Deep Learning in EMG-based Gesture Recognition

P. Tsinganos[1], B. Cornelis[2], J. Cornelis[2], B. Jansen[2] and A. Skodras[1]
[1]*University of Patras, Department of Electrical and Computer Engineering, 26504 Patras, Greece*
[2]*Vrije Universiteit Brussel, Department of Electronics and Informatics, 1050 Brussels, Belgium*

Keywords: sEMG, Gesture Recognition, Deep Learning, CNN.

Abstract: In recent years, Deep Learning methods have been successfully applied to a wide range of image and speech recognition problems highly impacting other research fields. As a result, new works in biomedical engineering are directed towards the application of these methods to electromyography-based gesture recognition. In this paper, we present a brief overview of Deep Learning methods for electromyography-based hand gesture recognition along with an analysis of a modified simple model based on Convolutional Neural Networks. The proposed network yields a 3% improvement on the classification accuracy of the basic model, whereas the analysis helps in understanding the limitations of the model and exploring new ways to improve the performance.

1 INTRODUCTION

Over the last decades there has been particular interest in gesture recognition for human-computer interaction (HCI). This particular combination finds many applications, including sign language recognition, robotic equipment control, virtual reality gaming, and prosthetics control (Cheok et al., 2017). Among the various sensor modalities that have been used to capture hand gesture information, electromyography (EMG) is considered more appropriate since it captures the muscle's electrical activity; the physical phenomenon that results in hand gestures. EMG data can be recorded either with invasive or non-invasive methods. Surface electromyography (sEMG) is a technique that measures muscle's action potential from the surface of the skin, contrary to invasive methods that penetrate the skin to reach the muscle.

A popular approach to sEMG-based gesture recognition consists of using pattern recognition methods derived from Machine Learning (ML) (Scheme and Englehart, 2011). Conventional ML pipelines include data acquisition, feature extraction, model definition and inference. Acquisition of sEMG signals involves one or more electrodes attached around the target muscle group. The features used for classification are usually hand-crafted by human experts and capture the temporal and frequency characteristics of the data. Typical features that have been used for sEMG pattern classification are shown in Table 1. These extracted features serve as the input to ML classifiers, such as k-Nearest Neighbors (kNN), Support Vector Machines (SVM), Multi-Layered Perceptron (MLP), Linear Discriminant Analysis (LDA), and Random Forests (RF), where the classifiers parameters are adjusted towards accurate classification.

Deep Learning (DL) is a class of ML algorithms that has revolutionized many fields of data analysis (Goodfellow et al., 2016). For example, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) were successfully deployed for image classification and speech recognition tasks, respectively. DL methods differ from conventional ML approaches in that feature extraction is part of the model definition, therefore obviating the need for hand-crafted features. Although these methods are not new (Goodfellow et al., 2016), they recently gained more attention due to the increased availability of abundant data and vast improvements in computing hardware allowing these computationally demanding methods to be executed in less time.

Motivated by the progress of DL methods we provide an overview of the application of these methods to sEMG pattern classification problems and propose modifications to a simple CNN model (Atzori et al., 2016). The comparison with the state of the art and the analysis of the results sheds light on how the architecture performs and allows for improvements to be made.

The remaining of the paper is organized as follows. In Section 2, we provide an overview of the related gesture recognition approaches. Section 3 gives

Tsinganos, P., Cornelis, B., Cornelis, J., Jansen, B. and Skodras, A.
Deep Learning in EMG-based Gesture Recognition.
DOI: 10.5220/0006960201070114
In *Proceedings of the 5th International Conference on Physiological Computing Systems (PhyCS 2018)*, pages 107-114
ISBN: 978-989-758-329-2

a detailed description of the proposed CNN architecture. The experiments performed for the evaluation of the model are presented in Section 4, while the results and a brief discussion are given in Section 5. Finally, in Section 6 we conclude the paper and outline our future work.

2 RELATED WORK

There exists a great body of literature on the problem of sEMG-based hand gesture recognition. One can discriminate between approaches that use conventional ML techniques and studies based on deep learning methods.

The most significant study on sEMG classification with traditional ML techniques is the work described in (Hudgins et al., 1993). For every 200ms segment of 2 channel sEMG signals, 5 time-domain features are extracted and fed to an MLP classifier, achieving an accuracy of 91.2% on the classification of 4 hand gestures. Later approaches based on this work improve the classification performance by using more features or different classifiers. In (Englehart and Hudgins, 2003), the same set of features is extracted from 4 channel sEMG signals and fed to an LDA classifier. The average accuracy obtained is greater than 90% and is further improved by applying a majority vote window to the predictions of the classifier. The work presented in (Castellini et al., 2009) achieves a 97.14% accuracy on the task of classifying 3 types of grasp motions using the RMS value from 7 electrodes as the input to an SVM classifier. In (Kuzborskij et al., 2012), a set of time- and frequency-domain features is extracted from 8 channel myoelectric signals and evaluated with various classifiers. This experiment is considered the first successful approach for the classification of a large number of hand gestures, since they achieve high accuracy (70-80%) on a set of 52 hand gestures (Ninapro dataset (Atzori et al., 2015)) using any of the proposed features and an SVM classifier with RBF kernel. This work was further improved in (Atzori et al., 2014) by considering linear combination of features and using a RF classifier resulting in an average accuracy of 75.32%. In (Gijsberts et al., 2014), different kernel classifiers were evaluated jointly on EMG and acceleration signals, improving the classification accuracy by 5%.

Considering the advancements of DL methods in the fields of image processing and speech recognition, many works have investigated their application to EMG-based hand gesture recognition. In (Shim and Lee, 2015) and (Shim et al., 2016), the authors propose a Deep Belief Network (DBN) classifier as a more effective model compared to a shallow MLP network trained with back-propagation. Time-domain features are extracted from segments of 2 channel EMG signals which are used to train the model in a layer-by-layer fashion, either with a greedy approach or using genetic algorithms, achieving an accuracy of 88.59% and 89.29% respectively on a set of 5 movements.

The first end-to-end DL architecture, however, was proposed by (Park and Lee, 2016). The authors built a CNN-based model for the classification of six common hand movements resulting in a better classification accuracy compared to SVM. In (Atzori et al., 2016), a simple CNN architecture based on 5 blocks of convolutional and pooling layers is used to classify a large number of gestures. The classification accuracy is comparable to those obtained with classical methods, though not higher than the best performance achieved on the same problem using a RF classifier. The works of (Geng et al., 2016) and (Wei et al., 2017) improve their results across various datasets incorporating dropout (Srivastava et al., 2014) and batch normalization (Sergey and Szegedy, 2015) techniques in their methodology. Apart from choosing different model architectures, other differences to previous works consist of using a high-density electrode array to capture EMG data. Using instantaneous EMG images, (Geng et al., 2016) achieves a 89.3% accuracy on a set of 8 movements, going up to 99.0% when using majority voting over 40ms windows. In (Wei et al., 2017), the observation is made that a small group of muscles play a significant role in some movements. Therefore, a multi-stream CNN architecture is employed, where the input is divided into smaller images that are separately processed by convolutional layers before being merged with fully connected layers. With this model the reported accuracy on the Ninapro dataset is improved by 7.2% (from 77.8% to 85%).

Later works deal with the problem of inter-subject classification, i.e. where the train and test data come from different subjects, either with recalibration ((Zhai et al., 2017)) or model adaptation ((Du et al., 2017), (Côté-Allard et al., 2018)). The performance of the network proposed in (Zhai et al., 2017), which takes as input downsampled spectrograms of EMG segments, is improved by updating the network weights using the predictions of previous sessions corrected by majority voting. In (Du et al., 2017) it is assumed that the weights of each layer of the network contain information that allows for differentiation between gestures, while the mean and variance of the batch normalization layers correspond to discriminating between sessions/subjects.

Table 1: Typical sEMG features.

Feature	Domain	Reference
Root Mean Square	time	(Castellini et al., 2009)
Variance	time	(Kuzborskij et al., 2012)
Mean Absolute Value	time	(Kuzborskij et al., 2012) (Atzori et al., 2014) (Hudgins et al., 1993) (Englehart and Hudgins, 2003)
Zero Crossings	time	(Atzori et al., 2014) (Hudgins et al., 1993) (Englehart and Hudgins, 2003)
Slope Sign Changes	time	(Atzori et al., 2014) (Hudgins et al., 1993) (Englehart and Hudgins, 2003)
Waveform Length	time	(Kuzborskij et al., 2012) (Atzori et al., 2014) (Hudgins et al., 1993) (Englehart and Hudgins, 2003)
Histogram	time	(Kuzborskij et al., 2012) (Atzori et al., 2014) (Hudgins et al., 1993) (Englehart and Hudgins, 2003)
Short Time Fourier Transform	frequency	(Kuzborskij et al., 2012) (Englehart et al., 1999)
Cepstral coefficients	frequency	(Kuzborskij et al., 2012)
Marginal Discrete Wavelet Transform	time-frequency	(Kuzborskij et al., 2012) (Atzori et al., 2014)

Therefore, they apply adaptive batch normalization (AdaBN) (Li et al., 2016), where only the normalization statistics are updated for each subject using a few unlabeled data. The results show improved performance compared to a model without adaptation. The authors of (Côté-Allard et al., 2018) use transfer learning techniques to exploit inter-subject data learned by a pre-trained source network. In their architecture, for each subject a new network is instantiated with weighted connections to the source network. Through this technique, which achieves an accuracy of 98.31% on 7 movements, predictions for a new subject are based both on previously learned information and subject-specific data.

3 PROPOSED MODEL

The problem of sEMG-based hand gesture recognition can be formulated as an image classification problem using CNNs, where the input sEMG image has a size of $H \times W \times 1$ (height×width×depth). Various approaches have been employed to construct an sEMG image. For example, in the works of (Geng et al., 2016), (Wei et al., 2017), and (Du et al., 2017), the instantaneous sEMG signals from a high density electrode array have been used, where the width and the height of the array match the dimensions of the image. In addition, sEMG images can be constructed with segments of sEMG signals using (overlapping) time-windows, in which case the width matches the number of electrodes and the height is equal to the window length (Atzori et al., 2016). Another approach is based on spectrograms using the STFT of sEMG segments, where for each channel of the EMG a spectrogram is created resulting in an image of size frequency×time-bins×channels (Zhai et al., 2017), (Côté-Allard et al., 2018).

In this paper, we adhere to the approach of (Atzori et al., 2016) and generate sEMG images with sliding windows. These images are created using a window length of 150ms and an overlap of 60%, i.e. 90ms, in order to make fair comparisons with previous works in the literature that use similar time-windows. Therefore, the input EMG image has a size of 15×10 (height × width), where the height dimension corresponds to the window length (i.e. 150ms sampled at 100Hz) and the width equals the number of electrodes.

The proposed CNN (depicted in Fig. 1) is based on the architecture proposed in (Atzori et al., 2016) with modifications to increase the models classification accuracy. The main adjustments in the architecture are the introduction of dropout (Srivastava et al., 2014) layers and the use of max pooling instead of average pooling, while the number of trainable parameters remains the same.

The CNN architecture has 4 hidden convolutional layers and 1 output layer. The first two hidden layers consist of 32 filters of size 1×10 and 3×3. The third consists of 64 filters of size 5×5. The fourth layer contains 64 filters of 5×5 size, whereas the last one is a G-way convolutional layer with 1×1 filters, where G is the number of gestures to be classified. Zero padding is applied before the convolutions of the hidden layers, which are followed by rectified linear unit (ReLU) non-linearities and dropout layer with a probability of 0.15 for zeroing the output of a hidden unit. In addition, a subsampling layer performs max

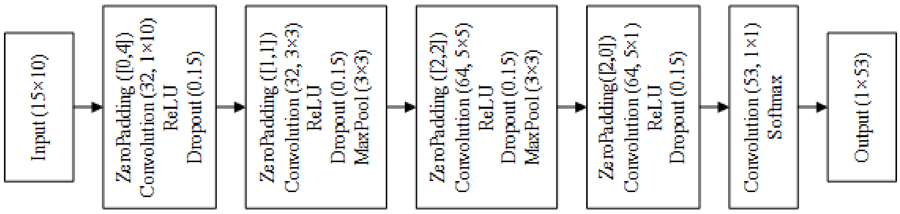

Figure 1: The proposed model architecture is based on the work of (Atzori et al., 2016) with modifications that were found to improve the classification accuracy.

pooling over a 3×3 window after the dropout of the second and third layers. Finally, the last convolutional layer is followed by a softmax activation function.

The weights were initialized with the Xavier initializer (Glorot and Bengio, 2010) and a weight decay (l_2 regularization) of 0.0002 was applied during training. Network parameters were identified via cross-validated random search and manual hyper-parameter tuning on a validation set composed of three subjects randomly selected from the first dataset (DB-1) of the Ninapro database (Atzori et al., 2014). This dataset contains 10 repetitions for each gesture, therefore approximately 2/3 of the repetitions was used as the train set and the remaining repetitions consisted the test set. In each fold of the cross-validation, EMG data from one repetition of the training set were used as test data and the rest repetitions for training. The hyper-parameter search space included weight decay, dropout rate, pooling method, kernel initializer, whereas stride and padding values were computed such that the size of the output tensor is correct. The search space along with the selected values are listed in Table 2. In addition, the proper optimizer parameters were found in the same fashion for each evaluation method.

The EMG signals were preprocessed as follows. Firstly, a 1st order 1 Hz low-pass Butterworth filter was applied as in previous studies on Ninapro database ((Atzori et al., 2016), (Geng et al., 2016)). Then, EMG data were segmented into overlapping windows of 150ms length and 90ms overlap, which can be considered as a form of data augmentation similar to image shifting. Additionally, data were augmented during training by adding Gaussian noise to each image with a signal to noise ratio (SNR) equal to 25dB.

Due to the recording process followed in the Ninapro database, each gesture repetition is followed by a rest phase, meaning that the majority of the images correspond to the 'rest' gesture. In addition, there are variations in the duration of the gesture repetitions, which affects the number of generated images. Therefore, accounting for the fact that gestures are not equally represented in the dataset, two steps are taken

Table 2: Hyperparameter tuning.

Parameter	Search space	Selected value
Weight decay	[0.0001, 0.001]	0.0002
Dropout	[0, 0.333]	0.15
Pool method	'max', 'average'	'max'
Kernel initializer	'glorot', 'he', 'normal', 'uniform'	'glorot'
Optimizer	'SGD', 'Adam'	'SGD'
Learning rate	[0.001, 0.1]	0.05
Learning schedule	'constant', 'step decay', 'exponential decay'	'step decay'
Epochs	[30,150]	100
Batch size	32, 64, 128, 256, 512, 1024	512

to deal with the imbalance problem. First, the EMG data of the 'rest' gesture are subsampled, such that the same number of repetitions is shared between all gestures. Secondly, during training the loss function is weighted such that the network pays more attention to under-represented gestures.

4 EXPERIMENTS

The proposed CNN architecture is evaluated on data from the Ninapro database that includes EMG data related to 53 hand movements of 78 subjects (11 transradial amputees, 67 intact subjects) divided into three datasets. The Ninapro DB-1 includes data acquisitions of 27 intact subjects (7 females, 20 males; 2 left handed, 25 right handed; age 28±3.4 years). The second dataset includes data acquisitions of 40 intact subjects (12 females, 28 males; 6 left handed, 34 right handed; age 29.9±3.9 years). The third dataset includes data acquisitions of 11 transradial amputees (11 males; 1 left handed, 10 right handed; age 42.36±11.96 years). More details about the database and the acquisition procedure can be found in (Atzori et al., 2016), and (Atzori et al., 2014). Table 3 and Table 4 summarize the information about the Ninapro database.

All the evaluations of the model were carried out on the Ninapro DB-1 using all the data available. This dataset is comprised of sEMG signals captured from 27 subjects using 10 electrodes, of which 8 are placed around the forearm and the other two are placed on the main activity spots of the large flexor and extensor muscles of the forearm (Atzori et al., 2014). To allow for a comparison with current literature, the data were split into train and test datasets following the approach described in (Atzori et al., 2016), i.e. repetitions 2,5, and 7 were used for testing and the rest for training. Hyperparameter tuning was performed using cross-validation on the training set. The model was evaluated by means of two experiments. The first one used the evaluation procedure described in (Atzori et al., 2016), while the second used the setting of (Geng et al., 2016). The assessment of the results, reported in Table 5, consists of the average accuracies on the train and test sets, the average of the top-3 test accuracies (i.e. the accuracy when any of the model 3 highest output probabilities match the expected gesture) and the test accuracy after majority voting on each gesture repetition (i.e. the repetition segment of a specific gesture is assigned the majority gesture label of the EMG images that correspond to that repetition). Additionally, the model performance is further evaluated by analyzing misclassifications per class, provided by the confusion matrix, and the accuracy over the gesture duration normalized time as in (Atzori et al., 2015).

In accordance with (Atzori et al., 2016), a model was trained using 7 repetitions and tested with the remaining 3 for each of the 27 subjects in the dataset. Each model is initialized with randomized weights and trained using stochastic gradient descent (SGD) for 100 epochs with 0.05 initial learning rate and a batch size of 512. The learning rate was reduced every 15th epoch by a factor of 50%.

The second experiment follows the setting of (Geng et al., 2016), which differs from the procedure of (Atzori et al., 2016) in that a pre-trained network is created using all the training data of all subjects and then a fine-tuned model is generated for each subject. The first model is initialized with randomized weights and trained using SGD for 100 epochs with 0.05 learning rate, and a batch size of 512. The learning rate was reduced every 15 epochs by a factor of 50%. The subject-specific models were initialized with the pre-trained network and the last two convolutional layers were fine-tuned using SGD optimizer for 30 epochs with a learning rate of 0.01 halved every 10th epoch, and a batch size of 128.

5 RESULTS AND DISCUSSION

For the problem of hand gesture recognition based on EMG, a DL approach is presented in this paper, which utilizes convolutional layers and learning methods that have been successfully applied to other domains. Compared to similar works evaluated on the same dataset, the proposed model outperforms the original network of (Atzori et al., 2016), while it is inferior to the more complex approaches of (Geng et al., 2016) and (Wei et al., 2017). Table 6 shows the comparison between these works under the same evaluation that was used in each paper. The model of (Geng et al., 2016) uses as input the instantaneous EMG images, i.e. 1×10 for the Ninapro DB-1, so the majority vote over 200ms is shown in parentheses, whereas the input image in the network of (Wei et al., 2017) is 20×10 pixels.

Apart from differences in the input, there are more model architecture dissimilarities. Both (Geng et al., 2016) and (Wei et al., 2017) incorporate batch normalization (Sergey and Szegedy, 2015) that allows for faster convergence, and fully connected layers that offer increased network capacity due to more trainable weights. In addition, the approach of (Wei et al., 2017) adopts a multi-stream pipeline where a number of EMG electrodes are processed separately and are then merged with fully connected layers. This split-and-merge approach enables learning the correlation between individual muscles and specific gestures leading to state-of-the-art accuracy of 85% on the Ninapro DB-1. However, we do not follow similar approaches in this paper in order to better understand how DL methods can be applied to sEMG data through a simpler network.

The proposed network is further evaluated through the loss graphs and an error analysis. Fig. 2 shows the loss graphs during training on the train and test sets, with coloring that corresponds to different subjects. It can be seen that decaying the learning rate helps the network parameters converge to a better optimum. When comparing the loss between the train and test sets, it is obvious that there is some degree of overfitting. However, applying more regularization (e.g. dropout, weight decay) does not decrease the test loss. Therefore, a different pipeline (e.g. preprocessing steps, data augmentation, different filter sizes) may reduce the generalization error of the network.

An error analysis was performed to better understand the performance of our model. The confusion matrix is calculated for each subject evaluation and in Fig. 3 the average is shown. Most misclassifications occur around the main diagonal and according to the class labels (Table 4) similar movements are

Table 3: The Ninapro dataset.

Dataset	Subjects	Movements	Electrodes	Sampling (Hz)
Dataset 1 (DB-1)	27	53	10	100
Dataset 2 (DB-2)	40	53	12	2000
Dataset 3 (DB-3)	11	53	12	2000

Table 4: Gestures label/number as in (Atzori et al., 2014).

Label	Gesture
0	Rest
1-12	Individual finger extension/flexion
13-20	Isometric/isotonic configurations
20-29	Wrist movements
30-52	Grasps and functional movements

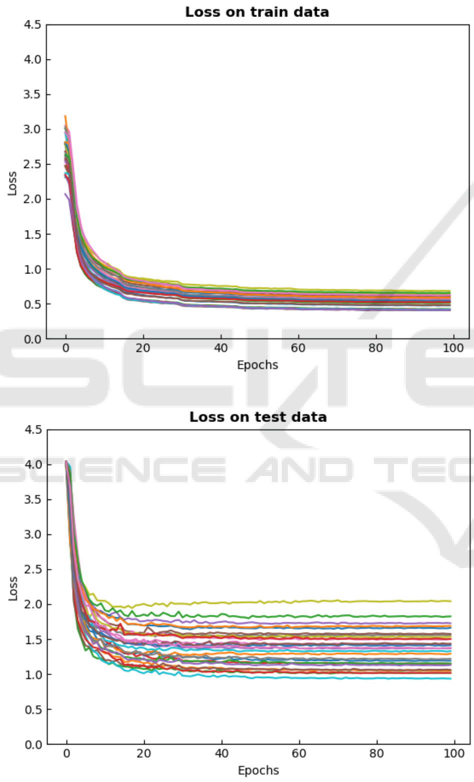

Figure 2: Loss value after each training epoch calculated on train set (up) and test set (down). Colors correspond to different subjects.

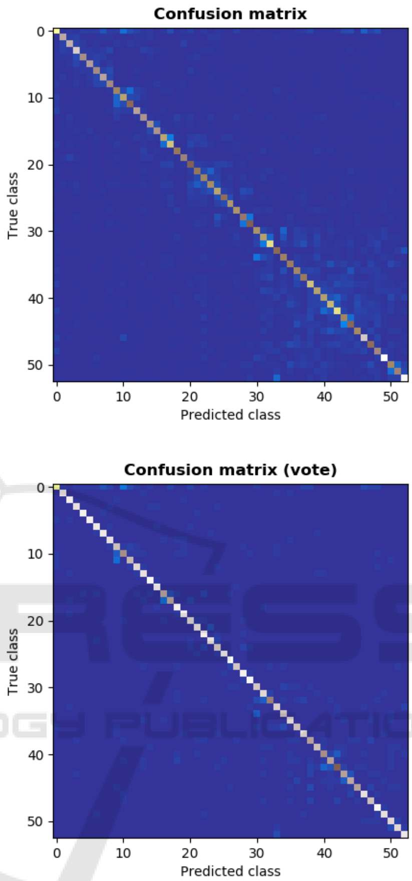

Figure 3: Confusion matrices based on the per image predictions (up) and majority voting predictions (down).

falsely categorized. That is expected considering the location of the EMG electrodes and the muscles that participate in each movement. For example, gesture labels '9', '11' represent the adduction and flexion of the thumb that are coordinated by the same forearm muscles. In addition, there is a concentration of errors in the low-right corner that corresponds to grasps and functional hand gestures that involve more muscles. Taking into account that each EMG image is a 150ms segment and the gesture repetition lasts 5s, we may conclude that for a given misclassification a proportion of the images will be similar between the two gestures. A possible explanation is that some groups of movements can be broken down into the same smaller movements. It is only when the full sequence of images is available that the network can decide which gesture is performed. Comparing the confusion matrices before and after the majority voting we see that most errors around the diagonal are reduced.

Another reason for the low accuracy is the fact that the errors are not evenly distributed on the duration of the entire gesture repetition. Fig. 4, which relates classification errors with the time-normalized movement duration, demonstrates that misclassifications are primarily concentrated in the beginning and at the completion of the movement. The reason for that

Table 5: Experimental results.

Setting	Train accuracy	Test accuracy	Top-3 accuracy	Vote accuracy
(Atzori et al., 2016)	83.03%	70.48%	87.06%	92.31%
(Geng et al., 2016)	81.21%	72.06%	88.06%	93.06%

Table 6: Comparison with other works.

Setting	This work	(Atzori et al., 2016)	(Geng et al., 2016)	(Wei et al., 2017)
(Atzori et al., 2016)	70.48%	66.59%	-	-
(Geng et al., 2016)	72.06%	-	76.10% (77.80%)	85%

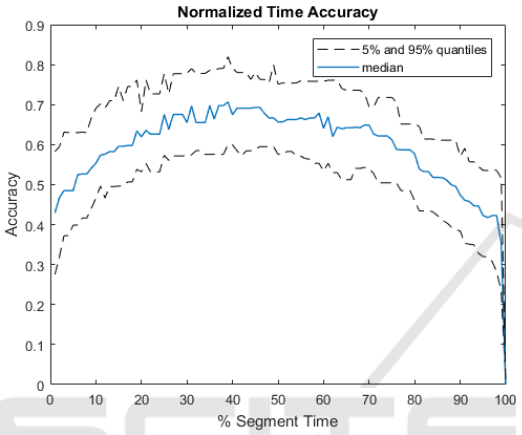

Figure 4: Plot of prediction accuracy against normalized time duration. It can be seen that at the start and completion of the gesture repetition the accuracy is lower.

is that during the recording session there is a gradual transition between rest, gesture and rest, in contrast to the discrete changes of the gesture labels. Consequently, accuracy is lower during these transition periods where the change in movement is not yet clearly evident from the input EMG signal (Atzori et al., 2015).

Overall, it is shown that a simple CNN architecture can be successful at the task of sEMG hand gesture recognition taking into account the chance level when classifying 53 gestures. Small modifications to the model parameters and the training process can boost the performance, whereas deeper and more complex networks yield the best performance. The inability of the proposed model to generalize well to unseen data needs to be addressed to facilitate further improvement. Finally, the use of small EMG segments accounts for much of the classification error assuming that a great amount of overlap happens between the EMG signals of gesture groups especially during their transitive periods. Therefore, majority voting over these small EMG segments provides a better evaluation metric.

6 CONCLUSIONS

This paper presented an overview of recent advances in the use of DL methods for EMG hand gesture classification, while improvements to existing architectures were discussed. The proposed model follows the work of (Atzori et al., 2016) and is compared to the state of the art. It improves on the basic model by 3%, yet the works of (Geng et al., 2016) and (Wei et al., 2017) outperform it under the same evaluation settings. As future work, we plan to investigate the utilization of time-frequency representations (e.g. Wavelet and Fourier transforms) as a preprocessing step, as well as more complex architectures based on RNNs to benefit from the temporal information in the data.

The implementation code is available at the following link https://github.com/DSIP-UPatras/PhyCS2018_paper.

ACKNOWLEDGEMENTS

This work was supported by the VUB-UPatras International Joint Research Group (IJRG) on ICT.

REFERENCES

Atzori, M., Cognolato, M., and Müller, H. (2016). Deep Learning with Convolutional Neural Networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Frontiers in Neurorobotics*, 10.

Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A., Elsig, S., Giatsidis, G., Bassetto, F., and Müller, H. (2014). Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific Data*, 1(140053).

Atzori, M., Gijsberts, A., Kuzborskij, I., Elsig, S., Hager, A., Deriaz, O., Castellini, C., Müller, H., and Caputo, B. (2015). Characterization of a benchmark database for myoelectric movement classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):73–83.

Castellini, C., Fiorilla, A., and Sandini, G. (2009). Multi-subject/daily-life activity EMG-based control of mechanical hands. *Journal of neuroengineering and rehabilitation*, 6:41.

Cheok, M., Omar, Z., and Jaward, M. (2017). A review of hand gesture and sign language recognition techniques. *International Journal of Machine Learning and Cybernetics*.

Côté-Allard, U. et al. (2018). Deep Learning for electromyographic hand gesture signal classification by leveraging transfer learning. *ArXiv e-prints*.

Du, Y., Jin, W., Wei, W., Hu, Y., and Geng, W. (2017). Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors*, 17(3):458.

Englehart, K. and Hudgins, B. (2003). A robust, real-time control scheme for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 50(7):848–854.

Englehart, K., Hudgins, B., Parker, P., and Stevenson, M. (1999). Classification of the myoelectric signal using time-frequency based representations. *Medical Engineering and Physics*, 21(6):431–438.

Geng, W., Du, Y., Jin, W., Wei, W., Hu, Y., and Li, J. (2016). Gesture recognition by instantaneous surface EMG images. *Scientific Reports*, 6(36571).

Gijsberts, A., Atzori, M., Castellini, C., Müller, H., and Caputo, B. (2014). Movement error rate for evaluation of Machine Learning methods for sEMG-based hand movement classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(4):735–744.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward Neural Networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Sardinia, Italy.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA.

Hudgins, B., Parker, P., and Scott, R. (1993). A new strategy for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 40(1):82–94.

Kuzborskij, I., Gijsberts, A., and Caputo, B. (2012). On the challenge of classifying 52 hand movements from surface electromyography. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 4931–4937. IEEE.

Li, Y., Wang, N., Shi, J., Liu, J., and Hou, X. (2016). Revisiting Batch Normalization for practical domain adaptation. *ArXiv e-prints*.

Park, K. and Lee, S. (2016). Movement intention decoding based on Deep Learning for multiuser myoelectric interfaces. In *2016 4th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–2. IEEE.

Scheme, E. and Englehart, K. (2011). Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use. *The Journal of Rehabilitation Research and Development*, 48(6):643–659.

Sergey, I. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network training by reducing internal covariate shift. *ArXiv e-prints*.

Shim, H., An, H., Lee, S., Lee, E., Min, H., and Lee, S. (2016). EMG pattern classification by split and merge deep belief network. *Symmetry*, 8(12):148.

Shim, H. and Lee, S. (2015). Multi-channel electromyography pattern classification using deep belief networks for enhanced user experience. *Journal of Central South University*, 22(5):1801–1808.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent Neural Networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Wei, W., Wong, Y., Du, Y., Hu, Y., Kankanhalli, M., and Geng, W. (2017). A multi-stream Convolutional Neural Network for sEMG-based gesture recognition in muscle-computer interface. *Pattern Recognition Letters*.

Zhai, X., Jelfs, B., Chan, R., and Tin, C. (2017). Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on Convolutional Neural Network. *Frontiers in Neuroscience*, 11:379–390.