lot of data, and the difficulty of EMG data collection.

In this thesis, we propose to directly use the filtered raw EMG signal in a time-series format as the data representation of each movement, which can save the overhead time of feature extraction in real time. More importantly, we propose data augmentation strategies in order toto address the remaining three challenges. First, length augmentation is used to pad zeros as neutral noise to make sure all movements are of the same length. Second, Gaussian noise without changing the original general data pattern is applied to generate synthetic data, so as to increase the number of training data for both minor classes and the whole training. Thereafter, the issues that come from limited training data and imbalanced data can be both handled.

Our empirical study using an EMG sensor applied on the arm of one human subject shows the above challenges in real-world applications. More importantly, comparing the prediction performance using the random forest algorithm, a competitive classification model, with or without data augmentation, we can observe the dramatic performance improvement in finger movement classification, which demonstrates the effectiveness of the proposed method for EMG signal classification. The performance increase indicates the feasibility of directly using signal sequences in a time series format for classification to save time of feature extraction.

5.2 Future Directions

It can be easily observed that the current empirical study is only conducted on one human subject. Therefore, it is important to demonstrate the proposed method on more human subjects. As a consequence, it is also interesting to investigate if different human subjects share the same signal patterns or differ in signal patterns even doing the same movement. This information will be very useful for authentication in cybersecurity.

After collecting more data from other human subjects or the same human subject, we can further investigate when data augmentation will help or not. Additionally, more complicated machine learning models can be involved. For example, Recurrent Neural Networks (RNNs) are famous for capturing temporal patterns. It will be interesting to study if RNNs can further help in finger movement classification using raw EMG signals in a time-series format.

Bibliography

[1] M Azlan Abu, Syazwani Rosleesham, Mohd Zubir Suboh, Mohd Syazwan Md Yid, Zainudin Kornain, and Nurul Fauzani Jamaluddin. Classification of emg signal for multiple hand gestures based on neural network. *Journal of Electrical Engineering and Computer Science*, 17(1):256–263, 2020.

[2] Yousef Al-Assaf. Surface myoelectric signal analysis: Dynamic approaches for change detection and classification. *IEEE Transactions on biomedical engineering*, 53(11):2248–2256, 2006.

[3] Claudio Castellini and Patrick van der Smagt. Surface emg in advanced hand prosthetics. *Biological cybernetics*, 100(1):35–47, 2009.

[4] Qingqing Li, Penghui Dong, and Jun Zheng. Enhancing the security of pattern unlock with surface emg-based biometrics. *Applied Sciences*, 10(2):541, 2020.

[5] Vijay R Mankar. Emg signal noise removal using neural netwoks. In *Advances in Applied Electromyography*. IntechOpen, 2011.

[6] Mamun Bin Ibne Reaz, M Sazzad Hussain, and Faisal Mohd-Yasin. Techniques of emg signal analysis: detection, processing, classification and applications. *Biological procedures online*, 8(1):11–35, 2006.

[7] S Sudarsan and E Chandra Sekaran. Design and development of emg controlled prosthetics limb. *Procedia engineering*, 38:3547–3551, 2012.

[8] Chang Wei Tan, Francois Petitjean, Eamonn Keogh, and Geoffrey I Webb. Time series classification for varying length series. *arXiv preprint arXiv:1910.04341*, 2019.

[9] Woon-Hong Yeo, Yun-Soung Kim, Jongwoo Lee, Abid Ameen, Luke Shi, Ming Li, Shuodao Wang, Rui Ma, Sung Hun Jin, Zhan Kang, et al. Multifunctional epidermal electronics printed directly onto the skin. *Advanced Materials*, 25(20):2773–2778, 2013.