Instinctive Real-time sEMG-based Control of Prosthetic Hand with Reduced Data Acquisition and Embedded Deep Learning Training

Zeyu Yang, Angus B. Clark, Digby Chappell, and Nicolas Rojas

Abstract— **Achieving instinctive multi-grasp control of prosthetic hands typically still requires a large number of sensors, such as electromyography (EMG) electrodes mounted on a residual limb, that can be costly and time consuming to position, with their signals difficult to classify. Deep-learning-based EMG classifiers however have shown promising results over traditional methods, yet due to high computational requirements, limited work has been done with in-prosthetic training. By targeting specific muscles non-invasively, separating grasping action into hold and release states, and implementing data augmentation, we show in this paper that accurate results for embedded, instinctive, multi-grasp control can be achieved with only 2 low-cost sensors, a simple neural network, and minimal amount of training data. The presented controller, which is based on only 2 surface EMG (sEMG) channels, is implemented in an enhanced version of the OLYMPIC prosthetic hand. Results demonstrate that the controller is capable of identifying all 7 specified grasps and gestures with 93% accuracy, and is successful in achieving several real-life tasks in a real world setting.**

I. Introduction

Commercial electromyography (EMG) controlled upper limb prostheses can cost from 25,000 to 75,000 USD [1]. Such a high cost means advanced prosthetic hands remain inaccessible to the majority of amputees. Existing commercially successful advanced myoelectric hands, such as the bebionic hand [2] and the Hero Arm [3], despite being controlled based on surface EMG (sEMG) signals, are cumbersome and unintuitive to use [4]. Often external buttons or other kinds of inputs are used in conjunction to enable more specialised control [3]. For example, the button is used to scroll through and select from a set of grasps/gestures, then the sEMG signal triggers the selected grasp. There is no or minimal classification on the EMG signal in these solutions, thus the same or a couple of muscle movements are used for various grasps.

The topic of EMG signal classification has been tackled in research for many years. Historically, traditional signal processing and statistical methods were unsuccessful, either having a low accuracy, or could only classify a couple of grasps [5]. Classical machine learning methods such as Support Vector Machines (SVMs) [6], [7], [8], Hidden Markov Models [7], and K-Nearest-Neighbours [9] achieved more satisfactory, but still limited, results. More recently, thanks to developments in deep learning, a lot of research progress has been made in the area. Convolutional neural networks (CNNs) have been used to classify EMG signals [10], [11]. Moreover, since voice and EMG signals share some similar characteristics by both being a continuous analog signal, researchers have also tried to apply speech recognition techniques for EMG signal classification. Examples include long-short term memory (LSTM) [12], [13], [14], [15], gated recurrent units (GRUs) [14], [15], and attention mechanism [15], [16]. However, most of these approaches are still far from commercial viability due to limited training data, lack of real-time performance [17], computational limitations in embedded systems, and high cost.

Embedded microcontrollers (MCUs) historically have not been powerful enough to satisfy the high computational needs of various deep learning techniques for EMG classification. Researchers have then relied on more traditional methods like finite state machines [18], linear discriminant

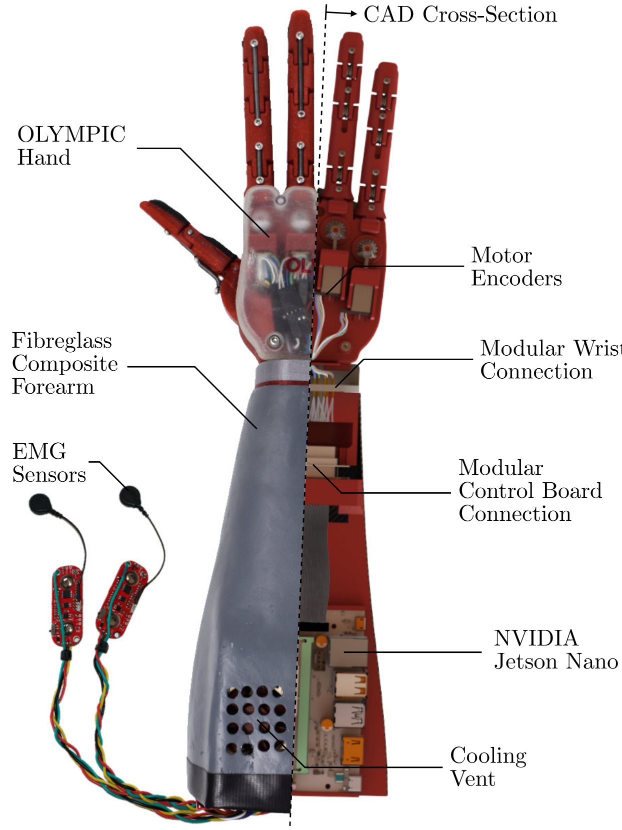

Fig. 1. The improved OLYMPIC prosthetic hand used in this work with embedded control system in the forearm. A CAD cross-section illustrates the internal structure and components of the prosthetic.

Zeyu Yang, Angus B. Clark, Digby Chappell, and Nicolas Rojas are with the REDS Lab, Dyson School of Design Engineering, Imperial College London, 25 Exhibition Road, London, SW7 2DB, UK (zeyu.yang18, a.clark17, d.chappell19, n.rojas)@imperial.ac.uk

Digby Chappell was supported by the UKRI CDT in AI for Healthcare http://ai4health.io (Grant No. EP/S023283/1).

analysis and multi layer perception [19], Kernel Fisher discriminant analysis [20], and SVM [21]. But recently, thanks to the improvements in computing power of MCUs, researchers have been able to implement embedded deep learning-based EMG classifiers and analyse its real-time performance. For example, Tam et al. [4] implemented a CNN based classifier on a Jetson Nano MCU, and Zanghieri et al. [22] implemented a temporal convolutional network based on a GAP8 MCU. Nguyen et al. [23] implemented a GRU based classifier on a Jetson Nano MCU, and additionally tested the classifier on a prosthetic hand to verify its practicability in a clinical/real-world scenario.

Despite this progress, among all of these embedded solutions, none of them train their neural networks on the embedded processor. The need for a powerful computer for the setup of the prosthetic hand adds to the cost and complexity of the system, and overall jeopardises suitability of use by different users. On top of this, while achieving great results, the EMG sensors used in previous works are either too expensive to afford [4] or too complicated to setup [22], sometimes even requiring surgery [23]. To allow advanced myoelectric hands become more wide spread among amputees, prosthetic hands that can achieve sophisticated control with cheaper, simpler sensors have to be developed. The challenge is that using such sensors typically results in noisy contaminated signals, making classification more troublesome.

This work focuses on the development of a low-cost, fully embedded sEMG-based prosthetic hand control platform using deep learning techniques, which address some of the limitations of previous solutions described above and the problems associated with the use of low-cost EMG sensors. Summarised in Fig. 1, the proposed control platform is implemented and tested on an enhanced version of the OLYMPIC prosthetic hand—improvements respect to the original version [24] include the addition of position and torque sensors, a redesigned wrist connection, and a forearm socket housing the circuits and MCU. The developed controller is fully trained on the MCU, requiring only two low-cost electrodes, and uses a data acquisition and augmentation method that minimises the data required to train the prosthetic hand controller.

II. Hardware Integration

The improved OLYMPIC hand with additional sensors and embedded control system in the added forearm is shown in Fig. 1. The hand was modified as such to enable the desired multi-grasp, EMG-based control with a minimum number of low-cost sensors in a small embedded solution. All hardware of the enhanced version can be manufactured with standard soldering iron and 3D printer, with a total cost of £384.6. Details of the modifications introduced are discussed next.

A. Electrical

The NVIDIA Jetson Nano 2GB was chosen as the MCU thanks to its powerful GPU (graphics processing unit) and support for popular deep learning platforms like Tensorflow.

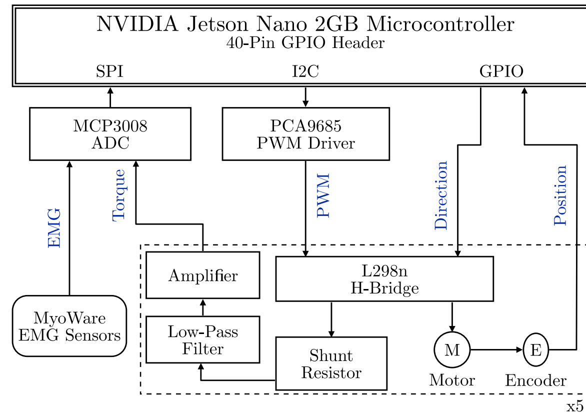

Fig. 2. Block diagram of the electronics for motor control and sensor feedback

It also had a 40-pin GPIO header, which enables digital communication with other peripheral devices. However, it does not have any analog input capability, and also lacks a hardware PWM generator.

The EMG sensor and motor torque signals, both new additions to the prosthetic to provide input from the users' residual limb and improved control of the fingers respectively, required analog input capability as both are analog signals. An MCP3008 analog to digital converter (ADC) with 8 input channels was used to provide this capability, enough for 2 EMG and 5 motor torque signals.

The ability to produce PWM signals was also crucial for driving the motor to achieve precise position and torque control. To provide these a PCA9685 PWM driver was selected. To drive the 5 motors of each finger, three L298n dual H-bridge were used.

Pololu Magnetic Encoders were chosen to provide position feedback of each motor to the Jetson Nano. The encoders are very compact, and can be directly soldered to the back of the existing motor with no additional modification to the hand. The encoder provides 6 ticks per rotation, and the existing motor takes 400 rotations to drive the finger from fully open to closed. Therefore, each finger can be controlled to a resolution of approximately 2400 positions.

Torque of a brushed DC motor is proportional to its current draw, thus to measure torque a simple method is to measure the current draw. To achieve this, a shunt resistor (0.1 Ohm) is placed in series with the motor and the voltage across the resistor is measured, which allows us to calculate the current according to Ohm's law. The voltage signal is then passed through a RC (resistor-capacitor) low pass filter (500 Hz cut off) to remove switching frequency of the PWM, and a non-inverting amplifier (gain of 75), before final signal input to the MCP3008 ADC. The overall electrical system diagram is shown in Fig. 2.

B. Control

With electrical current and position sensing capabilities, each motor in the hand can be controlled to either a reference current (torque) or a reference position. Both torque and position control are achieved by adding proportional integral

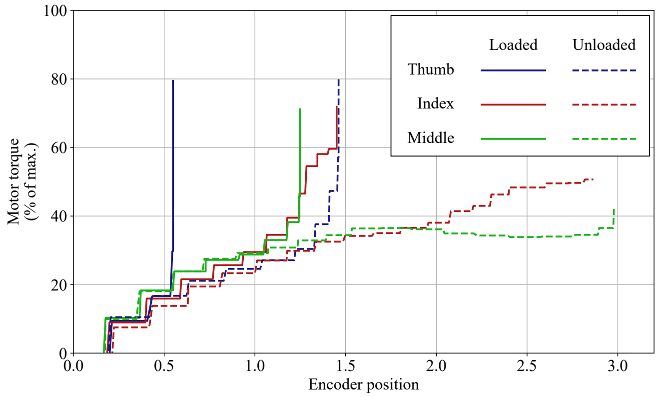

Fig. 3. Motor torque vs encoder position during tripod grip, solid line - loaded (grasping an object), dashed line - unloaded (freely closing). From top to bottom: thumb, index and middle finger.

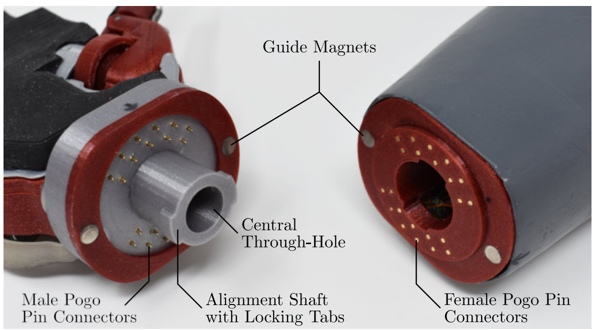

Fig. 4. Modular wrist design, highlighting the significantly increased number of pogo pins (original 10, new 22) to provide the necessary control signals.

(PI) controllers, and both output direction and PWM duty-cycle values to control the motors.

Monitoring the current and position of each motor means contacts with each finger can be detected. Fig. 3 shows the motor torque and position of the thumb, index finger, and middle finger of the hand during a tripod grip. As seen, the torque of each finger of the motor spikes when contact is made, either with an object or the palm of the hand, and the motor is unable to rotate, so encoder position halts. This enables the grasping of more delicate objects.

C. Mechanical

With the additions of new sensors to the hand, the modular wrist no longer had enough electrical connections to support all of the motors and their signals. A new wrist was designed and constructed using pogo-pins at the modular interface to ensure a reliable electrical connection. The pogo-pins were rotated to be co-linear with the forearm to enable the close spacing of the pins, and female pogo-pins were positioned in the forearm socket to increase the connection quality. To lock the hand in place once connected, a rotary shaft with locking tabs was utilised. Inserted at a 90° offset, the hand could then be rotated to its correct position where it would then be locked in place due to a positioned seat for the locking tabs. Additional guide magnets were introduced to the internal faces of the modular wrist to further increase this locking connection, and to increase the locking 'click' felt by the user when assembling the hand. The implemented wrist and its components can be viewed in Fig. 4.

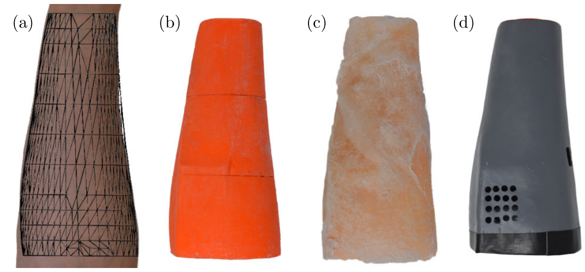

Fig. 5. The moulding process for the OLYMPIC forearm. **(a)** Forearm with overlaid scanned 3D model, **(b)** 3D printed forearm mould, **(c)** Fibreglass and epoxy resin composite shell, and **(d)** finished painted socket with installed modular wrist and port and fan cut-outs.

A socket was constructed to house the electronics, using the forearm of the original hand model participant. First, a scan of the forearm was captured and area desired extracted. This was modified to include port and fan cut outs, then 3D printed as a mould. A fibreglass and epoxy resin composite lay-up was formed using this mould, which was then finished aesthetically and functionally to include the electronics and modular wrist. The manufacturing process of the socket can be seen in Fig. 5. Following both of these modifications, the mass of the hand and forearm combined is now 1.426 kg.

III. Deep Learning Grasp Classification

A. Targeted Grasps and Gestures

The muscle activity corresponding to 7 different grasps and gestures are targeted for classification: Power Grasp, Tool Grasp, Tripod, Lateral Pinch, Index Finger Point, Open Palm, and relaxed, selected based on the GRASP taxonomy [25]. As shown in Fig. 6, the set consists of a balanced mixture of precision grasps, power grasps, and gestures, thus provides a foundation to perform a variety of day to day tasks. In addition, an idle state is also needed for the system to recognise when there is no grasping action at all.

B. Classification Method

Instead of trying to recognise the EMG signal when a grasp is active (steady state), we target a change of state (hold action) and a common release action that separates different grasp types. The neural network will try to recognise the hold action for each grasp, and all grasps share the release action to return the hand to the relaxed position.

The hold action of a grasp consists of 2 sections: the movement of fingers to meet in a particular position; and the static force applied on the fingers when holding an object. However, a gesture only has the finger movement section, since there is no need to apply a constant force on the fingers to hold an object. By using this insight, we can then discern between grasps and gestures, being able to classify a larger set of grasps and gestures with only 2 EMG channels. The muscle activity pattern recorded for each action is congruent to the desired motion of the prosthesis, allowing us to achieve intuitive control.

Fig. 6. The default, grasping, and gesture configurations implemented on the OLYMPIC hand. **(a)** Relax (default position), **(b)** Open Palm and **(c)** Index Finger Point (gestures), **(d)** Tripod Grasp, **(e)** Power Grasp, **(f)** Tool Grasp, and **(g)** Lateral Pinch (grasps).

The ML problem here is multi-class classification. The classification uses a window method, in which the input of the classifier is a 1.5 second snapshot of EMG signal, and tries to determine which class the signal belongs to. The 1.5s duration ensures the EMG signal of the previous grasp, transition movement, and the new grasp to be recorded, as one of the goal is to eliminate the need of labelling the data afterwards.

C. EMG Sensor and Signal Preprocessing

The MyoWare Muscle Sensor was chosen for several of its advantages. With a small profile of 52 mm by 20 mm, the sensor is lightweight and compact for a portable prosthetic hand. The sensor requires a single 3.3 V or 5 V power supply, and only draws 10 mA, which are all suitable for using battery power. The sensor also pre-processes the raw EMG signal in an analogue fashion by its on-board electronics, which saves computing power on the microcontroller. The raw EMG signal from the 2 electrodes feed into a precision differential amplifier, followed by 2 amplifiers, a rectifier, and an integrator. The output is the envelope of the raw EMG signal, which directly inputs to the Jetson Nano MCU. The targeted areas of the forearm are anterior and posterior regions, corresponding to the bulk muscle responsible for flexion and extension of the fingers of the hand, respectively.

Crossing the analogue and digital domain, the Jetson Nano samples the incoming EMG envelope at 200 Hz. The raw EMG signal used has a bandwidth of 500 Hz [26]. The integrator circuit on the MyoWare sensor cutoff the output envelope signal bandwidth at 12.5 Hz. According to the Nyquist sampling theorem, we need to sample at 25 Hz in this case to ensure no information will be lost. Sampling at 200 Hz is well above the minimal requirement and ensures no information is lost.

D. Training Data

30 repetitions for each of the 8 classes (7 grasps + Idle state) were collected as the training data. The user was be asked to perform the grasping action within the 1.5 second time window. Data augmentation is used to generate more training data in order to improve the generalisation ability of the model. Three data augmentation strategies, inspired from computer vision/image recognition, are used: random shift, random zoom, and random contrast. These strategies used to deal with images of an object under different camera angle or lighting condition. In this case they each corresponding to the start time, speed, and force variations of the grasping action of the user during the data collections.

E. Neural Network Architecture

The first layer of the CNN is the data augmentation layer, consisting of the aforementioned 3 data augmentation strategies. There are 2 advantages of applying data augmentation as an input layer over directly producing modified copies of original data: it saves storage space on the micro-controller, and exploits the computing power of GPU during training. Following is a convolutional layer of 8 filters with a kernel size of 2 * 30. The size of the filter extracts spatial information of the 2 input channels, and temporal information of 30 consecutive samples. The features extracted by the convolutional layer are processed by 2 dense layers with sizes of 128 and 64 respectively. ReLU was used as the activation function between all the convolutional and dense layers. Finally, a dense layer of size 8 and a softmax activation function completes the classification of the 8 classes.

F. Roll-out

The EMG data, sampled at 200 Hz, is input to a 300 sample FIFO, corresponding to the 1.5 second window. The trained neural network is inferred every 20 new samples of the EMG signal, i.e. a new prediction is generated every 0.1 seconds. The predictions feed into a decision making chain. First, the inference must have a confidence greater than 70% in order for the prediction to be valid. Next, 3 consecutive identical predictions are needed. Then, the prediction needs to be feasible, i.e. a grasp needs to be unlocked by entering the relax state before another grasp. If the 3 criterion are all satisfied, the result will be sent to the motor controller to perform the actions with the OLYMPIC hand. Finally, predictions of the following second are blocked to avoid post-grasping-action signal leading to incorrect results. The overall signal processing flow can be seen in Fig. 7.

IV. Evaluation

A. Experiment 1 - Data collection and training

First, we would like to understand the time needed for the data collection procedure. We also would like to verify

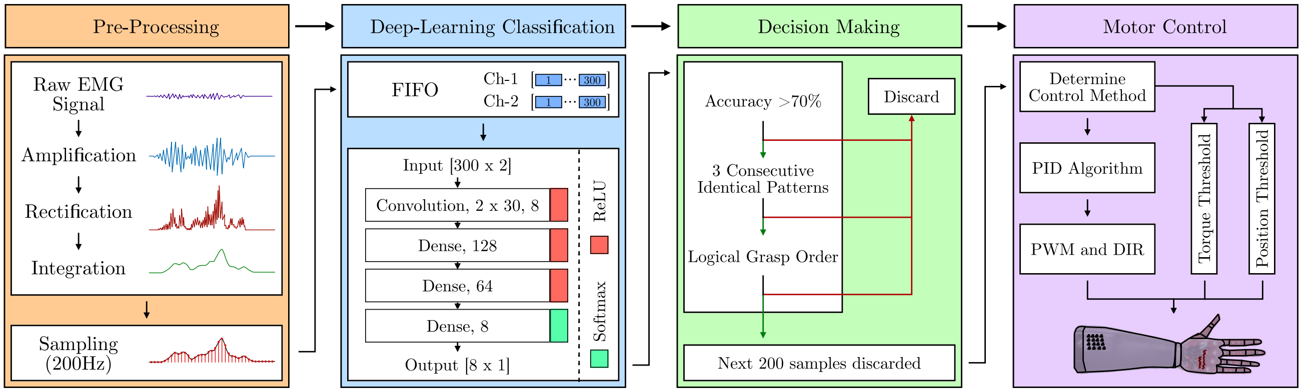

Fig. 7. Signal processing flow showing key stages of Pre-processing, Deep Learning Classification, Decision Making, and Motor Control

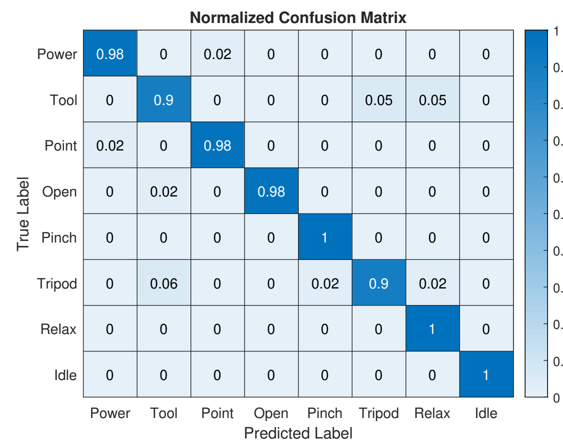

Fig. 8. Average confusion matrix of the 5 trained participant classifiers making predictions on their respective validation datasets.

the categorisation ability of the classifier by analysing the validation accuracy and the confusion matrix. 5 non-disabled participants were asked to perform 30 repetitions of each grasp in turn. A 4s interval before and between the 30 repetitions was used to allow the participant to rest and prepare for the next repetition. The participants were asked to perform the grasp or gesture action approximately 0.5s into the recording period, which ensures the whole action is recorded. The participants were asked to perform the grasp and gestures exactly as they normally do in real life, and try to be consistent for all the repetitions. The real-time EMG signal is displayed in front of the participants to help with the consistency. For the relax state, the participants are asked to shake their hands several times, as they would to relax a stiff tired muscle. The entire data collection process took less than 30 minutes to complete for each participant.

The collected data were randomly split into training and validation sets at a ratio of 7:3. The model trained with a categorical cross-entropy loss function, and Adam optimiser with learning rate of 1e-4. The model trained for 30 epochs with a batch size of 8.

B. Experiment 2 - Roll-out of the classifier

The performance of the classifier in terms of both accuracy and latency was tested on a real-time continuous EMG signal. Participants performed 90 grasps in total each in a random pre-determined order, with 15 repetitions for the 6 targeted grasps and gestures. Participants were asked to perform the grasps in the order, and then release the grasp to the relax state, at a pace they feel comfortable. Upon a wrong prediction, the participants needed to redo the motion until a correct prediction was made.

The Jetson Nano ran the real-time classification, with the 10 Hz inference frequency, continuously outputting the most up-to-date prediction. The number of tries it took to get a correct prediction, and the time delay between the end of hand movement to the correct prediction were recorded.

V. Results

A. Experiment 1 - Training result of the neural network

On average of the five participants, the validation accuracy of the classifier converges at around 20 epochs, at a value of 96.68%, with total training time of 54 seconds. Fig. 8 shows the average confusion matrix of the classifiers trained on each participant, making predictions on their respective validation datasets. As seen, all of the grasps have a sensitivity greater than 90%, with lateral pinch and idle showing greatest sensitivity and specificity. Tripod and tool grips show the worst performance, although sensitivity is still above 90%.

B. Experiment 2 - Roll out of the classifier

During roll out, participants achieved at 79.11% accuracy on first attempt at performing each action, and 96.22% accuracy with a second try. However, more interestingly, we observed an increase in accuracy throughout the 90 repetitions. Shown in Fig. 9, if we look at the accuracy over a 30 repetitions window through the 90 repetitions, we see the one try average accuracy started at 66%, and gradually increased to 93% by the end.

The average latency of a correct grasp is at 1.097 seconds, with a standard deviation of 0.307 seconds. The value is expected due to the setup protocol, as the participants were asked to perform the grasping action 0.5 seconds into the 1.5 second window. The total time for the participants to finish the 90 repetitions was no longer than 20 minutes.

Finally, the controller was demonstrated in practice by performing daily tasks with each of the targeted motions, as shown in Fig. 10. During these tasks, the prosthesis was

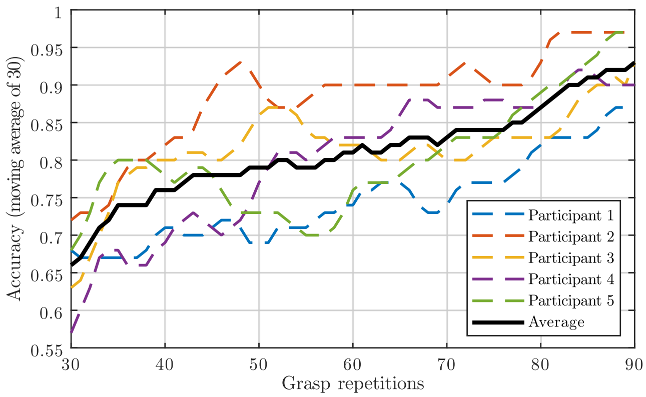

Fig. 9. Average prediction accuracy over 90 test repetitions, smoothed with a 30 sample moving average.

mounted directly below a non-disabled user's forearm with an offset block.

VI. Discussion

We have achieved a state-of-the-art classification accuracy of 93% in the real-time test, with just two hobbyist level surface EMG sensors. The window technique has allowed the CNN architecture to be very simple, thus can be trained locally on the microcontroller. We also don't need to manually label the data, partially thanks to the automated data collection procedure, but also because we do not need to align the grasping action with time. Doing all the computing locally and without the need of a professional to label the data makes the setup and use of the hand more versatile.

However, there is a drawback of the method we used, which is the latency between the grasping action and a correct prediction is rather long, at 1.097 seconds. To make a correct prediction, the classifier must wait for the EMG signal to propagate through the FIFO to the correct time stamp. This problem arises because we rely on a larger time window to acquire a more detailed input, rather than more sensors. One method to reduce this problem is to decrease the window size, i.e. from 1.5 to 1 second, but it would more difficult to ensure the entire grasping action will be captured in the window.

For data collection, and therefore for calibration, the amount of training data needed is minimal. The size of all the training data are less than 2MB, which is suitable for an embedded micro-controller. More importantly, the time needed to collect all the training data is only 30 minutes; together with the automated guidance, the data collection procedure is quick and user friendly. Thanks to the simple CNN architecture, the training time of the CNN is less than 60 s. This makes the controller suitable for use for real prosthetic hand users, since all data used to train the controller is recorded directly from the user. This allows the controller to account for heterogeneity in muscle activity post-amputation, and means that users would be able to record muscle activity patterns that are most suited to their residual limb during rehabilitation.

The intuitive control leads to a fast learning curve. After a 20 minute practice period of 90 repetitions, the participants

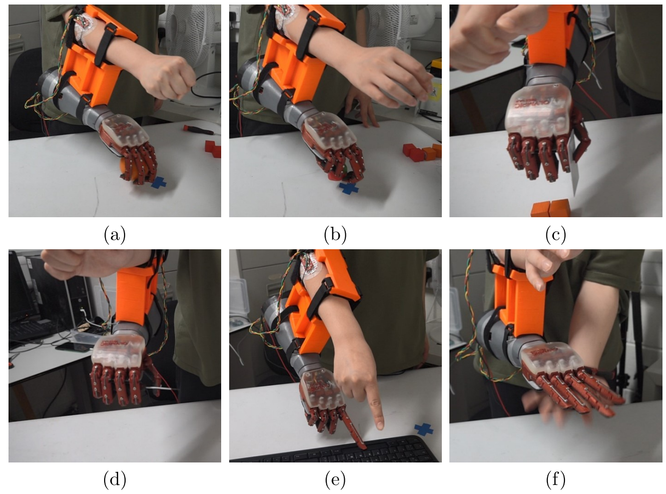

Fig. 10. Demonstration of the hand in a real-life setting: **(a)** Power Grasp: An orange, **(b)** Tripod: A strawberry, **(c)** Lateral Pinch: A card, **(d)** Tool Grasp: A screw driver, **(e)** Index Finger Point: Pressing Keyboard, **(f)** Open Palm: Applaud,

were able to achieve a state-of-the-art 93% accuracy, which has been achieved previously with more precise sensors in greater quantity [4], [22], [23]. During this practice period, users perform actions in the context of the controller; each action must be followed by a relax action, which, observationally, accounts for much of the learning. The total time needed to setup the hand for a new user can be completed in less than one hour. With the automated guidance, the entire process can be done without specialised supervision from an instructor. The quick and easy setup, together with a low cost of manufacture, pave the way to make advanced myoelectric hands accessible to more people.

VII. Conclusions & Future Work

In this paper we have presented a low cost embedded sEMG based intuitive grasping prosthetic hand control system. All the electrical and mechanical hardware are compact and can be fit inside a forearm socket. Data collection, CNN training and classification, and motor control all runs on the embedded Jetson Nano 2GB micro-controller. Experiments showed the entire system is easy to setup, achieves state-of-the-art classification accuracy, and successfully achieves several real-life tasks, despite only using 2 electrodes.

Future work will investigate reducing latency of classification and integrating the sensing capabilities of the hand with the myoelectric control system. The sensed torque and positing information could be useful for a haptic feedback system. Although currently powered by an external power supply, the system was also designed with battery powered usage in mind. The whole system works at 5V, and draws a maximum of 4A when motors are heavily loaded. There is also space left in the forearm socket where rechargeable batteries can be placed. If the weight of the battery is significant, or does not have enough power density to power the hand for extended periods, an option could be to place the battery in a waist bag.

REFERENCES

[1] L. Resnik, M. R. Meucci, S. Lieberman-Klinger, C. Fantini, D. L. Kelty, R. Disla, and N. Sasson, "Advanced upper limb prosthetic devices: implications for upper limb prosthetic rehabilitation," *Archives of physical medicine and rehabilitation*, vol. 93, no. 4, pp. 710–717, 2012.

[2] "Bebionic hand." [Online]. Available: https://www.ottobock.co.uk/prosthetics/upper_limbs_prosthetics/product-systems/bebionic-hand/

[3] "Meet the hero arm," Aug 2021. [Online]. Available: https://openbionics.com/hero-arm/

[4] S. Tam, M. Boukadoum, A. Campeau-Lecours, and B. Gosselin, "A fully embedded adaptive real-time hand gesture classifier leveraging hd-semg and deep learning," *IEEE transactions on biomedical circuits and systems*, vol. 14, no. 2, pp. 232–243, 2019.

[5] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 50, no. 7, pp. 848–854, 2003.

[6] A. Alkan and M. Günay, "Identification of emg signals using discriminant analysis and svm classifier," *Expert systems with Applications*, vol. 39, no. 1, pp. 44–47, 2012.

[7] M. Rossi, S. Benatti, E. Farella, and L. Benini, "Hybrid emg classifier based on hmm and svm for hand gesture recognition in prosthetics," in *2015 IEEE International Conference on Industrial Technology (ICIT)*. IEEE, 2015, pp. 1700–1705.

[8] M. Tavakoli, C. Benussi, P. A. Lopes, L. B. Osorio, and A. T. de Almeida, "Robust hand gesture recognition with a double channel surface emg wearable armband and svm classifier," *Biomedical Signal Processing and Control*, vol. 46, pp. 121–130, 2018.

[9] M. Haris, P. Chakraborty, and B. V. Rao, "Emg signal based finger movement recognition for prosthetic hand control," in *2015 Communication, Control and Intelligent Systems (CCIS)*. IEEE, 2015, pp. 194–198.

[10] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers in neurorobotics*, vol. 10, p. 9, 2016.

[11] J. A. George, M. R. Brinton, C. C. Duncan, D. T. Hutchinson, and G. A. Clark, "Improved training paradigms and motor-decode algorithms: results from intact individuals and a recent transradial amputee with prior complex regional pain syndrome," in *2018 40th annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 3782–3787.

[12] P. Xia, J. Hu, and Y. Peng, "Emg-based estimation of limb movement using deep learning with recurrent convolutional neural networks," *Artificial organs*, vol. 42, no. 5, pp. E67–E77, 2018.

[13] M. Jabbari, R. N. Khushaba, and K. Nazarpour, "Emg-based hand gesture classification with long short-term memory deep recurrent neural networks," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2020, pp. 3302–3305.

[14] M. Simão, P. Neto, and O. Gibaru, "Emg-based online classification of gestures with recurrent neural networks," *Pattern Recognition Letters*, vol. 128, pp. 45–51, 2019.

[15] A. Samadani, "Gated recurrent neural networks for emg-based hand gesture classification. a comparative study," in *2018 40th annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 1–4.

[16] 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.

[17] N. Parajuli, N. Sreenivasan, P. Bifulco, M. Cesarelli, S. Savino, V. Niola, D. Esposito, T. J. Hamilton, G. R. Naik, U. Gunawardana, *et al.*, "Real-time emg based pattern recognition control for hand prostheses: a review on existing methods, challenges and future implementation," *Sensors*, vol. 19, no. 20, p. 4596, 2019.

[18] H. Wu, M. Dyson, and K. Nazarpour, "Arduino-based embedded system for myoelectric hand prostheses," in *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2020, pp. 1–4.

[19] S. A. Raurale, J. McAllister, and J. M. del Rincon, "Real-time embedded emg signal analysis for wrist-hand pose identification," *IEEE Transactions on Signal Processing*, vol. 68, pp. 2713–2723, 2020.

[20] S. Raurale, J. McAllister, and J. M. del Rincon, "Emg wrist-hand motion recognition system for real-time embedded platform," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1523–1527.

[21] F. Ryser, T. Bützer, J. P. Held, O. Lambercy, and R. Gassert, "Fully embedded myoelectric control for a wearable robotic hand orthosis," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2017, pp. 615–621.

[22] 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*, vol. 14, no. 2, pp. 244–256, 2019.

[23] A. T. Nguyen, M. W. Drealan, D. K. Luu, M. Jiang, J. Xu, J. Cheng, Q. Zhao, E. W. Keefer, and Z. Yang, "A portable, self-contained neuroprosthetic hand with deep learning-based finger control," *Journal of neural engineering*, vol. 18, no. 5, p. 056051, 2021.

[24] L. Liow, A. B. Clark, and N. Rojas, "Olympic: a modular, tendon-driven prosthetic hand with novel finger and wrist coupling mechanisms," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 299–306, 2019.

[25] T. Feix, J. Romero, H.-B. Schmiedmayer, A. M. Dollar, and D. Kragic, "The grasp taxonomy of human grasp types," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 66–77, 2016.

[26] L. Shaw and S. Bagha, "Online emg signal analysis for diagnosis of neuromuscular diseases by using pca and pnn," *International Journal of Engineering Science and Technology (IJEST)*, vol. 4, no. 10, pp. 4453–4459, 2012.