Temporal Variability Analysis in sEMG Hand Grasp Recognition using Temporal Convolutional Networks

Temporal Variability Analysis in sEMG Hand Grasp Recognition using Temporal Convolutional Networks

Marcello Zanghieri*, Simone Benatti*, Francesco Conti*†, Alessio Burrello*, Luca Benini*†
* Department of Electronic and Electric Engineering, University of Bologna, Italy
† Integrated Systems Laboratory, ETH Zurich, Switzerland
Email: marcello.zanghieri2@unibo.it, simone.benatti@unibo.it, f.conti@unibo.it,
alessio.burrello@unibo.it, lbenini@iis.ee.ethz.ch

***Abstract*—Hand movement recognition via surface electromyographic (sEMG) signal is a promising approach for the advance in Human-Computer Interaction. However, this field has to deal with two main issues: (1) the long-term reliability of sEMG-based control is limited by the variability affecting the sEMG signal (especially, variability over time); (2) the classification algorithms need to be suitable for implementation on embedded devices, which have strict constraints in terms of power budget and computational resources. Current solutions present a performance over-time drop that makes them unsuitable for reliable gesture controller design. In this paper, we address temporal variability of sEMG-based grasp recognition, proposing a new approach based on Temporal Convolutional Networks, a class of deep learning algorithms particularly suited for time series analysis and temporal pattern recognition. Our approach improves by 7.6% the best results achieved in the literature on the NinaPro DB6, a reference dataset for temporal variability analysis of sEMG. Moreover, when targeting the much more challenging inter-session accuracy objective, our method achieves an accuracy drop of just 4.8% between intra- and inter-session validation. This proves the suitability of our setup for a robust, reliable long-term implementation. Furthermore, we distill the network using deep network quantization and pruning techniques, demonstrating that our approach can use down to 120× lower memory footprint than the initial network and 4× lower memory footprint than a baseline Support Vector Machine, with an inter-session accuracy degradation of only 2.5%, proving that the solution is suitable for embedded resource-constrained implementations.**

I. Introduction

Decoding muscular activation analyzing surface electromyographic (sEMG) signal is a widely accepted method for advanced Human-Machine Interfaces (HMI) design. [1]. This method enables several applications, such as robot interaction, industrial robot control, game and mobile interfaces, and control of poliarticulated prostheses [2], [3].

Several approaches are gaining traction to decode hand gestures by analyzing sEMG activation patterns, ranging from blind source separation of the motoneuron activation [3] to machine learning-based approaches [4]. Such solutions surpass 80% accuracy on classifying several hand gestures (from 4 to 12), making them suitable for the design of a human-machine interface. However, the sEMG signal is affected by high variability, originating from perspiration, changes in the skin-to-electrode interface, user's adaptation or fatigue, and especially from electrode shifts in multi-day usage [5], [6], [7]. These variability sources are a major limit for the long-term use and reliability of EMG-based gesture recognition. For instance, in [6], an inter-session (morning-to-afternoon) accuracy loss higher than 25% was observed, while in [8] it was shown a 30% accuracy drop with a single training session and multiple classification sessions.

To tackle these issues, the proposed solutions mostly rely on training dataset augmentation, as shown in [5], where one subject performs 10 discrete gestures on 121 sessions over 21 days, and the training dataset collected on the latest 5 sessions reduces the average error rate from 27% down to 13%. Such a solution represents a step towards a robust gesture control interface, but the performance drop and the lack of generalization are still hampering the deployment of these solutions in reliable, commercially available systems. Moreover, increasing training dataset is labor-intensive and requires active user engagement, which could be an obstacle for widespread adoption.

In recent years, Deep Learning (DL) techniques have been successfully used in biosignal application scenarios, also providing a promising perspective for hand gesture recognition, since some works reach an accuracy comparable with the SoA approaches [9], [10]. One of the significant advantages of DL approaches is that they need no handrafted feature extraction, since deep models learn the best features intrinsically at training time. This is especially convenient for signals characterized by a very high variability like the sEMG, resulting in an improved generalization after more extensive training, which is particularly useful in multi-day performance analysis.

In this work, we address the challenge of the inter-session generalization of sEMG-based grasp recognition using the Temporal Convolutional Network (TCN), a DL method which is gaining widespread traction in classification of temporal series data. Exploiting the very challenging NinaPro Database 6 [6], specifically utilized to investigate the algorithm adaptation over multiple sessions, we tested several multi-session training set compositions to minimize the accuracy loss over non-training data. Moreover, we evaluate our TCN-based approach against other well-established ML methods to explore the performance drop over multi-day testing sessions. Finally, we evaluated the network quantization to allow the network deployment on platforms with a reduced memory budget (i.e., $< 512\,\mathrm{kB}$). Our work brings a twofold contribution:

- We demonstrate that our approach can yield an inter-session classification accuracy of 49.4% on the NinaPro Database 6, improving by 7.6% the results achieved in literature, and outperforming by 4.4% the results yielded by a RBF-SVM based on RMS feature, a widely used baseline for gesture recognition. Our solution reaches 4.8% accuracy drop on unseen data after 5 training sessions, demonstrating the feasibility of a reliable and robust controller.
- We verify that the implemented network can be distilled

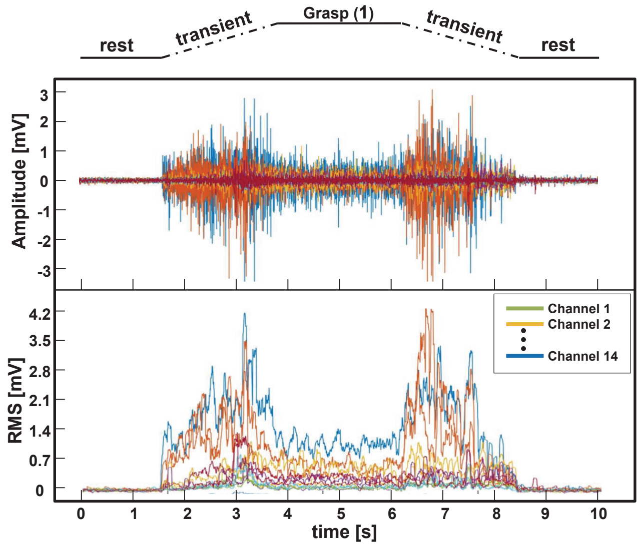

Fig. 1: sEMG signal of the first grasp of NinaPro DB6, preceded and followed by the rest position. The beginning and the end of the grasp exhibit the strong transients typical of the dataset.

using data quantization and pruning, which allow deployment on resource-constrained real-time platforms, a fundamental requirement in next-generation design for wearable HMI. We show that a model which can fit a $512\,\mathrm{kB}$ memory, yet achieves 1.9% higher inter-session accuracy than the SVM, exhibiting $4\times$ lower memory footprint than the SVM and $120\times$ lower memory footprint compared to the complete network.

II. Materials and Methods

A. EMG signal

The electromyographic (EMG) signal [11], [12], [13] is the biopotential that arises from the current generated by the flow of ions through the membrane of the muscular fibers. Hence, it is a significant indicator of muscular activity. The origin of this potential is the electrical stimulus that starts from the central nervous system and passes through the motoneurons of the muscular tissue, giving rise to the Action Potentials (APs). The typical amplitude and bandwidth of the EMG signal are $10\,\mu\mathrm{V} \div 10\,\mathrm{mV}$ and $\sim 2\,\mathrm{kHz}$, respectively. Furthermore, the EMG is affected by several noise sources, including motion artifacts, floating ground noise, crosstalk, and power line interference [14]. These make the EMG a very challenging signal.

EMG signal can be collected either via invasive or non-invasive techniques. In this work, we focus on surface electromyography (sEMG), a non-invasive method based on surface electrodes placed on the skin surface. APs are detected via an instrumentation amplifier with the positive and negative terminals connected to two metal plates positioned on the surface of the skin: the sEMG signal is the superposition of all the APs detected by the amplifier [7]. For Human-Machine Interfaces, basing gesture recognition on sEMG analysis is among the most promising approaches, because non-invasiveness is a main requirement in several HMI scenarios.

B. NinaPro Database 6

The Non-Invasive Adaptive hand Prosthetics Database 6 (NinaPro DB6) [6] is a public sEMG dataset exploring the robustness of sEMG-based hand gesture recognition over time. The dataset comprises 10 able-bodied subjects (3 females, 7 males, average age of 27 ± 6 years). For each subject, data were recorded in 10 sessions (5 days, twice a day: morning and afternoon), each entailing 12 repetitions of 7 grasps. The grasps were chosen from the rehabilitation and robotics fields, selecting movements typical of the Activities of Daily Living. Each repetition lasts approximately $6\,\mathrm{s}$, followed by $2\,\mathrm{s}$ of rest. The sEMG signals were acquired with 14 Delsys Trigno sEMG Wireless electrodes on the higher half of the forearm, sampling at $2\,\mathrm{kHz}$.

Figure 1 shows the sEMG trace of a grasping gesture, indicating the *rest*, *transient*, and *steady grasp* stages. It is noteworthy that the gesture execution exhibits, before and after the steady contraction, strong transient stages, characterized by signal amplitudes and RMS up to $5\times$ greater than the steady signal.

Remarkably, the NinaPro DB6 has been under-exploited in literature [6], [15], as it is made very challenging by the similar nature of the grasps, and by the aforementioned strong transient states caused by the impulsive movements. Results obtained by Palermo et al. [6], applying Random Forest (RF) on Waveform Length (WL), showed high inter-session (morning-to-afternoon) accuracy loss: training on morning data (for each day, for each subject) yields a 52.4% average accuracy on morning data, but a 25.4% average accuracy when validating on afternoon data. These results were improved by Cene et al. [15], who used Extreme Learning Machines (ELM), achieving 69.8% accuracy on the morning sessions used for training, and 41.8% accuracy in the afternoon. This represents, to the best of our knowledge, the State of the Art (SoA) on the NinaPro DB6. Its main limitation is that the temporal variability is not properly addressed since the setup is limited to single-session training. In this work, we improve the SoA with a new setup, including multi-session training.

C. Temporal Convolutional Networks

In this work, we approach the sEMG signal as a time series, employing a small Temporal Convolutional Network to classify fixed-size sEMG time windows. This is significantly different from previous works in sEMG-based gesture recognition, which are framed as single-sample classification [7] or image classification [9].

Temporal Convolutional Networks (TCNs) are sequential models capable to learn time dependencies in the input signal. TCNs represent the SoA in many tasks of sequence modeling, surpassing Recurrent Neural Networks (RNNs) [16], which are more complicated and expensive to train. The idea behind TCNs resides in the 1d-convolutional layers (Conv. layers) operating along time. The two main features of this network topology are: *i) causality*: each output y_n only depends on convolutions on earlier inputs x_i with $i < n$; and *ii) dilation*: a fixed step d is inserted between the filter inputs. Hence, a TCN's Conv. layer has the form of

$$\mathbf{y}_n^{(o)} = \mathrm{Conv}\left(\mathbf{x}\right) = \sum_{l=0}^{L-1} \sum_{i=0}^{K-1} \mathbf{W}_i^{(o,l)} \cdot \mathbf{x}_{n-d\,i}^{(l)}$$

$$\mathbf{y}_n^{(o)} = \mathrm{Conv}\left(\mathbf{x}\right) = \sum_{l=0}^{L-1} \sum_{i=0}^{K-1} \mathbf{W}_i^{(o,l)} \cdot \mathbf{x}_{n-\boldsymbol{d}\cdot i}^{(l)}$$

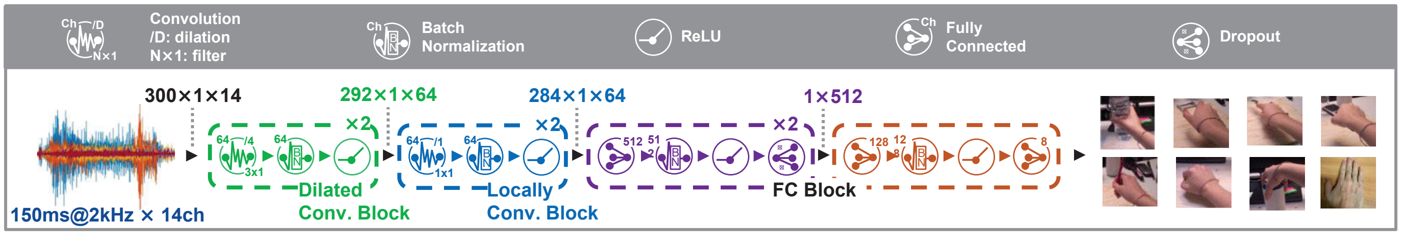

Fig. 2: Temporal Convolutional Network architecture. Three main blocks are used: the *Dilated Conv. Block* for temporal feature extraction, the *Locally Conv. Block* for local feature extraction, and the *FC Block* for the final classification.

with $\mathbf{x}$ input feature map and $\mathbf{y}$ output feature map, $\mathbf{W}$ the filter weights, n the time index, L the number of input channels, o the output channel, K the filter size, and d the dilation. For classification tasks, a sequence of time-distributed fully connected (FC) layers on top of the Conv. layers compute the final label.

Figure 2 displays the architecture of the proposed TCN:
1) the *Dilated Conv. Block* is composed of 2 Conv. layers with $3 \times 1 \times 64$ filters, and dilation 4, to exploit the temporal sequence extracting information on the time-relationship of the samples;
2) the *Locally Conv. Block* is composed of 2 Conv. layers with $1 \times 1 \times 64$ filters, to capture the cross-information of the different channels;
3) finally, the *FC Block* is made of 3 FC layers, with dropout ($p_{\text{dropout}} = 0.5$) to help regularization [17], which flatten the input information to assign a label to the input sequence.
All layers have ReLU non-linearity as activation function, and are equipped with Batch-Normalization to counter the internal covariate shift [18].

III. Experimental Results

In this work, we address temporal variability and generalization to new sessions by developing a multi-session setup, which we use to compare a set of conventional ML algorithms, namely Linear Discriminant Analysis (LDA), Random Forest (RF) and Radial Basis Function-kernel Support Vector Machine (RBF-SVM), against DL models, namely the proposed TCN and its distilled versions.

To analyze the ability to generalize to never-seen sessions, we adopt an incremental training protocol, where 1 to a maximum of 5 sessions are used for training, and the remaining 5 for testing. In this sequential scenario, training sessions always precede testing sessions, preserving temporal coherence.

We employ an internal 2-fold cross-validation, stratified (i.e., each fold contains an equal number of grasp repetitions from each training session). The 2-fold cross-validation is necessary to evaluate our algorithm also on the same sessions used for training (i.e., without the temporal variability). Hence, we use alternately one fold of the training sessions to train the model and the other to test it.

We evaluate our method on two criteria: (1) the *intra-session* validation accuracy, calculated as the average accuracy on the fold not used for training (alternately), and (2) the *inter-session* validation accuracy, computed as the average accuracy on sessions 6-to-10, never included in the training.

A. Classical Machine Learning

First, we apply our multi-session training setup on three well-established ML approaches: LDA, RF and RBF-SVM, implemented with Python 3.5 and Scikit-learn (version 0.20.0) [19] using the modules `sklearn.discriminant_analysis`, `sklearn.ensemble` and `sklearn.svm`, respectively. All are fed the Root Mean Square (RMS) of the 14-channel sEMG signal, computed on $60\,\text{ms}$ time windows. This approach was chosen as a baseline as it is a widely used setup.

For all the three classifiers, adding training sessions gradually improves the recognition of the last 5 never-seen sessions, with the 5-session training yielding the best accuracy, as reported in Table I. This behavior is due to the regularizing effect of multi-session training: showing the classifier more heterogeneous data improves generalization.

In particular, the RBF-SVM (trained with `C=1` and `gamma='scale'`) yields the best results, shown in the left panel of Figure 3. With 5-session training, the SVM achieves an average intra-session validation accuracy of $(51.4 \pm 0.6)\%$ and an average inter-session validation accuracy of $(45.0 \pm 0.8)\%$, thus with a 6.4% accuracy drop on never-seen sessions. Sweeping from 1-session to 5-session training increases the intra-session validation accuracy by 1.1% and the inter-session validation accuracy by 8.2%, and reduces the inter-session accuracy drop by 7.1%. This inter-session validation accuracy outperforms by 3.2% the SoA represented by [15] mentioned in Section II.

B. Temporal Convolutional Network

We implement our TCN-based approach using Python 3.5 and the PyTorch 1.1 framework [20], specialized for DL development. The TCN is fed with $150\,\text{ms}$ sliding ($15\,\text{ms}$) time windows of the raw 14-channel sEMG signal, in sharp contrast with the classical ML methods. The TCN is trained with cross-entropy as loss function, and stochastic gradient descent for 20 epochs (minibatch size 64) with initial learning rate 0.001, divided by 10 after epochs 9 and 19. L_2-regularization is applied with PyTorch `weight_decay=1e-4`, corresponding to $\lambda = 5 \cdot 10^{-5}$.

Even though the training is done off-line, our setup fully allows for online inference, since our TCN neither relies on feature extraction nor on the removal of sEMG transients (as instead popular in both inter-subject [21] and inter-session [5], [22], [23] studies, including the SoA on NinaPro DB6 [15].

The recognition accuracies obtained adding training sessions are shown in the right panel of Figure 3, which provides a detailed comparison between the RBF-SVM and the TCN.

Multi-session training gradually increases the performance on the last 5 never-seen sessions, minimizing the inter-session accuracy drop, with the best results for the 5-session training strategy. This means that including more sessions enables the TCN to leverage more heterogeneous samples to learn a more robust representation. This behavior is the same as for the classical ML algorithms but with higher accuracy. Passing

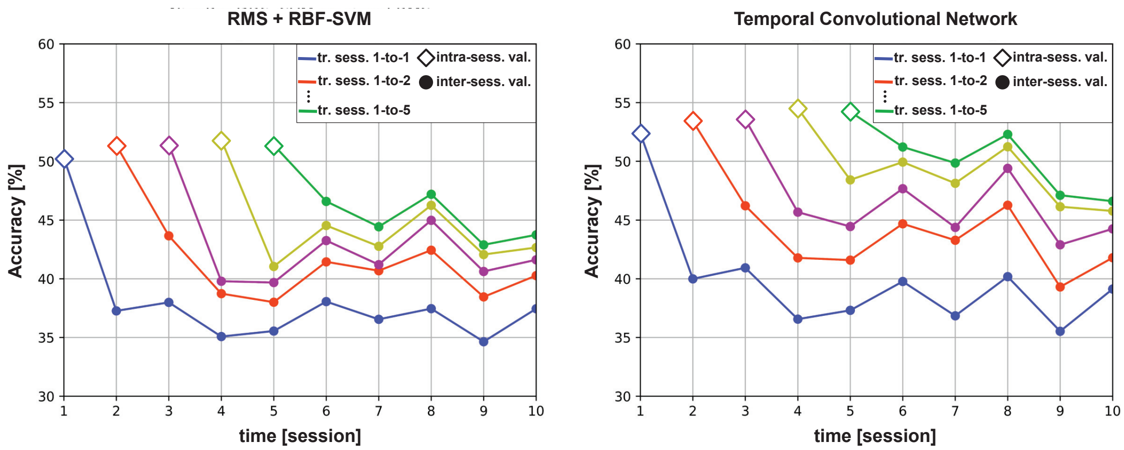

Fig. 3: Classification accuracy of the baseline SVM and our proposed TCN on the different sessions of NinaPro DB6. The 5 multi-session training strategies incrementally improve the generalization to never-seen sessions, with better results for the TCN.

from 1-session to 5-session training improves the intra-session validation accuracy by 1.8% and the inter-session validation accuracy by 11.1%, and reduces the inter-session accuracy drop by 9.3%. The 5-session training achieves an average intra-session validation accuracy of $(54.2 \pm 0.6)\%$ (2.8% higher than the SVM), and an average inter-session validation accuracy of $(49.4 \pm 0.9)\%$ (4.4% higher than the SVM). The inter-session validation accuracy is better-than-SoA by 7.6% [15].

Our lower-than-SoA intra-session accuracy is due to the fact that [15] (1) uses single-session training with feature extraction, prone to overfitting to the single sessions; (2) applies very aggressive signal filtering over 200 ms time windows, thus significantly smoothing the transients discussed in Section II; and (3) embeds outlier removal into the algorithm, thus excluding the transients from the accuracy. For a fair comparison, we evaluated our RBF-SVM and TCN discarding the transients, obtaining results comparable to the SoA. Validated on the steady segments, the RBF-SVM yielded 69.2% intra-session accuracy (just 0.6% below SoA) and 60.4% inter-session accuracy, while the TCN achieved 71.3% intra-session accuracy (1.5% above SoA) and 65.0% inter-session accuracy.

The improvement we achieve in inter-session accuracy is due to the successful regularization provided by multi-session training, as for the classical ML algorithms. Moreover, the higher results compared to classical ML corroborate the initial assumption that TCNs are able to achieve superior generalization thanks to their higher ability to process raw data and to handle the variability of the sEMG signal between sessions.

C. TCN distillation

On top of evaluating the accuracy performance of our TCN-based approach, we discuss the memory requirements of the proposed solution to understand how it can be deployed on a resource-constrained platform. We distill three new networks from the initial one, applying (1) a stride factor s in the first two convolutional layers ($s_1 = 2$ in the first and $s_2 = 4$ in the second), (2) 16-bit quantization for the Conv. layers and 8-bit quantization for the FC layers, and (3) the pruning of the network weights.

Table I reports the memory occupancy and the accuracy of the different configurations of the network, compared to the baseline SVM. Remarkably, introducing the strides, quantizing, and pruning, we only lose 4.7% intra-session and 2.5%

TABLE I: Memory footprint and best intra-/inter-session accuracy of the proposed methods compared to the SVM baseline.

	Intra-/inter-session accuracy	Memory footprint
State of the Art [15]		
ELM	69.8%[1] / 41.8%[1]	n.a.
Classical ML		
LDA	47.5% / 38.3%	negligible
RF	46.3% / 43.4%	negligible
RBF-SVM	51.4% / 45.0%	1.3 MB
RBF-SVM on steady	69.2% / 60.4%	1.3 MB
Proposed DL methods		
Full TCN	54.2% / 49.4%	38.8 MB (30×)
Full TCN on steady	71.3% / 65.0%	38.8 MB (30×)
Full TCN q. and prun.	53.0% / 49.3%	2.0 MB (1.5×)
Strided TCN	52.0% / 48.7%	6.0 MB (4.6×)
Strided TCN q. and prun.	49.5% / 46.9%	0.33 MB (0.25×)

[1]With transient removal by outlier rejection (see Section III-B).

inter-session accuracy. This lowest-area configuration requires $120\times$ less memory than the initial configuration, and $4\times$ lower memory footprint compared to the SVM, yet demonstrating an inter-session accuracy higher than the SVM. In addition, neural networks can benefit from a far better parallelization compared to SVMs, leading to lower delay in the recognition.

IV. Conclusion

In this work, we addressed the temporal variability affecting the inter-session generalization of sEMG-based grasp recognition. We propose a new approach that uses a Temporal Convolutional Networks (TCN), a novel deep learning algorithm for time-series analysis. Our approach, validated on the NinaPro Database 6, proves that the best training set composition is the one including the highest number of sessions, producing an inter-session classification accuracy of 49.4.%, surpassing by 4.4% the results from a reference SVM, and by 7.6% the results achieved in literature. The accuracy drop of just 4.8% between intra- and inter-session validation allows the design of a robust long-term sEMG controller. Moreover, we distilled the TCN using deep network quantization and pruning techniques, showing that our approach reaches as little as $120\times$ lower memory footprint than the starting full network, with an inter-session accuracy decrease of only 2.5%, which makes it very suitable for implementation on resource-constrained platforms.

Future work will explore other network topologies, as well as the reliability of the approach on a more extended dataset and the aggressive optimization for low-power embedded computing platforms.

V. Acknowledgments

This work has been partially supported by the European H2020 FET project OPRECOMP under Grant 732631, and by project EC Horizon-2020 ALOHA under Grant 780788.

References

[1] R. Meattini, S. Benatti, U. Scarcia, D. De Gregorio, L. Benini, and C. Melchiorri, "An sEMG-Based Human-Robot Interface for Robotic Hands Using Machine Learning and Synergies," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 8, no. 7, pp. 1149–1158, 2018.

[2] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, pp. 131–153, 2019.

[3] D. Farina and A. Holobar, "Characterization of human motor units from surface emg decomposition," *Proceedings of the IEEE*, vol. 104, no. 2, pp. 353–373, 2016.

[4] J. Yousefi and A. Hamilton-Wright, "Characterizing emg data using machine-learning tools," *Computers in biology and medicine*, vol. 51, pp. 1–13, 2014.

[5] P. Kaufmann, K. Englehart, and M. Platzner, "Fluctuating emg signals: Investigating long-term effects of pattern matching algorithms," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE, 2010, pp. 6357–6360.

[6] F. Palermo, M. Cognolato, A. Gijsberts, H. Muller, B. Caputo, and M. Atzori, "Repeatability of grasp recognition for robotic hand prosthesis control based on sEMG data," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, jul 2017, pp. 1154–1159. [Online]. Available: https://ieeexplore.ieee.org/document/8009405/

[7] B. Milosevic, E. Farella, and S. Benatti, "Exploring Arm Posture and Temporal Variability in Myoelectric Hand Gesture Recognition," *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, vol. 2018-August, pp. 1032–1037, 2018.

[8] S. Benatti, E. Farella, E. Gruppioni, and L. Benini, "Analysis of robust implementation of an EMG pattern recognition based control," in *International Conference on Bio-inspired Systems and Signal Processing 2014 (BIOSIGNALS 2014)*. ScitePress, 2014, pp. 45–54.

[9] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng, "A novel attention-based hybrid cnn-rnn architecture for semg-based gesture recognition," *PloS one*, vol. 13, no. 10, p. e0206049, 2018.

[10] M. Zanghieri, S. Benatti, A. Burrello, V. Kartsch, F. Conti, and L. Benini, "Robust real-time embedded emg recognition framework using temporal convolutional networks on a multicore iot processor," *IEEE Transactions on Biomedical Circuits and Systems*, pp. 1–1, 2019.

[11] L. G. Tassinary, J. T. Cacioppo, and E. J. Vanman, "The Skeletomotor System : Surface," 1985, no. January 1990.

[12] C. J. De Luca, "The Use of Surface Electromyography," *Journal of applied biomechanics*, vol. 13, no. July 1993, pp. 1–38, 1997.

[13] R. M. Rangayyan, *Biomedical Signal Analysis: A Case-Study Approach*. IEEE/Wiley, New York, NY, 2002.

[14] M. Tomasini, S. Benatti, B. Milosevic, E. Farella, and L. Benini, "Power Line Interference Removal for High-Quality Continuous Biosignal Monitoring with Low-Power Wearable Devices," *IEEE Sensors Journal*, vol. 16, no. 10, pp. 3887–3895, 2016.

[15] V. H. Cene, M. Tosin, J. Machado, and A. Balbinot, "Open Database for Accurate Upper-Limb Intent Detection Using Electromyography and Reliable Extreme Learning Machines," *Sensors (Basel, Switzerland)*, vol. 19, no. 8, 2019.

[16] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[17] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[18] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *ArXiv*, vol. abs/1502.03167, 2015.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[20] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS-W*, 2017.

[21] M. Atzori, A. Gijsberts, S. Heynen, A.-G. M. Hager, O. Deriaz, P. V. der Smagt, C. Castellini, B. Caputo, and H. Müller, "Building the NINAPRO Database: A Resource for the Biorobotics Community - HES SO Valais publications - Aigaion 2.0," *Proceedings of the IEEE International Conference on Biomedical Robotics and Biomechatronics*, p. 51, 2012. [Online]. Available: http://publications.hevs.ch/index.php/publications/show/1172

[22] S. Amsüss, L. P. Paredes, N. Rudigkeit, B. Graimann, M. J. Herrmann, and D. Farina, "Long term stability of surface EMG pattern classification for prosthetic control," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, pp. 3622–3625, 2013.

[23] J. He, D. Zhang, N. Jiang, X. Sheng, D. Farina, and X. Zhu, "User adaptation in long-term, open-loop myoelectric training: Implications for EMG pattern recognition in prosthesis control," *Journal of Neural Engineering*, vol. 12, no. 4, 2015.