Source-Free Random Forest Model Calibration for Myoelectric Control

Xinyu Jiang, Chenfei Ma, and Kianoush Nazarpour*

Authors are with School of Informatics, The University of Edinburgh, Edinburgh, United Kingdom.

* Correspondence: kianoush.nazarpour@ed.ac.uk

Abstract.

Objective: Most existing machine learning models for myoelectric control require a large amount of data to learn user-specific characteristics of the electromyographic (EMG) signals, which is burdensome. Our objective is to develop an approach to enable the calibration of a pre-trained model with minimal data from a new myoelectric user. *Approach:* We trained a random forest model with EMG data from 20 people collected during the performance of multiple hand grips. To adapt the decision rules for a new user, first, the branches of the pre-trained decision trees were pruned using the validation data from the new user. Then new decision trees trained merely with data from the new user were appended to the pruned pre-trained model. *Results:* Real-time myoelectric experiments with 18 participants over two days demonstrated the improved accuracy of the proposed approach when compared to benchmark user-specific random forest and the linear discriminant analysis models. Furthermore, the random forest model that was calibrated on day one for a new participant yielded significantly higher accuracy on day two, when compared to the benchmark approaches, which reflects the robustness of the proposed approach. *Significance:* The proposed model calibration procedure is completely source-free, that is, once the base model is pre-trained, no access to the source data from the original 20 people is required. Our work promotes the use of efficient, explainable, and simple models for myoelectric control.

Keywords: myoelectric control, electromyography, random forest.

1. Introduction

Many myoelectric control systems use machine learning to map the electromyographic (EMG) signals [1, 2, 3, 4, 5] to control commands for human-machine interfaces, e.g. prosthesis [6, 7, 8, 9, 10, 11, 12] and virtual keyboards [13, 14]. Most modern myoelectric control machine learning models require a large amount of data from a user to learn a bespoke and user-specific map [7, 15, 16]. In real-life settings, machine learning models that are flexible and require minimal data from a new user are preferable [17]. Additionally, such models are expected to be robust in long-term (multi-day) applications without the need for frequent re-calibration.

Domain adaptation minimises the distribution mismatch between training data (source domain) and testing data (target domain) [18]. In myoelectric control applications, this technique can be used for a model to learn extra knowledge from source data collected from other subjects, so that the demand for data collection from a new target user is reduced. For instance, Vidovic et al. [19] employed a covariate shift adaptation algorithm to align the statistical metrics (e.g., the mean value and covariance) of source and target domains. These statistical metrics informed the training of a linear discriminant analysis (LDA) model. Wang et al. [20] developed a multi-user myoelectric control model using discriminative canonical correlation analysis by jointly projecting the feature sets for source and target domains into a low-dimensional, uniform-style subspace. Jiang et al. [21] proposed a correlation-based data weighting (CORW) scheme to measure the level of feature distribution shift between the target testing user and each training user and then assign different weights to data from different training subjects accordingly. One common disadvantage of the above approaches is the need to access the data in the source domain. EMG data contain diverse private and sometimes sensitive information, e.g., the status of neurological diseases [22] or personal identity [23]. Therefore, in most practical cases, the source EMG data should be strictly protected, due to regulations such as the European General Data Protection Regulation (GDPR) and user concerns.

Large pre-trained models have proven effective in many machine learning tasks and shown their impressively strong generalisation capability, e.g., the Generative Pre-trained Transformer 4 (GPT-4) [24] and the segment anything model (SAM) [25] in computer vision (CV). However, such large pre-trained models are mainly built on differentiable and parameterised deep neural networks. The pre-training of such large models with back-propagation relies heavily on extremely large datasets and significant computing infrastructure. Despite their excellent performance, such pre-trained models have a highly complex architecture and are used as a black-box module. In myoelectric control applications, large pre-trained models are not available up to date in the academic and clinical sectors, mainly because collecting an extremely large and labelled dataset is expensive and time-consuming. Additionally, for most applications of myoelectric control, the model should be implemented in a portable and wearable system with limited computing and battery resources. Moreover, the employed model is expected to be explainable [26, 27, 28]. Therefore, we asked: "Is it possible to create a pre-trained myoelectric control model that can be tuned using a small dataset for a new user, is explainable, and can classify the EMG signals in real-time with limited computing resources, e.g., memory?"

Standard, off-the-shelf random forest (RF) models have not been generally viewed as the most powerful model for myoelectric control [29]. However, RF models have proved effective in dealing with problems with small sample sizes [30]. They can capture complex nonlinear relations between samples. They exhibit robustness on outliers and they can be easily parallelised for computational efficiency [31, 32]. Additionally, the decision tree model, which is the base learner in RF, is recognised as one of the most explainable models [33, 28]. The above merits motivated us to explore new possibilities with RF models and its potentials for myoelectric control applications.

We propose a myoelectric control model with a user-generic RF model that is pre-trained using data from 20 users. When a new (i.e., 21st) user is introduced, we use 1s of new EMG data for each gesture to fine-tune the pre-trained RF model, upon a series of tree pruning, branch grafting, and tree appending procedures. We show that our approach improves model accuracy and across-day generalisability significantly. Crucially, the proposed model tuning is completely source-free, that is we do not need the source data (from the original 20 participants) to tune the pre-trained model for a new user. Both offline validations and the real-time implementation demonstrated the superiority of our method compared with the standard user-specific RF

and the benchmark LDA models.

2. Methods

2.1. Ethical approval

All subjects signed an informed consent form which was approved by the local ethics committee at the University of Edinburgh (reference number: 2019/89177). We conducted two experiments.

2.2. Experiment 1

Experiment 1 was offline. We collected EMG data from 20 able-bodied subjects (12 males, 8 females, age range: 22–43). Eight electrodes were equally spaced across the circumference of the forearm, as presented in Figure 1. The EMG data were collected using Delsys Trigno sensors (Delsys, Inc.), with a sampling rate of 2000 Hz and bandpass filter of 10–500 Hz. Each subject performed six hand gestures, as presented in Figure 1. For each gesture, 10 repetitions were performed in 10 successive trials, each six seconds long. In each trial, subjects were required to mimic an instructed target hand gesture and then hold the gesture until the trial ended. Signals recorded in the first two seconds of each trial were removed to get rid of the transient period. The EMG signals within the last four seconds were retained. There was a five-second resting period between trials. Data from Experiment 1 was used to pre-train the RF model.

2.3. Experiment 2

Experiment 2 was a real-time myoelectric control experiment. We recruited 18 *new* able-bodied subjects (11 males, 7 females, age range: 22–28). Experiment 2 was conducted on two successive days. On the first day, the experiment consisted of two sections, namely, calibration and testing. In the calibration section, subjects were asked to perform each hand gesture only once in a 2s trial. Each two-second trial afforded the participants one second to shape their hand so that they could hold the gesture reliably during the second one-second period. Only the data during the latter period was used for model tuning, or training a user-specific model from scratch (as a benchmark).

The testing section comprised five blocks. In each block, five repetitions of each gesture were instructed in a pseudo-randomised way. Subject had two seconds and five minutes for rest between trials and between blocks, respectively. On the second day, subjects started the testing section without any further tuning.

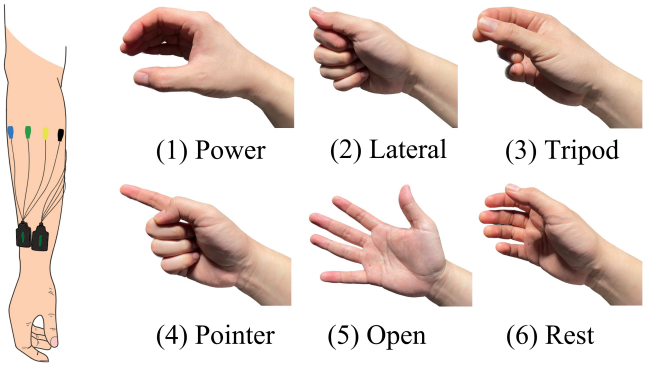

Figure 1: Electrode positions and involved gestures in our experiment.

2.4. Feature Extraction

In both experiments, we used a window of 200 ms, sliding at 100 ms, for feature extraction. Ten features were extracted, which described the EMG data from three complementary aspects. First, the mean absolute value (MAV), waveform length (WL), zero crossings (ZC) and slope sign changes (SSC) of the EMG signals, combined with root mean square (RMS), were used as energy descriptors [34, 35, 36]. Second, the skewness [37, 38, 39] of the EMG signals was extracted as a distribution descriptor. Third, the mean frequency (MNF), median frequency (MDF), peak frequency (PKF), and variance of central frequency (VCF) were extracted as spectrum descriptors [40]. We will demonstrate the complementary roles of these features through an ablation experiment in the Results Section.

2.5. The Proposed RF Model

2.5.1. Pre-training the RF model Data from Experiment 1 was used for pre-training an RF model. The model had 200 decision trees. A subject-wise feature normalisation was performed on the pre-training dataset. Specifically, the extracted features from each subject were normalised separately to a mean value of zero and a standard deviation of one to align the basic statistical metrics of feature distributions from different subjects. The allocation of pre-training and testing subjects varies with different validation settings and will be described later.

2.5.2. Pruning and Grafting Decision Trees Given a pre-trained RF model, the follow-up calibration procedure involved three strategies, namely, pruning, grafting, and appending decision trees, as presented in Figure 2. Pruning is a widely used strategy to simplify a model and at the same time improve its generalisability [41]. Grafting, that is, growing new branches on a leaf node to further specify the decision rule. Compared to

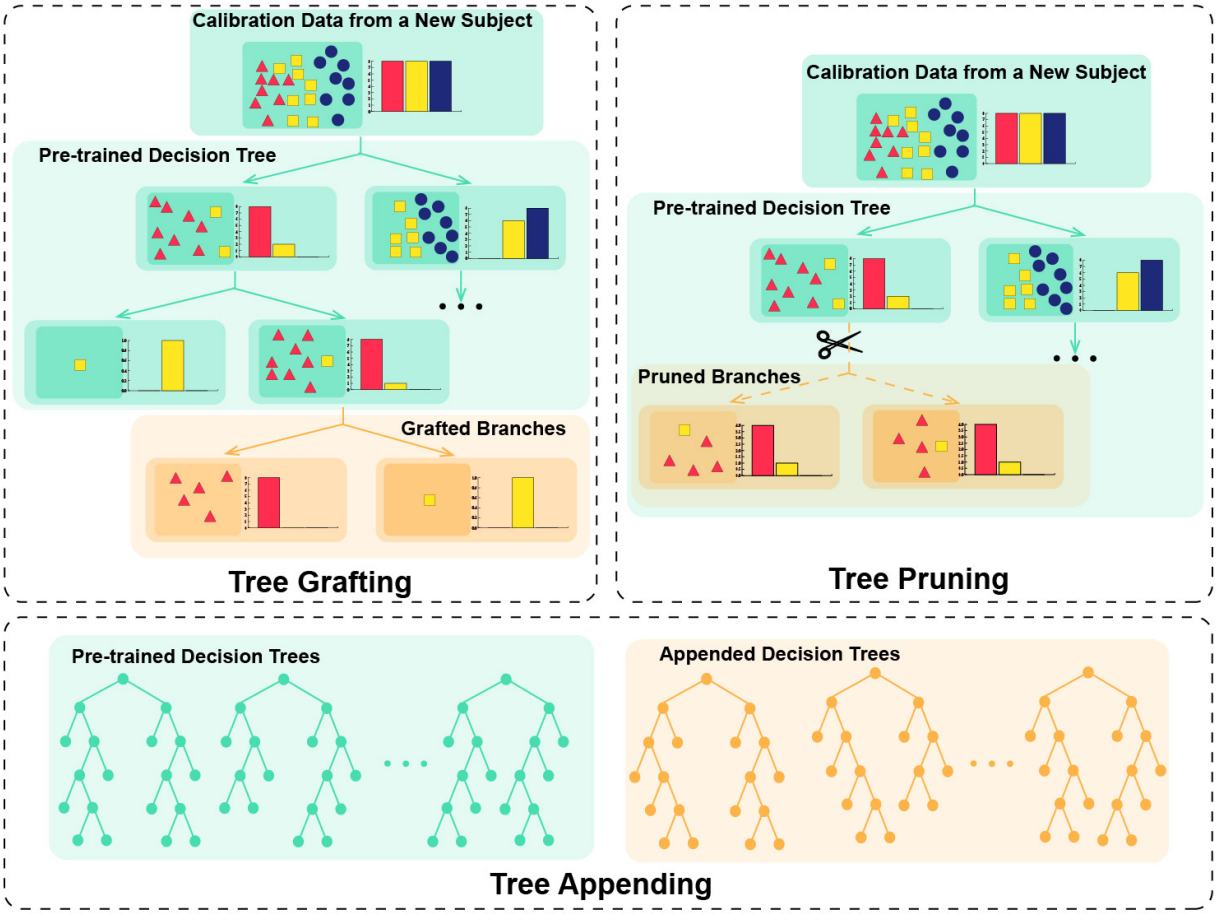

Figure 2: Tree grafting, tree pruning, and tree appending for RF model tuning.

Algorithm 1 Decision Tree Pruning

Input: $Tree, \mathcal{S}_{target}$
d_{max}=maxDepth($Tree$)
$d_{inspect} = d_{max}$ %inspect the deepest nodes %Repeat
while $d_{inspect} > 0$ **do**
  $nodes$=findNodes($d = d_{inspect}$)
  $parentNodes$=findParents($nodes$)
  for each $u_i \in parentNodes$ **do**
    if Error(Prune($Tree, u_i$),$\mathcal{S}_{target}$) <Error($Tree, \mathcal{S}_{target}$) **then**
      $Tree$ =Prune($Tree, u_i$)
    else
      continue
    end if
  end for
  $d_{inspect} = d_{inspect} - 1$ %inspect shallower nodes
end while%Until $h_{inspect} = 0$
Output: $Tree$

Algorithm 2 Decision Tree Grafting

Input: $Tree, \mathcal{S}_{target}$
$leafNodes$=findLeaves($Tree$)
for each $u_i \in leafNodes$ **do**
  % find samples falling into each node u_i
  $\mathcal{S}_i$=findSampleInNode($Tree, u_i, \mathcal{S}_{target}$)
  % get the set of ground truth labels of $\mathcal{S}_i$
  $\mathcal{L}_i$=unique(label($\mathcal{S}_i$)) %$\mathcal{L}_i$=$\{l_i^1, \ldots, l_i^m\}$
  if length($\mathcal{L}_i$)> 1 **then**
    %further purify labels via an additional tree
    $\mathcal{S}_{subset} = \{s_k \in \mathcal{S}_{target}$, if label($s_k$)$\in \mathcal{L}_i\}$
    $TreeGraft$=train($\mathcal{S}_{subset}$,label($\mathcal{S}_{subset}$))
    $Tree$ =Graft($Tree, TreeGraft, u_i$)
  else if length($\mathcal{L}_i$)==1 && $l_i^1 \neq l_i^{predict}$ **then**
    %flipping the prediction label of the node u_i
    $l_i^{predict} = l_i^1$
  else
    continue
  end if
end for
Output: $Tree$

pruning, grafting has attracted less attention in previous studies, but both the pruning and grafting strategies

have been proven effective in fine-tuning a decision tree in different application scenarios [42, 43, 44]. In this work, we applied both strategies and compared their performances to provide a comprehensive benchmark for future research.

For pruning, we applied a bottom-up strategy. Let's assume a pre-trained decision tree $Tree$ with a node u_i at depth d_i, where the depth of the root node is zero and the depth of a leaf node with the longest decision path is d_{max}. Using samples from a target new user ($\mathcal{S}_{target}$) as the validation dataset, the error rate of a decision tree was estimated and denoted as $Error(Tree, \mathcal{S}_{target})$. When performing the pruning operation on a node, all its children nodes (and children of children, if any) were removed from the tree. All samples falling into the removed children nodes now fall into their parent nodes, which are the new leaf nodes. The prediction label of the new leaf node after decision tree pruning was defined as the mode of ground truth labels of all validation samples falling into this node. In our bottom-up strategy, we first found the parent nodes of the deepest leaf nodes (denoted as $parentNodes = \{u_1, \ldots, u_i, \ldots, u_n\}$) with their depth equal to d_{max}. Only if the estimated error rate decreased after pruning these deepest nodes, the pruning operation was performed. The same inspection procedure was repeated on nodes at lower depths until the root node was inspected. The pseudo-code of our pruning strategy is presented in Algorithm 1.

As for grafting, given a set of leaf nodes, $leafNodes = \{u_1, ..., u_i, ..., u_n\}$, we first constructed a set of validation samples from the new target subject falling into each leaf node u_i, denoted as $\mathcal{S}_i = \{s_i^1, \ldots, s_i^k, \ldots, s_i^K\}$. The set of involved ground truth labels of the sample set $\mathcal{S}_i$ was defined as $\mathcal{L}_i = \{l_i^1, \ldots, l_i^m\}, m \leq N_{class} = 6$. The predicted label of a leaf node u_i was defined as $l_i^{predict}$. If $m > 1$ (that is, validation samples with more than one label fell into a leaf node u_i) or $m = 1$ but $l_i^1 \neq l_i^{predict}$, then the decision rule related to the leaf node u_i is not reliable and therefore should be fine-tuned. Concretely, the decision rule related to a leaf node was fine-tuned either by grafting a new decision tree to the leaf node to further purify the sample labels (in the case of $m > 1$), or by directly flipping the predicted label (in the case of $m = 1$ but $l_i^1 \neq l_i^{predict}$). The pseudo-code for our grafting strategy is presented in Algorithm 2.

The decision tree pruning and grafting are opposite operations. Performing either operation is the first step of our model calibration method, and their performances will be compared in our analyses.

2.5.3. Appending Decision Trees After performing either pruning or grafting on the pre-trained model, we appended an additional 200 decision trees trained merely on data from the new target user. This constructed an RF model with 400 trees. By appending new decision trees, the decision rules of the calibrated RF model consider both the generalised data distribution from a large number of pre-training users and the specific data distribution from the new user.

2.6. Baseline Machine Learning Models

Two other machine learning models were implemented to provide baseline performances. First, a user-specific LDA model was trained using data from the new target user. LDA was selected in our work because it is considered as a gold standard for the state-of-the-art EMG pattern recognition models [8, 45, 46]. Second, a user-specific standard RF model was also trained using data from the new target user. The baseline user-specific RF model also consisted of 400 decision trees, the same as the proposed model.

2.7. Validation Methods

For the data collected from 20 subjects in experiment 1, we performed an offline leave-one-subject-out cross-validation. For each subject, data from the other 19 subjects were allocated into the pre-training dataset to pre-train an RF model. For the 6 gestures $\times$ 10 trials/gesture = 60 trials from the testing subject, only data in the first trial of each gesture were used to calibrate the pre-trained model. To simulate the most challenging validation using minimal calibration data from the testing subject, only data within a 1s duration (the last 1s duration out of the total 4s valid duration) of a trial were allocated into the calibration dataset to fine-tune the pre-trained RF. The choice of using only 1s duration of a trial in the calibration dataset is also the same as the counterpart in our subsequent online experiment - experiment 2, where only data within a 1s holding period were available in each calibration trial. For the other two baseline user-specific RF and LDA models, the same calibration dataset was used to train a brand-new user-specific model, and all other configurations were the same. Overall, in our offline validations, we compared the performances of four models, namely, RF calibrated via grafting and appending decision trees, RF calibrated via pruning and appending decision trees, standard RF, and standard LDA.

For our real-time study in experiment 2, an RF model was first pre-trained using all offline data from all 20 subjects of experiment 1. When a new user started the experiment on the first day, data in the 1s holding duration of one trial for each gesture was collected in the calibration session, and used to calibrate the pre-trained RF model. Standard user-specific RF and LDA models were also trained using the same

calibration dataset. In the testing part of experiment 2, the three models, namely RF calibrated via pruning and appending decision trees, the user-specific RF and the user-specific LDA, were implemented and gave their three predictions in each sliding window. Accuracy was evaluated on all predictions in 10 windows within the 1s holding period of a testing trial. On the second day of experiment 2, no model calibration was performed, and the testing section directly started using the same three models trained/calibrated on the first day.

Experiment 2 was conducted in an open-loop mode, that is the participants did not see the outcome of the decoders. As such any observed differences between the three decoders is wholly because of the decoding capacity of the model. The experiment was run on a laptop (CPU: 11th Gen Intel(R) Core(TM) i5-1145G7 @ 2.60GHz).

2.8. Ablation Experiment

Considering many key components were included in our method, we performed an ablation experiment with each component removed from the processing pipeline, to show the necessity and contribution of each individual component. These key components are: (1) energy descriptors, (2) distribution descriptors, (3) spectrum descriptors, (4) subject-wise feature normalization, (5) pruned decision trees, and (6) appended decision trees. Note that when removing pruned decision trees or appended decision trees, we always keep the same number of total decision trees, namely 400 trees, by doubling the other types of decision trees.

2.9. Statistical Analyses

To compare the performances of different models (≥ 3 groups), the Friedman test was first performed. The Nemenyi post-hoc test, a multi-comparison test to identify pair-wise group differences, was then applied. Significant differences were claimed if $p < 0.05$ was obtained.

3. Results

3.1. Experiment 1

Results of the first experiment are presented in Figure 3. RF with pruning and appending strategies and RF with grafting and appending strategies yielded an accuracy of 88.3% and 88.2%, respectively, both significantly higher than the outcome of LDA. By definition, pruning leads to smaller models with fewer parameters. Therefore, we selected decision tree pruning rather than grafting in our real-time implementation.

Additionally, considering the final fine-tuned RF model involves many components, we ran an offline ablation experiment to examine the effect of removing each individual component. Results were presented in Figure 4. Removing each individual component from the overall framework would lead to lower accuracy, indicating the necessity of all these components in the whole framework. Significance was observed when removing the energy features, pruned decision trees, or appended decision trees, demonstrating the important roles of these components.

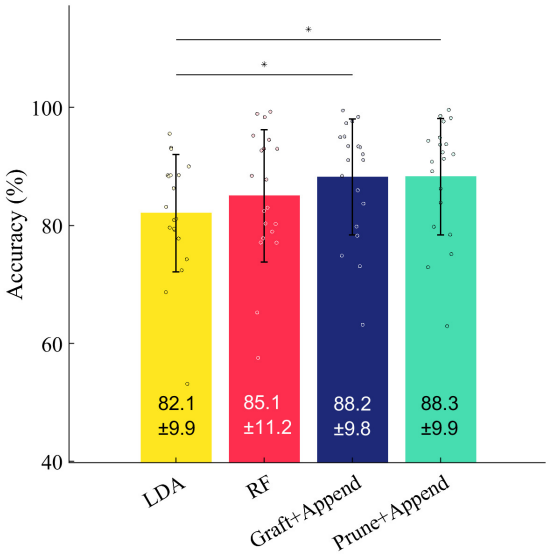

Figure 3: Results in offline validations. Symbol $*$ denotes a significant difference between the two groups.

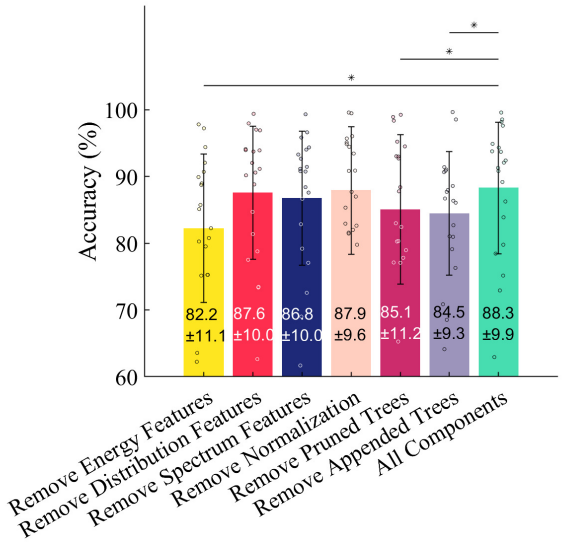

Figure 4: Results in offline ablation experiments. Symbol $*$ denotes a significant difference between the two groups.

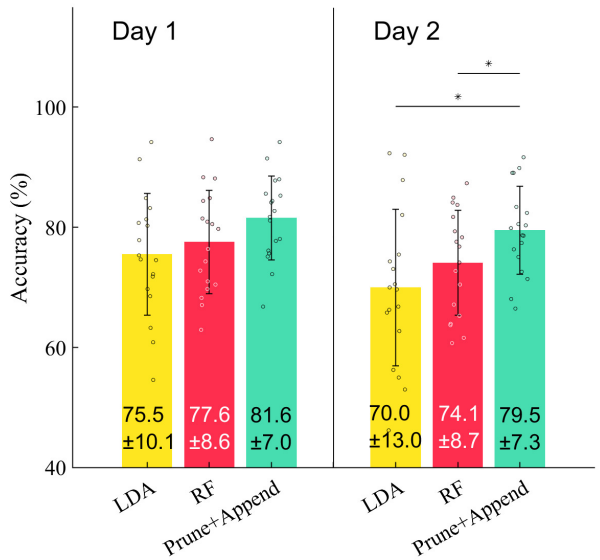

Figure 5: Results in real-time implementation in experiment 2. Symbol $*$ denotes a statistical significance.

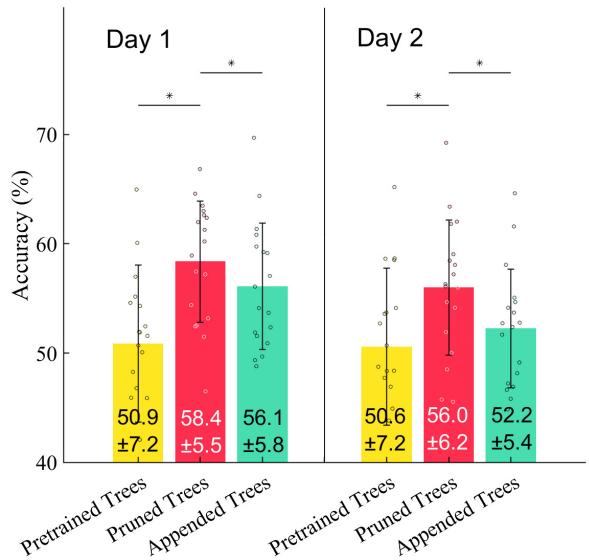

Figure 6: Average accuracy of different decision trees. Results were obtained from each individual decision tree without an ensemble. Symbol $*$ denotes a significant difference between the two groups.

3.2. Experiment 2

Results of our real-time implementations are presented in Figure 5. On the first day, RF calibrated with pruning and appending decision trees achieved the highest average accuracy of 81.5% but the accuracy improvement over the standard LDA (75.5%) and standard RF (77.6%) is not significant. On the second day, RF calibrated via pruning and appending decision trees significantly outperformed both the standard LDA and the standard RF. This results verifies the the superiority of our method in an inter-day application.

The final pre-trained and then fine-tuned RF model involves two groups of decision trees, i.e., the pruned decision trees and the appended decision trees. For the pruned decision trees, because they were pre-trained on a large dataset from a large number of subjects, each decision tree consists of $\sim$ 925 nodes. For the appended decision trees trained on a small dataset from a new subject, each decision tree comprised $\sim$ 15 nodes. Accordingly, the whole model consists of $925 \times 200 + 15 \times 200 = 188,000$ nodes. Four parameters were saved in each node, i.e., the index of the left child (2 bytes), the index of the right child (2 bytes), the index of the selected feature in a node (1 byte) and the threshold of the selected feature (2 bytes), with a size of 7 bytes each node. Therefore, the size of the whole model is 188,000 nodes $\times$ 7 bytes/node = 1.26 MB. Other memory usage in the whole processing pipeline is 190.1 ± 19.2 KB (peak memory).

4. Discussions

We proposed a source-free calibration method for an RF-based myoelectric control model based on pruning and appending decision trees, which outperformed standard LDA and RF models, especially in an inter-day study. The improved performance of RF by pruning and appending decision trees can be explained from two aspects:

- improving the average accuracy of individual decision trees and
- improving the ambiguity (or, diversity) among all decision trees,

which are two important factors determining the overall performance of an ensemble model [47, 48]. Accordingly, we evaluated the average accuracy of each individual decision tree and the distribution of their predictions, presented in Figure 6 and Figure 7, respectively. First, as shown in Figure 6, by pruning each pre-trained decision tree, the average accuracy of the pruned decision trees was significantly higher than the pre-trained and appended decision trees. Second, as shown in Figure 7, the predictions of the pre-trained trees and pruned trees were similar so their predictions were distributed near each other. However, the predictions of the appended decision trees were far from the other trees. These results demonstrated that, by appending new user-specific decision trees, the ambiguity between decision trees was improved, so that the predictions from different decision trees can complement each other.

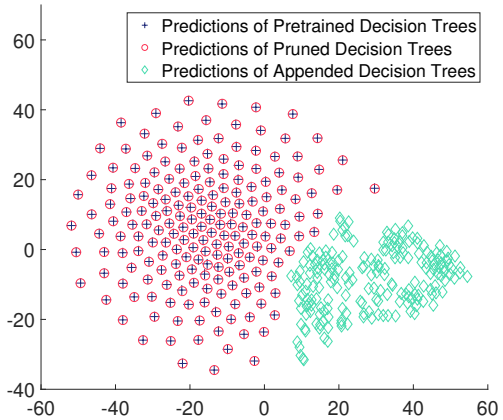

Figure 7: Visualization of predictions of different decision trees in a 2-dimensional space via t-Distributed Stochastic Neighbor Embedding (t-SNE) [49]. Predictions in this figure were drawn from a representative subject on the second day. Predictions on testing samples were used as visualization variables and all decision trees were used as visualization data points. Data from other subjects follow a similar pattern. A single data point represents a decision tree.

A high average accuracy and a high ambiguity of decision trees together improve the overall performance of the RF model.

We also compared the effectiveness of the grafting and pruning operations on decision trees. Decision tree grafting could further specify the decision rules of a pre-trained decision tree based on the feature distribution of a new user; so that the new decision rules can be better adapted to the new user. By using data from a new user as the validation dataset, pruning could simplify the decision rules. Both grafting and pruning were effective and did not show significantly different performance. We selected decision tree pruning in our real-time implementation because it led to smaller models.

All hyper-parameters were tuned and determined in our offline validations on data collected in experiment 1 and directly used in our real-time implementation in experiment 2 without additional hyper-parameter tuning. One important hyper-parameter is the number of decision trees in the pre-trained RF (200 trees) and appended RF (200 trees). We selected these values because further increasing the number of trees did not contribute to a substantial performance improvement but reduced the computational efficiency. Another hyper-parameter is the number of samples to draw from the pre-training dataset to pre-train each decision tree. Considering the pre-training dataset from multiple subjects is relatively large, drawing the same number of bootstrap samples as the total size of the pre-training dataset would lead to an extremely large decision tree with over-complex decision rules. Therefore, in our work, only 7% samples were randomly drawn from the pre-training dataset via bootstrap to pre-train each decision tree. For user-specific decision trees, considering the size of the calibration dataset from the new target user is small, the number of bootstrap samples to train each user-specific decision tree was the same as the size of the calibration dataset. The tuning of model hyper-parameters in our real-time implementation is pre-determined and completely independent of the data collected in our online experiment.

Table 1: Computation time in each processing step. No parallel computing was employed.

Processing Step	Computation Time
Feature Extraction	2.8±0.2 ms
Classification	6.2±0.6 ms
Total	9.0±0.7 ms

The significant performance improvement of our method on the second day is another important finding. In inter-day applications of myoelectric control models, the model performance would normally degrade due to the joint effects of multiple factors, e.g. the electrode shift, skin-electrode impedance, etc. By training a new model merely on data from the new user, the model is prone to overfitting on the small dataset from the new user on the first day and degraded performance on the second day. By first pre-training a model and then fine-tuning the model on the new user, the model could adapt to the data of the new user and at the same time retain the generalisation capability learned from a large pre-training dataset. Additionally, the above finding may make even broader contributions, because the pre-training mechanism is expected to improve the inter-day generalisation capability for most machine learning models.

We evaluated the memory requirements and computational efficiency of our method. For memory requirements, we evaluated the size of model parameters and other memory usage in computing (the peak memory). For computational efficiency, we evaluated the computation time of feature extraction and decision-making, respectively. The whole processing pipeline can be divided into two steps, namely, the feature extraction step and the classification step. The computation time in each processing step is presented in Table 1. The feature extraction and classification steps require 2.82 ms and 6.27 ms processing time, respectively, with a total of 9.08 ms for the whole processing pipeline. The processing time of 9.08 ms is short enough for online processing of an EMG

segment with a 200 ms window length and 100 ms sliding step. Furthermore, the average model size is 1.26 MB with an additional average memory of 190.1 KB for computations in the whole processing pipeline. These metrics demonstrate the potential of our method for mobile computing scenarios with low-cost computational resources. In practical applications, a trade-off between the computational time and the memory requirement can be achieved via parallel computing, depending on the priority of different requirements in different application scenarios. RF is a model highly suitable for parallel computing because the processing procedures of different decision trees are independent of each other. This property increases the flexibility of the RF model in real-world applications with different computational platforms.

Our work is an initial attempt to demonstrate the superiority of pre-trained and fine-tuned RF over the standard RF in myoelectric control applications. RF has not been considered as one of the most powerful models in myoelectric control. Therefore, little attention has been paid to RF models and there are very few attempts on advanced algorithms for improved RF models [29]. Here we demonstrated the high potential of RF models empowered by several key components such as decision tree pre-training, pruning and appending. Our very recent study [28] also demonstrated the high explainability and robustness of decision tree-based models in myoelectric control. We hope these attempts can attract renewed attention on the efficient, explainable, and powerful decision tree-based models within the myoelectric control community, and expect more interesting and inspiring properties of RF to be discovered in follow-up studies.

5. Conclusions

We proposed a myoelectric control model based on random forest, using only 1s of EMG data per gesture from the new user to fine-tune a pre-trained model. Both offline and real-time experiments showed the superiority of our method compared to the standard RF and LDA models. In our inter-day study, our model trained with the data from the first day yielded a significantly higher accuracy compared to other models, without any calibration needed on the second day. Further evaluations also demonstrate the low memory requirements and high computational efficiency of our method.

Acknowledgments

This work is supported by the Engineering and Physical Sciences Research Council (EPSRC), UK, (grant number: EP/R004242/2).

References

[1] Nan Zhao, Bolun Zhao, Gencai Shen, Chunpeng Jiang, Zhuangzhuang Wang, Zude Lin, Lanshu Zhou, and Jingquan Liu. A robust hd-semg sensor suitable for convenient acquisition of muscle activity in clinical post-stroke dysphagia. *Journal of Neural Engineering*, 20(1):016018, jan 2023.

[2] Lahiru N Wimalasena, Jonas F Braun, Mohammad Reza Keshtkaran, David Hofmann, Juan Álvaro Gallego, Cristiano Alessandro, Matthew C Tresch, Lee E Miller, and Chethan Pandarinath. Estimating muscle activation from emg using deep learning-based dynamical systems models. *Journal of Neural Engineering*, 19(3):036013, may 2022.

[3] Yue Wen, Sangjoon J Kim, Simon Avrillon, Jackson T Levine, François Hug, and José L Pons. Toward a generalizable deep cnn for neural drive estimation across muscles and participants. *Journal of Neural Engineering*, 20(1):016006, jan 2023.

[4] Xuan Zhang, Le Wu, Xu Zhang, Xiang Chen, Chang Li, and Xun Chen. Multi-source domain generalization and adaptation toward cross-subject myoelectric pattern recognition. *Journal of Neural Engineering*, 20(1):016050, feb 2023.

[5] Daniela Souza de Oliveira, Andrea Casolo, Thomas G Balshaw, Sumiaki Maeo, Marcel Bahia Lanza, Neil R W Martin, Nicola Maffulli, Thomas Mehari Kinfe, Bjoern M Eskofier, Jonathan P Folland, Dario Farina, and Alessandro Del Vecchio. Neural decoding from surface high-density emg signals: influence of anatomy and synchronization on the number of identified motor units. *Journal of Neural Engineering*, 19(4):046029, aug 2022.

[6] Yanan Diao, Qiangqiang Chen, Yan Liu, Linjie He, Yue Sun, Xiangxin Li, Yumin Chen, Guanglin Li, and Guoru Zhao. A fuzzy granular logistic regression algorithm for semg-based cross-individual prosthetic hand gesture classification. *Journal of Neural Engineering*, 20(2):026029, apr 2023.

[7] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. Feature reduction and selection for emg signal classification. *Expert Systems with Applications*, 39(8):7420–7431, 2012.

[8] Ali H. Al-Timemy, Guido Bugmann, Javier Escudero, and Nicholas Outram. Classification of finger movements for the dexterous hand prosthesis control with surface electromyography. *IEEE Journal of Biomedical and Health Informatics*, 17(3):608–618, 2013.

[9] Farshad Khadivar, Vincent Mendez, Carolina Correia, Iason Batzianoulis, Aude Billard, and Silvestro Micera. Emg-driven shared human-robot compliant control for in-hand object manipulation in hand prostheses. *Journal of Neural Engineering*, 19(6):066024, dec 2022.

[10] Pranav Mamidanna, Jakob L Dideriksen, and Strahinja Dosen. Estimating speed-accuracy trade-offs to evaluate and understand closed-loop prosthesis interfaces. *Journal of Neural Engineering*, 19(5):056012, sep 2022.

[11] Federica Barberi, Francesco Iberite, Eugenio Anselmino, Pericle Randi, Rinaldo Sacchetti, Emanuele Gruppioni, Alberto Mazzoni, and Silvestro Micera. Early decoding of walking tasks with minimal set of emg channels. *Journal of Neural Engineering*, 20(2):026038, apr 2023.

[12] Philip P Vu, Alex K Vaskov, Christina Lee, Ritvik R Jillala, Dylan M Wallace, Alicia J Davis, Theodore A Kung, Stephen W P Kemp,

Deanna H Gates, Cynthia A Chestek, and Paul S Cederna. Long-term upper-extremity prosthetic control using regenerative peripheral nerve interfaces and implanted emg electrodes. *Journal of Neural Engineering*, 20(2):026039, apr 2023.

[13] Md Abdur Rahim and Jungpil Shin. Hand movement activity-based character input system on a virtual keyboard. *Electronics*, 9(5), 2020.

[14] Md. Rokib Raihan and Mohiuddin Ahmad. Developing wearable human–computer interfacing system based on emg and gyro for amputees. In Mohiuddin Ahmad, Mohammad Shorif Uddin, and Yeong Min Jang, editors, *Proceedings of International Conference on Information and Communication Technology for Development*, pages 291–304, Singapore, 2023. Springer Nature Singapore.

[15] Feiyun Xiao, Jingsong Mu, Jieping Lu, Guangxu Dong, and Yong Wang. Real-time modeling and feature extraction method of surface electromyography signal for hand movement classification based on oscillatory theory. *Journal of Neural Engineering*, 19(2):026011, mar 2022.

[16] Asim Waris, Imran K. Niazi, Mohsin Jamil, Kevin Englehart, Winnie Jensen, and Ernest Nlandu Kamavuako. Multiday evaluation of techniques for emg-based classification of hand motions. *IEEE Journal of Biomedical and Health Informatics*, 23(4):1526–1534, 2019.

[17] Katarzyna Szymaniak, Agamemnon Krasoulis, and Kianoush Nazarpour. Recalibration of myoelectric control with active learning. *Frontiers in Neurorobotics*, 16, 2022.

[18] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. A brief review of domain adaptation. In Robert Stahlbock, Gary M. Weiss, Mahmoud Abou-Nasr, Cheng-Ying Yang, Hamid R. Arabnia, and Leonidas Deligiannidis, editors, *Advances in Data Science and Information Engineering*, pages 877–894, Cham, 2021. Springer International Publishing.

[19] Marina M.-C. Vidovic, Han-Jeong Hwang, Sebastian Amsüss, Janne M. Hahne, Dario Farina, and Klaus-Robert Müller. Improving the robustness of myoelectric pattern recognition for upper limb prostheses by covariate shift adaptation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(9):961–970, 2016.

[20] Jinqiang Wang, Dianguo Cao, Yang Li, Jiashuai Wang, and Yuqiang Wu. Multi-user motion recognition using semg via discriminative canonical correlation analysis and adaptive dimensionality reduction. *Frontiers in Neurorobotics*, 16, 2022.

[21] Xinyu Jiang, Berj Bardizbanian, Chenyun Dai, Wei Chen, and Edward A. Clancy. Data management for transfer learning approaches to elbow emg-torque modeling. *IEEE Transactions on Biomedical Engineering*, 68(8):2592–2601, 2021.

[22] Machiel J Zwarts, Gea Drost, and Dick F Stegeman. Recent progress in the diagnostic use of surface emg for neurological diseases. *Journal of Electromyography and Kinesiology*, 10(5):287–291, 2000.

[23] Ashirbad Pradhan, Jiayuan He, and Ning Jiang. Performance optimization of surface electromyography based biometric sensing system for both verification and identification. *IEEE Sensors Journal*, 21(19):21718–21729, 2021.

[24] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4, 2023.

[25] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[26] Paras Gulati, Qin Hu, and S. Farokh Atashzar. Toward deep generalization of peripheral emg-based human-robot interfacing: A hybrid explainable solution for neurorobotic systems. *IEEE Robotics and Automation Letters*, 6(2):2650–2657, 2021.

[27] Hyunin Lee, Dongwook Kim, and Yong-Lae Park. Explainable deep learning model for emg-based finger angle estimation using attention. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:1877–1886, 2022.

[28] Xinyu Jiang, Kianoush Nazarpour, and Chenyun Dai. Explainable and robust deep forests for emg-force modeling. *IEEE Journal of Biomedical and Health Informatics*, pages 1–12, 2023.

[29] Tao Zhou, Olatunji Mumini Omisore, Wenjing Du, Lei Wang, and Yuan Zhang. Adapting random forest classifier based on single and multiple features for surface electromyography signal recognition. In *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–6, 2019.

[30] Yanjun Qi. *Random Forest for Bioinformatics*, pages 307–323. Springer US, Boston, MA, 2012.

[31] Trevor Hastie, Jerome Friedman, and Robert Tibshirani. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, Berlin, 2009.

[32] Gérard Biau and Erwan Scornet. A random forest guided tour. *TEST*, 25(2):197–227, Jun 2016.

[33] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Comput. Surv.*, 51(5), aug 2018.

[34] B. Hudgins, P. Parker, and R.N. Scott. A new strategy for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 40(1):82–94, 1993.

[35] Fuqiang Ye, Bibo Yang, Chingyi Nam, Yunong Xie, Fei Chen, and Xiaoling Hu. A data-driven investigation on surface electromyography based clinical assessment in chronic stroke. *Frontiers in Neurorobotics*, 15, 2021.

[36] K. Englehart and B. Hudgins. A robust, real-time control scheme for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 50(7):848–854, 2003.

[37] Egon L. van den Broek, Marleen H. Schut, Joyce H. D. M. Westerink, Jan van Herk, and Kees Tuinenbreijer. Computing emotion awareness through facial electromyography. In Thomas S. Huang, Nicu Sebe, Michael S. Lew, Vladimir Pavlović, Mathias Kölsch, Aphrodite Galata, and Branislav Kisačanin, editors, *Computer Vision in Human-Computer Interaction*, pages 52–63, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[38] Kianoush Nazarpour, Ahmad R. Sharafat, and S. Mohammad P. Firoozabadi. Application of higher order statistics to surface electromyogram signal classification. *IEEE Transactions on Biomedical Engineering*, 54(10):1762–1769, 2007.

[39] Rami N. Khushaba, Ahmed Al-Ani, and Adel Al-Jumaily. Orthogonal fuzzy neighborhood discriminant analysis for multifunction myoelectric hand control. *IEEE Transactions on Biomedical Engineering*, 57(6):1410–1419, 2010.

[40] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. Feature reduction and selection for emg signal classification. *Expert Systems with Applications*, 39(8):7420–7431, 2012.

[41] F. Esposito, D. Malerba, G. Semeraro, and J. Kay. A comparative analysis of methods for pruning decision trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):476–491, 1997.

[42] Floriana Esposito, Donato Malerba, and Giovanni Semeraro. Simplifying decision trees by pruning and grafting: New results (extended abstract). In Nada Lavrac and Stefan Wrobel, editors, *Machine Learning: ECML-95*, pages 287–290, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[43] Geoffrey I. Webb. Decision tree grafting from the all-tests-but-one partition. In *16th International Joint Conference on Artificial Intelligence Proceedings*, volume 2 of *IJCAI International Joint Conference on Artificial Intelligence*, pages 702–707, December 1999. International Joint Conference on Artificial Intelligence 1999, IJCAI 1999 ; Conference date: 31-07-1999 Through 06-08-1999.

[44] Noam Segev, Maayan Harel, Shie Mannor, Koby Crammer, and Ran El-Yaniv. Learn on source, refine on target: A model transfer learning framework with random forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1811–1824, 2017.

[45] Ning Jiang, Thomas Lorrain, and Dario Farina. A state-based, proportional myoelectric control method: online validation and comparison with the clinical state-of-the-art. *Journal of NeuroEngineering and Rehabilitation*, 11(1):110, Jul 2014.

[46] Martyna Stachaczyk, S. Farokh Atashzar, and Dario Farina. Adaptive spatial filtering of high-density emg for reducing the influence of noise and artefacts in myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(7):1511–1517, 2020.

[47] Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1994.

[48] Zhi-Hua Zhou and Ji Feng. Deep forest. *National Science Review*, 6(1):74–86, 10 2018.

[49] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.