Recurrent Convolutional Neural Networks as an Approach to Position-Aware Myoelectric Prosthesis Control

Heather E. Williams, Ahmed W. Shehata, Michael R. Dawson, Erik Scheme, Jacqueline S. Hebert, and Patrick M. Pilarski

Abstract*—*Objective: **Persons with normal arm function can perform complex wrist and hand movements over a wide range of limb positions. However, for those with transradial amputation who use myoelectric prostheses, control across multiple limb positions can be challenging, frustrating, and can increase the likelihood of device abandonment. In response, the goal of this research was to investigate convolutional neural network (RCNN)-based position-aware myoelectric prosthesis control strategies. *Methods:* Surface electromyographic (EMG) and inertial measurement unit (IMU) signals, obtained from 16 non-disabled participants wearing two Myo armbands, served as inputs to RCNN classification and regression models. Such models predicted movements (wrist flexion/extension and forearm pronation/supination), based on a multi-limb-position training routine. RCNN classifiers and RCNN regressors were compared to linear discriminant analysis (LDA) classifiers and support vector regression (SVR) regressors, respectively. Outcomes were examined to determine whether RCNN-based control strategies could yield accurate movement predictions, while using the fewest number of available Myo armband data streams. *Results:* An RCNN classifier (trained with forearm EMG data, and forearm and upper arm IMU data) predicted movements with 99.00% accuracy (versus the LDA's 97.67%). An RCNN regressor (trained with forearm EMG and IMU data) predicted movements with R^2 values of 84.93% for wrist flexion/extension and 84.97% for forearm pronation/supination (versus the SVR's 77.26% and 60.73%, respectively). The control strategies that employed these models required fewer than all available data streams. *Conclusion:* RCNN-based control strategies offer novel means of mitigating limb position challenges. *Significance:* This research furthers the development of improved position-aware myoelectric prosthesis control.**

***Index Terms*—Recurrent convolutional neural networks, electromyography, inertial measurement units, limb position effect, myoelectric, pattern recognition, prosthesis, prosthesis control.**

I. Introduction

MYOELECTRIC prostheses are designed to restore lost upper limb motor function for individuals with amputation. Recreating the coordinated movements of a natural human wrist and hand, however, remains a challenge for those with transradial amputations. In response, researchers have developed control strategies that use pattern recognition models to predict and execute a user's movement intent [1]. Electromyography (EMG) is currently the most commonly used input source for prosthesis control [2], whereby EMG signals generated by muscle contractions in a user's residual limb are captured by electrodes embedded in a device socket. Despite yielding reliable device movements in research environments, precise decoding of movement intent from EMG signals can be unreliable when a wide range of limb positions are introduced by users during daily activities [3].

This significant challenge to myoelectric prosthesis control is known as the "limb position effect" [4]. Often, detected surface EMG control signals are altered when a user's limb is in a position different from that in which the prosthesis controller was trained (usually a comfortable, low position) [4]. Resulting EMG signal variations can cause prosthesis control to degrade and unexpected prosthetic wrist and hand movements to occur. Researchers have investigated various methods of mitigating this problem, including the use of intramuscular electrodes [5], [6], high-density surface electrode arrays [7], [8], and wearable limb position sensors [9]–[14]. However, a reliable and practical position-aware control solution has yet to be found. As such, continued research is required.

Manuscript received June 30, 2021; revised November 12, 2021 and December 20, 2021; accepted December 24, 2021. Date of publication January 5, 2022; date of current version June 20, 2022. This work was supported in part by the Natural Science and Engineering Research Council (NSERC) PGS-D, in part by the Alberta Innovates, Alberta Advanced Education, the Sensory Motor Adaptive Rehabilitation Technology (SMART) Network, the Smart Technology (ST) Innovations at the University of Alberta, NSERC DG RGPIN-2015-03646, NSERC DG RGPIN-2019-05961, NSERC DG RGPIN-2020-04776, in part by the Alberta Machine Intelligence Institute (Amii), and in part by the Canada Research Chairs Program. *(Correspondence author: Heather E. Williams.)*

Heather E. Williams is with the Department of Biomedical Engineering, Faculty of Engineering, University of Alberta, Canada (e-mail: heather.williams@ualberta.ca).

Ahmed W. Shehata, Michael R. Dawson, and Jacqueline S. Hebert are with the Division of Physical Medicine and Rehabilitation, Department of Medicine, University of Alberta, Canada.

Erik Scheme is with the Institute of Biomedical Engineering, University of New Brunswick, Canada.

Patrick M. Pilarski is with the Division of Physical Medicine and Rehabilitation, Department of Medicine, University of Alberta, Canada, and also with the Alberta Machine Intelligence Institute (Amii), Canada.

This article has supplementary downloadable material available at https://doi.org/10.1109/TBME.2022.3140269, provided by the authors.

Digital Object Identifier 10.1109/TBME.2022.3140269

Various pattern recognition approaches have been explored to address the limb position effect on end effector control [9]–[13], [15]–[23]. Broadly, pattern recognition approaches have included Statistical Models and Neural Networks (including deep learning), each of which can use either classification or regression techniques [3], [22]. Typically, classification models (classifiers) and regression models (regressors) map EMG signal features, which are extracted from raw EMG data, to predict intended end effector movements [24], [25]. Classifiers map signal features to one of a *discrete* set of known classes (categories) of degrees of freedom (DOFs), offer control over multiple DOFs, but do not provide proportional control over device movement velocity or simultaneous control over multiple DOFs [2]. Conversely, regressors can map signal features to *continuous* velocity values for each DOF (proportional to input signal strength), offer simultaneous control of separate DOFs [2], but tend to be less robust than classifiers due to the increased complexity of their predictions [24]. Whether classification or regression is used for control, all models require a device training routine to be undertaken by the user, to inform pattern learning [26]. Although more training data generally yields stronger models, long training routines are cumbersome for the user [2], [27]. Overall, not only does the chosen pattern recognition model influence the resulting device control, the duration of its required training routine, the time needed to train the model and make predictions, and the complexity of the model algorithm are also considerations.

Statistical models apply probability theory to learn patterns in data and are currently more often employed in position-aware prosthesis control research than deep learning neural network alternatives [3]. Some researchers have collected EMG data across multiple limb positions to inform statistical classifiers [9], [12], [13], [19], [20], while others have added positional information (quaternions or accelerometer data) to take limb orientation into account [9]–[13]. Statistical regressors have not been as extensively explored as classifiers in device control literature [24]. Nevertheless, both statistical classifiers and regressors offer the benefits of being straightforward to implement and having low computational costs [25]. Each such model requires representative feature extraction from EMG signals. This means that assumptions must be made regarding which features best inform movement prediction [25], [28], [29]. To avoid possibly making ill-informed or erroneous assumptions, researchers have also begun to explore the benefits of neural network methods for EMG-based control [17], [30].

Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) may yield improved prosthetic movement prediction accuracies over statistical approaches, given the advantages that they offer. The first such advantage is that CNNs can predict intended movements from raw EMG signals (rather than from extracted features) [31]–[33]. This means that, given sufficient data, new features can be automatically learned, thereby avoiding the need for feature engineering. Another advantage is that CNNs offer the ability to combine a high volume of data from multiple sensors [30], [34]. This suggests that CNNs may prove to be effective towards learning the complex features of combined EMG and inertial measurement unit (IMU) signals across multiple limb positions. Furthermore, as time-domain features are commonly used for prosthesis control [25], recurrent neural networks (RNNs), which leverage the temporal behaviour of signals [35], might also be beneficial towards solving the limb position effect. Given that recurrent convolutional neural networks (RCNNs) can harness the collective advantages presented by CNNs and RNNs, they too offer a promising research direction for improving device control.

Compared to statistical approaches, few studies have explored using RCNN or CNN-based models for prosthesis control. Xia *et al.* examined the use of RCNNs, with raw EMG data, for the prediction of shoulder position (irrespective of end effector function) [35]. Their proposed model yielded higher predictive accuracy than an alternative statistical regressor (support vector regression, SVR). Amongst other things, this research demonstrated that an RCNN can indeed learn features from raw EMG data to inform limb position. Ameri *et al.* confirmed that a CNN can be used with raw EMG data to predict wrist movement, and yielded offline and real-time performances better than those of an SVR [31]. More recently, Bao *et al.* used an RCNN to extract EMG features for the prediction of wrist motion [36]. This solution outperformed CNN-only approaches during complex wrist movements, and further supports the predictive potential of RCNN models. In 2018, Phinyomark and Scheme reviewed the potential for developing more advanced applications of EMG pattern recognition using deep learning approaches [30]. Collectively, the abovementioned studies recommended the continued pursuit of deep learning, including combining CNNs with RNNs, further optimizing model architectures, conducting more online testing of such models, or testing with larger datasets [31], [35], [36].

The goal of this study was to investigate the novel use of position-aware RCNN-based myoelectric prosthesis control strategies, towards solving the "limb position effect" problem. To this end, this study examined device control strategies that: (1) combined EMG and IMU input data streams to inform prosthesis movements and limb positions, respectively; and (2) used RCNN models to make movement predictions from these data. For each RCNN model under investigation, resulting movement predictions were compared to those of commonly used statistical models, so that potential improvements could be ascertained. The criteria by which position-aware control strategies were evaluated included their movement prediction accuracy, along with the number of EMG and/or IMU data streams that they required. Based on these criteria, this study identified two promising RCNN-based myoelectric prosthesis control strategies that were found to be consistently accurate across multiple limb positions.

II. Methods

A. Participants

A total of 19 participants with no upper-body pathology or recent neurological or musculoskeletal injuries were recruited. The data from 3 participants were incomplete and as such not used for this study. Of the remaining 16 participants, 3 had previous experience with EMG control, all had normal

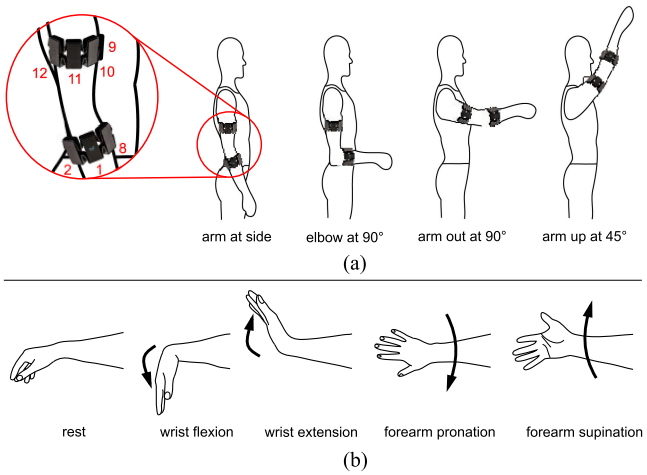

Fig. 1. (a) The placement of each Myo armband, along with limb positions used in data collection, and (b) movements used in data collection.

or corrected to normal vision, 15 were right-handed, 8 were male, and the mean age was 26.4 ± 8.7 years (± 1 standard deviation). Each participant provided written informed consent, as approved by the University of Alberta Health Research Ethics Board (Pro00086557).

B. Experimental Setup

Two Myo gesture control armbands (Thalmic Labs, Kitchener, Canada) were used to collect EMG and IMU data. Each armband contained 8 surface electrodes and an IMU. Each surface electrode collected 1 EMG data stream (sampled at 200 Hz). Each IMU collected 10 limb position data streams (3 accelerometer, 3 gyroscope, and 4 quaternion, all sampled at 50 Hz). Using Myo Connect software, the EMG and IMU data were streamed into Matlab. Hardware and software limitations required that each Myo armband be connected to a separate laptop, so two Lenovo ThinkPad laptops were employed. A custom Matlab script, running on one laptop, captured streamed data from one Myo armband and simultaneously displayed onscreen instructions for a participant to follow. At the same time, another custom script ran on the second laptop to record data from the second Myo armband.

C. Data Collection

Each participant donned two Myo armbands on their self-identified dominant arm, as shown in Fig. 1(A). One was worn on their forearm, with a mean distance of 6.0 ± 1.9 cm distal to the olecranon, and *electrode 1* on the lateral side of their forearm. The second armband was worn on their upper arm, with a mean distance of 12.0 ± 2.4 cm proximal to the olecranon, and *electrode 9* on the anterior side of their upper arm, over the biceps muscle.

Participants followed onscreen instructions, performing various movements in 4 limb positions, as described below.

- *Movements included:* rest (relaxed), wrist flexion, wrist extension, forearm pronation, and forearm supination, as shown in Fig. 1(B). These movements are functionally important for individuals with transradial amputation [37]. Notably, the hand open and close movements were not included in this study, given that wrist flexion and extension can instead be used to control the opening and closing of a *prosthetic* hand [31], [38]–[41]. Similarly, forearm pronation and supination can be used to control *prosthetic* wrist rotation.
- *Limb positions included:* arm at side, elbow bent at 90°, arm out in front at 90°, and arm up at 45° from vertical, as shown in Fig. 1(A).

Data collection consisted of 6 trials: 3 static trials and 3 dynamic trials. Rest time was provided between each trial.

- Trials 1–3 (**static**) required participants to perform various movements (shown in Fig. 1(B)) using sustained isotonic muscle contractions. All movements were held for 5 seconds, separated by 5 seconds of rest. The movements were repeated in each of the 4 limb positions (shown in Fig. 1(A)). Participants were instructed to perform each muscle contraction at a moderate effort that could be sustained for 5 seconds.
- Trials 4–6 (**dynamic**) required participants to perform movements that oscillated either between wrist flexion and extension or forearm pronation and supination. The timing of these oscillations was demonstrated onscreen (5 cycles with a period of 4 seconds). These oscillations were repeated in each of the 4 limb positions (shown in Fig. 1(A)).

D. Data Pre-Processing

The EMG data from each Myo armband were filtered using a high pass filter at 20 Hz (to remove movement artifacts), as well as a notch filter at 60 Hz (to remove electrical noise). Then, the IMU data streams were resampled to 200 Hz using linear interpolation to align them with the corresponding EMG data. The resulting data from the two Myo armbands were synchronized.

The static trials were segmented into movements (rest, wrist flexion, wrist extension, forearm pronation, and forearm supination). For the dynamic trials, target sinusoids were generated to represent movement oscillations. Given that an offset was evident between participants' movements and onscreen oscillations, their sinusoids were corrected as follows: forearm EMG signal peaks were identified and used to fit a sine wave to represent wrist flexion/extension oscillations, whereas forearm IMU signal peaks and valleys (specifically from the accelerometer) were used to fit a sine wave to represent forearm pronation/supination oscillations. The resulting target sinusoids were then used to segment the dynamic trials into movements in each DOF (wrist flexion/extension, forearm pronation/supination).

Next, for the purposes of the RCNN models under investigation and their comparative statistical models, data were segmented further into windows (160-millisecond with a 40-millisecond offset). For the statistical models, time-domain features were then calculated for each EMG or IMU channel, in each window. These included 4 commonly-used EMG features

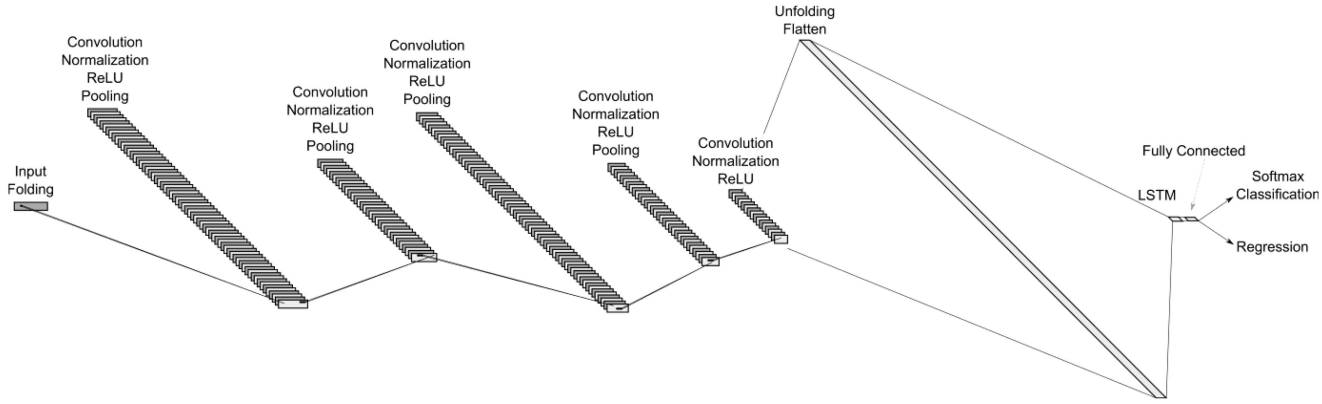

Fig. 2. RCNN architecture: sequence input layer (Input); sequence folding layer (Folding); 4 blocks of convolution, batch normalization (Normalization), rectified linear unit (ReLU), and maximum pooling (Pooling); 1 block of convolution, batch normalization, and ReLU; sequence unfolding layer (Unfolding); flatten layer; long short-term memory (LSTM) layer; fully connected layer; and finally either (1) softmax and classification layers, or (2) a regression layer. Figure made using NN-SVG [43].

(mean absolute value, waveform length, Willison amplitude, and zero crossings [42]) and 1 IMU feature (mean value). For the RCNN models, time-domain features were not calculated and instead, filtered signal data remained in each window.

E. RCNN and Comparative Statistical Models

1) RCNN Models – Architecture: Matlab software was used to implement the RCNN models. Bayesian optimization automatically determined the number of convolution layers, number of filters, filter size, pooling size, and patience required in this study. Optimization was performed in two steps: first, the number of layers along with each hyperparameter being optimized were determined using a broad range of values; thereafter, values were refined using a narrower range (centered at earlier optimized values).

Our resulting RCNN models had architectures that consisted of 27 (classification) or 26 (regression) layers, as shown in Fig. 2 [43]. In these models, a sequence input layer first received and normalized the training data. Then, a sequence folding layer was used, allowing convolution operations to be performed independently on each window. This was followed by a block of 4 layers: a convolution, a batch normalization, a rectified linear unit (ReLU), and a maximum pooling layer. This block was repeated 3 more times. Each of the 4 maximum pooling layers had a pooling size of 1x2. A block of 3 layers followed: a convolution, a batch normalization, and a ReLU layer.

- *For limb position classification:* the optimal number of filters in the convolution layers were determined to be 32, 32, 32, 64, and 64, respectively, and each had a filter windows size of 1x3.
- *For movement classification and regression:* the optimal number of filters in the convolution layers were determined to be 64, 32, 64, 32, and 16, respectively, and each had a filter window size of 1x5.

Subsequent layers included a sequence unfolding layer (to restore the sequence structure), a flatten layer, a long short-term memory (LSTM) layer, and a fully connected layer. Finally, either (1) a softmax layer and classification layer were used, or (2) simply a regression layer was used. To prevent overfitting, a patience parameter was set that triggered early stopping.

2) Comparative Statistical Models: Given that linear discriminant analysis (LDA) is commonly used in prosthesis control research [9]–[11], this study opted to use LDA classifiers for comparisons to both RCNN limb position classifiers and RCNN movement classifiers. The chosen LDA discriminant type was pseudo-linear, since columns of zeros were occasionally present in rest classes for some features (including Willison amplitude and zero crossings).

SVR regressors were used for comparisons to RCNN movement regressors, as per earlier research [31], [35]. The SVR regressors used a linear kernel for input data mapping, given that it yielded the most accurate movement predictions in earlier pilot work (compared to radial basis function and polynomial kernel alternatives). This pilot work was based on EMG and IMU data from multiple limb positions. The kernel scale parameter was automatically optimized by Matlab software and no kernel offset was used.

F. Classification and Regression

This study explored models that predicted limb positions and movements. Three model specifications (S1–S3) were investigated, in addition to a comparative baseline model. All model specifications were substantiated by earlier research:

S1 – Model trained with EMG data from all limb positions [9], [12], [13], [19], [20]

S2 – Model trained with EMG and IMU data from all limb positions [9]–[11], [13]

S3 – Models trained with EMG data at each limb position, with subsequent predictions occurring in a **2-staged sequence**: 1st, a limb position was classified using IMU data; 2nd, a corresponding model (trained at that specified limb position) predicted a wrist movement using EMG data [9]–[12]

Baseline – Model trained with EMG data from arm at side

Note that S1 and the Baseline require only EMG data, whereas S2 and S3 require both EMG and IMU data. Specifications S1–S3 and the Baseline were each implemented using an RCNN classifier, an RCNN regressor, an LDA classifier, and an SVR regressor (16 models total). The training and testing of each model were performed in Matlab using an Intel Core i9-9900K CPU (3.60 GHz).

1) Limb Position Classification: RCNN limb position classifiers were compared to LDA limb position classifiers in this study. The RCNN classifier inputs were signals from each window and the LDA classifier inputs were time-domain features from each window. Both classifiers outputted a predicted limb position class (shown in Fig. 1(A)) for each window. Limb position classifiers were trained with Trials 1–2 (static) data from a participant and subsequently tested using Trial 3 (static) data from that same participant. This approach was motivated by current myoelectric prosthesis use, wherein the user must train their device controller before it can predict movement intent.

Prosthesis control research has shown that the use of numerous data streams (EMG and/or IMU) can result in longer machine learning processing times and/or increased hardware costs [10]. Taking these drawbacks into consideration, the specific data stream *types* that would most accurately inform limb position were initially investigated. Data from both Myo armbands were used in this investigation, with the RCNN and LDA limb position classifiers trained and tested using the following data stream combinations:

- All EMG and IMU data streams from both Myo armbands
- All EMG data streams from both Myo armbands
- All IMU data streams (quaternions, gyroscope, and accelerometer) from both Myo armbands
- Only accelerometer data streams [9], [12], [23] from both Myo armbands

Note that gyroscope and quaternion data streams were not investigated independently. Earlier pilot work revealed that accelerometer data better informed limb position in comparison to gyroscope and/or quaternion data.

2) Movement Classification: RCNN movement classifiers were compared to LDA movement classifiers. As with limb position classification, the RCNN movement classifier inputs were signals from each window and the LDA classifier inputs were time-domain features from each window. Both the RCNN and LDA movement classifiers outputted a predicted movement class (shown in Fig. 1(B)) for each window. Movement classifiers were trained with Trials 1–2 data from a participant, and subsequently tested using Trial 3 data from that same participant.

Movement classifiers were trained, tested, and compared under model specifications S1–S3. The predictive accuracies of these classifiers were compared to those of a baseline classifier (BC), trained with only EMG data collected with each participant's arm at their side (as per standard prosthesis training). Additionally, to minimize the number of data streams necessary for movement classification, each classifier was trained with the following combinations:

- Data (EMG and, when applicable, IMU) from only the forearm Myo armband
- Data from both Myo armbands
- EMG data from the forearm and IMU data from both Myo armbands (when applicable)

3) Movement Regression: RCNN movement regressors were compared to SVR movement regressors. The RCNN and SVR movement regressors used the same inputs as did the RCNN and LDA movement classifiers, respectively. However, the RCNN and SVR regressors outputted *continuous* movement predictions, denoting muscle activation intensity for each DOF (flexion/extension and pronation/supination) in each window. DOF range *endpoints* included:

- full flexion = -1, full extension = 1
- full pronation = -1, full supination = 1

Within each DOF range, 0 indicated rest. Notably, a single RCNN regressor was capable of yielding movement prediction values for both DOFs simultaneously. In comparison, two SVR regressors were required to yield the same movement predictions, given that a single SVR regressor can only predict movements for one DOF.

RCNN and SVR movement predictions were then post-processed: (1) they were smoothed using the prediction from the previous window via a moving average filter [44]; (2) predictions between -0.2 and 0.2 were suppressed to 0 [45]; and (3) predictions greater than 1 or less than -1 were clipped to 1 or -1, respectively.

Movement regressors were trained with Trials 4–5 (dynamic) data from a participant and subsequently tested using Trial 6 (dynamic) data from that same participant. Movement regressors were trained, tested, and compared under model specifications S1–S3. The predictive accuracies of these regressors were compared to those of a baseline regressor (BR), trained with only EMG data collected with each participant's arm at their side (as per standard prosthesis training). For S3, when RCNN movement regression was investigated, RCNN limb position classification was used (that is, S3's models were all RCNN). Conversely, when SVR movement regression was investigated, LDA limb position classification was used (that is, S3's models were all statistical). As detailed in the previous Movement Classification section, each movement regressor under S1–S3 and the Baseline was trained with the same three combinations of data streams.

G. Outcome Measures and Statistical Analysis

1) Prediction Accuracy Calculations: *Limb Position Classifiers:* The predicted limb positions performed by the participants were compared to actual limb position classes, with resulting Trial 3 accuracies presented in the Results section as percentages (averaged across participants).

Movement Classifiers: The predicted movements performed by the participants were compared to actual movement classes, with resulting Trial 3 accuracies presented in the Results section as percentages (averaged across participants).

Movement Regressors: Unlike movement classifiers, which were trained with static data (discrete values), movement regressors were trained and tested with data from dynamic trials (continuous values). As such, the prediction accuracy of movement regressors was determined using R^2 (coefficient of

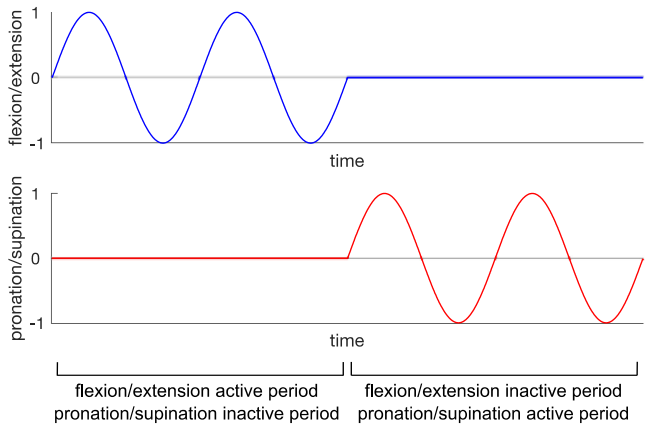

Fig. 3. Active and inactive periods for each degree of freedom (DOF). An active period is when movements are observed in a given DOF, and an inactive period is when that DOF is in rest while movements are observed in the other DOF.

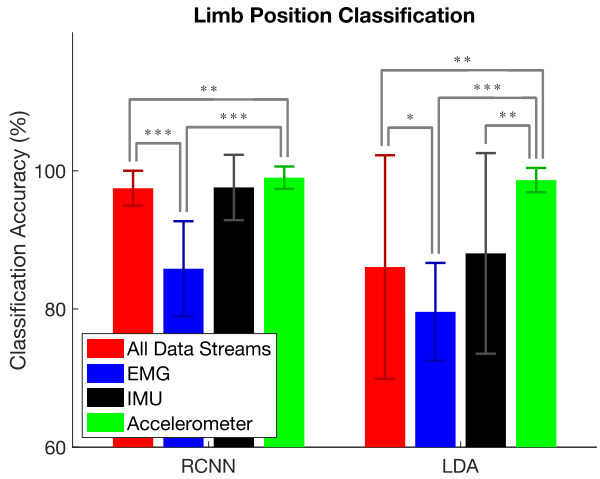

Fig. 4. Mean limb position classification accuracy (across participants) using RCNN and LDA classification for each combination of data streams: all EMG and IMU data streams from both Myo armbands (red); all EMG data streams from both Myo armbands (blue); all IMU data streams from both Myo armbands (black); and only accelerometer data streams from both Myo armbands (green). One standard deviation of each classification accuracy is shown with error bars, and significant prediction accuracy differences across combinations of data streams are indicated with asterisks ($*$: $p < 0.05$, $**$: $p < 0.01$, $***$: $p < 0.001$).

determination) calculations. Recall that the dynamic data consisted of movements that oscillated between either wrist flexion and extension or forearm pronation and supination – that is, in only one DOF at a time. Given this, two kinds of movement periods (or states) occurred for each DOF: active periods, wherein movements were observed in that DOF, and inactive periods, wherein rest occurred in that DOF while movements were observed in the other DOF, as shown in Fig. 3.

- For active periods, R^2 values were calculated by comparing movement predictions to the target sinusoids. Resulting Trial 6 values are presented in the Results section as percentages (averaged across participants).
- For inactive periods, however, R^2 values could not be calculated. This is because the actual movements in those periods form a horizontal line at 0 (see Fig. 3), with R^2 becoming an invalid measure of fit. As such, standard deviations of the movement predictions were calculated instead [45], to reveal the amount of predictive variation. A low standard deviation indicated a high prediction accuracy (that is, one with minimal unwanted movement predictions). Resulting Trial 6 values are presented in the Results section (averaged across participants).

The Kolmogorov-Smirnov test was conducted and revealed that all prediction accuracies did not follow a normal distribution. Therefore, the non-parametric Friedman's Analysis of Variance and post-hoc Wilcoxon signed-rank tests were used to identify significant prediction accuracy differences across combinations of data streams (for a given classifier or regressor).

2) Confusion Matrices: Of the movement classifiers and regressors under investigation, the best-performing were further analyzed using confusion matrices. When creating confusion matrices for regressors, predictions and target sinusoid values were categorized into rest, flexion, extension, pronation, and supination using the following rules: (1) values between -0.2 and 0.2 in both DOFs were categorized as rest (in accordance with the chosen post-processing threshold); and (2) for all non-rest values, flexion/extension predictions were compared to pronation/supination predictions, whereby the greater absolute values between them were used to identify movement categories. When creating confusion matrices for the classifiers under further investigation, such categorization was unnecessary.

3) Overall Comparisons: Finally, to facilitate direct comparisons between all movement classifiers and regressors under investigation, their root mean square errors (RMSEs) were calculated. RMSE provided a measure of the deviation between predicted and target values. Other studies have used similar measures to compare the performance of classifiers and regressors [46]–[48]. To calculate RMSE, movement classification predictions and actual movement classes were converted to values of -1, 0, or 1 in each DOF.

III. Results

A. Limb Position Classification

The mean limb position classification accuracies (across participants) of the RCNN and LDA classifiers, using four combinations of data streams from both Myo armbands, are shown in Fig. 4 and in Appendix A. Notably, both the RCNN and LDA classifiers predicted limb positions most accurately when the IMU's accelerometer data alone were used (99.01% for RCNN, 98.66% for LDA; a 0.35% difference between these).

The mean training and prediction times of the RCNN and LDA classifiers, using the same four combinations of data streams from both Myo armbands, are shown in Appendix B. In addition to yielding the highest prediction accuracies, both the RCNN and LDA classifiers resulted in decreased training times when only accelerometer data were used (RCNN: 1.68 minutes, LDA: 38.48 milliseconds) versus when all data streams were used (RCNN: 2.52 minutes, LDA: 89.19 milliseconds). Of note, all

TABLE I
REGRESSION STANDARD DEVIATIONS IN INACTIVE PERIODS

Specification	Data Streams	Flexion/ Extension Standard Deviation (%)		Pronation/ Supination Standard Deviation (%)	
		RCNN	SVR	RCNN	SVR
BR	Forearm EMG	10.76	4.85	16.01	13.50
	Both EMG	8.08	5.02	21.15	16.36
S1	Forearm EMG	5.74	3.47	11.14	7.59
	Both EMG	5.89	3.53	8.80	9.73
S2	Forearm EMG + Accel	4.20	3.19	5.11	8.10
	Both EMG + Accel	4.74	3.25	5.76	4.05
	Forearm EMG + Both Accel	4.50	3.13	5.39	4.13
S3	Forearm EMG + Accel	8.81	4.56	11.02	13.71
	Both EMG + Accel	6.56	4.97	8.91	13.82
	Forearm EMG + Both Accel	7.70	4.23	11.02	13.21

Mean standard deviations (across participants) using the RCNN flexion/extension regression, SVR flexion/extension regression, RCNN pronation/supination regression, and SVR pronation/supination regression, under each specification: the baseline regressor (BR), specification 1 (S1), specification 2 (S2), and specification 3 (S3). Standard deviations are provided for each combination of data streams: data from only the forearm Myo armband; data from both Myo armbands; and EMG data from the forearm and accelerometer (Accel) data from both Myo armbands (when applicable).

TABLE II
MOVEMENT PREDICTION ACCURACY SUMMARY

Model and Specification			Root Mean Square Error (calculated with different data streams)		
			Forearm EMG + Accel	Both EMG + Accel	Forearm EMG + Both Accel
Classifier	BC	RCNN	0.36	0.49	
		LDA	0.40	0.50	
	S1	RCNN	0.24	0.20	
		LDA	0.27	0.24	
	S2	RCNN	0.11	0.14	0.09
		LDA	0.16	0.15	0.15
	S3	RCNN	0.23	0.22	0.22
		LDA	0.24	0.23	0.24
Regressor	BR	RCNN	0.34	0.36	
		SVR	0.37	0.36	
	S1	RCNN	0.26	0.24	
		SVR	0.36	0.34	
	S2	RCNN	0.20	0.20	0.20
		SVR	0.28	0.27	0.28
	S3	RCNN	0.27	0.25	0.27
		SVR	0.34	0.31	0.34

Root mean square error across participants for each movement prediction method (classification or regression), specification (BC, BR, S1, S2, and S3), type of model (RCNN, LDA, and SVR), and combination of data streams (data from only the forearm Myo armband; data from both Myo armbands; and EMG data from the forearm and accelerometer (Accel) data from both Myo armbands, when applicable). Root mean square errors less than 0.22 are highlighted in green.

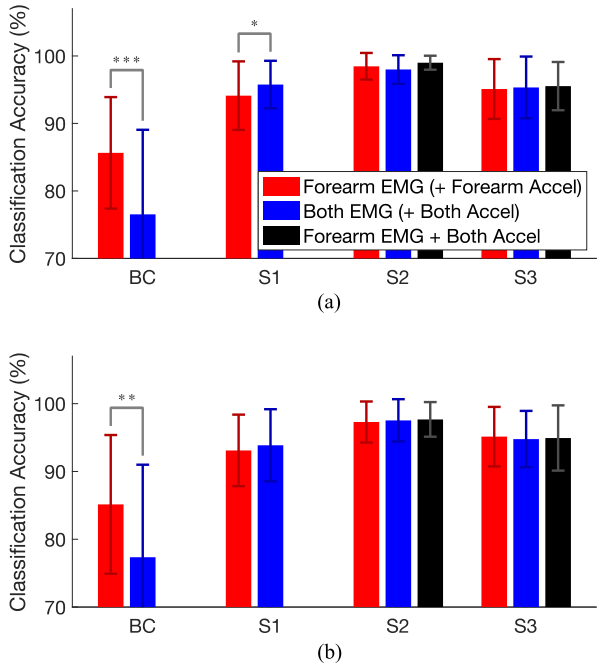

Fig. 5. Mean movement classification accuracy (across participants) using (a) RCNN classification and (b) LDA classification, under each classification specification: the baseline classifier (BC), specification 1 (S1), specification 2 (S2), and specification 3 (S3). Accuracies are provided for each combination of data streams: data from only the forearm Myo armband (red); data from both Myo armbands (blue); and EMG data from the forearm and accelerometer (Accel) data from both Myo armbands (when applicable, black). One standard deviation of each classification accuracy is shown with error bars, and significant prediction accuracy differences across combinations of data streams are indicated with asterisks ($*$: $p < 0.05$, $**$: $p < 0.01$, $***$: $p < 0.001$).

classifiers took less than 1 millisecond per prediction, which is well below the 100-millisecond threshold for optimal controller delay [49] (although, admittedly, the computer used in this study was much faster than an embedded processor in a myoelectric prosthesis).

Given these results, for subsequent movement classification and regression investigations, the quaternion and gyroscope data streams from the IMU were eliminated. Furthermore, the limb position classifier in model specification S3 used only accelerometer data streams.

B. Position-Aware Movement Classification: S1–S3

The mean movement classification accuracies (across participants) of the RCNN and LDA classifiers, under each specification and using three combinations of data streams, are shown in Fig. 5 and in Appendix A. As expected, the baseline RCNN classifier and baseline LDA classifier yielded the least accurate movement predictions (approximately 85% for each, when using only forearm Myo armband data streams). Overall, the RCNN classifier under S2, trained with EMG data from the forearm Myo armband and accelerometer data from both Myo armbands, yielded the most accurate movement predictions (99.00%). The LDA classifier under S2 using the same training data predicted movements with a slightly lower accuracy (97.67%). So, in comparison, this RCNN classifier was 1.33% more accurate than the

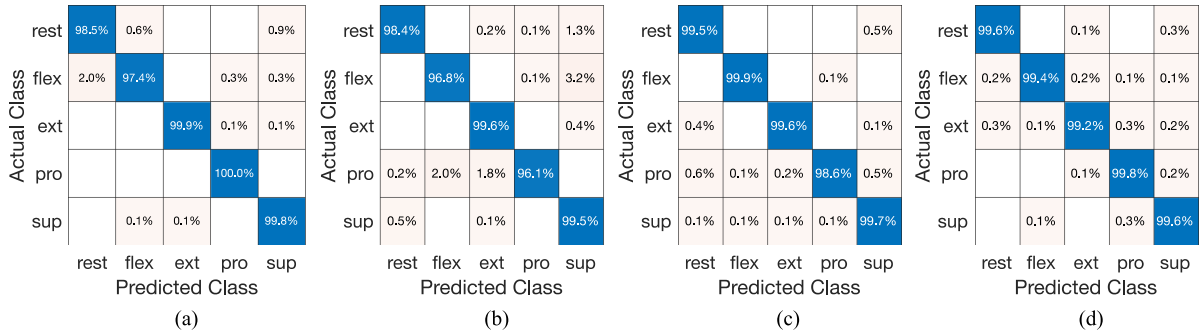

Fig. 6. RCNN movement classification confusion matrices, across participants for (a) arm at side, (b) elbow at 90°, (c) arm out at 90°, and (d) arm up at 45° under S2 using EMG data from the forearm Myo armband and accelerometer data from both Myo armbands. Movement classes are rest, flexion (flex), extension (ext), pronation (pro), and supination (sup).

corresponding LDA classifier. Generally, most of the position-aware RCNN *and* LDA classifiers yielded movement prediction accuracies over 95%, especially those under S2 and S3.

The mean training and prediction times of the RCNN and LDA classifiers, under each specification and using three combinations of data streams, are shown in Appendix B. On average, the RCNN classifiers under S1 and S2 took approximately 2 minutes to train, whereas RCNN classifiers under S3 took 9 minutes to train. The LDA classifiers under S1 and S2 took approximately 26 milliseconds to train, whereas LDA classifiers under S3 took 84 milliseconds to train. When comparing training times across specifications, RCNN and LDA classifiers under S3 required more time than classifiers under other specifications. Of note, all classifiers took less than 6 milliseconds per prediction, which is well below the 100-millisecond threshold for optimal controller delay [49].

Given that the RCNN classifier under S2, trained with EMG data from the forearm Myo armband and accelerometer data from both Myo armbands, predicted movements most accurately, its predictions were further investigated using confusion matrices for each limb position, as shown in Fig. 6. The RCNN classifier under S2's movement prediction accuracy was found to be consistent across all limb positions, with a roughly equal proportion of errors across classes.

C. Position-Aware Movement Regression: S1–S3

Recall that this study used two outcome measures to assess movement regression predictive accuracy: R^2 values during active periods and standard deviations during inactive periods. The mean R^2 values (across participants) of the RCNN and SVR movement regressors, under each specification and using three combinations of data streams, are shown in Fig. 7 and in Appendix A. The corresponding mean standard deviations are presented in Table I. For both flexion/extension and pronation/supination DOFs, the RCNN regressor under S2 yielded the highest R^2 values during active periods and the lowest standard deviations during inactive periods (compared to standard deviations of predictions made with the other RCNN regressors). Overall, the RCNN regressor under S2, trained with EMG and accelerometer data from the forearm Myo armband, yielded high R^2 values for both DOFs (84.93% for flexion/extension and 84.97% for pronation/supination), while reducing the required number of data streams. Conversely, the SVR regressor under S2, also using EMG and accelerometer data from the forearm Myo armband, yielded much lower R^2 values (77.26% for flexion/extension and 60.73% for pronation/supination). The RCNN regressor had R^2 values that were 7.67% greater in flexion/extension and 24.24% greater in pronation/supination than those of the corresponding SVR regressor.

When comparing standard deviations, the RCNN regressor under S2 had a flexion/extension standard deviation of 4.20% and a pronation/supination standard deviation of 5.11%. Conversely, the corresponding SVR regressor had standard deviations of 3.19% and 8.10%, for these same movements. The RCNN regressor had a flexion/extension standard deviation 1.01% higher than that of the SVR regressor, and a pronation/supination standard deviation 2.99% lower than that of the SVR regressor.

The mean training and prediction times of the RCNN and SVR regressors under each specification, using three combinations of data streams, are shown in Appendix B. On average, the RCNN regressors under S1 and S2 took approximately 1 minute to train, whereas RCNN under S3 took 3 minutes to train. The SVR regressors under S1 and S2 took approximately 21 seconds to train, whereas SVR regressors under S3 took 8 seconds to train. Of note, all regressors took less than 6 milliseconds per prediction, which is well below the 100-millisecond threshold for optimal controller delay [49].

Given that the RCNN regressor under S2, trained with EMG and accelerometer data from the forearm Myo armband, predicted movements most accurately, its predictions were further investigated. These predictions were categorized into movement classes (rest, flexion, extension, pronation, and supination), and the resulting confusion matrices for each limb position were generated, as shown in Fig. 8. The RCNN regressor under S2's movement prediction accuracy was found to be consistent across all limb positions, but most errors were related to rest.

D. Results Summary

A comparative summary of the classifiers and regressors that were investigated in this study is presented in Table II, wherein the RMSE was calculated for all movement predictions.

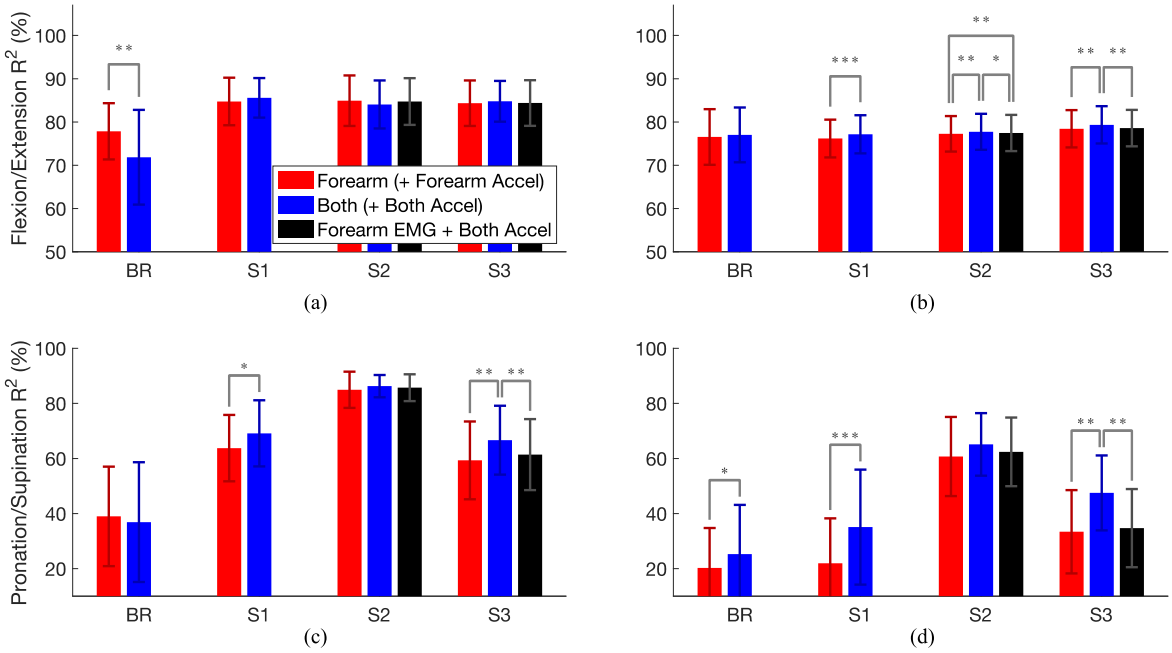

Fig. 7. Mean movement regression R^2 values (across participants) using (a) RCNN flexion/extension regression, (b) SVR flexion/extension regression, (c) RCNN pronation/supination regression, and (d) SVR pronation/supination regression, under each specification: the baseline regressor (BR), specification 1 (S1), specification 2 (S2), and specification 3 (S3). R^2 values are provided for each combination of data streams: data from only the forearm Myo armband (red); data from both Myo armbands (blue); and EMG data from the forearm and accelerometer (Accel) data from both Myo armbands (when applicable, black). One standard deviation of each R^2 value is shown with error bars, and significant prediction accuracy differences across combinations of data streams are indicated with asterisks ($*$: $p < 0.05$, $**$: $p < 0.01$, $***$: $p < 0.001$).

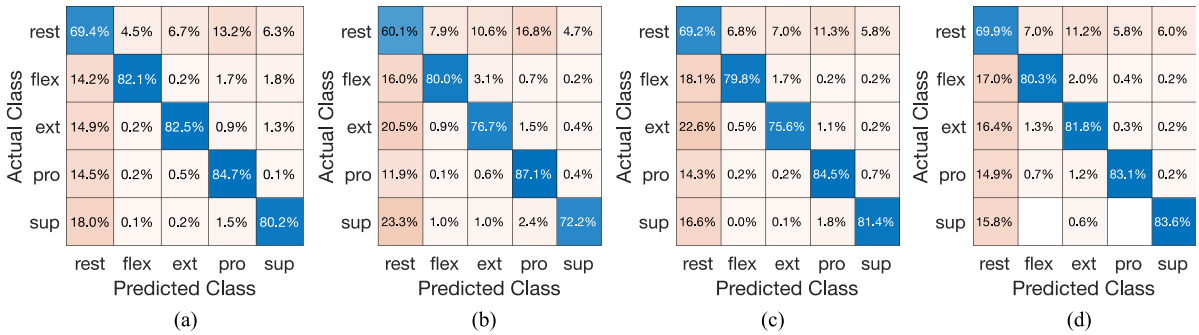

Fig. 8. RCNN movement regression confusion matrices, across participants for (a) arm at side, (b) elbow at 90°, (c) arm out at 90°, and (d) arm up at 45° under S2 using data from the forearm Myo armband. Movement classes are rest, flexion (flex), extension (ext), pronation(pro), and supination (sup).

Table II identifies the position-aware control strategies that most accurately predicted movements — those with an RMSE less than 0.22 (this threshold was chosen as it represents the 70th percentile of accuracy).

Overall, the best classifier was determined to be the RCNN classifier under S2, and the best regressor was the RCNN regressor under S2 – both yielded the most accurate movement predictions, while using fewer than all available data streams.

IV. Discussion

The goal of this study was to investigate RCNN-based position-aware myoelectric prosthesis control strategies, using combined EMG and IMU input data streams. EMG signals primarily informed intended movements, whereas IMU signals primarily provided context about limb position. Classifiers and regressors used these signals to make position-aware movement predictions. Recall that three model specifications were explored:

- S1 – Model (classifier or regressor) trained with EMG data from all limb positions
- S2 – Model trained with EMG and IMU data from all limb positions
- S3 – Models trained with EMG data at each limb position, with subsequent predictions occurring in a 2-staged sequence: 1st, a limb position was classified using IMU data; 2nd,

a corresponding model (trained at that specified limb position) predicted a wrist movement using EMG data

For this study, a favourable position-aware myoelectric prosthesis control strategy was considered to be one where a classifier or regressor yielded accurate movement predictions, using the fewest possible data streams.

A. Position-Aware Classification

This study corroborates and extends the findings of earlier prosthesis control strategy research that likewise used classifiers under model specifications S1–S3. Such research yielded improved movement predictions compared to a baseline classifier [9]– [11]. Fougner *et al.* found that LDA classification under S2 yielded the most accurate movement predictions [9], whereas Geng *et al.* concluded that LDA classification under S3 proved to be the most accurate [10], [11].

Of the position-aware classifiers under S1–S3 that were investigated in this study, **the most promising was the RCNN classifier under S2 (with EMG data from the forearm Myo armband and accelerometer data from both Myo armbands)**. It yielded the highest movement prediction accuracy (99.00%, versus the LDA's at 97.67%) while requiring a reduced number of data streams. The success of this classifier under S2 is consistent with Fougner *et al.*'s observations [9]. Notably, classifiers under S1 performed less accurately compared to those under S2 because accelerometer data (and consequently limb position information) was not included under S1. Additionally, classifiers under S3 performed less accurately than those under S2, likely because the classification sequence of S3 (with two stages) introduced the potential to compound errors.

This study's RCNN classifier under S2 yielded more accurate movement predictions than did classifiers in earlier research. As such, this work offers encouraging results towards solving the limb position effect.

B. Position-Aware Regression

To our knowledge, only one other study has implemented a regression-based device control strategy in the context of addressing the limb position effect. Park *et al.* employed a position decoder to accomplish position-independent regression, and tested their resulting predictive device control outcomes through real-time experimentation [22]. They predicted movements with smaller R^2 values than those of this study, but caution should be taken when comparing their real-time results to those of this offline work.

Of the position-aware regressors under S1–S3 that were investigated in this study, **the most promising was the RCNN regressor under S2 (with EMG and accelerometer data from the forearm Myo armband)**. It yielded the highest movement prediction accuracy (with R^2 values of 84.93% for wrist flexion/extension and 84.97% for forearm pronation/supination, versus the SVR's at 77.26% and 60.73%, respectively). It also required a reduced number of data streams. However, this RCNN regressor predicted movements with lower accuracies than the investigated classifiers. This is in keeping with previous research that found regression to be less accurate than classification, due to the increased complexity of regression predictions (continuous values for each DOF) [24]. Despite being lower in predictive accuracy than classification, regression may offer increased functionality, through both simultaneous and proportional control, and as such might outperform classification in real-time experimentation [48].

Of the errors that contributed to the decreased accuracy of the RCNN regressor under S2, the majority occurred around the rest periods, as evidenced in Fig. 8. These errors can be categorized as either false negatives (falsely predicting rest) or false positives (falsely predicting a movement instead of rest). False negatives occurred more frequently. Notably, false negatives can be considered acceptable in prosthesis control; that is, simply perceived as device responsiveness latency by users [51]. The detected false negatives may have resulted from prediction suppression, whereby prediction values between -0.2 and 0.2 were set to 0. Note that in future work, this suppression threshold can be adjusted. Finally, both false negatives and false positives may have been caused by offsets between the participants' movements and the sinusoids chosen to represent these movements. Undoubtedly, without a perfect match between sinusoids and movements, slight inaccuracies can be expected.

To mitigate the occurrence of such inaccuracies, participants' movements must closely track training sinusoids. It is a common research practice to have participants follow an onscreen training target (such as a moving cursor or virtual hand) [45], [52]–[54]. But this practice can result in the introduction of participant movement delays. This study corrected the delay between onscreen movement instructions and participants' actual movements by using generated sinusoids for both wrist flexion/extension and forearm pronation/supination. To accomplish this, peak muscle contractions were extracted from the EMG signal data and used to produce wrist flexion/extension sinusoids, whereas accelerometer signals were used to generate forearm pronation/supination sinusoids. Despite making the necessary movement corrections in this study through the use of sinusoids, offsets may have still been present (although presumably smaller than without such corrections.

To further reduce the occurrence of movement offsets, modifications could be made to the data collection methods for the regression training routine. For example, if a participant were to follow an onscreen sinusoid overlayed with their real-time EMG signals [55] (that is, afforded visual feedback), more accurate instruction adherence would likely result. That same sinusoid could then be used as a precise training target (as opposed to extracting muscle and position signals' peaks for sinusoids). Additionally, if participants were required to complete a practice dynamic trial before data collection, the precision with which they follow the target sinusoid would likely improve.

C. Promising RCNN Outcomes

As expected, the RCNN-based control strategies investigated in this study predicted movements more accurately than statistical-based alternatives (which was especially evident when comparing RCNN and SVR regressors). This may be because RCNNs offer the advantage of learning new features

from complex input data. Other studies have investigated the use of engineered feature sets to address the limb position problem, and as such did not harness this advantage [17], [21], [23]. Despite yielding position-aware movement predictions using engineered features, their models did not perform quite as well as this study's RCNN classifier under S2. Although these studies examined more extensive limb position ranges and movements, their lower predictive accuracies may suggest that for position-aware myoelectric control, *learning* new features with RCNNs may be favourable over using *engineered* features. Naturally, further research is required to confirm this.

D. Limitations

Limitations in this study included: the requirement for training routines with long durations; the number of limb positions and wrist movements used for training and testing models was not exhaustive; models were only tested on the 3rd or 6th trials; more training data may be required for accurate results in other limb positions; and only static limb positions were employed in this study (training with continuous limb positions may improve predictive accuracy [13], [19]).

Notably, regressors were tested using data from oscillations in one DOF at a time. This testing method does not demonstrate model performance during simultaneous muscle contractions in two DOFs and consequently cannot translate directly to activities of daily living. Furthermore, data from only *isotonic* muscle contractions were recorded, rather than data resulting from isometric contractions (which are used to control a prosthesis). The performance of the models presented in this study may differ when isometric contractions are used.

Finally, the feasibility of implementing RCNN-based prosthesis control strategies using existing hardware was not investigated in this study. However, as the capabilities of onboard prosthetic device processors continue to improve, it is expected that implementation might well be possible in the near future.

E. Future Work

Future work will focus on real-time testing of the promising RCNN-based control strategies (RCNN classifier and regressor under S2) presented in this study. Upcoming research will include real-time testing of these control strategies with both non-disabled participants (using a simulated prosthesis) and myoelectric prosthesis users. Testing using a simulated or actual prosthesis will require participants to use isometric contractions for device control. Participants will carry out functional tasks that simulate activities of daily living. These tasks will also allow for the assessment of regression control for simultaneous movements.

Although the movement prediction accuracy of myoelectric control strategies may not always correlate with their real-time performance [11], [18], a reduction in the limb position effect can be expected in real-time experimentation (given that participants will have visual feedback and will be able to adjust their muscle contractions accordingly [44]). Improvements to the regression training routine that were gleaned from this study will be implemented in future work. Additionally, as RCNN classifiers under S2 required training routines with long durations (relative to the baseline classifiers), a generalized RCNN classifier will be investigated, with the goal of eliminating the training routine (and consequently model training time) altogether.

V. Conclusion

This study has identified two promising position-aware myoelectric prosthesis control strategies towards solving the "limb position effect" problem:

1) An RCNN classifier trained with EMG and accelerometer (IMU) data (captured from participants across multiple limb positions) predicted movements best, while requiring a reduced number of data streams; and
2) An RCNN regressor trained with EMG and accelerometer data (captured from participants across multiple limb positions) performed much better than an SVR regressor, although not as accurately as the aforementioned RCNN classifier. It also required fewer than all available data streams.

It is expected that both of these RCNN-based control strategies will likewise yield accurate, position-aware movement predictions in real-time experimentation. As such, results of this research are anticipated to improve the usability of myoelectric devices for individuals with amputation, particularly when faced with the challenges of the "limb position effect".

Appendix

Appendix A (see supplementary material) contains tables with the results shown in Figs. 4, 5, and 7. Appendix B (see supplementary material) contains tables with the time required to train and test all classifiers and regressors.

Appendix A Prediction Accuracies

This appendix contains the mean limb position classification accuracies, mean movement classification accuracies, and mean movement regression R^2 values resulting from this study.

Appendix B Training and Prediction Times

This appendix contains the mean limb position classification training and prediction times, movement classification training and prediction times, and movement regression training and prediction times resulting from this study.

References

[1] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *J. Rehabil. Res. Dev.*, vol. 48, no. 6, pp. 643–659, 2011, doi: 10.1682/Jrrd.2010.09.0177.

[2] P. Geethanjali, "Myoelectric control of prosthetic hands: State-of-the-art review," *Med. Devices Evid. Res.*, vol. 9, pp. 247–255, 2016, doi: 10.2147/MDER.S91102.

[3] E. Campbell, A. Phinyomark, and E. Scheme, "Current trends and confounding factors in myoelectric control: Limb position and contraction intensity," *Sensors (Switzerland)*, vol. 20, no. 6, 2020, Art. no. 1613, doi: 10.3390/s20061613.

[4] E. Scheme *et al.*, "Examining the adverse effects of limb position on pattern recognition based myoelectric control," in *Proc. 32nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2010, pp. 6337–6340. doi: 10.1109/IEMBS.2010.5627638.

[5] H. A. Dewald *et al.*, "Stable, three degree-of-freedom myoelectric prosthetic control via chronic bipolar intramuscular electrodes: A case study," *J. Neuroeng. Rehabil.*, vol. 16, p. 147, 2019, doi: 10.1186/s12984-019-0607-8.

[6] M. Ortiz-Catalan, B. Håkansson, and R. Brånemark, "An osseointegrated human-machine gateway for long-term sensory feedback and motor control of artificial limbs," *Sci. Transl. Med.*, vol. 6, no. 257, 2014, Art. no. 257re6, doi: 10.1126/scitranslmed.3008933.

[7] A. Boschmann and M. Platzner, "Towards robust HD EMG pattern recognition: Reducing electrode displacement effect using structural similarity," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2014, pp. 4547–4550. doi: 10.1109/EMBC.2014.6944635.

[8] A. Radmand, E. Scheme, and K. Englehart, "High-density force myography: A possible alternative for upper-limb prosthetic control," *J. Rehabil. Res. Dev.*, vol. 53, no. 4, pp. 443–456, 2016, doi: 10.1682/JRRD.2015.03.0041.

[9] A. Fougner *et al.*, "Resolving the limb position effect in myoelectric pattern recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 19, no. 6, pp. 644–651, Dec. 2011, doi: 10.1109/TNSRE.2011.2163529.

[10] Y. Geng, P. Zhou, and G. Li, "Toward attenuating the impact of arm positions on electromyography pattern-recognition based motion classification in transradial amputees," *J. Neuroeng. Rehabil.*, vol. 9, p. 74, 2012, doi: 10.1186/1743-0003-9-74.

[11] Y. Geng *et al.*, "Improving the robustness of real-time myoelectric pattern recognition against arm position changes in transradial amputees," *Biomed Res. Int.*, vol. 2017, 2017, Art. no. 5090454, doi: 10.1155/2017/5090454.

[12] A. Radmand, E. Scheme, and K. Englehart, "A characterization of the effect of limb position on EMG features to guide the development of effective prosthetic control schemes," in *Proc. 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2014, pp. 662–667. doi: 10.1109/EMBC.2014.6943678.

[13] W. Shahzad *et al.*, "Enhanced performance for multi-forearm movement decoding using hybrid IMU–SEMG interface," *Front. Neurorobot.*, vol. 13, p. 43, 2019, doi: 10.3389/fnbot.2019.00043.

[14] A. Radmand, E. Scheme, and K. Englehart, "On the suitability of integrating accelerometry data with electromyography signals for resolving the effect of changes in limb position during dynamic limb movement," *J. Prosthetics Orthot.*, vol. 26, no. 4, pp. 185–193, 2014, doi: 10.1097/JPO.0000000000000041.

[15] R. J. Beaulieu *et al.*, "Multi-position training improves robustness of pattern recognition and reduces limb-position effect in prosthetic control," *J. Prosthetics Orthot.*, vol. 29, no. 2, pp. 54–62, 2017.

[16] J. L. Betthauser *et al.*, "Limb position tolerant pattern recognition for myoelectric prosthesis control with adaptive sparse representations from extreme learning," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 4, pp. 770–778, Apr. 2018, doi: 10.1109/TBME.2017.2719400.

[17] A. K. Mukhopadhyay and S. Samui, "An experimental study on upper limb position invariant EMG signal classification based on deep neural network," *Biomed. Signal Process. Control*, vol. 55, 2020, Art. no. 101669, doi: 10.1016/j.bspc.2019.101669.

[18] A. Gigli, A. Gijsberts, and C. Castellini, "Natural myocontrol in a realistic setting: A comparison between static and dynamic data acquisition," in *Proc. IEEE Int. Conf. Rehabil. Robot. (ICORR)*, 2019, pp. 1061–1066. doi: 10.1109/ICORR.2019.8779364.

[19] E. Scheme, K. Biron, and K. Englehart, "Improving myoelectric pattern recognition positional robustness using advanced training protocols," in *Proc. 33rd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, 2011, pp. 4828–4831. doi: 10.1109/IEMBS.2011.6091196.

[20] A. Boschmann and M. Platzner, "Reducing the limb position effect in pattern recognition based myoelectric control using a high density electrode array," in *Proc. ISSNIP Biosignals Biorobotics Conf. (BRC)*, 2013, pp. 1–5. doi: 10.1109/BRC.2013.6487548.

[21] R. N. Khushaba *et al.*, "Towards limb position invariant myoelectric pattern recognition using time-dependent spectral features," *Neural Netw.*, vol. 55, pp. 42–58, 2014, doi: 10.1016/j.neunet.2014.03.010.

[22] K.-H. Park, H.-I. Suk, and S.-W. Lee, "Position-independent decoding of movement intention for proportional myoelectric interfaces," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 9, pp. 928–939, Sep. 2016, doi: 10.1109/TNSRE.2015.2481461.

[23] Y. Yu *et al.*, "Attenuating the impact of limb position on surface EMG pattern recognition using a mixed-LDA classifier," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, 2017, pp. 1497–1502. doi: 10.1109/ROBIO.2017.8324629.

[24] A. D. Roche *et al.*, "Prosthetic myoelectric control strategies: A clinical perspective," *Curr. Surg. Rep.*, vol. 2, p. 44, 2014, doi: 10.1007/s40137-013-0044-8.

[25] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7420–7431, 2012, doi: 10.1016/j.eswa.2012.01.102.

[26] S. Asht and R. Dass, "Pattern recognition techniques: A review," *Int. J. Comput. Sci. Telecommun.*, vol. 3, no. 8, pp. 25–29, 2012.

[27] F. Cordella *et al.*, "Literature review on needs of upper limb prosthesis users," *Front. Neurosci.*, vol. 10, p. 209, 2016, doi: 10.3389/fnins.2016.00209.

[28] A. Phinyomark *et al.*, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Syst. Appl.*, vol. 40, no. 12, pp. 4832–4840, 2013.

[29] A. Phinyomark, R. N. Khushaba, and E. Scheme, "Feature extraction and selection for myoelectric control based on wearable EMG sensors," *Sensors (Switzerland)*, vol. 18, no. 5, 2018, Art. no. 1615, doi: 10.3390/s18051615.

[30] A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data Cogn. Comput.*, vol. 2, no. 3, p. 21, 2018, doi: 10.3390/bdcc2030021.

[31] A. Ameri *et al.*, "Regression convolutional neural network for improved simultaneous EMG control," *J. Neural Eng.*, vol. 16, no. 3, 2019, Art. no. 036015, doi: 10.1088/1741-2552/ab0e2e.

[32] 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," *Front. Neurorobot.*, vol. 10, p. 9, 2016, doi: 10.3389/fnbot.2016.00009.

[33] U. Côté-Allard *et al.*, "Interpreting deep learning features for myoelectric control: A comparison with handcrafted features," *Front. Bioeng. Biotechnol.*, vol. 8, p. 158, 2020, doi: 10.3389/fbioe.2020.00158.

[34] W. Wang *et al.*, "Sensor fusion for myoelectric control based on deep learning with recurrent convolutional neural networks," *Artif. Organs*, vol. 42, no. 9, pp. E272–E282, 2018, doi: 10.1111/aor.13153.

[35] P. Xia, J. Hu, and Y. Peng, "EMG-Based estimation of limb movement using deep learning with recurrent convolutional neural networks," *Artif. Organs*, vol. 42, no. 5, pp. E67–E77, 2017, doi: 10.1111/aor.13004.

[36] T. Bao *et al.*, "A CNN-LSTM hybrid framework for wrist kinematics estimation using surface," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 88–94, 2020.

[37] D. J. Atkins, D. C. Y. Heard, and W. H. Donovan, "Epidemiologic overview of individuals with upper-limb loss and their reported research priorities," *J. Prosthetics Orthot.*, vol. 8, no. 1, pp. 2–11, 1996, doi: 10.1097/00008526-199600810-00003.

[38] A. W. Shehata *et al.*, "Mechanotactile sensory feedback improves embodiment of a prosthetic hand during active use," *Front. Neurosci.*, vol. 14, p. 263, 2020, doi: 10.3389/fnins.2020.00263.

[39] L. F. Engels *et al.*, "When less is more – discrete tactile feedback dominates continuous audio biofeedback in the integrated percept while controlling a myoelectric prosthetic hand," *Front. Neurosci.*, vol. 13, p. 578, 2019, doi: 10.3389/fnins.2019.00578.

[40] H. E. Williams *et al.*, "Myoelectric prosthesis users and non-disabled individuals wearing a simulated prosthesis exhibit similar compensatory movement strategies," *J. Neuroeng. Rehabil.*, vol. 18, p. 72, 2021.

[41] J. V. V. Parr *et al.*, "Visual attention, EEG alpha power and T7-Fz connectivity are implicated in prosthetic hand control and can be optimized through gaze training," *J. Neuroeng. Rehabil.*, vol. 16, no. 1, p. 52, 2019, doi: 10.1186/s12984-019-0524-x.

[42] N. Parajuli *et al.*, "Real-Time EMG based pattern recognition control challenges and future implementation," *Sensors (Switzerland)*, vol. 19, no. 20, 2019, Art. no. 4596.

[43] A. LeNail, "NN-SVG: Publication-ready neural network architecture schematics," *J. Open Source Softw.*, vol. 4, no. 33, p. 747, 2019, doi: 10.21105/joss.00747.

[44] H.-J. Hwang, J. M. Hahne, and K.-R. Müller, "Real-time robustness evaluation of regression based myoelectric control against arm position change and donning/doffing," *PLoS One*, vol. 12, no. 11, 2017, Art. no. e0186318, doi: 10.1371/journal.pone.0186318.

[45] A. Ameri *et al.*, "Support vector regression for improved real-time, simultaneous myoelectric control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 6, pp. 1198–1209, Nov. 2014, doi: 10.1109/TNSRE.2014.2323576.

[46] A. W. Shehata, E. J. Scheme, and J. W. Sensinger, "Evaluating internal model strength and performance of myoelectric prosthesis control strategies," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 5, pp. 1046–1055, May 2018, doi: 10.1109/TNSRE.2018.2826981.

[47] A. W. Shehata, E. J. Scheme, and J. W. Sensinger, "Audible feedback improves internal model strength and performance of myoelectric prosthesis control," vol. 8, 2018, Art. no. 8541, doi: 10.1038/s41598-018-26810-w.

[48] J. M. Hahne, M. Markovic, and D. Farina, "User adaptation in myoelectric man-machine interfaces," *Sci. Rep.*, vol. 7, 2017, Art. no. 4437, doi: 10.1038/s41598-017-04255-x.

[49] T. R. Farrell and R. F. Weir, "The optimal controller delay for myoelectric prostheses," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 1, pp. 111–118, Mar. 2007, doi: 10.1109/Tnsre.2007.891391.

[50] Y. Teh and L. J. Hargrove, "The effects of limb position and external load on offline myoelectric pattern recognition control," in *Proc. IEEE RAS EMBS Int. Conf. Biomed. Robot. Biomechatronics (BioRob)*, 2020, pp. 661–666, doi: 10.1109/BioRob49111.2020.9224333.

[51] S. Tam *et al.*, "Intuitive real-time control strategy for high-density myoelectric hand prosthesis using deep and transfer learning," *Sci. Rep.*, vol. 11, 2021, Art. no. 11275, doi: 10.1038/s41598-021-90688-4.

[52] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, "Use of probabilistic weights to enhance linear regression myoelectric control," *J. Neural Eng.*, vol. 12, no. 6, 2015, Art. no. 066030, doi: 10.1088/1741-2560/12/6/066030.

[53] L. H. Smith, T. A. Kuiken, and L. J. Hargrove, "Evaluation of linear regression simultaneous myoelectric control using intramuscular EMG," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 4, pp. 737–746, Apr. 2016, doi: 10.1109/TBME.2015.2469741.

[54] J. A. George *et al.*, "Inexpensive surface electromyography sleeve with consistent electrode placement enables dexterous and stable prosthetic control through deep learning," in *Proc. Myoelectric Controls Symp. (MEC)*, 2020, pp. 103–106, [Online]. Available: http://arxiv.org/abs/2003.00070

[55] N. Malesevic *et al.*, "A database of multi-channel intramuscular electromyogram signals during isometric hand muscles contractions," *Sci. Data*, vol. 7, p. 10, 2020, doi: 10.1038/s41597-019-0335-8.