

Article

sEMG-Based Continuous Estimation of Finger Kinematics via Large-Scale Temporal Convolutional Network

Chao Chen ^{1,†}, Weiyu Guo ^{1,2,†}, Chenfei Ma ¹ , Yongkui Yang ^{1,*} , Zheng Wang ^{1,*}  and Chuang Lin ³

¹ Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518000, China; chao.chen@siat.ac.cn (C.C.); guoweiyu96@gmail.com (W.G.); googlcfma@gmail.com (C.M.)

² School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

³ School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China; linchuang_78@126.com

* Correspondence: yk.yang@siat.ac.cn (Y.Y.); zheng.wang@siat.ac.cn (Z.W.)

† These authors contributed equally to this work.

Abstract: Since continuous motion control can provide a more natural, fast and accurate man–machine interface than that of discrete motion control, it has been widely used in human–robot cooperation (HRC). Among various biological signals, the surface electromyogram (sEMG)—the signal of actions potential superimposed on the surface of the skin containing the temporal and spatial information—is one of the best signals with which to extract human motion intentions. However, most of the current sEMG control methods can only perform discrete motion estimation, and thus fail to meet the requirements of continuous motion estimation. In this paper, we propose a novel method that applies a temporal convolutional network (TCN) to sEMG-based continuous estimation. After analyzing the relationship between the convolutional kernel’s size and the lengths of atomic segments (defined in this paper), we propose a large-scale temporal convolutional network (LS-TCN) to overcome the TCN’s problem: that it is difficult to fully extract the sEMG’s temporal features. When applying our proposed LS-TCN with a convolutional kernel size of 1×31 to continuously estimate the angles of the 10 main joints of fingers (based on the public dataset Ninapro), it can achieve a precision rate of 71.6%. Compared with TCN (kernel size of 1×3), LS-TCN (kernel size of 1×31) improves the precision rate by 6.6%.

Keywords: temporal convolutional network; human–robot cooperation; surface electromyogram; continuous motion estimation



Citation: Chen, C.; Guo, W.; Ma, C.; Yang, Y.; Wang, Z.; Lin, C.; sEMG-Based Continuous Estimation of Finger Kinematics via Large-Scale Temporal Convolutional Network. *Appl. Sci.* **2021**, *11*, 4678. <https://doi.org/10.3390/app11104678>

Academic Editor: Bruno Baruaque Zanon

Received: 29 March 2021

Accepted: 18 May 2021

Published: 20 May 2021

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Although intelligent robots can perform highly-intensive work in harsh environments, they still cannot complete autonomous decision-making in complex situations [1], especially in medical treatment [2,3] and military scenarios [4]. Human–robot cooperation (HRC) systems with high efficiency are promising solutions for performing these tasks safely and reliably. Hence, developing a new generation of HRC systems that are more natural, fast and direct has become a hot research topic. Finding a quicker and more natural interactive interface that does not require any additional learning process is one the most significant aims of research for developing a new generation of HRC system. In other words, in an efficient HRC system, the machine should be able to understand human intentions quickly and accurately. Meanwhile, humans should not bear any new physical or mental burdens.

Currently, the signals used for intention recognition in HRC systems can be divided into two categories, i.e., non-physiological signals and physiological signals. Among them, non-physiological signals are widely used in daily life—for instance, in images, videos and forms of mechanical input (keyboards and control buttons). However, non-physiological signal-based systems suffer from poor real-time performance, and the signal collection

equipment is often inconvenient to carry around [5]. On the other hand, physiological signals—such as sEMG—have characteristics that directly reflect human intentions, and these physiological signals are easy to collect [6]. The surface electromyogram (sEMG) is generated by action neurons in muscle. It is the signal of an action potential superimposed on the surface of the skin through time and space. The sEMG contains rich information of motor intentions [7], and can be collected in a non-invasive way. In addition, since the action potential is generated before the muscle's movement, the external information transmission can be completed 30 ms to 150 ms ahead of the actual action. The human hand—the most frequently used body part for external interactions and one of the most complex organs—can provide abundant interactive signals for HRC [8]. Compared with hands' other movements, finger movements are more delicate and complex, involving many small deep muscles and more than 20 joint degrees of freedom [9]. Hence, it is still challenging to estimate finger movement.

At present, there are two methods for extracting the motion intentions of sEMG signals. One is to use a classification algorithm to classify the sEMG to generate discrete motion information, which can be used as switch signals in HRC [10–12]. However, this simple classification method cannot meet the requirements of HRC for our daily use. The other one—which is known as sEMG-based continuous estimation—is to use nonlinear models to extract continuous motion intention information (such as the angle of motion joint at each moment), which is more natural and more accurate than the classification algorithm-based method [1,13,14]. Hence, in the rest of this paper, we focus on the research of sEMG-based continuous estimation.

Traditionally, most of motion intention estimation methods have adopted conventional machine learning algorithms to decode EMG/sEMG signals and perform artificial feature selection. Jiang et al. proposed a synchronous proportional multi-degree of freedom (DOF) EMG control method based on sparse constrained non-negative matrix factorization [15]. This method further expands the researchers' thinking in pattern recognition, but still cannot meet the actual needs in terms of the number of DOFs and the complexity of recognizable gestures. Xiloyannis et al. used a Gaussian process to estimate hand motion [16]. The Gaussian process defines the prior function. After observing certain function values, they can be converted into posterior functions through algebraic operations. However, in theory, the Gaussian process will lose its validity in high-dimensional space. Clancy et al. estimated the elbow joint torque produced by sEMG through linear and nonlinear dynamic models [17]. However, these methods using traditional machine learning algorithms cannot meet the requirements of current HRC scenarios in terms of accuracy and real-time responses [18].

In recent years, researchers began to focus on continuous motion estimation based on deep learning. These methods are mainly based on advanced time-oriented machine learning methods or deep learning methods. Alique et al. [19] proposed a neural network based approach to predict the mean cutting force in milling progress. Precup et al. [20] developed Takagi Sugeno-Kang (TSK) fuzzy models, which are evolved by an incremental online identification algorithm. Matía et al. [21] investigated the fuzzy Kalman filter (FKF) and improved its implementation by reformulating uncertainty representation.

Smith et al. proposed an artificial neural network to estimate the angles of five metacarpophalangeal joints [22]. This work introduced neural networks into the field of continuous motion estimation and verified their feasibility. However, due to the limitations of neural network development at that time, this approach can only estimate simpler gestures. Muceli et al. proposed a method based on multilayer perceptrons (MLP) to estimate the motion of multiple joints at the same time [23]. This method divides the sEMG into segments and estimates the joint angle values corresponding to each segment of the sEMG. However, the relevance of the input before and after is not considered, which makes it difficult for the accuracy rate to meet the actual demand. To solve this problem, the recurrent neural network (RNN) has been used for EMG control [24,25]. The RNN can analyze the time correlation between multiple inputs, which further improves the accuracy

of the model. However, the application scenarios of continuous motion estimation are mostly in edge devices, and it is difficult to meet the demand of RNN for computing power, resulting in poor real-time performance.

In this paper, we propose a large-scale temporal convolutional network (LS-TCN) to continuously estimate the angles of the 10 main joints of the finger in real time. LS-TCN achieved the estimation accuracy of 71.6%, which is an improvement of accuracy over traditional methods by 6.6%.

The rest of the paper is organized as follows. We explain the dataset and our methodology in Section 2. Results and discussion are presented in Section 3. Section 4 concludes the paper.

2. Methodology

2.1. Data Set

In order to fairly compare this with other methods, the public database Ninapro DB2 was chosen. Ninapro [26] is a publicly available multi-mode database, designed for facilitating the research of artificial intelligence robots and prosthetic hands. Ninapro includes EMG, kinematics, inertia, eye tracking, visual, clinical and neurocognitive data. Ninapro's data are widely used by scientific researchers for machine learning, robotics, medicine and neurocognitive science.

We chose 8 subjects out of the database, and those 8 subjects cover all subjects' information as much as possible. The ranges of height, weight and age were 154–187 cm, 50–90 kg and 24–35; there were 5 males and 3 females; and regarding dominant hand, for 6 subjects it was the right and for 2 the left, respectively. Since grasping movements are the most commonly used hand movements in daily lives, we selected 6 types of grasping movements, as shown in Figure 1.

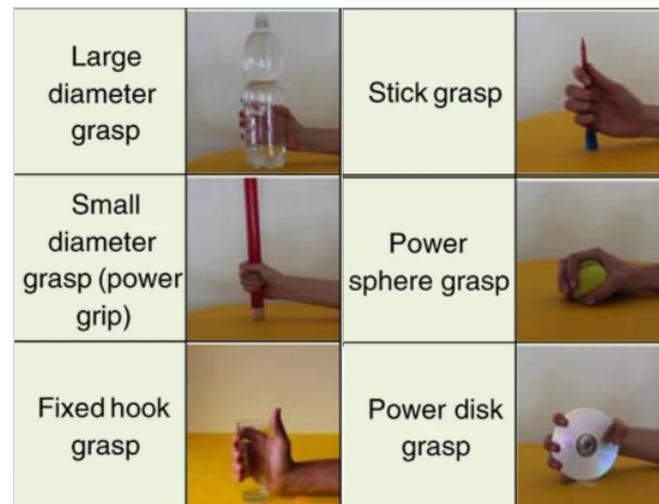


Figure 1. The 6 types of grasping movements we selected for experiments from Ninapro [26].

Note that we selected only 6 grasping movements, because continuous estimation tasks are more challenging than classification tasks in terms of modeling, especially when simultaneously estimating 10 joint angles as we did in this subject. To possess both good fitting capabilities and real-time performance, we could not adopt many grasping movements for modeling. Otherwise there would have been many parameters in the model, such that real time performance could not be achieved. We will design light-weight models for more movements in future studies.

We selected the 6 movements based on the shapes and diameters of the objects grasped. The shapes included a cylinder, a ball and a flat object. The diameters included large, medium and small-diameter objects. The hand joint ranges and the coordination mechanisms of the selected movements were different such that they could be used for modeling.

Ninapro DB2 used a 22-sensor CyberGloveII data-glove to measure hand kinematics, and it adopted Delsys Trigno wireless system, including 12 wireless sEMG electrodes, to collect sEMG signals. We used a 12-channel sEMG to estimate 10 main joint angles. We chose the proximal interphalangeal point (PIP) and the metacarpophalangeal point (MCP) as estimated joints, because they are the main active joints in the grasping movement. These 10 joint angles we selected are shown in Figure 2.



Figure 2. The 10 joint angles we selected for experiments.

2.2. Data Processing

The hand kinematics movement was collected at 20 Hz and resampled to 2000 Hz to synchronize with the sEMG signals. The sEMG signals and hand joint angle signals were divided into fragment sequences of 100 ms duration, and the sliding step-length was 0.5 ms. The commonly used feature extraction methods in EMG processing include root mean square value (RMS) [27], mean square value (MSV) [28], envelopes [29], etc. In this paper, RMS was employed as the feature extraction approach, due to its abundant information content and uncomplicated computation process. The RMS feature extraction used a 100 ms processing window size with 0.5 ms stride length. The RMS could be calculated as:

$$RMS = \sqrt{\frac{1}{N} \sum_i^n (n_i - \bar{n})^2} \quad (1)$$

where n_i represents the values in the window, and \bar{n} is the mean value of the window; N is the length of the window.

2.3. Parameters for Evaluation

The Pearson correlation coefficient (PCC) is commonly used to measure whether two sequences are on a line or not, and to measure the linear relationship between distance variables [30]. Here, we used it to measure the correlation between the actual joint angle and the estimated joint angle. Its calculation formula is as follows:

$$PCC = \frac{\sum_{i=1}^N (\theta_{est} - \bar{\theta}_{est})(\theta_{real} - \bar{\theta}_{real})}{\sqrt{\sum_{i=1}^N (\theta_{est} - \bar{\theta}_{est})^2} \sqrt{\sum_{i=1}^N (\theta_{real} - \bar{\theta}_{real})^2}} \quad (2)$$

where θ_{est} , $\bar{\theta}_{est}$, θ_{real} and $\bar{\theta}_{real}$ are the value of estimated joint angle, the mean value of estimated joint angles, the value of real joint angle and the mean value of real joint angles, respectively. The PCC value is between -1 and 1 , which can be used to evaluate the performance of the algorithm. The closer the PCC value is to 1 , the more similar the predicted finger trajectory is to that of the actual movement, and the higher the accuracy of the estimation can reach.

We use root mean square error (RMSE) to evaluate the numerical error of amplitude between predicted joint angles and actual joint angles. It can be described as:

$$RMSE = \sqrt{\frac{1}{N} \sum_i^N (\theta_{est} - \theta_{real}^-)^2} \quad (3)$$

2.4. Applying Tcn to Sems-Based Continuous Estimation

The temporal convolutional network (TCN) was initially designed by Bai et al. [31] for sequence modeling tasks. Their experimental results showed that TCN outperforms canonical recurrent networks (such as RNN, LSTM and GRU) across a diverse range of sequence modeling tasks (such as Sequential MNIST, Music JSB Chorales and Word-level PTB) [31]. The main architectural elements in the TCN are dilated causal convolution (modified from causal convolution) and residual connections.

As shown in Figure 3, causal convolution only looks back at a history with a size linearly proportional to the network's depth. Differently from the traditional convolution neural network, causal convolution can not see the future data. In other words, it is unidirectional structure, not bidirectional. Thanks to this, causal convolution ensures that the model only uses the time series before the moment when doing forecast.

In order to extract the features of longer time series, the TCN uses a modified causal convolution called dilated causal convolution [31], as shown in Figure 4a. It can extract longer time series at the same depth. Differently from the causal convolution, dilation causal convolution allows the input of convolution to have interval sampling. The interval between sampling points of convolutional kernel is determined by d , whose value generally increases with the depth of the layer. This means that the receptive field increases exponentially with the network's depth. Therefore, for a certain receptive field, the depth of the network with dilated causal convolution is significantly less than that with causal convolution.

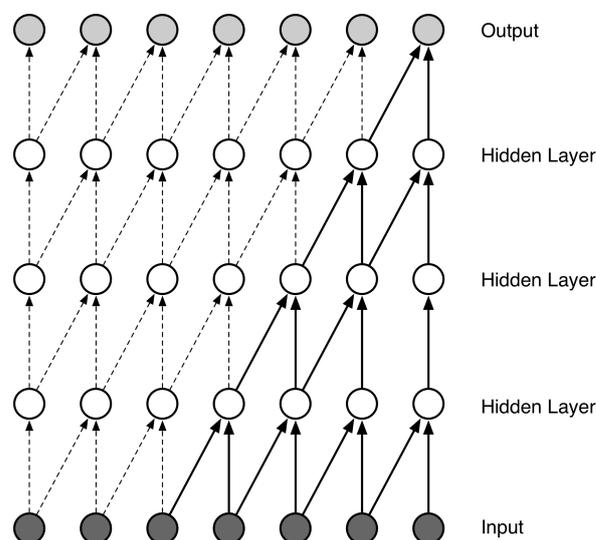


Figure 3. The structure of causal convolution, where each circle is a neuron. Solid arrows are used to depict the data flow of the right-most output neuron, and dashed arrows represent other data flow.

In order to make the network's error transfer across layers and effectively prevent the gradient disappearing, the TCN constructs a residual block to replace one layer of convolution. As shown in the Figure 4b, the residual block contained two layers of convolution and nonlinear mapping. In each layer, weightnorm and dropout were added to regularize the network.

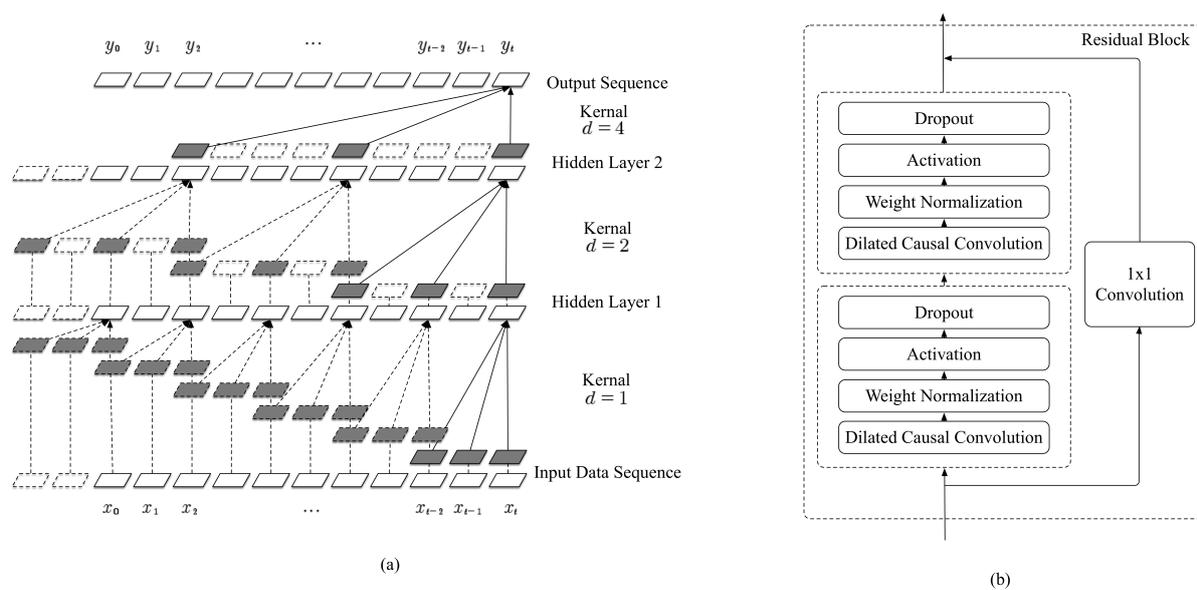


Figure 4. The structure of dilated convolution and residual connections, where each parallelogram is a neuron, solid arrows are used to depict the data flow of the right-most neuron and dashed arrows represent other data flow. (a) Dilated convolution; (b) Residual connections.

As sEMG is one type of sequence modeling, TCN can be adopted to extract sEMG's features. In this paper, we propose a novel method that applies TCN to the sEMG-based continuous estimation. When directly applying this TCN to continuously estimating the angles of the 10 main joints of the finger (based on Ninapro dataset), it can only get a terrible precision rate (i.e., the Pearson correlation coefficient, PCC), 65%, which will be explained in the following section.

2.5. The Large-Scale Temporal Convolutional Network

The depth of the network and the convolutional kernel's size are two determining factors for the accuracy of the deep learning network. Therefore, in this subsection, we will discuss how to improve the precision rate of the TCN for sEMG-based continuous estimation of finger kinematics, by considering the depth and convolutional kernel size of TCN. Finally, we propose our large-scale temporal convolutional network (LS-TCN).

With an increase in the depth of a deep learning network, the extracted features will become more and more abstract. If we simply deepen the network, the details of the underlying information will be lost, especially the temporal features. Considering that the continuous motion estimation requires the details of the sEMG signal, we limited the number of layers of network to 5 layers.

After the analysis of the influences of the depth and convolutional kernel size on the network precision rate and parameter size (Section 3), we created the large-scale temporal convolutional network (LS-TCN). This LS-TCN is a 5-layer network with a convolutional kernel size of 1×31 , and the convolutional channels are [32, 64, 64, 32, 10]. Following the convolution layer, 2 dense layers (256, 10) are used to complete the mapping from feature space to target value.

3. Results and Discussion

3.1. Experimental Setup

We built all models on the PyTorch [32] platform to compare their performance. Mean square error (MSE) was adopted as the loss function, which has excellent performance in regression tasks. Adam was used as the optimizer with a learning rate of 0.0001. We used the public dataset Ninapro for predicting the angles of 10 joints in 6 kinds of grasping motion. The first 60 percent and the last 40 percent of each movement were used for training and testing, respectively.

3.2. Movement Data

Movements are characterized by the angles of joints, and these joints involve the use of different muscles, whose movements can be estimated using the sEMG signals [33]. A specific movement consists of a set of joint angles, which corresponds to a set of sEMG signals. We adopted 12 channels of sEMG signals to predict 10 joint angles of 5 fingers, with two joints from one finger (as depicted in Figure 2). Consequently, to evaluate the effectiveness of continuous movement estimation, we predicted the joint angles with sEMG signals and compared our prediction results with those of other methods in Section 3.4.

3.3. Kernel Size Optimization

In order to find the optimized convolutional kernel size of the network (other architectural elements are kept same as the TCN) for sEMG-based continuous estimation, we have explored the influence of different convolutional kernel size on the network accuracy.

From the experimental results (as shown in Figure 5), it can be seen that when we expanded the convolutional kernel, the correlation coefficient increased until achieving the highest peak at 82.06% (where kernel size is 31), and then fell back. This could be explained as follows. In sEMG, there is a strong correlation between the points within a certain period of time, which we call atomic segments.

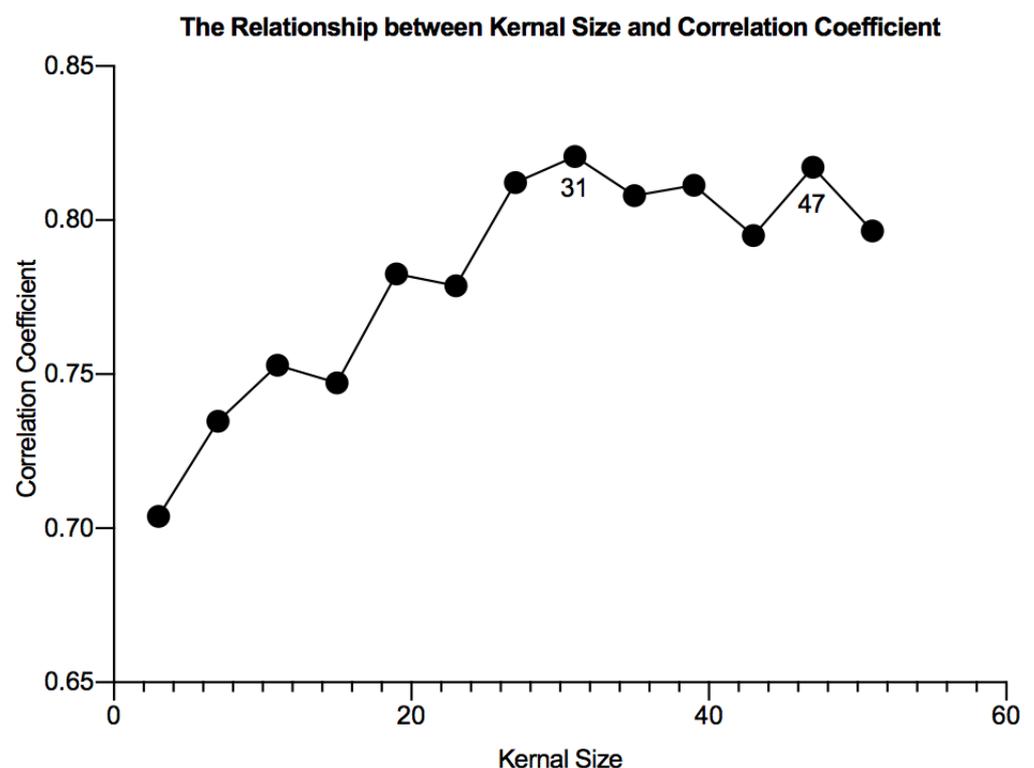


Figure 5. The relationship between kernel size and PCC.

As shown in Figure 6, we define the atomic segment as the shortest time sequence in which sEMG can express effective information. When the convolutional kernel is smaller than the atomic segment, such as 1×3 in TCN, it is difficult to obtain sEMG's information. This is caused by the fine-grained, long sequence information contained in the sEMG, and a shallow network with a small convolutional kernel cannot obtain enough temporal features. Similar situations have appeared in image segmentation. Peng et al. [34] found that a large convolutional kernel has better performance for image segmentation (pixel classification). This further proves that a large convolutional kernel is helpful for maintaining the underlying details.

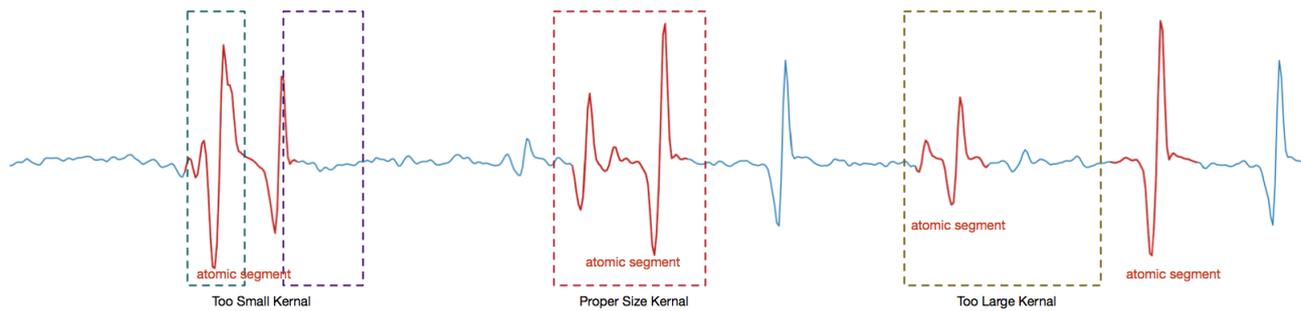


Figure 6. The atomic segment is defined as the shortest time sequence in which sEMG can express effective information. When the convolutional kernel is too small, it is difficult to obtain effective information. If the convolutional kernel is too large, it will contain redundant information, which increases the difficulty for the network to learn efficient information and increase the number of network parameters.

On the other hand, if the convolutional kernel is too large (such as larger than 31 in Figure 5), it will contain non-strongly correlated and redundant information, which will not help improve the network accuracy but will increase the number of network parameters. Therefore, when setting the convolutional kernel size equal to or slightly larger than the length of the sEMG's atomic segment, the network will maximize the network precision rate while maintaining a minimum network parameter size.

3.4. Performance Comparison

The experimental results are shown in Figure 7. It shows that our proposed LS-TCN can achieve an accuracy (measured by correlation coefficient) of 71.6% for sEMG-based continuous motion estimation (six common gripping actions in Ninapro in this case). Compared with TCN [31], the accuracy of LS-TCN was improved by 6.6%. As shown in Figure 8, LS-TCN can achieve the best average RMSE performance. For subjects 4 and 6, it was not the best, because the convolution structure imore easily produces jitter than the method with the recurrent structure. However, the convolution structure is easy to accelerate in hardware and thus can provide better real-time performance [35], which was demonstrated in our previous work [36]. Note that individual factors such as low muscle mass or obesity may lead to poor performance of the model.

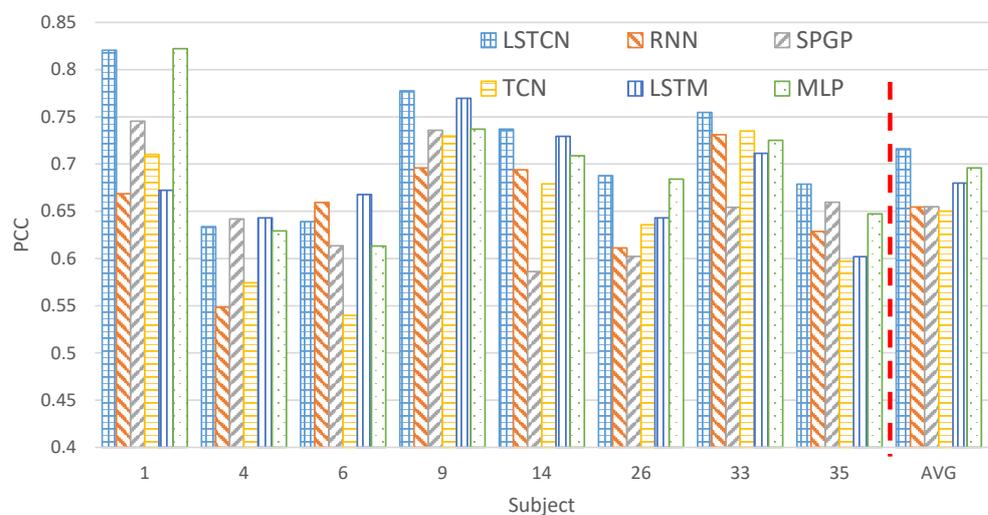


Figure 7. Summary of the PCC of 8 subjects. PCC denotes the correlation between predicted joint angles and real joint angles. The higher, the better.

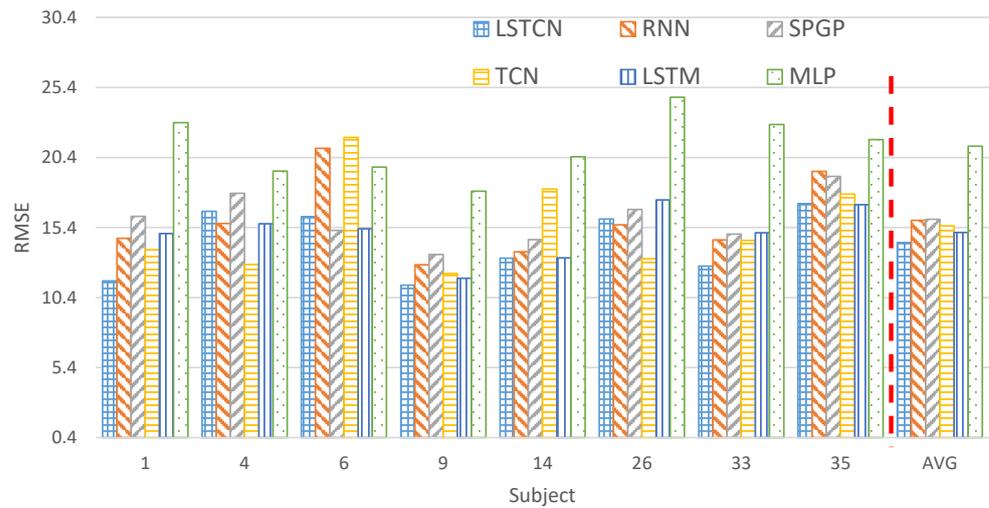


Figure 8. Summary of the RMSE of 8 subjects. RMSE denotes the root mean square error between predicted joint angles and real joint angles. The lower, the better.

Figures 9–11 display orange real joint angles from measurements and blue predicted joint angles estimated from sEMG signals using different methods. Although the blue predicted values look similar to sEMG signals, they are actually predicted joint angles. Two joints were used for each finger (as depicted in Figure 2), and therefore there are 10 subfigures (indexed from 1 to 10) for five fingers in total. Note that for each movement, there was a significant joint angle amplitude variation, which resulted in a peak or a valley. In each subfigure, two peaks or two valleys denote the same movement, because every movement was performed twice with the same duration in our test dataset. Thus there are 12 peaks/valleys for selected six grasping movements. Subfigure 1 of Figure 9 illustrates the division of six movements with two repetitions.

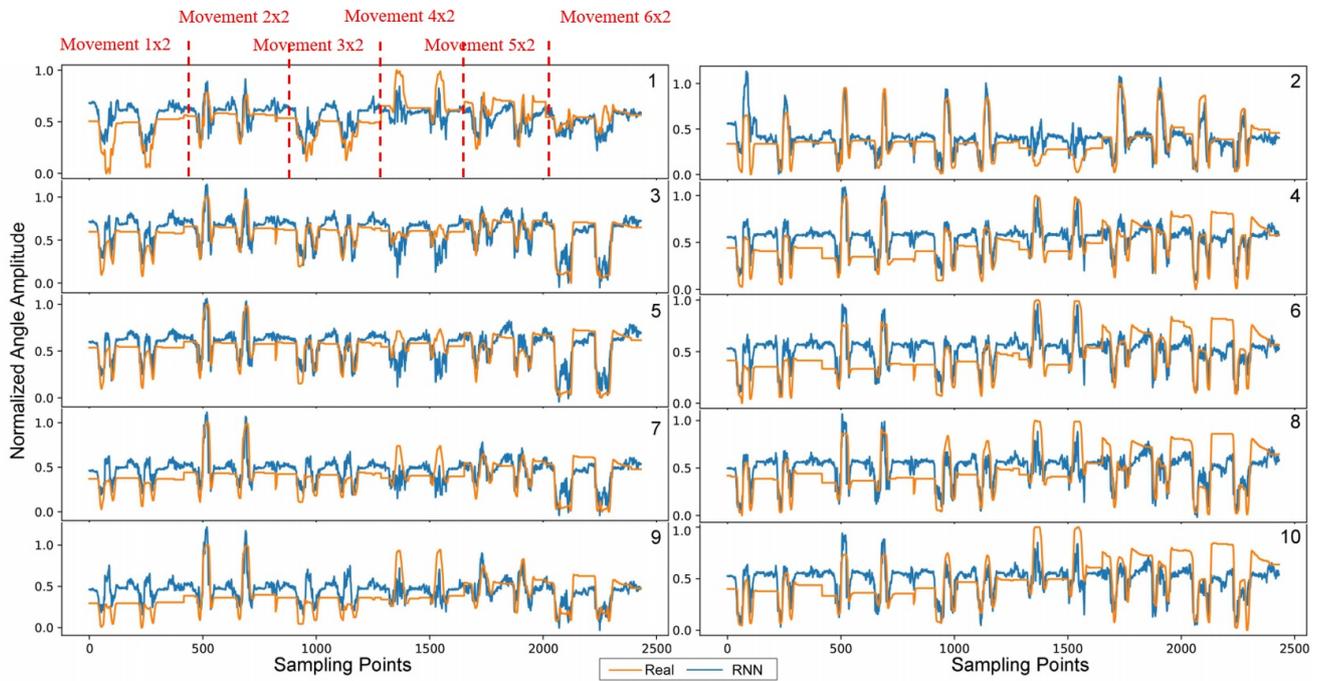


Figure 9. The continuous amplitudes of real and predicted joint angles using RNN. There are 10 subfigures for 5 fingers (i.e., two joints per finger as depicted in Figure 2), and each subfigure denotes the result of a finger joint, where the x-axis represents the sampling point index and the y-axis denotes normalized joint angles. Each movement was performed twice, and thus every two peaks or valleys represents a movement; i.e., 6 grasping movements were characterized by 12 peaks/valleys in each subfigure. The annotations of subfigure 1 indicate the division of 6 movements with 2 repetitions.

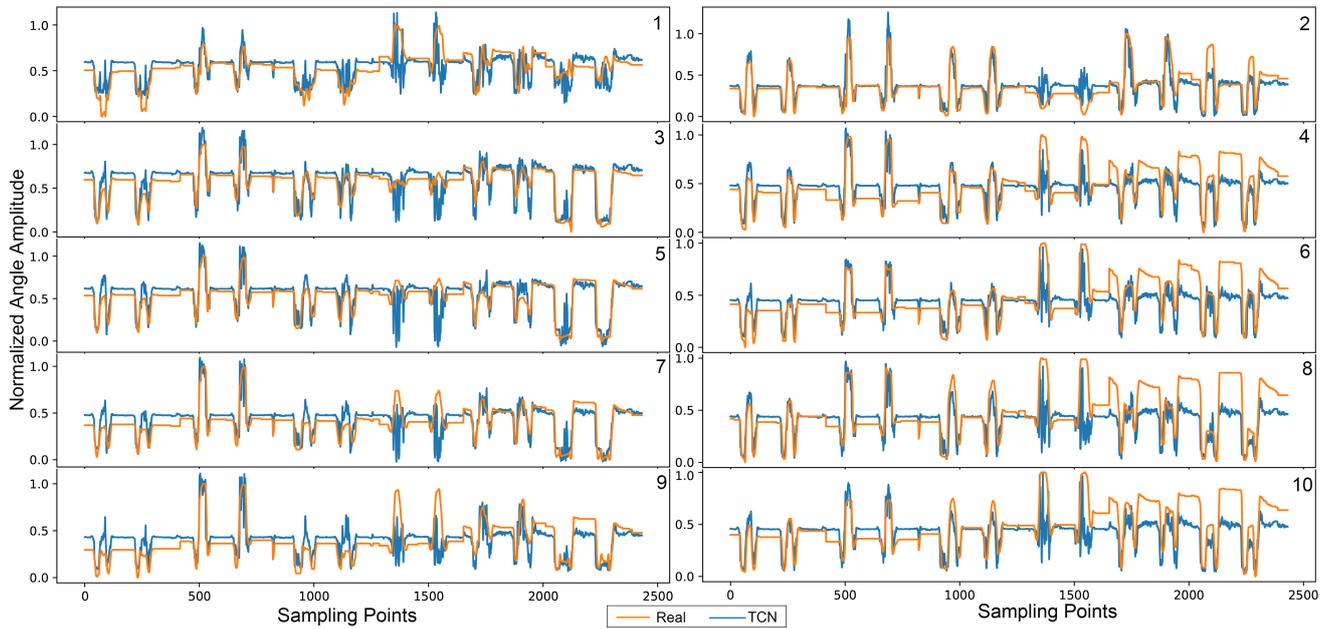


Figure 10. The continuous amplitudes of real and predicted joint angles using TCN (kernel = 3). There are 10 subfigures for 5 fingers (i.e., two joints per finger as depicted in Figure 2), and each subfigure denotes the result of a finger joint, where the x-axis represents the sampling point index and the y-axis denotes normalized joint angles. Each movement was performed twice, and thus every two peaks or valleys represents a movement; i.e., 6 grasping movements were characterized by 12 peaks/valleys in each subfigure.

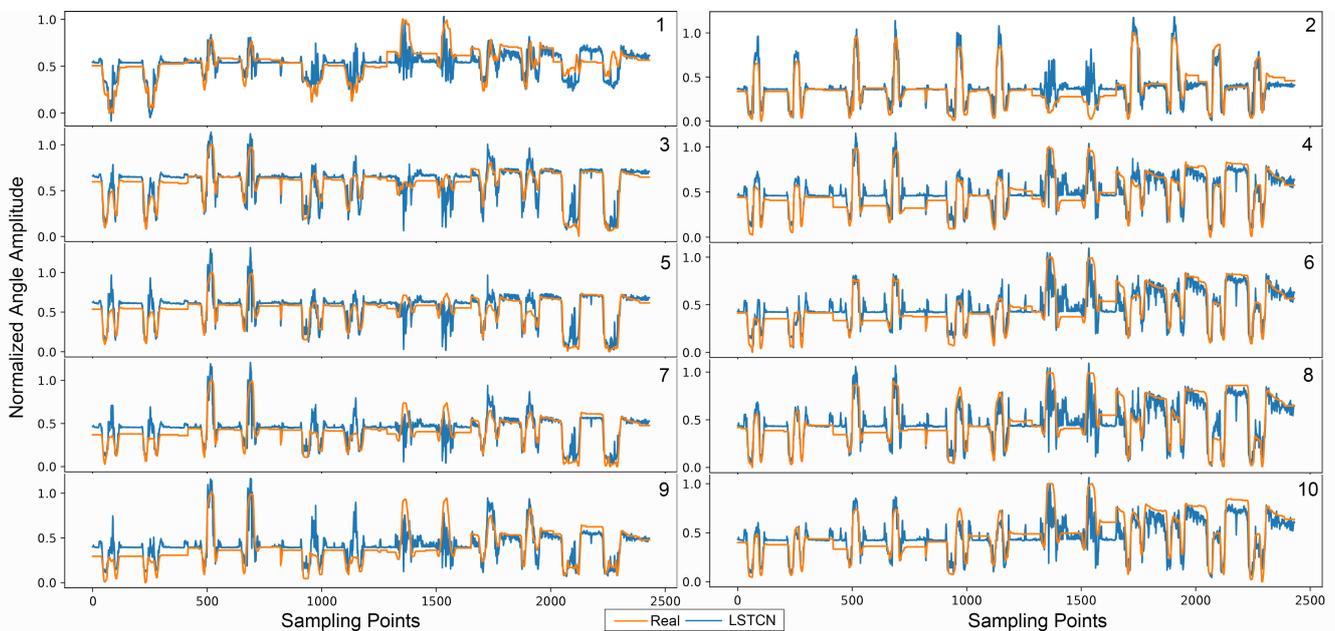


Figure 11. The continuous amplitudes of real and predicted joint angles using LS-TCN. There are 10 subfigures for 5 fingers (i.e., two joints per finger as depicted in Figure 2), and each subfigure denotes the result of a finger joint, where the x-axis represents the sampling point index and the y-axis denotes normalized joint angle. Each movement was performed twice, and thus two peaks or two valleys represents one repeated movement; i.e., 6 grasping movements are characterized by 12 peaks/valleys in each subfigure.

From Figures 9–11, we can see that the joint angle curve predicted by LS-TCN is closer to the real curve, especially in movements 5 and 6 (the last four peaks/valleys). In addition, we can see that the estimation for movement 4 (i.e., the Power Sphere Grasping) was the worst and vibrated most among the six movements for all the three models (including the RNN, the TCN and the proposed LS-TCN). This was caused by the fact that the sampled joint angle of movement 4 varied dramatically between different repetitions, and this problem may be solved by adding a real-time smoothing algorithm at the end of the methods. For other movements, since the sampled joint angle was much more stable than that of movement 4 in different repetitions, the performance was far better.

This paper explored the influences of the kernel size and the network depth of TCN on accuracy. We found that the convolutional kernel size we chose allows the minimum effective information of sEMG, and small numbers of layers are beneficial to continuous motion estimation based on sEMG. Then we proposed LS-TCN based on the experimental results and verified the performance of LS-TCN, TCN, RNN and SPGP when extracting the continuous motion information from sEMG. Although LS-TCN improved the accuracy by 6.6% compared with TCN, there are still several problems to be solved in practical applications of human–computer interaction. First, the current model leverages personal data for training and lacks generality. One possible solution is to train the general model with a large number of subjects and adjust it with transfer learning, which we will try in our future work. Second, collecting stable and high-quality EMG signals is still difficult for current studies. For example, dry electrodes are prone to displacement and wet electrodes are not easy to wear. Third, the stability of the prediction angle needs to be further improved, which may be solved by adding an implementation smoother. Forth, we plan to find optimal parameterization of our network by leveraging advanced technologies in our future work, such as nature-inspired optimization algorithms [37] and multi-objective optimization [38].

4. Conclusions

In this paper, we proposed LS-TCN for sEMG-based continuous motion estimation. We used it for predicting the angles of 10 joints in six kinds of grasping motion. By discussing the influences of network depth and convolutional kernel size on the prediction accuracy, we found that if the convolutional kernel's size is close to the length of the atomic segment, the prediction accuracy of the network will be optimized. Based on TCN, we proposed the LS-TCN whose convolutional kernel size is 1×31 . Finally, we tested the LS-TCN with six common gripping actions on the Ninapro dataset, and the accuracy was 71.6%, which proves that LS-TCN has good prospects for application in sEMG-based continuous motion estimation.

Author Contributions: Conceptualization, W.G. and C.L.; methodology, W.G. and C.C.; experiments, Y.Y.; visualization, C.M.; result analysis, C.C. and Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science and Foundation of China (NSFC 61902355 and 61702493), the Key-Area Research and Development Program of Guangdong Province (grant number 2019B010155003) and the Guangdong Basic and Applied Basic Research Foundation (grant number 2020B1515120044).

Institutional Review Board Statement: Not Applicable

Informed Consent Statement: Not Applicable.

Data Availability Statement: The data presented in this study are openly available in Ninapro at doi:10.1111/aor.13004 [26].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bi, L.; Feleke, A.; Guan, C. A review on EMG-based motor intention prediction of continuous human upper limb motion for human-robot collaboration. *Biomed. Signal Process. Control* **2019**, *51*, 113–127. [[CrossRef](#)]
2. Morimoto, T.K.; Hawkes, E.W.; Okamura, A.M. Design of a Compact Actuation and Control System for Flexible Medical Robots. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1579–1585. [[CrossRef](#)] [[PubMed](#)]
3. Kuo, C.M.; Chen, L.C.; Tseng, C.Y. Investigating an innovative service with hospitality robots. *Int. J. Contemp. Hosp. Manag.* **2017**, *29*, 1305–1321. [[CrossRef](#)]
4. Marchant, G.E.; Allenby, B.; Arkin, R.C.; Borenstein, J.; Gaudet, L.M.; Kittrie, O.; Lin, P.; Lucas, G.R.; O'Meara, R.; Silberman, J. International governance of autonomous military robots. In *Handbook of Unmanned Aerial Vehicles*; Springer: Dordrecht, The Netherlands, 2015; pp. 2879–2910. [[CrossRef](#)]
5. Szabo, R.; Gontean, A. Controlling a robotic arm in the 3D space with stereo vision. In Proceedings of the 2013 21st Telecommunications Forum Telfor, TELFOR 2013-Proceedings of Papers, Belgrade, Serbia, 26–28 November 2013; pp. 916–919. [[CrossRef](#)]
6. Kopniak, P.; Kaminski, M. Natural interface for robotic arm controlling based on inertial motion capture. In Proceedings of the 2016 9th International Conference on Human System Interactions, HSI 2016, Portsmouth, UK, 6–8 July 2016; pp. 110–116. [[CrossRef](#)]
7. Phinyomark, A.; Quaine, F.; Charbonnier, S.; Serviere, C.; Tarpin-Bernard, F.; Laurillau, Y. EMG feature evaluation for improving myoelectric pattern recognition robustness. *Expert Syst. Appl.* **2013**, *40*, 4832–4840. [[CrossRef](#)]
8. Lin, J.; Wu, Y.; Huang, T.S. Modeling the constraints of human hand motion. In Proceedings of the Workshop on Human Motion, HUMO 2000, Austin, TX, USA, 7–8 December 2000; pp. 121–126. [[CrossRef](#)]
9. Kapandji, I.A. The Physiology of the Joints. Vol. 1, Upper Limb. *Postgrad. Med. J.* **1971**, *47*, 140–140. [[CrossRef](#)]
10. Micera, S.; Carpaneto, J.; Raspopovic, S. Control of Hand Prostheses Using Peripheral Information. *IEEE Rev. Biomed. Eng.* **2010**, *3*, 48–68. [[CrossRef](#)] [[PubMed](#)]
11. Zanghieri, M.; Benatti, S.; Burrello, A.; Kartsch, V.; Conti, F.; Benini, L. Robust Real-Time Embedded EMG Recognition Framework Using Temporal Convolutional Networks on a Multicore IoT Processor. *IEEE Trans. Biomed. Circuits Syst.* **2020**, *14*, 244–256. [[CrossRef](#)] [[PubMed](#)]
12. Côté-Allard, U.; Fall, C.L.; Drouin, A.; Campeau-Lecours, A.; Gosselin, C.; Glette, K.; Laviolette, F.; Gosselin, B. Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2019**, *27*, 760–771. [[CrossRef](#)]
13. Jiang, N.; Dosen, S.; Muller, K.R.; Farina, D. Myoelectric control of artificial limbs: is there a need to change focus? [In the Spotlight]. *IEEE Signal Process. Mag.* **2012**, *29*, 148–152. [[CrossRef](#)]
14. Kapelner, T.; Vujaklija, I.; Jiang, N.; Negro, F.; Aszmann, O.C.; Principe, J.; Farina, D. Predicting wrist kinematics from motor unit discharge timings for the control of active prostheses. *J. NeuroEng. Rehabil.* **2019**, *16*, 1–11. [[CrossRef](#)]
15. Jiang, N.; Englehart, K.B.; Parker, P.A. Extracting simultaneous and proportional neural control information for multiple-dof prostheses from the surface electromyographic signal. *IEEE Trans. Biomed. Eng.* **2009**, *56*, 1070–1080. [[CrossRef](#)] [[PubMed](#)]
16. Xiloyannis, M.; Gavriel, C.; Thomik, A.A.C.; Faisal, A.A. Gaussian Process Autoregression for Simultaneous Proportional Multi-Modal. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1785–1801. [[CrossRef](#)]
17. Clancy, E.A.; Liu, L.; Liu, P.; Moyer, D.V. Identification of constant-posture EMG-torque relationship about the elbow using nonlinear dynamic models. *IEEE Trans. Biomed. Eng.* **2012**, *59*, 205–212. [[CrossRef](#)] [[PubMed](#)]
18. Ahsan, M.R.; Ibrahimy, M.I.; Khalifa, O.O. EMG signal classification for human computer interaction: A review. *Eur. J. Sci. Res.* **2009**, *33*, 480–501.
19. Alique, A.; Haber, R.; Ros, S.; Gonzalez, C. A neural network-based model for the prediction of cutting force in milling process. A progress study on a real case. In Proceedings of the 2000 IEEE International Symposium on Intelligent Control, Held Jointly with the 8th IEEE Mediterranean Conference on Control and Automation (Cat. No.00CH37147), Patras, Greece, 19 July 2000; pp. 121–125. [[CrossRef](#)]
20. Precup, R.E.; Teban, T.A.; Abu, A.; Borlea, A.B.; Zamfirache, I.A.; Petriu, E.M. Evolving Fuzzy Models for Prosthetic Hand Myoelectric-Based Control. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 4625–4636. [[CrossRef](#)]
21. Matía, F.; Jiménez, V.; Alvarado, B.P.; Haber, R. The fuzzy Kalman filter: Improving its implementation by reformulating uncertainty representation. *Fuzzy Sets Syst.* **2021**, *402*, 78–104. [[CrossRef](#)]
22. Smith, R.J.; Tenore, F.; Huberdeau, D.; Etienne-Cummings, R.; Thakor, N.V. Continuous decoding of finger position from surface EMG signals for the control of powered prostheses. In Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS'08 - "Personalized Healthcare through Technology", Vancouver, BC, Canada, 20–25 August 2008; pp. 197–200. [[CrossRef](#)]
23. Muceli, S.; Farina, D. Simultaneous and proportional estimation of hand kinematics from EMG during mirrored movements at multiple degrees-of-freedom. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2012**, *20*, 371–378. [[CrossRef](#)]
24. Wang, C.; Guo, W.; Zhang, H.; Guo, L.; Huang, C.; Lin, C. sEMG-based continuous estimation of grasp movements by long-short term memory network. *Biomed. Signal Process. Control* **2020**, *59*, 101774. [[CrossRef](#)]
25. Xia, P.; Hu, J.; Peng, Y. EMG-Based Estimation of Limb Movement Using Deep Learning With Recurrent Convolutional Neural Networks. *Artif. Organs* **2017**. [[CrossRef](#)]

26. Atzori, M.; Muller, H. The Ninapro database: A resource for sEMG naturally controlled robotic hand prosthetics. In Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, Milan, Italy, 25–29 August 2015; Volume 2015-Novem, pp. 7151–7154. [[CrossRef](#)]
27. Willmott, C.J.; Matsuura, K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Clim. Res.* **2005**, *30*, 79–82. [[CrossRef](#)]
28. Ryait, H.S.; Arora, A.S.; Agarwal, R. SEMG signal analysis at acupuncture points for elbow movement. *J. Electromyogr. Kinesiol.* **2011**, *21*, 868–876. [[CrossRef](#)] [[PubMed](#)]
29. Supuk, T.G.; Skelin, A.K.; Cic, M. Design, development and testing of a low-cost sEMG system and its use in recording muscle activity in human gait. *Sensors* **2014**, *14*, 8235–8258. [[CrossRef](#)] [[PubMed](#)]
30. Gooch, J.W. Pearson Correlation Coefficient. In *Encyclopedic Dictionary of Polymers*; Springer: Berlin/Heidelberg, Germany, 2011; p. 990. [[CrossRef](#)]
31. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271. 1803.
32. Ketkar, N. *Introduction to PyTorch BT-Deep Learning with Python: A Hands-on Introduction*; Apress: Berkeley, CA, USA, 2017; pp. 195–208. [12](#). [[CrossRef](#)]
33. Atzori, M.; Gijssberts, A.; Castellini, C.; Caputo, B.; Hager, A.G.M.; Elsig, S.; Giatsidis, G.; Bassetto, F.; Müller, H. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Sci. Data* **2014**, *1*, 1–13. [[CrossRef](#)] [[PubMed](#)]
34. Peng, C.; Zhang, X.; Yu, G.; Luo, G.; Sun, J. Large kernel matters - Improve semantic segmentation by global convolutional network. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2016; pp. 1743–1751. [[CrossRef](#)]
35. Chen, Y.H.; Krishna, T.; Emer, J.S.; Sze, V. Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks. *IEEE J. Solid-State Circuits* **2017**, *52*, 127–138. [[CrossRef](#)]
36. Wang, Z.; Zhou, L.; Xie, W.; Chen, W.; Su, J.; Chen, W.; Du, A.; Li, S.; Liang, M.; Lin, Y.; et al. Accelerating hybrid and compact neural networks targeting perception and control domains with coarse-grained dataflow reconfiguration. *J. Semicond.* **2020**, *41*, 022401. [[CrossRef](#)]
37. Precup, R.E.; David, R.C. *Nature-Inspired Optimization Algorithms for Fuzzy Controlled Servo Systems*; Butterworth-Heinemann: Oxford, UK, 2019; p. iv. [[CrossRef](#)]
38. Haber, R.E.; Beruvides, G.; Quiza, R.; Hernandez, A. A Simple Multi-Objective Optimization Based on the Cross-Entropy Method. *IEEE Access* **2017**, *5*, 22272–22281. [[CrossRef](#)]