Mapping Intrinsic and Extrinsic Muscle Myoelectric Activity During Natural Dynamic Movements into Finger and Wrist Kinematics using Deep Learning Prediction Models

Marcus Panchal, Simone Tanzarella, Moon Ki Jung and Dario Farina *Fellow, IEEE*

***Abstract*— Objective: We investigate the use of high-density surface EMG (HDsEMG) recordings of intrinsic hand muscles, along with those from extrinsic muscles, on finger and wrist kinematic prediction performance. We incorporate these HDsEMG signals using a framework based on a custom hybrid convolutional-recurrent deep learning model. Methods: Five healthy subjects performed a wide variety of motion tasks activating multiple degrees of freedom of the wrist and fingers. During the tasks, HDsEMG signals were recorded from extrinsic and intrinsic muscles of the hand while motion capture technology tracked the hand/wrist kinematics. A convolutional-recurrent model architecture was designed and trained on the recorded dataset, incorporating both residual connections as well as inception convolutional structures. Results: The proposed model led to greater regression accuracy over the simultaneous prediction of 12 joint angles (CC, MAE and RMSE of 0.850, 4.84 degrees and 11.2 degrees respectively) than previously proposed mapping models, when incorporating both intrinsic and extrinsic muscle signals. The inclusion of both sets of hand muscles also led to statistically greater performance than the same model trained on only extrinsic muscle data. Conclusion: We show accurate predictions of hand/wrist kinematics from combined extrinsic and intrinsic hand muscle myoelectric activity, using a convolutional-recurrent hybrid deep learning model. This greater performance is replicated over several subjects and across multi-degree of freedom motion tasks. Significance: Our developed system (electrode setup and deep neural networks) can be translated into a compact wearable interface in the future for medical as well as consumer applications.**

***Index Terms*— HDsEMG, Deep Learning, Hand Kinematics, LSTM, Convolutional Neural Networks**

I. Introduction

Man-machine interfaces enable the decoding/encoding of sensorimotor information to control external devices or modulate neural structures [1]. Applications range from gaming [2], [3], remotely operated devices [4], and medical systems, such as exoskeletons [5] and active prostheses [6]. When used to decode motor intention, man-machine interfaces are designed to map biological/neural signals into degrees of freedom.

The decoding of user intent may be performed from a variety of biosignals. The most commonly used signal for man-machine interfacing is the electromyographic (EMG) signal [7] - a measure of the depolarisation of the muscle fibres recruited during movements. The surface EMG (sEMG) signal is non-invasive and simple to record [7]. It indirectly reflects the neural drive sent from the spinal cord to the muscles [8], [9]. While sEMG signal data have been conventionally measured with a few electrodes, more recent approaches consist of the recording of tens to hundreds of sEMG signals concurrently, i.e. high-density sEMG (HDsEMG) [10]. HDsEMG provides a two-dimensional spatial and one-dimensional temporal recording [10].

Classic approaches to myoelectric control involve manual feature extraction from the raw sEMG signal followed by a machine learning algorithm to learn the mapping between the signal features and the intended control outcome [8], [11]. The mapping of sEMG to upper limb kinematics has classically been performed by either classification or regression techniques [6]. Regression techniques used for sEMG processing include linear regression methods [12], auto-encoders [13], hidden Markov models [14], and artificial neural networks [15] algorithms.

Deep learning algorithms can learn important features within a training dataset without the need for these features to be hand-engineered [16]. Lower-level features are usually learned by the initial layers, and are then combined hierarchically to form higher level features/representations of the data in the later layers of the model [16]. The depth of these networks, which are usually built up of several layers of the same neural units, is an important factor of influence for performance [17]. Deep learning has been used for kinematic prediction from sEMG signals in several areas of research, with improved accuracy compared to classical machine learning techniques [18]. Typical architectures use either convolutional layers [18], [19], with kernel weights that convolve over multi-dimensional inputs (spatial and temporal axes), or recurrent networks [19], [20], which allow previous time step predictions to influence the current prediction. These two architectures have also been

Marcus Panchal is with the University of Leicester Medical School, Leicester, UK (e-mail: mop2@student.le.ac.uk) and the Department of Bioengineering, Imperial College London, London, UK (e-mail: mop18@ic.ac.uk).

Simone Tanzarella, Moon Ki Jung and Dario Farina are with the Department of Bioengineering, Imperial College London, London, UK (e-mail: simone.tanzarella17@imperial.ac.uk; m.jung@imperial.ac.uk; d.farina@imperial.ac.uk).

used together [21].

The input data used in deep learning approaches for hand kinematic prediction has generally focused on the measurement of sEMG signals from extrinsic hand muscles, since the main application of these models is the control of upper limb prostheses by trans-radial amputees [22]. However, in addition to the extrinsic hand muscles, the intrinsic muscles also contribute to hand fine motor control [23], especially with regards to control of the thumb kinematics, which provide a large degree of hand dexterity [24]. The measure of intrinsic muscles for myocontrol has relevance in general purpose control systems, e.g. gaming or interaction with virtual reality environments.

Here, we investigate a deep neural network mapping approach for the prediction of hand/wrist kinematics when recording HDsEMG signals from both intrinsic and extrinsic hand muscles (Figure 1). The application of this work can lead to the development of a general man-machine interface for consumer electronics as well as medical applications.

II. Methods

A. Data Collection

1) High Density sEMG Recording Setup: Overall, five, right-handed, participants (age: 27.2 $\pm$ 3.3 yrs; weight: 74.6 $\pm$ 7.1 kg; height: 179 $\pm$ 6.7 cm) volunteered for the experiments. Their hand/wrist kinematics and extrinsic/intrinsic hand muscle HDsEMG data were recorded while performing various hand motion tasks. All subjects were required to complete a participation and ethics consent form in accordance with the declaration of Helsinki and approved by the Imperial College London Research Ethics Committee (approval no. 18IC4685).

HDsEMG grid signals were recorded using a Quattrocento EMG amplifier (OT Bioelettronica^{TM}, Turin, Italy) at a sampling frequency of 2048Hz. Six 64-channel HDsEMG grids (384 channels in total) were used, in the same configuration described by Tanzarella et al [23], [25], with the electrodes of each grid arranged in a 13x5 spatial orientation (Figure 2a). Two larger grids with 8-mm inter-electrode distance (IED) were placed over the extrinsic flexor/extensor muscles. Four grids with 4-mm IED were placed over the first dorsal interosseous (FDI); the other three dorsal interossei (II DI, III DI, and IV DI); the thenar muscle group; and the abductor digiti minimi.

2) Motion Capture: Hand/wrist kinematics were recorded using motion capture technology to produce kinematic data at a sampling frequency of 250Hz. A total of 19 motion capture markers were used at the following locations on the right hand of each subject (Figure 2c) to allow the requisite joint angles to be calculated: Medial epicondyle of elbow (EpicM); Lateral epicondyle of elbow (EpicL); Radial styloid (Rad); Ulnar styloid (Uln); Metacarpo-phalangeal (MCP) joint of thumb (T3); Inter-phalangeal (IP) joint of thumb (T2); Tip of the thumb (T1); MCP joint of index finger (I3); Proximal inter-phalangeal (PIP) joint of index finger (I2); Distal inter-phalangeal (DIP) joint of index finger (I1); MCP joint of middle finger (M3); PIP joint of middle finger (M2); DIP joint of middle finger (M1); MCP joint of ring finger (R3); PIP joint of ring finger (R2); DIP joint of ring finger (R1); MCP joint of little finger (L3); PIP joint of little finger (L2); DIP joint of little finger (L1).

Eight motion capture cameras, utilising BTS motion capture software (BTS S.p.A., Milan, Italy), were used to record the marker positions. The subject was seated while performing the hand motion tasks, with their arm suspended horizontally using a custom-built metal rig (Figure 2b). Motion capture and HDsEMG recordings were synchronized via a custom control program written in the MATLAB programming language.

3) Recording Protocol: The motion tasks performed were grouped into four categories: single degree of freedom (DoF) motions (flexion/extension of each MCP/PIP/wrist joint and abduction/adduction of the thumb MCP joint), two-DoF motions (flexion/extension of each MCP joint while the wrist joint was either flexed or extended), natural motions (hand grasp, three/two finger pinch), and complex motions (10 random sign language letters).

It was hypothesised that by using a wide variety of motions, including a mixture of single and multi DoFs, the developed models would not only learn simple single DoF control but would also show increased performance in multi-DoF prediction, utilising and combining the information learned for single DoF tasks hierarchically rather than starting from the beginning. The wide variety of tasks also increased the sampling of the HDsEMG input and kinematic output space, improving the ability to generalise to varying DoF motions.

In total, 11 different types of hand motion tasks were recorded for each subject, with each task lasting for either 1 min or 30 s depending on the motion (Supplementary Material 1A). Tasks consisted of 5 consecutive joint motion repetitions, with each repetition lasting 6 seconds, which included the joint being held at the extreme of its range of movement for 1 second. Supplementary Material 1A provides further details of the motion tasks performed by each subject.

B. Data Processing

Processing of the HDsEMG and kinematic data was performed with the Python programming language [26], utilising the SciPy and Numpy scientific packages.

1) sEMG Pre-Processing: Each HDsEMG channel was filtered (2^{nd} order Butterworth) between 20 Hz and 500 Hz, to remove noise and artifacts from the signal. The signal of each HDsEMG grid was then scaled so that the amplitude of each channel of the grid was between -1 and 1 using min-max scaling (Eq. 1). Scaling enabled improved model training performance [27], as constraining the input space allows for smaller gradients during backpropagation, resulting in stable training with less chance of diverging model weights. Due to artefacts/outliers in the recordings, the scaling method (Eq. 1) was modified to use $x_{grid_max/min} = \mu_{grid} \pm (\delta * \sigma_{grid})$ as the maximum and minimum amplitude values, where $\delta = 20$ was determined empirically and $grid_max/min$ is the maximum/minimum signal value for a single sEMG electrode grid. Any signal amplitudes outside this new maximum and minimum range were bounded at 1 or -1, respectively. This resulted in a wider range of amplitudes after scaling, maintaining a large sampling of the input space.

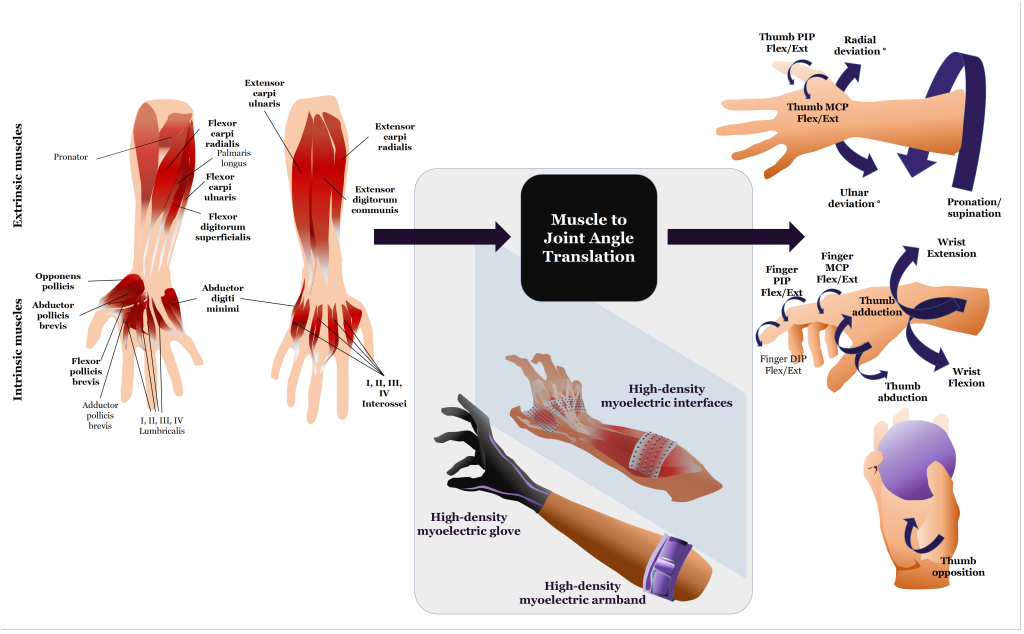


Fig. 1. Intrinsic and extrinsic muscles of the hand from which HDsEMG data were recorded. The degrees of freedom predicted by our model are also indicated.

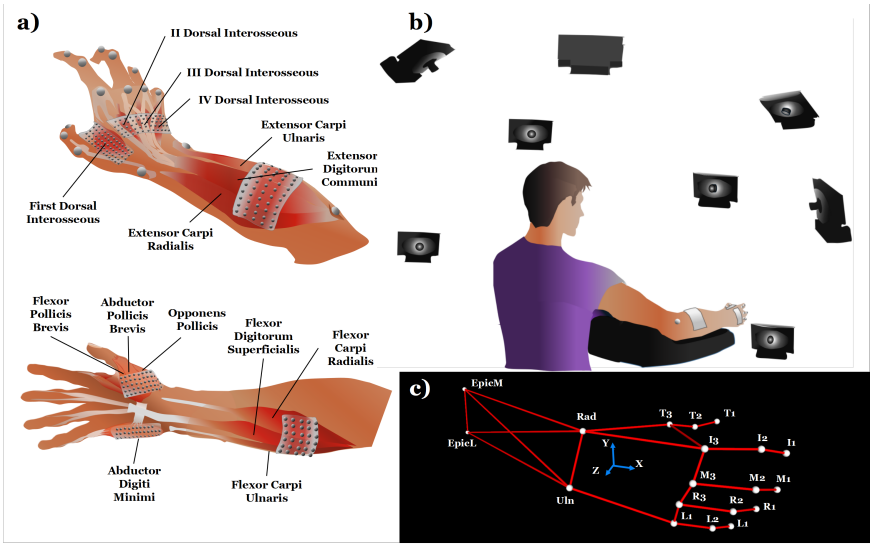


Fig. 2. Experimental setup for High Density Surface EMG recordings and the Motion Capture. **a)** The positioning and orientation of the 6 HDsEMG Grids over the hand and forearm. The names of the underlying muscles being recorded are written above each grid for reference. **b)** The positioning of the subject with respect to the motion capture cameras, which were specifically placed in order to capture a volume of space around the right hand and forearm. The subject was allowed to rest their arm upon a custom-made support during the experiment to avoid fatigue. **c)** The motion capture markers used to capture landmarks of the hand and forearm as visualised in a wireframe model during post processing of the motion capture data. The white spheres indicate the markers and the red connecting lines are the anatomical relationships between them.

$$x_{scaled} = 2 * \frac{(x - x_{grid_min})}{(x_{grid_max} - x_{grid_min})} - 1 \qquad (1)$$

2) Motion Capture Processing: The motion capture kinematic data were processed to extract a time series of joint angles from the 3D positional data. The first step involved manual marker editing, consisting of labelling and tracking 3D marker positions through time and removing artifacts from each marker. Additional automated processing was performed on the motion capture kinematics data before the calculation of joint angles (Supplementary Material 1B).

In total 12 joint angles (x10 finger and thumb MCP/PIP flexion-extension angles, x1 thumb MCP abduction-adduction angle and x1 wrist flexion-extension angle) were calculated through vector analysis from the post-processed motion capture kinematics marker data (Supplementary Material 1C). Angles were calculated in radians, limiting the range of values to a small band $(-\pi, \pi)$, for the same reasons min-max scaling was applied to the HDsEMG signal.

C. Deep Learning Model Architecture

1) Architecture Overview: To evaluate the effect of incorporating the intrinsic muscle HDsEMG signal into deep learning predictive frameworks, we devised a deep learning based model architecture that contains neural network layers based on the current literature. As previous work in the literature has only focused on deep learning models utilising extrinsic muscle data, we therefore had to design a new custom model architecture to be able to incorporate the HDsEMG grid signals from multiple spacially separated locations (intrinsic and extrinsic) appropriately.

Our design firstly involves an initial model to extract features from the input HDsEMG signal time window of each electrode recording grid (Figure 3). Once features were extracted from each grid, they were concatenated and passed through a final combinatory deep learning model (Figure 3) to form a prediction of 12 output joint angles. This final model design is the result of significant experimentation with several different layer architecture types and methods for combining the signals from both intrinsic and extrinsic HDsEMG electrode grids. This included Transformer Decoder based designs; models that combined all the grids into one single input matrix; models only using LSTM or Convolutional layers; and models with the input sEMG grid signals dimensionally reduced through PCA.

2) Input Grid Feature Extractor Architecture: Inputs to the model are three dimensional, consisting of the two spatial axes of the 13x5 electrode grids and the time window of the signal as the third axis. As the average duration for a motor unit action potential (MUAP) is 10-15ms [28], a time window length of 200ms was chosen so that it allowed multiple MUAP's to be present. Moreover, this interval accounted for the electromechanical delay between stimulation and contraction of the muscles [18], [29], [30].

The feature extractor models consisted of several 3D/2D Inception Resnet blocks (Figure 3). These blocks performed 1x1x1, 3x3x3 and 5x5x5 kernel convolutions simultaneously on the input to the block, with the output features then concatenated together and a final 1x1x1 convolution applied to reduce the output dimension [31] (Figure 3). This allowed the optimum kernel size for each block to be learned, as well as enabled a greater number of features (including different feature lengths) to be learned from a single block [31]. We assumed that the learned features were related to the motor unit action potential spike trains innervating the underlying muscles, as many blind source separation algorithms for decomposing HDsEMG signals into spike trains rely on convolutional operators [32]. However, this relationship is not explored further within this work.

Residual connections [33] within the Inception Resnet blocks propagate the input data of the block to the output (Figure 3). This allows a large number of neural network layers without detriment to the training process. A Squeeze and Excite layer was also incorporated into each Inception Resnet block, as described in [34] (Figure 3), allowing us to add additional complexity to the model, which significantly improved performance with little computational overhead [34].

The initial Inception Resnet blocks implemented 3D kernel convolutions over the spatial and temporal dimensions of the sEMG grid input. This changed to 2D kernel convolutions in the later blocks (with the kernel filters modified accordingly by the removal of the last axis) when the size of one of the spatial axes of the data became reduced to 1. The stride of the kernel filters was kept to 1, with zero padding being applied to the inputs of the layer in all blocks. A leaky ReLU activation function was applied after each convolutional layer, with $\alpha = 0.1$, as suggested in previous work [18]. This function does not saturate, allowing the propagation of negative layer output values through the network.

Between each Inception Resnet block, a Batch Normalisation layer was used to normalise the output data from the block by statistical re-centering and re-scaling of the features. This significantly speeds up the training time while improving the model performance [35]. Average pooling layers were used on both spatial and temporal axes to reduce the dimensionality of the data passing through the model. This constrained memory resources and removed unnecessary information by forcing lower dimensional features to be extracted.

A different weighted feature extractor model, with the same architecture, was used for the HDsEMG signal from each grid to allow for different temporal and spatial features to be learned. This aimed to account for varying signal qualities at different anatomical locations.

3) Feature Combination Architecture: The features extracted from each extractor model were then concatenated together along the spatial axis to form a single data structure which was the input into a final combination model architecture. This was designed to coalesce the information contained from the features of each recording grid signal and form a prediction of the joint angle configuration. The feature combination model (Figure 3) consisted of two recurrent LSTM layers [36], followed by two fully connected dense neural network layers. The first LSTM layer returns a sequence of 128 dimensional features while the second only returns a single feature of dimension 64. Dropout, with a probability of inputs being dropped of $p = 0.2$, was used within these layers to reduce overfitting. The final fully connected layer has a linear activation function to allow a continuous, unbounded and non-wrapped set of angles to be predicted. This output vector was passed through a 2nd order low-pass Butterworth filter, removing frequencies above 2 Hz, and thereby providing a smooth output signal for control systems.

4) Model Variants: In addition to the model described above, which uses both extrinsic and intrinsic HDsEMG signals as input (**InceptionConvLSTM**), we also trained models that used only extrinsic HDsEMG signals as input (**InceptionExtrinsic**); only intrinsic HDsEMG signals as input (**InceptionIntrinsic**); and that had more convolutional layers in each Inception Resnet block (**InceptionLargerModel**). This allowed us to observe the effect of including the intrinsic HDsEMG signal on model performance and ascertain whether the size and depth of our devised networks were optimal for the problem, while keeping the architecture of the base model layers constant. We do not train models devised by previous works in the literature as the difference in model architecture would not allow for

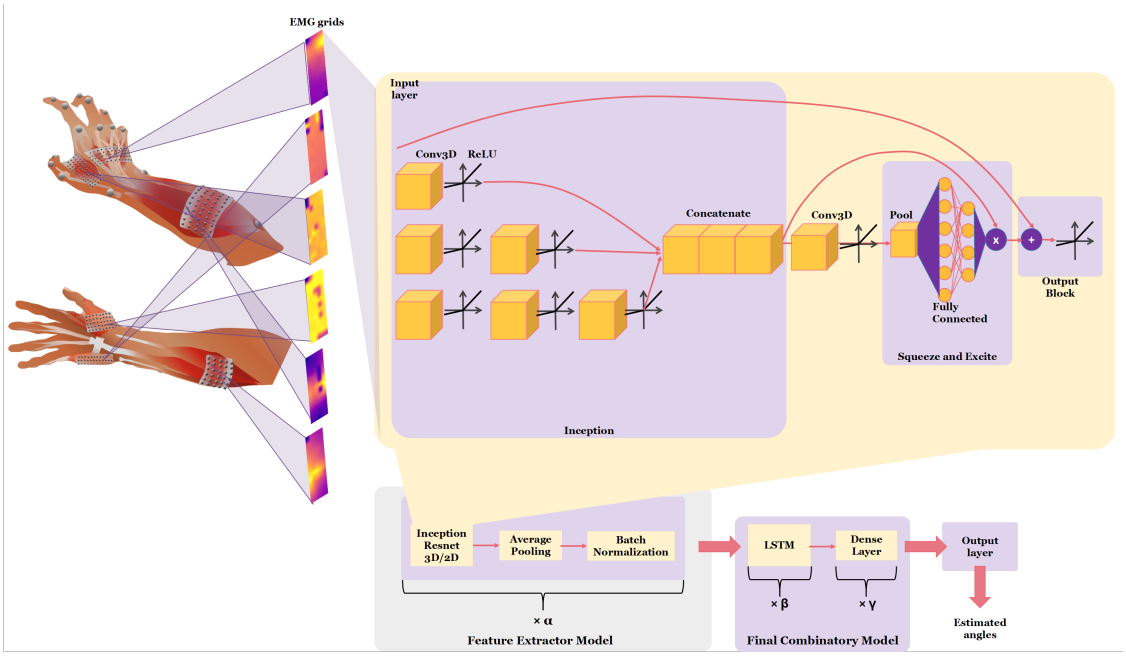


Fig. 3. Overview of the base model architecture. Each grid input is passed through its own feature extractor model consisting of a number (α) of inception resnet 3D/2D blocks [31]. The final combinatory model consists of a number (β) of LSTM layers followed by a number (γ) of dense fully connected layers.

fair comparison with our models, which were designed with the intention of utilising both sets of hand muscle activity for input.

We also experimented with how best to amalgamate features extracted from HDsEMG signals recorded at distinctive, spatially separated, anatomical sites. We therefore also implemented a different version of our base model where the feature extractor models directly predict only the angles that their input muscle HDsEMG signal was responsible for actuating (**InceptionModifiedCombination** i.e, grid 4 would only predict thumb joint angles due to this grid only measuring HDsEMG signal from the thenar muscles). These predicted angles from each grid were then concatenated together respectively (e.g, thumb angle predictions from grid 4 were combined with thumb angle predictions from grid 6 on the forearm) and passed through three fully connected dense layers in order to combine them into a final prediction for each angle. We hypothesised this method could improve performance, as the features extracted from each recording grid would lie in the same feature space, simplifying the model's ability to combine information from each anatomical location. To predict angles from each grid, the feature extractor models for this variant were modified to include two LSTM layers and a single fully connected dense neural network layer at the end.

Supplementary Material 1D shows a concise summary of all the various models trained.

5) Training Protocol: All models were developed using the Tensorflow [37] Python package, with all data processing, training and evaluation code available in a public Github repository (https://github.com/MarcusPanchal/sHDEMG-DeepLearning-Paper). To increase the size of the training data, a sliding window was applied over the time axis of the data. A 200 ms window was shifted by 2 ms increments over the HDsEMG and motion capture data, increasing the number of input data samples to approximately 3,750,000. During training the input data was batched into 64 samples with the order of training samples randomised.

The dataset was split into training and validation data using the first 80% (first 4 Repetitions) of each motion task recording for training and the last 20% (last 1 Repetition) for validation, so that both datasets contained data from all subjects and all motion tasks recorded. Performing cross-validation for our models would have been preferred, however due to the extremely long model training times (on the order of days), this was not feasible.

Training was performed using the Imperial College, Research Computing Service GPU cluster, giving access to eight NVIDIA Quadro RTX6000 GPUs per model training job.

Model training utilised a supervised learning approach, with the HDsEMG signals as the input and the calculated motion capture joint angles as the desired output vectors. The ADAM optimiser [38] was chosen due to its ability to stabilise, and improve training speed and performance. Mean squared error was used as the loss function due to its prevalence among regression tasks and its greater penalisation of larger errors.

Each model was trained until the validation loss after 5 consecutive epochs had not decreased, stopping training before overfitting of the model occurred. Regularisation techniques, such as L2 weighting and dropout, were also initially used within the feature extractor models in anticipation of reducing model overfitting, however their inclusion led to poor model performance and so were removed. Dropout within the LSTM layers of the feature combination architecture was the only regularisation technique retained.

The training of the InceptionLargerModel model (our largest deep learning model) took on average 23.1 Hours to perform

1 epoch of training, with an input batch size of 64.

The prediction of kinematic data, using the same eight GPU hardware setup, enabled batches of 64 kinematic outputs to be predicted in approximately 1 s on average, equating to 125 ms per kinematic output due to parallel processing.

D. Evaluation Metrics

1) Classification: While the main focus of our work is the continuous regression of kinematic joint angles, we first undertook an analysis of the classification performance of our trained models. We looked at the ability of our models to predict, from their regression predictions, whether a particular joint (MCP/PIP) or digit was active within a particular task. We performed this analysis only on the single and two-DOF tasks. Classification predictions were obtained from the regression prediction data through thresholding of the standard deviation for each angle over the time axis, with this prediction being compared to the true values obtained from knowledge of the particular task being performed. This analysis was performed using an automated pipeline, with optimal thresholds for each class determined through the maximisation and minimisation of the true positive and false positive rates, respectively.

We used 2 commonly implemented classification metrics to quantify their performance: Accuracy and F1 Score.

2) Regression: We conducted our main analysis of the regression predictions generated by all five models trained over the five subjects using the following metrics to quantify performance:

- **Correlation Coefficient** (**CC** - Equation 2) - The Correlation Coefficient is a value that expresses quantitatively how correlated two variables are, which in the case of our experiment are the predicted and actual joint angles. The coefficient varies between $+1$ (Total Positive Correlation) and -1 (Total Negative Correlation) with the larger the positive value the better the performance of the model.

$$CC = \frac{\Sigma(x - x_\mu)(y - y_\mu)}{\sqrt{\Sigma(x - x_\mu)^2}\sqrt{\Sigma(y - y_\mu)^2}} \tag{2}$$

- **Mean Absolute Error** (**MAE** - Equation 3) - The Mean Absolute Error is an easily interpretable regression metric, expressed in original units, that assesses the mean of the absolute differences between the actual and predicted joint angle values, with the size of the differences weighted equally.

$$MAE = \frac{\Sigma|x_{predicted} - x_{actual}|}{N} \tag{3}$$

- **Root Mean Squared Error** (**RMSE** - Equation 4) - The Root Mean Squared Error is a common regression metric that aims to show the aggregated difference between the actual and predicted values in original units, with heavier weighting on larger differences.

$$RMSE = \sqrt{\frac{\Sigma(x_{predicted} - x_{actual})^2}{N}} \tag{4}$$

- **Normalised MAE/RMSE** (**NMAE/NRMSE** - Equation 5) - The MAE and RMSE can both be normalised to allow for easier comparison of model performance between individual joint angles, which each have different ranges within the validation task recordings. This normalisation is done via the mean interquartile range (IQR) of each joint angle across all task recordings.

$$NRMSE = \frac{RMSE}{IQR_{ActualJointAngles}} \tag{5}$$

3) Statistical Analysis: We use the one-tailed non-parametric Wilcoxon Signed-Rank test to statistically evaluate whether the InceptionLargerModel, which combines extrinsic and intrinsic HDsEMG signals, produces better metric performance than the InceptionExtrinsic model, which only includes extrinsic hand muscle data. Statistical analysis of overall performance between the two models (III-B.1) uses the metric (averaged over all the predicted joints) for each task per subject as the samples ($n = 136$). Statistical analysis of the performance between the two models for predicted joints (Supplementary Material 1F) uses the metric (averaged over all tasks and subjects) for each predicted joint as the samples ($n = 12$). All p-values are corrected through Bonferroni correction and effect sizes (ES) are determined through the matched pairs rank-biserial correlation [39].

III. Results

A. Classification Performance

Full classification results can be found in Supplementary Material 1E. All 5 model combinations performed at a high classification accuracy, with the best accuracy and F1 score achieved by the InceptionIntrinsic Model (0.96 and 0.90 respectively). The ring finger classification had lower accuracy than the other digits (best values for Accuracy and F1 score respectively - **Index:** 0.96, 0.92; **Middle:** 0.96, 0.92; **Ring:** 0.91, 0.79; **Little:** 0.99, 0.97; **Thumb:** 0.96, 0.94). While the accuracy scores for both the MCP and PIP joints were very high, the F1 scores differed significantly, with the F1 scores for the PIP joints being considerably smaller. This can be attributed in part to natural movement of the PIP joints during MCP joint tasks, which were not encoded in the true values due to our method of automated classification analysis. With this in mind, the InceptionLargerModel provided greater accuracy and F1 scores for the PIP joints compared with the other smaller models.

We have generated a ROC curve (Figure 4) for each model using micro-averaging, with all models achieving an area under the curve (AUC) of 0.98, illustrating a high classification power.

It is worth noting that the analysis of the classification performance of our models is heavily influenced by the method used to extract classification predictions from the continuous joint angle predictions. While we chose to use the simple technique of calculating the standard deviation to form classification predictions from the continuous predicted data, more advanced methods may lead to greater accuracy but that is beyond the scope of this exploratory analysis.

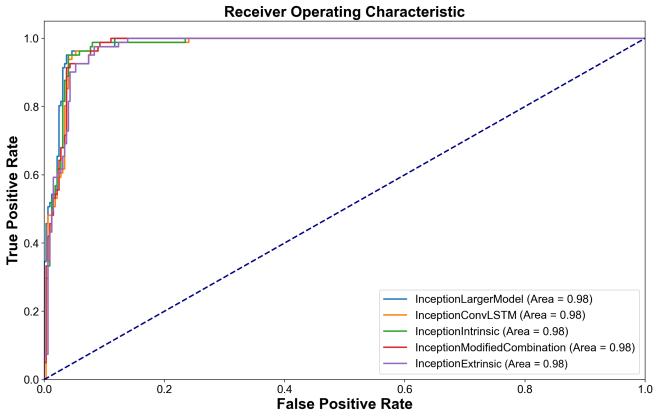


Fig. 4. ROC curves for the classification of digit tasks for the single and two degree of freedom validation data, for each model trained and evaluated. As this is a multi-class classification, these curves are generated using the micro-averaging technique over each prediction class.

B. Regression Performance

1) Inter-Model Performance Analysis: The overall results for each model variant are shown in Table I and Figure 5a. Overall, the InceptionLargerModel variant outperformed the other variant models with a CC, MAE, NMAE, RMSE and NRMSE of 0.850, $4.84°$, 0.410, $11.2°$ and 0.986 respectively.

The use of only the extrinsic grids in the InceptionExtrinsic model yields results that are worse in performance when compared to the InceptionLargerModel (CC: $p < 0.00001$; $ES = 0.560$, MAE: $p < 0.00001$; $ES = -0.672$, RMSE: $p < 0.00001$; $ES = -0.539$), suggesting that the addition of the intrinsic grids adds useful value to the regression performance. Full statistical analysis investigating the improved performance of the InceptionLargerModel over the InceptionExtrinsic model is shown in Supplementary Material 1F.

Moreover, the InceptionConvLSTM model outperformed the InceptionModifiedCombination model, despite both models being of similar depth. This indicates that while the Convolutional-LSTM Hybrid architecture is able to learn an effective transformation and mapping of the data, the less constrained method of combining the feature data from each anatomically separated electrode grid in the InceptionConvLSTM model allowed this model to improve its performance over the other.

The models that used only extrinsic or only intrinsic muscles also resulted in similar performance to each other, with the InceptionIntrinsic variant performing less well compared with the InceptionExtrinsic variant.

2) Inter-Joint Performance Analysis: The regression analysis for each kinematic joint and each model variant is shown in Figure 5b. The InceptionLargerModel variant outperformed all other variant models, including the InceptionExtrinsic model, for nearly all joint angles except for the Index, Middle and Ring PIP joints and the Wrist where the smaller InceptionConvLSTM model had better performance. This implies that the increased complexity of the larger model is beneficial and that the added intrinsic muscle signal improves performance even for wrist angles that are not directly influenced by intrinsic muscle action. The greater performance of the InceptionLargerModel against the InceptionExtrinsic model over all joint angles is also revealed to be much greater for the Index, Middle, Little and Thumb digits than for the Ring digit (Supplementary Material 1F).

Although the InceptionIntrinsic model generally performed worse than InceptionExtrinsic, it did perform better for the Thumb PIP and MCP joints. This highlights that the information within the intrinsic muscles can provide added improvement on thumb kinematic prediction, which is especially relevant given the importance of thumb position in shaping and defining the human grip [24].

While for the MAE metric, joint angles for all digits were predicted with similar performance, joint angles for the Index and Middle digits were better predicted than for the Ring, Little and Thumb when considering the normalised MAE which accounts for the range of joint angles in the validation recordings. MCP joints were also predicted significantly better than their PIP counterparts for all digits.

Figure 6 illustrates the comparison between the actual and predicted joint angles using the InceptionLargerModel, which was the best performing model overall, for Subject 1 performing three separate motion tasks. The predicted joint angles followed the actual angle trajectories, with some discrepancies.

3) Inter-Motion Performance Analysis: The regression analysis for each motion group and each model variant is shown in Figure 5c. As seen previously, the InceptionLargerModel variant outperformed all other variant models for all motion groups, except for the natural motions and sign language tasks, where the InceptionConvLSTM and InceptionModifiedCombination model performed marginally better respectively.

The fact that natural motions were predicted with similar MAE performance as one and two degree of freedom tasks can be explained by the high correlation of joint angles in these tasks, which could be a factor in simplifying the task of regression for the model.

4) Inter-Subject Performance Analysis: The regression analysis of our best performing model (InceptionLargerModel), using the validation data for each individual subject, is presented in Table II. The performance for three subjects was much greater than the overall performance of this model, with RMSE error for subjects 1, 2 and 4 below 10 degrees. This indicates that our devised model is able to generalise over several subjects while still achieving high prediction performance. The contrasting performance for subjects 3 and 5 can potentially be explained by poorer kinematic data quality for some of the tasks compared with the data of the other subjects.

IV. Discussion

We have achieved the full mapping of hand/wrist kinematics from the myoelectric features of the major muscles actuating the wrist and fingers. We achieved this goal by training a custom deep learning architecture for the prediction of hand kinematics, using the HDsEMG signals from both extrinsic and intrinsic hand muscles. Including both sets of hand muscles within our model has produced accurate predictions

TABLE I
OVERALL MODEL METRICS

Model Name	Correlation Coefficient	MAE (Degrees)	NMAE	RMSE (Degrees)	NRMSE
InceptionLargerModel	$\mathbf{0.850 \pm 0.0098}$	$\mathbf{4.84 \pm 0.337}$	$\mathbf{0.410 \pm 0.0292}$	$\mathbf{11.2 \pm 0.654}$	$\mathbf{0.986 \pm 0.0591}$
InceptionConvLSTM	0.846 ± 0.0102	4.97 ± 0.330	0.420 ± 0.0286	11.3 ± 0.657	0.991 ± 0.0591
InceptionIntrinsic	0.808 ± 0.0128	5.82 ± 0.413	0.462 ± 0.0345	13.1 ± 0.774	1.11 ± 0.0689
InceptionModifiedCombination	0.810 ± 0.0102	5.74 ± 0.303	0.489 ± 0.0262	11.5 ± 0.617	1.02 ± 0.0554
InceptionExtrinsic	0.817 ± 0.0115	5.53 ± 0.347	0.467 ± 0.0298	11.8 ± 0.648	1.04 ± 0.0584

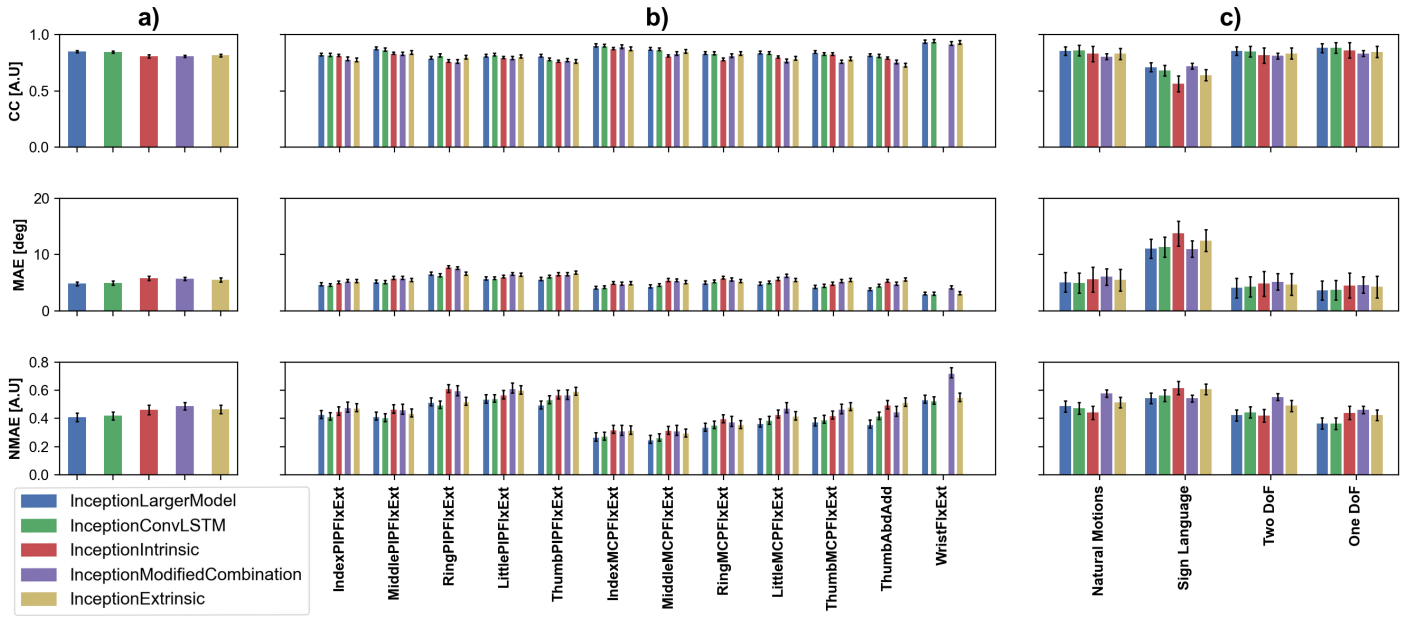


Fig. 5. CC, MAE and NMAE results for the models overall **(a)**, for each joint individually **(b)**, and for each type of motion task **(c)**. The overall and motion metrics are both averaged over all subjects and all joint angles, whereas the indivdual joint metrics are averaged over only the subjects.

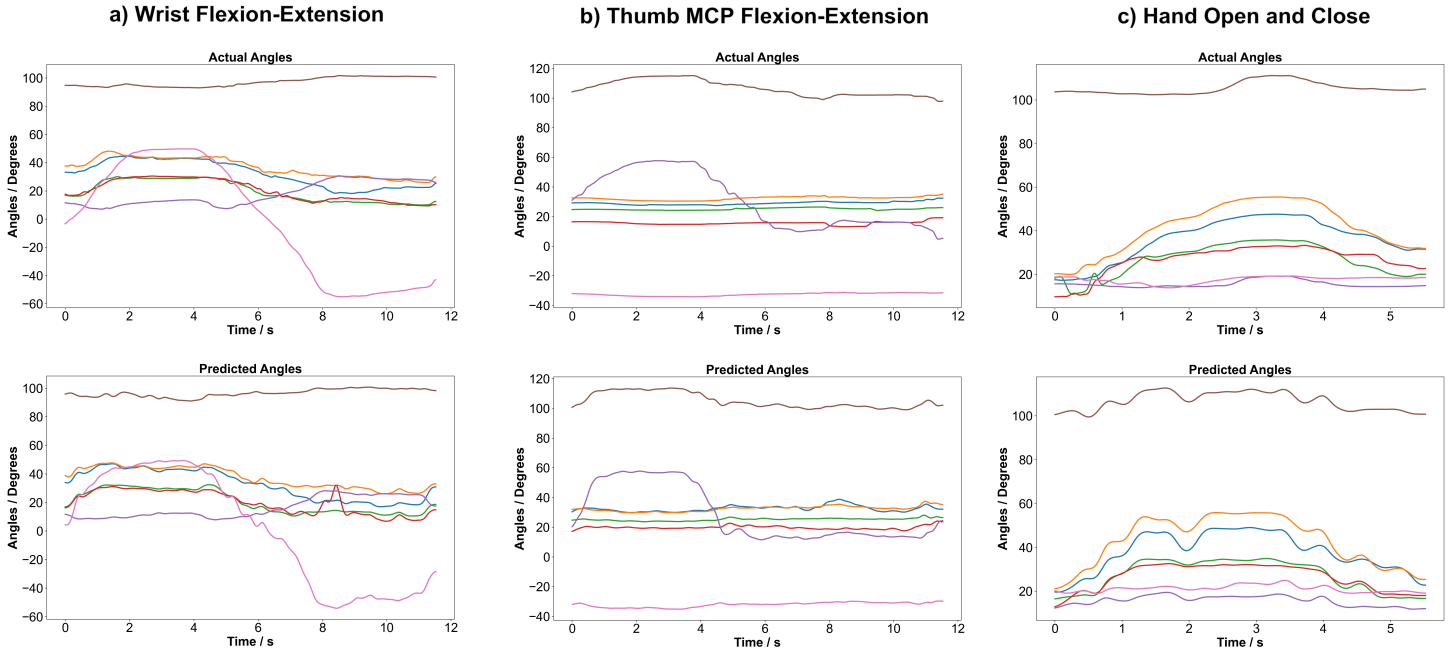


Fig. 6. Actual and predicted joint angles using the validation data of a single subject (Subject 1) for three motion tasks using the InceptionLargerModel: **a)** Wrist flexion-extension, **b)** Thumb MCP flexion-extension and **c)** Hand Open and Close. Each colour highlights a different joint angle: **Blue** - Index MCP flexion-extension; **Orange** - Middle MCP flexion-extension; **Green** - Ring MCP flexion-extension; **Red** - Little MCP flexion-extension; **Purple** - Thumb MCP flexion-extension; **Brown** - Thumb MCP abduction-adduction; **Pink** - Wrist flexion-extension.

TABLE II
INTER-SUBJECT PERFORMANCE METRICS

Subject	Correlation Coefficient	MAE (Degrees)	NMAE	RMSE (Degrees)	NRMSE
1	$\mathbf{0.890 \pm 0.0147}$	$\mathbf{3.281 \pm 0.1882}$	$\mathbf{0.3649 \pm 0.0286}$	$\mathbf{5.65 \pm 0.394}$	$\mathbf{0.644 \pm 0.0561}$
2	$\mathbf{0.854 \pm 0.0251}$	$\mathbf{5.3068 \pm 0.5855}$	$\mathbf{0.3451 \pm 0.0392}$	$\mathbf{9.67 \pm 1.09}$	$\mathbf{0.639 \pm 0.0755}$
3	0.834 ± 0.0183	5.4224 ± 0.5211	0.4687 ± 0.0398	15.8 ± 2.74	1.15 ± 0.147
4	$\mathbf{0.859 \pm 0.0227}$	$\mathbf{3.8365 \pm 0.366}$	$\mathbf{0.3864 \pm 0.0443}$	$\mathbf{7.60 \pm 0.678}$	$\mathbf{0.825 \pm 0.105}$
5	0.794 ± 0.0199	7.1503 ± 0.7184	0.5478 ± 0.0620	15.3 ± 1.89	1.12 ± 0.121

replicated over five subjects and for a wide range of tasks, including natural motions commonly used in daily life.

Through our experiments we have therefore shown that the kinematic regression performance, when using deep learning model architectures is superior when including both intrinsic and extrinsic muscles as input, compared to only extrinsic muscles, as hypothesised. As the spatial resolution of signal recording in relation to muscle size/volume is the same between electrode recording grids for extrinsic and intrinsic muscle groups, we believe this allows us to perform valid comparisons in performance despite using different numbers of channels for the various modified input variations of the models.

Previous work using the same dataset focused on the decomposition of the HD-sEMG signals to analyze the synergistic organization of the spinal motor neurons, grouping the muscles similarly as in this study (i.e All muscle groups together, Extrinsic only and Intrinsic only) ([40], Chapter 4). This previous study found that intrinsic muscles can capture the inherent information of single finger control much better than extrinsic muscles, while the extrinsic muscles are able to capture the important aspects of wrist motor control. This further validates our results in this current work.

In previous studies, similar regression performance has been attempted, either using sEMG recorded only from extrinsic muscles or by regressing over only a small number of joint angles (usually just the angles of the wrist). Recently, Guo et al [29] developed a Long Exposure LSTM architecture for the prediction of only digit joint angles from single channel electrodes recorded from the forearm. Given the performance that we observe for our models, we believe that our approach might outperform their architecture, especially as they only predict over a smaller set of joint angles and tasks and have smaller CC and larger RMSE metrics than our devised models.

Our system also compares well with other novel systems aimed at predicting wrist and finger kinematics, including the Deep Kalman filter approach by Bao et al [41]. This system utilises a probabilistic approach in combination with common deep learning architectures, such as the LSTM [36], however as the RMSE for our devised model is smaller than their results for both wrist and finger kinematic predictions, we believe that our model potentially achieves an improved performance over this design.

While we did train and evaluate our models over several subjects, with promising results across 3 of the 5 subjects, we have mainly focused our analysis on the impact of including both intrinsic and extrinsic sEMG signals on prediction performance. A valuable domain to explore in further work is the ability of our model to transfer its performance to new subjects. Ideally this may be achieved through initially pre-training our model on a large number of subjects followed by a fine-tuning learning phase on a small amount of collected data from a target subject. This approach however would require a much larger cohort of subjects in order to validate both learning phases effectively. We encourage the exploration of transfer learning in future studies, further extending our framework to more subjects and more repetitions per tasks.

We did observe a clear increase in performance between models that used both intrinsic and extrinsic muscle data and those which only included extrinsic or intrinsic muscle input data, especially for digits with more intrinsic hand control such as the thumb. The magnitude of this increase however was not as much as was expected given the important function of the intrinsic muscles in fine motor control of the hand. Further to this, the increase in size/depth of our network (InceptionLargerModel) did little to increase the performance over the standard model (InceptionConvLSTM).

We therefore further conclude that while including both intrinsic and extrinsic muscle HDsEMG signals does lead to increased performance, applications with significant constraints on electrode and hardware use could still use only extrinsic forearm muscle recordings and still achieve reasonable regression performance. We aim to continue work to improve the architecture of our models to further increase the performance benefit of combining input data from both sets of hand muscles.

We also found similar results for models that used only intrinsic or only extrinsic muscle activity. This suggests that for any application where applying electrodes to the hand would be more convenient than on the forearm, a glove with a high-density distribution of EMG electrodes could enable similar accuracy as a forearm band. We believe that this result can provide a rationale for the creation of new glove-like interfaces, which could also integrate other sensor types, including flex and tactile capacitive sensors on the fingertips. These systems could be of particular interest in studies with hemiplegic post-stroke patients, where the hand presents a compromised capacity of movement, but the myoelectric activity can still be measured [42].

A final further outcome of this work is the collected dataset itself. To the best of our knowledge, this is a unique dataset in so far that it contains for each subject, the activity of 14 hand muscles (both extrinsic and intrinsic) recorded using 384 HDsEMG channels. This dataset also includes a wide variety of natural dynamic motion tasks for which the HDsEMG data was recorded, mapped by labelled kinematics data covering the

excursion of the main joints of the fingers and wrist. Thus, this dataset is suitable to be used in future works with deep learning models, as well as the study of control systems utilising motor unit spike trains, which can be extracted from the HDsEMG grids using blind source separation techniques [32].

A. Limitations

A limitation of our recording design involves the number of motion capture markers used for recording. Nineteen markers were used in total which allowed measurement of the MCP, PIP for each digit and the wrist angles. This number was chosen in part to reduce the inaccuracies and post-processing time of the recorded motion capture data. More markers, however, could be used in the future with a configuration similar to Ma'touq et al [43], allowing more accurate calculations of thumb joint angles, especially in the abduction-adduction plane.

We also acknowledge that the number of subjects recorded for our dataset is smaller than many of the popular open source datasets used in other works in the literature [29], due to the complexity of our experimental design. The novelty of our dataset further makes it hard to compare the performance of our work with other approaches in the literature, that use varying amounts of subjects, task motions and sEMG channels. In the future we would like to collect further data from additional subjects to increase the size of our dataset and further validate the use of intrinsic muscle HDsEMG data with our devised deep learning models. This includes the addition of other tasks related to intrinsic muscle control, such as the abduction/adduction of the finger MCP joints, whose joint angles were not extracted/predicted in our current work due to our focus on currently actuated degrees of freedom in prostheses and orthotics. It is also important to consider that while the abduction/adduction of finger MCP joints may be included in future generations of robotic hand prosthetics, intrinsic muscle activity is irrelevant for control of these prostheses, while of more interest in the actuation of exoskeletons for the hand.

The computational systems available were only able to train models of a certain size, due to GPU memory constraints. This impacted our ability to train larger models, as well as making it unfeasible to perform cross-validation analysis due to the lengthy training times for our models, most on the order of days.

It has been previously shown that systems with latency of less than 125ms provide good real time control of myoelectric prostheses [44]. While our computational time does meet this criterion, this does not include electrode recording/processing and mechanical actuation latencies which, when included, would breach this optimal threshold. With the improvement in GPU hardware and network architecture implementation, we believe that this issue will be easily solved in the future.

B. Future Applications

While the majority of the literature focuses on medical applications of myoelectric control systems, we envisage that our devised model can be implemented in any generic control system that requires a more intuitive, flexible and realistic control interface. Furthermore, we believe that the development and optimisation of novel textile based EMG sensors [45], [46] will allow compact and reliable sEMG wearable devices to be manufactured, that can be combined with deep learning control systems as have been described in our work. This will give users access to more convienient, and versatile myoelectric control of a wide variety of devices.

V. Conclusion

We have for the first time implemented a custom deep learning model for the simultaneous prediction of a large number of degrees of freedom of the hand, using HDsEMG data measured from both extrinsic and intrinsic hand muscles. We have shown greater performance when combining input data from both sets of hand muscles, as well as our final model outperforming current state of the art approaches. We propose that our model could be integrated within general purpose control systems through the use of upcoming wearable systems [45], [46].

Author contributions statement

All authors conceived the study. M.P and S.T conducted the data recording as well as performed the processing of the data-set with input from M.K.J on kinematic post-processing. Model creation and training was performed by M.P along with evaluation and results analysis. M.P, S.T, M.K.J and D.F wrote and revised the manuscript.

The authors declare that there are no competing interests.

References

[1] X. Navarro, T. B. Krueger, N. Lago, S. Micera, T. Stieglitz, and P. Dario, "A critical review of interfaces with the peripheral nervous system for the control of neuroprostheses and hybrid bionic systems," *Journal of the Peripheral Nervous System*, vol. 10, no. 3, pp. 229–258, 2005.

[2] B. Blankertz, M. Tangermann, C. Vidaurre, S. Fazli, C. Sannelli, S. Haufe, C. Maeder, L. E. Ramsey, I. Sturm, G. Curio *et al.*, "The berlin brain–computer interface: non-medical uses of BCI technology," *Frontiers in neuroscience*, vol. 4, p. 198, 2010.

[3] L.-D. Liao, C.-Y. Chen, I.-J. Wang, S.-F. Chen, S.-Y. Li, B.-W. Chen, J.-Y. Chang, and C.-T. Lin, "Gaming control using a wearable and wireless EEG-based brain-computer interface device with novel dry foam-based sensors," *Journal of neuroengineering and rehabilitation*, vol. 9, no. 1, pp. 1–12, 2012.

[4] H. F. Hassan, S. J. Abou-Loukh, and I. K. Ibraheem, "Teleoperated robotic arm movement using electromyography signal with wearable myo armband," *Journal of King Saud University-Engineering Sciences*, vol. 32, no. 6, pp. 378–387, 2020.

[5] Z. Lu, K.-y. Tong, H. Shin, S. Li, and P. Zhou, "Advanced myoelectric control for robotic hand-assisted training: outcome from a stroke patient," *Frontiers in Neurology*, vol. 8, p. 107, 2017.

[6] P. Geethanjali, "Myoelectric control of prosthetic hands: state-of-the-art review," *Medical Devices (Auckland, NZ)*, vol. 9, p. 247, 2016.

[7] N. Jiang, S. Dosen, K.-R. Muller, and D. Farina, "Myoelectric control of artificial limbs—is there a need to change focus?[in the spotlight]," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 152–150, 2012.

[8] D. Farina, A. Holobar, R. Merletti, and R. M. Enoka, "Decoding the neural drive to muscles from the surface electromyogram," *Clinical neurophysiology*, vol. 121, no. 10, pp. 1616–1623, 2010.

[9] R. Merletti, A. Holobar, and D. Farina, "Analysis of motor units with high-density surface electromyography," *Journal of electromyography and kinesiology*, vol. 18, no. 6, pp. 879–890, 2008.

[10] D. F. Stegeman, B. U. Kleine, B. G. Lapatki, and J. P. Van Dijk, "High-density surface EMG: Techniques and applications at a motor unit level," *Biocybernetics and biomedical engineering*, vol. 32, no. 3, pp. 3–27, 2012.

[11] N. Nazmi, A. Rahman, M. Azizi, S.-I. Yamamoto, S. A. Ahmad, H. Zamzuri, and S. A. Mazlan, "A review of classification techniques of EMG signals during isotonic and isometric contractions," *Sensors*, vol. 16, no. 8, p. 1304, 2016.

[12] T. Kitamura, N. Tsujiuchi, and T. Koizumi, "Hand motion estimation by EMG signals using linear multiple regression models," in *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2006, pp. 1339–1342.

[13] I. Vujaklija, V. Shalchyan, E. N. Kamavuako, N. Jiang, H. R. Marateb, and D. Farina, "Online mapping of EMG signals into kinematics by autoencoding," *Journal of neuroengineering and rehabilitation*, vol. 15, no. 1, p. 21, 2018.

[14] N. Malešević, D. Marković, G. Kanitz, M. Controzzi, C. Cipriani, and C. Antfolk, "Decoding of individual finger movements from surface EMG signals using vector autoregressive hierarchical hidden markov models (varhhmm)," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 2017, pp. 1518–1523.

[15] L. Bi, C. Guan *et al.*, "A review on EMG-based motor intention prediction of continuous human upper limb motion for human-robot collaboration," *Biomedical Signal Processing and Control*, vol. 51, pp. 113–127, 2019.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[18] T. Bao, A. Zaidi, S. Xie, and Z. Zhang, "Surface-EMG based wrist kinematics estimation using convolutional neural network," in *2019 IEEE 16th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2019, pp. 1–4.

[19] A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data and Cognitive Computing*, vol. 2, no. 3, p. 21, 2018.

[20] M. Hioki and H. Kawasaki, "Estimation of finger joint angles from sEMG using a recurrent neural network with time-delayed input vectors," in *2009 IEEE International Conference on Rehabilitation Robotics*. IEEE, 2009, pp. 289–294.

[21] T. Bao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "A CNN-LSTM hybrid framework for wrist kinematics estimation using surface electromyography," *arXiv preprint arXiv:1912.00799*, 2019.

[22] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. C. Aszmann, "The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 4, pp. 797–809, 2014.

[23] S. Tanzarella, S. Muceli, A. Del Vecchio, A. Casolo, and D. Farina, "A high-density surface EMG framework for the study of motor neurons controlling the intrinsic and extrinsic muscles of the hand," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 2307–2310.

[24] G. Cotugno, K. Althoefer, and T. Nanayakkara, "The role of the thumb: Study of finger motion in grasping and reachability space in human and robotic hands," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1061–1070, 2016.

[25] S. Tanzarella, S. Muceli, A. Del Vecchio, A. Casolo, and D. Farina, "Non-invasive analysis of motor neurons controlling the intrinsic and extrinsic muscles of the hand," *Journal of Neural Engineering*, vol. 17, no. 4, p. 046033, 2020.

[26] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[27] C. M. Bishop *et al.*, *Neural networks for pattern recognition*. Oxford university press, 1995.

[28] W. F. Ganong, *Review of medical physiology*. Mcgraw-hill, 1995.

[29] W. Guo, C. Ma, Z. Wang, H. Zhang, D. Farina, N. Jiang, and C. Lin, "Long exposure convolutional memory network for accurate estimation of finger kinematics from surface electromyographic signals," *Journal of Neural Engineering*, vol. 18, no. 2, p. 026027, 2021.

[30] P. R. Cavanagh and P. V. Komi, "Electromechanical delay in human skeletal muscle under concentric and eccentric contractions," *European journal of applied physiology and occupational physiology*, vol. 42, pp. 159–163, 1979.

[31] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," *arXiv preprint arXiv:1602.07261*, 2016.

[32] C. Chen, Y. Yu, S. Ma, X. Sheng, C. Lin, D. Farina, and X. Zhu, "Hand gesture recognition based on motor unit spike trains decoded from high-density electromyography," *Biomedical Signal Processing and Control*, vol. 55, p. 101637, 2020.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[34] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[36] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[37] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, 2016, pp. 265–283.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[39] D. S. Kerby, "The simple difference formula: An approach to teaching nonparametric correlation," *Comprehensive Psychology*, vol. 3, pp. 11–IT, 2014.

[40] S. Tanzarella, "A non-invasive human-machine interfacing framework for investigating dexterous control of hand muscles," Ph.D. dissertation, Imperial College London, 2021.

[41] T. Bao, Y. Zhao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "A deep kalman filter network for hand kinematics estimation using sEMG," *Pattern Recognition Letters*, vol. 143, pp. 88–94, 2021.

[42] S. E. Williams, K. C. Koch, and C. Disselhorst-Klug, "Non-invasive assessment of motor unit activation in relation to motor neuron level and lesion location in stroke and spinal muscular atrophy," *Clinical Biomechanics*, vol. 78, p. 105053, 2020.

[43] J. Ma'touq, T. Hu, and S. Haddadin, "Sub-millimetre accurate human hand kinematics: from surface to skeleton," *Computer methods in biomechanics and biomedical engineering*, vol. 21, no. 2, pp. 113–128, 2018.

[44] T. R. Farrell and R. F. Weir, "The optimal controller delay for myoelectric prostheses," *IEEE Transactions on neural systems and rehabilitation engineering*, vol. 15, no. 1, pp. 111–118, 2007.

[45] S. Kim, S. Lee, and W. Jeong, "EMG measurement with textile-based electrodes in different electrode sizes and clothing pressures for smart clothing design optimization," *Polymers*, vol. 12, no. 10, p. 2406, 2020.

[46] L. Guo, L. Sandsjö, M. Ortiz-Catalan, and M. Skrifvars, "Systematic review of textile-based electrodes for long-term and continuous surface electromyography recording," *Textile Research Journal*, vol. 90, no. 2, pp. 227–244, 2020.