Improved Training Paradigms and Motor-decode Algorithms: Results from Intact Individuals and a Recent Transradial Amputee with Prior Complex Regional Pain Syndrome

Jacob A. George, *Student Member, IEEE,* Mark R. Brinton, *Member, IEEE*, Christopher C. Duncan, Douglas T. Hutchinson, and Gregory A. Clark

Abstract— **Working towards improved neuromyoelectric control of dexterous prosthetic hands, we explored how differences in training paradigms affect the subsequent online performance of two different motor-decode algorithms. Participants included two intact subjects and one participant who had undergone a recent transradial amputation after complex regional pain syndrome (CRPS) and multi-year disuse of the affected hand. During algorithm training sessions, participants actively mimicked hand movements appearing on a computer monitor. We varied both the duration of the hold-time (0.1 s or 5 s) at the end-point of each of six different digit and wrist movements, and the order in which the training movements were presented (random or sequential). We quantified the impact of these variations on two different motor-decode algorithms, both having proportional, six-degree-of-freedom (DOF) control: a modified Kalman filter (MKF) previously reported by this group, and a new approach – a convolutional neural network (CNN). Results showed that increasing the hold-time in the training set improved run-time performance. By contrast, presenting training movements in either random or sequential order had a variable and relatively modest effect on performance. The relative performance of the two decode algorithms varied according to the performance metric. This work represents the first-ever amputee use of a CNN for real-time, proportional six-DOF control of a prosthetic hand. Also novel was the testing of implanted high-channel-count devices for neuromyoelectric control shortly after amputation, following CRPS and long-term hand disuse. This work identifies key factors in the training of decode algorithms that improve their subsequent run-time performance.**

I. Introduction

In the United States alone, 1.6 million individuals have lost a limb due to either dysvascular diseases like diabetes (54%) or to trauma (45%) [1]. For nearly one in every 200 individuals, limb-loss is a life-long struggle with chronic pain, depression, and functional disability [1]–[3]. The current standard-of-care for upper-limb amputees is unsatisfactory, and as a result, up to 50% of upper-limb amputees abandon their prostheses [4], citing ineffective control as a primary reason [5].

Even though the physical hand is missing after an amputation, most transradial amputees still retain the neural circuits and forearm musculature that are used to control the hand. These residual biopotentials can be recorded using peripheral nerve interfaces or electromyographic (EMG) implants and can then be used to estimate an amputee's motor intent. In this work, we use a modified Kalman filter (MKF) [6] and a convolutional neural network (CNN) to extract motor intent from the neural and EMG signals that still exist after long-term amputation of the hand. We then explore how variations in the initial training data impact run-time performance for these two motor-decode algorithms.

A variety of different algorithms exist for translating neural and EMG signals into motor intent. They generally fall into the broad categories of Wiener filters, population vectors, probabilistic methods, and recursive Bayesian decoders [7]. One of the more widely used motor-decode algorithms is the Kalman filter (KF), which is linear in nature, and as a result, is somewhat limited in its ability to accurately predict motor intent based off highly non-linear neural and EMG signals. Decoding algorithms capable of capturing these non-linearities, like neural networks, have primarily been used to predict hand grasps (i.e., pattern recognition) as opposed to continuously predicting the location of each individual degree of freedom (DOF) on the hand (i.e., regression). Recent work has begun exploring the capabilities of more advanced neural networks in providing independent and proportional control over each DOF in the hand [8], [9], and some offline analyses suggests that neural networks, with the proper training, can exceed the performance of a standard KF [9]. However, it is unclear how the performance of neural networks compares with the performance of modified KFs, particularly during online tasks when there is a participant in the loop, actively controlling the prosthesis.

We therefore sought to compare the performance of a MKF and CNN algorithm directly and to explore how variations in training paradigms affect their subsequent online performance. An additional question is the extent to which these algorithms might provide effective control for an individual with a recent transradial amputation after complex regional pain syndrome (CRPS) and long-term hand disuse.

II. Methods

A. Human Subjects

A total of three human participants were used in this study: two intact subjects and one participant who had undergone a transradial amputation 5-6 months prior to the present experiments, after CRPS and multi-year disuse of the affected hand. All participants were male and between the ages of 24 and 48. One of the intact participants was a co-author on this study. The two intact participants had 32 single-ended surface electrodes placed on their upper forearm for recording EMG signals from extrinsic hand muscles. The transradial amputee had three, 100-electrode Utah Slanted Electrode Arrays (USEAs; Blackrock Microsystems, Salt Lake City, UT, USA) implanted into his residual nerves; two in the median nerve and one in the ulnar nerve. Thirty-two intramuscular electromyographic recording electrodes (iEMGs; Ripple LLC, Salt Lake City, UT, USA) were also implanted in the residual arm muscles of the amputee. Additional information regarding the devices and implantation procedure can be found in [6], [10].

Informed consent and experimental protocols were carried out in accordance with the University of Utah Institutional Review Board.

B. Signal Acquisition

Surface EMG from the two non-amputee subjects and iEMG from the transradial amputee were collected and processed in the same manner, as described in [6], [10].

Although a total of three USEAs were implanted into the residual nerves of the amputee, only two were used (both median USEAs) in this study due to software limitations. The neural signals recorded from these USEAs were processed approximately as described in [6], [10].

C. Initial Training Data

In order to train the MKF and the CNN, recordings from sEMGs or iEMGs and USEAs were collected while the participant mimicked a set of preprogrammed hand movements performed by a virtual Modular Prosthetic Limb (MSMS; Johns Hopkins Applied Physics Lab, Baltimore, MD) with their own phantom hand (Fig. 1). These hand movements included individuated movements of each DOF of the virtual prosthetic hand (flexions/extensions of D1, D2, and D3; wrist flexion/extension; wrist pronation/supination; thumb abduction/adduction) as well as two combination movements (simultaneous flexion of D1, D2 and D3; simultaneous extension of D1 and D2), for a total of 14 movements. D4 and D5 were excluded in this task because most physical prostheses on the market couple these DOFs to D3. For example, the index finger on the prosthetic hand would flex, and the participant would actively attempt to flex their index finger simultaneously. The signals recorded during this task, as well as the kinematic position of each individual DOF of the prosthetic hand, served as the training data for the motor-decode algorithm. To avoid complications due to the participant's reaction time, we aligned the kinematic positions with the recorded signals by shifting the kinematic positions by a lag that was determined by cross-correlation. This alignment was performed uniformly across all trials of a given training condition.

Figure 1: Training and testing of real-time proportional six-DOF motor-decode algorithms for a virtual prosthetic hand. This participant had undergone a transradial amputation of his right hand 5-6 months prior to experimental sessions, due to long-term complex regional pain syndrome (CRPS) and disuse of the affected hand. At the time of surgery, the participant also received implants of three 100-electrode USEAs in his residual arm nerves proximal to the elbow, plus a 32-channel assembly of electromyographic (EMG) electrodes in his residual extrinsic hand muscles in his forearm. The implanted devices allowed recordings of neuromyoelectric signals during subsequent experimental sessions. EMG recordings were obtained from the two intact subjects with temporary surface EMG electrodes. During training sessions, participants mimicked the movements of the virtual hand with their phantom hand (amputee participant) or intact hand (two intact participants). During subsequent testing sessions, participants attempted to reach and hold a target position (0.5 of full flexion) for a particular DOF while maintaining all other DOFs at their resting positions.

The initial training data included four trials for each individual movement. For the short-hold training dataset (our previous standard training condition), all four trials for each individual movement were performed sequentially, one after another, and the total duration of each individual movement was 1.5 s (made up of a 0.7 s flexion/extension away from the resting hand position, a 0.1 s hold-time at the maximum distance away from the rest position, and then another 0.7 s extension/flexion back to the rest position).

To increase the duration of the hold-time at the end-point of each movement, we increased the total duration of each individual movement to 6.4 s (made up of a 0.7 s flexion/extension away from the resting hand position, a 5-s hold-time at the maximum distance away from the rest position, and then another 0.7 s extension/flexion back to the rest position).

To vary the order in which movements occurred during training, we modified the training so that the individual hand movements appeared in either a sequential order (our previous standard training condition) or a random order. The participant was given brief, verbal notification of each upcoming movement to avoid an excessive delay between the motion of the virtual hand and the participants active movement.

D. Motor-decode Algorithms

Two motor-decode algorithms were implemented in MATLAB 2017B: the MKF previously outlined in [6], and a convolutional neural network that will be described in detail in this section. The MKF was trained using all of the acquired training signals described above. For the CNN, 50% of the data was used for training and the remaining 50% was used for validation. Training automatically terminated once the root-mean-squared-error (RMSE) on the validation data increased (i.e., stopping early before the CNN overfit the training data). The CNN was trained using a Stochastic Gradient Descent with Momentum solver with an initial learning rate of 0.001.

The CNN input was an Nx10 image consisting of N features (528 for sEMG data and 720 for iEMG + Neural data) sampled at the current time and nine previous time points (samples are acquired at 30 Hz). The CNN architecture (Fig. 2) consisted of a single convolutional layer, two fully-connected layers, ReLu activation between layers and a regression output. A 1x5 kernel was used for convolution, such that the convolution was only across time and not across the feature set. A total of 10 convolutional filters were used to produce a Nx6x10 output feature map. The output of the convolutional layer was then passed through a ReLu activation layer before being passed to the first fully-connected layer. The output of the first fully-connected layer was also passed through a ReLu activation layer before being passed to the second fully-connected layer. Both fully-connected layers consisted of 2N neurons, and thus the total size of the network depended on the number of input features used. The output of the second fully-connected layer

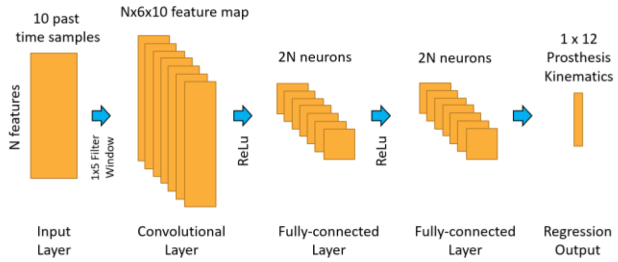

Figure 2: Architecture of CNN. For the two intact participants, the input features consisted of 528 myoelectric signals recorded via surface EMG electrodes. For the amputee participant, the input features consisted of 528 myoelectric signals recorded via intramuscular EMG assemblies and 196 neural signals recorded from multiple, small sets of motor fibers with intrafascicular electrodes. See text for additional details.

was then fed into a final fully-connected layer that produced a total of 12 regression outputs, one for each possible DOF in the prosthetic hand. For this study, the output of only six of the 12 possible DOFs were used.

The final output of the CNN was modified using a 20% threshold, similar to what was used for the MKF in our previous work [6].

E. Real-time, Extended Hand-Matching Task

To quantify the run-time performance of the motor-decode algorithms, the participants performed an extended hand-matching "test" task, in which they actively controlled the virtual MSMS hand and attempted to move select DOF(s) to a target location. In contrast with the end-point positions used for training the algorithms (the maximum unidirectional range of the virtual prosthetic hand), the test target positions were located at 50% of the unidirectional maximum range in order to evaluate proportional control. The participant was instructed to hold the selected DOF(s) as close as possible to the target location for as long as possible. The subject had visual feedback confirming when each DOF was within ±5% of the full bidirectional movement range for the desired target position for the DOF(s) to be moved, and for the rest positions for the other DOFs to be held stationary.

Each test trial lasted 7 s, and there was a 10-s wait time between trials to avoid fatigue for the amputee (1 s for intact individuals). A total of 17 movements were tested with this extended hand-matching task. These included the 14 movements that were part of the original training dataset, and an additional three novel, untrained movements (simultaneous extension of D1, D2 and D3; simultaneous flexion of D1 and D2; and simultaneous flexion of D1, D2 and D3).

For each participant, we performed three experiments on three separate datasets: 1) Extended-hold (5-s) vs. brief-hold (1-s) training for MKF vs. CNN; 2) Randomized vs. sequential training for MKF vs. CNN; and 3) Extended-hold and randomized training for MKF vs. CNN. Each of these datasets was collected in one or two experimental sessions. Collections of initial training data were performed first, followed by pseudorandom, counterbalanced testing of the different experimental conditions for both the MKF and CNN in order to avoid order effects. For each individual movement, a total of four trials were collected for the intact participant (two trials for the amputee participant to avoid excessive fatigue).

F. Performance Metrics

We measured the run-time performance of the motor-decode algorithms using three metrics: 1) the mean longest continuous-hold duration within the desired 10%-error window around the target location; 2) RMSE of each DOF compared to the target position during intended movements; and 3) the RMSE of each DOF compared to the natural resting position during unintended movements (i.e., cross-talk). To calculate RMSE during intended and unintended movements, devoid of the participant's reaction time, we aligned the kinematic output from the motor-decode algorithms with the desired target location by shifting the kinematic output by a lag that was determined by cross-correlation. This alignment was applied across all experimental conditions for a given session, so that there would be no bias affecting one experimental condition more than another.

Outliers in the performance metrics (more than 1.5 interquartile ranges above the upper quartile or below the lower quartile) were removed from the data. Aggregate data were generated by pooling the results from the three participants. A three-way ANOVA (factors: training hold-time, movement order, and decode method) was performed for each participant individually for each of the three performance metrics (continuous-hold time, intended movement RMSE, and cross-talk RMSE), treating each subject as a separate case-study. Subsequent pairwise comparisons were performed using a Holm-Šídák-Bonferroni correction for multiple comparisons.

III. Results

A. Increasing Hold-time During Training Improved Performance for All Participants

Increasing the duration of the hold-time significantly improved performance on each of the three performance metrics for each of the three participants individually (all nine p's < 0.001, three-way ANOVAs; Fig. 3, pairwise comparisons after sequential training).

B. Training Movement Order Had Only Modest and Variable Impacts on Performance

The order of training movements (random or sequential) had only a relatively modest and variable impact on performance. With some exceptions in either direction, the pattern of results for the three performance metrics for individual subjects after random training was similar to that after sequential training. For example, increasing the training hold-time improved the MKF performance in nine of nine cases after sequential training (sign comparisons for the three performance metrics for the three participants), and in eight of nine cases after random training. Similarly, for the CNN decode, increasing the training hold-time improved performance in nine of nine cases after sequential training (sign comparisons for the three performance metrics for the three participants), and in eight of nine cases after random training. Finally, after extended-hold training, the effects of decode type (MKF or CNN) on performance were the same for random or sequential training in all nine possible cases (sign comparisons for the three performance metrics for the three participants). Given this similarity, analyses reported herein focus primarily on results from sequential training.

C. MKF Outperformed CNN for Targeted Movements under Extended-hold Training Conditions, but CNN had Less Cross-talk

For all three participants, the type of motor-decode algorithm (MKF or CNN) had a significant effect on all three performance metrics (all nine p's <0.001, three-way ANOVAs), although the effect of decode type was different for the three performance metrics (Fig. 3). Additionally, the effect of motor-decode type was often significantly greater after training with extended hold-times than after training with brief hold-times ($p < 0.05$ in seven of nine cases for the three performance metrics for the three subjects, with similar but not significant trends for the other two instances; Fig. 3).

For the continuous hold-time and intended movement RMSE metrics, increasing the training hold-time typically improved performance of the MKF algorithm preferentially, relative to effects on CNN performance. After training with brief hold-times (0.1 s), the MKF and CNN performed comparably overall both in the aggregate data and for all three participants for these two performance metrics (Fig. 3, top two rows; no significant differences in any of the eight pairwise comparisons between MKF and CNN after brief-hold (0.1-s) training). In contrast, after training with extended hold-times (5 s), the MKF significantly outperformed the CNN overall for both performance metrics in the aggregate

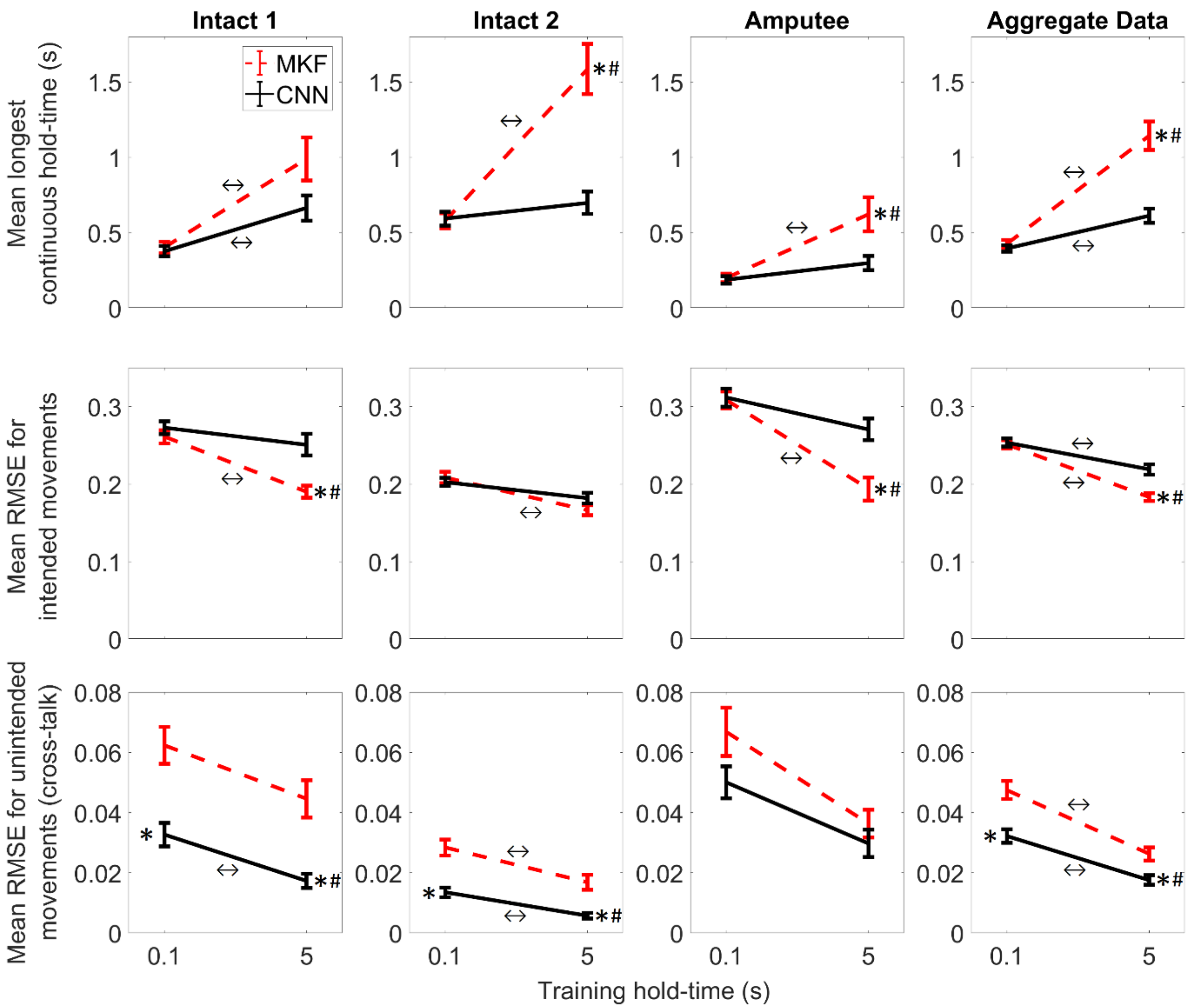

Figure 3: Mean (± SEM) longest continuous-hold times (top row), RMSE for intended movements (middle row), and RMSE for unintended movements (i.e., crosstalk) (bottom row) for the modified Kalman filter (MKF) and convolutional neural network (CNN) under brief-hold (0.1-s) and extended-hold (5-s) training conditions. Results are presented separately for each of the three participants, and for aggregate data pooled across subjects (columns). Training with extended hold-times improved performance on all three metrics. After extended-hold but not brief-hold training, the MKF decode significantly outperformed the CNN decode on two performance metrics: continuous hold-time and RMSE for intended movements (Aggregate Data; similar patterns for individual participants). In contrast, for RMSE cross-talk, the CNN outperformed the MFK decode overall and for the two intact subjects, regardless of the training hold-time. Within each panel, * indicates a statistically significant performance difference between MFK and CNN performance for a particular training hold-time condition (using Holm-Šídák-Bonferroni corrections for the six possible multiple comparisons within each panel); # indicates significant performance difference for that experimental condition, compared with all three other training conditions; and ↔ indicates a significant performance difference between 0.1-s and 5-s training condition for that decode type.

data, and for most (four of six) cases for individual subjects. Further, in the aggregate data, the performance in the extended-hold, MKF-decode condition was significantly better than the performance in each of the other three experimental conditions for both the continuous hold-time and intended movement RMSE metrics (Fig. 3, multiple pairwise comparisons).

In contrast, for the cross-talk RMSE performance metric, results for decodes were nearly the opposite: the CNN significantly outperformed the MKF (i.e., exhibited less cross-talk) after both brief- and extended-hold training, both for the aggregate data across all three subjects and for the two intact participants individually (Fig. 3, bottom row; six of six pairwise comparisons). Results for the amputee showed a strong similar trend that did not quite reach statistical significance after corrections for multiple comparisons (see Discussion).

IV. Discussion

A. Primary Findings and Potential Mechanisms

Increasing the hold-time during training improved performance across all three metrics for all three participants. For the CNN, we attribute this improvement in performance to the ability of the non-linear neural network to capture the temporal variation and non-linearities of EMG signals acquired during extended muscle contractions. We hypothesize that sampling the EMG signals across longer durations of contraction allows the CNN to attribute multiple, different signals to a single kinematic position.

This same reasoning cannot be applied as readily to the improvements seen with the MKF, given that the MKF is built from a linear Kalman filter. Instead, we propose that the MKF's improved performance stems from an overall reduction in Kalman gain, which, for all participants, was significantly less (all *p's* < 0.05, paired t-tests) when trained on extended (5-s) hold-times, relative to brief (0.1-s) hold-times. With a lower Kalman gain, the MKF puts more emphasis on the current state estimate relative to the new state predicted by the incoming EMG/neural feature data. We hypothesize that training on extended hold-times introduces temporal variation to the signals, making the features less reliable predictors of the kinematic state, which in turn results in a lower Kalman gain. The drop in Kalman gain then serves as a smoothing function, removing jitter from the EMG signal but reducing its overall temporal responsiveness. Additional tests that quantify speed and responsiveness against accuracy and reliability (Fitts' Law) should be performed to evaluate this potential mechanism.

The variable, participant-specific improvements associated with the order in which movements were presented during training might have arisen from the introduction of within-sample variations. Randomizing the movement order may result in within-sample variations due to differences in neural activation patterns between initial motor planning and repeated motor action, or due to the hand posture changes after individual movements. In principle, the CNN could use this variation to improve its overall generalization and robustness. Similarly, the MKF could have seen improved performance if the introduction in within-sample variance resulted in a substantive reduction in Kalman gain. Additional analyses are needed to determine if the variance was consistently different between the random and sequential movement orders.

Under the extended-hold training condition, the MFK outperformed the CNN for both the continuous hold-time and intended movement RMSE. However, the CNN had less cross-talk RMSE than did the MKF. Thus, what decode algorithm performs "best" depends in part on which performance metric is used for evaluation. Which is more important: accuracy of the intended movement, or less interference on other, unintended movements? Preferences may vary across tasks and individuals. Additionally, performance on other metrics not explicitly evaluated here (e.g., speed; cognitive effort; generalization to novel, untrained movements) may also influence individual preferences.

Overall, this CNN resulted in a noisier motor decode than did the MKF. Although the CNN was better able to isolate individual DOFs, movement of the isolated DOFs was more jittery, resulting in higher RMSE and shorter continuous hold-times. This factor, coupled with the smoothing capabilities of a decreased Kalman gain and the incorporation of thresholds into our MFK, may help explain why the MKF often outperformed the CNN in the present studies.

Direct comparisons among the motor-decode algorithms used here and those reported in previous works is difficult due to differences in the number of controllable DOFs and the complexity of the task. In order to effectively compare the performance across algorithms, clinical hand dexterity tasks should be performed in the near future [11].

B. Caveats and Future Work

The present data set should be extended to confirm generalizability and reproducibility of results across participants, and across training sessions for individual participants. The two intact participants were members of our research laboratory (including one contributing author) and may not be representative of a general population. Only one person with an amputation was tested. Further, the aggregate data were pooled across all three participants. That aggregation approach allows for the evaluation of results obtained with this subject pool, but precludes estimates of effects across subjects and generalization to other individuals.

Although the present limited dataset did not demonstrate powerful effects of training movement order (random or sequential), further investigations may identify circumstances in which one approach or the other would be advantageous.

The testing sessions used a target position different from that used to train the algorithms. Nonetheless, proportional control should be more fully explored by testing at a variety of different target positions.

The RMSE values here reflect the error calculated after temporal alignment of the decode output and the target position, which was intended to reduce effects of factors such as participants' reaction times. RMSE values calculated without this alignment would be slightly larger, but the same correction was applied across the different experimental conditions within a session to avoid biasing results.

The present results represent only an initial attempt to implement a CNN in real time. This CNN differs from a previously reported CNN [9] in terms of network architecture, input features, number of training trials, use of thresholds, testing metric (offline vs online), and training algorithm. In future work, training algorithms that result in preferential improvement to CNNs, such as Dataset Aggregation [9], along with other possible modifications and more sophisticated architectures [8], may yield improved performance for CNNs relative to MKFs. Our MFK has been improved over years of use and iterations, providing it unequal advantages relative to this first iteration of a relatively simple CNN. We do not conclude from our initial comparisons examined here that in general MKFs will necessarily continue to outperform improved CNNs (or other decode algorithms).

C. Motor-decode Performance after Recent Hand Amputation, Prior CRPS and Long-term Hand Disuse

Direct comparisons between the intact subjects and the amputee are difficult due to multiple confounding factors. The decodes for the amputee used two types of motor signals: myoelectric signals recorded via intramuscular EMG assemblies, and neural signals recorded from multiple, small sets of motor fibers with intrafascicular electrodes. In contrast, decodes for intact subjects used only signals recorded with surface EMG electrodes, which typically have greater crosstalk and lower quality than do signals recorded from implanted devices, and also have less long-term stability compared with signals from intramuscular EMG electrodes. Nonetheless, performance of intact subjects was nearly always better than performance of the amputee (23 of 24 direct comparisons for sequential training, $p < 0.00001$, binomial test).

Several factors likely adversely affected the performance of the amputee participant. Notably, prior to his amputation, this participant had suffered from CRPS that led to long-term disuse of his affected arm. Consequently, his arm muscles were weak and fatigued readily, and his myoelectric signals were relatively small, which decreased the signal-to-noise ratio for EMG recordings. Residual pain from CRPS, post-operative pain, and phantom pain may have further inhibited his movements. Additionally, data collection sessions for this participant involved half the trials per condition to avoid muscle fatigue, reducing statistical power.

Nonetheless, even shortly after the amputation, and with only modest training, the amputee participant was successfully able to achieve real-time, proportional control of a six-DOF hand. This high level of dexterous performance exceeds the capabilities of current commercially available myoelectric prostheses. His successes, despite multiple challenges and adversities, provide a striking and poignant demonstration of the practical benefits of the present approaches, and suggest that these benefits may be extended to a wider population of subjects, including those with recent amputations and with prior CRPS.

V. Conclusion

This work highlights the importance of optimizing training paradigms to yield online improvements in motor-decode algorithms. In addition, this work demonstrates novel use of a CNN by an amputee for real-time, proportional six-DOF control of a prosthetic hand. Furthermore, it presents a unique case-study in which an individual with CRPS and multi-year disuse of their intact hand, underwent an amputation and received neuromyoelectric implants simultaneously, and then subsequently controlled an advanced prosthetic hand.

Acknowledgment

This work was funded by: DARPA, BTO, Hand Proprioception and Touch Interfaces program, Space and Naval Warfare Systems Center, Pacific, Contract No. N66001-15-C-4017; NSF Award No. ECCS-1533649; and NSF GRFP Award No. 1747505; additional support provided by the Veterans Administration and the University of Utah.

References

[1] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the prevalence of limb loss in the united states: 2005 to 2050," *Arch. Phys. Med. Rehabil.*, vol. 89, no. 3, pp. 422–429, Mar. 2008.

[2] C. G. Bhuvaneswar, L. A. Epstein, and T. A. Stern, "Reactions to amputation: recognition and treatment," *Prim. Care Companion J. Clin. Psychiatry*, vol. 9, no. 4, pp. 303–308, 2007.

[3] D. M. Desmond and M. MacLachlan, "Affective distress and amputation-related pain among older men with long-term, traumatic limb amputations," *J. Pain Symptom Manage.*, vol. 31, no. 4, pp. 362–368, Apr. 2006.

[4] E. A. Biddiss and T. T. Chau, "Upper limb prosthesis use and abandonment: A survey of the last 25 years," *Prosthet. Orthot. Int.*, vol. 31, no. 3, pp. 236–257, Sep. 2007.

[5] E. Biddiss and T. Chau, "Upper-limb prosthetics: critical factors in device abandonment," *Am. J. Phys. Med. Rehabil.*, vol. 86, no. 12, pp. 977–987, Dec. 2007.

[6] D. M. Page *et al.*, "Motor control and sensory feedback enhance prosthesis embodiment and reduce phantom pain after long-term hand amputation," *Front. Hum. Neurosci.*, in review.

[7] K. Anam, A. A. Rosyadi, B. Sujanarko, and A. Al-Jumaily, "Myoelectric control systems for hand rehabilitation device: A review," in *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2017, pp. 1–6.

[8] P. Xia, J. Hu, and Y. Peng, "Emg-based estimation of limb movement using deep learning with recurrent convolutional neural networks: emg-based estimation of limb movement," *Artif. Organs*, Oct. 2017.

[9] H. Dantas, V. J. Mathews, S. M. Wendelken, T. S. Davis, G. A. Clark, and D. J. Warren, "Dataset aggregation-based movement decoders for emg signals," *IEEE Trans. Biomed. Eng.*, in review.

[10] S. Wendelken *et al.*, "Restoration of motor control and proprioceptive and cutaneous sensation in humans with prior upper-limb amputation via multiple Utah Slanted Electrode Arrays (USEAs) implanted in residual peripheral arm nerves," *J. NeuroEngineering Rehabil.*, vol. 14, p. 121, Nov. 2017.

[11] J. L. Segil, M. Controzzi, R. F. ff. Weir, and C. Cipriani, "Comparative study of state-of-the-art myoelectric controllers for multigrasp prosthetic hands," *J. Rehabil. Res. Dev.*, vol. 51, no. 9, pp. 1439–1454, Oct. 2014.