34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

# On the Challenge of Classifying 52 Hand Movements from Surface Electromyography

Ilja Kuzborskij, Arjan Gijsberts, and Barbara Caputo

*Abstract*— **The level of dexterity of myoelectric hand prostheses depends to large extent on the feature representation and subsequent classification of surface electromyography signals. This work presents a comparison of various feature extraction and classification methods on a large-scale surface electromyography database containing 52 different hand movements obtained from 27 subjects. Results indicate that simple feature representations as Mean Absolute Value and Waveform Length can achieve similar performance to the computationally more demanding marginal Discrete Wavelet Transform. With respect to classifiers, the Support Vector Machine was found to be the only method that consistently achieved top performance in combination with each feature extraction method.**

## I. Introduction

Active upper-limb prostheses have been under development for several decades with the ultimate aim of restoring most of the hands original functionality and appearance. These prostheses are commonly controlled by measuring original motor commands from the patients stump surface using surface electromyography (sEMG). Gradually, this technology has moved from control of a single prosthesis function, such as "open-close" grasp, to multifunction prostheses. Despite these significant technological advancements and commercial availability, acceptance among amputees has been found to be lacking [1]. Reasons for this include difficulty in controlling them and an insufficient level of dexterity for daily-life tasks.

Recent developments in mechatronics and robotics have demonstrated that mechanically dexterous prostheses are not only feasible, but also available on a commercial level (e.g., Touch Bionics' i-Limb). It follows that the issue of insufficient functionality is due primarily to the myoelectric control, rather than the prosthetic hardware itself. Research attention to increase acceptance of active prostheses should therefore focus on extracting relevant information from the myoelectric signals and subsequent classification thereof in terms of movement classes. Even though significant work has already been presented in this direction, it is not clear which methods are superior in this application domain and whether these can achieve satisfactory performance. Moreover, empirical validation of these methods is typically restricted to demonstrating feasibility on a proprietary dataset containing only a handful of movements acquired from an equally limited number of subjects.

This work is partially supported by the Swiss National Science Foundation Sinergia project Non-Invasive Adaptive Prosthetics (NinaPro).

I. Kuzborskij, A. Gijsberts, and B. Caputo are with the Idiap Research Institute, Centre Du Parc, Rue Marconi 19, CH-1920 Martigny, Switzerland ilja.kuzborskij@idiap.ch

We attempt to address this issue by presenting a comparison of common feature extraction and classification methods on the newly acquired and publicly available NinaPro dataset [2], [3]. A primary advantage of this dataset is that it contains 52 hand movements acquired from 27 healthy subjects. Aside from producing a direct quantitative comparison of various popular methods, we also intend to assess whether current state-of-the-art methods are in fact able to attain satisfactory levels of performance on this challenging setting. Lastly, the presented results also form a baseline to measure progress of future developments in dexterous myoelectric control.

The present paper is organized as follows. Section II contains a concise overview of related work on feature extraction and classification methods that have been used for myoelectric control of prostheses. A representative selection of these methods form the base of comparison and these methods are described in Section III. The experimental setup, including a description of the dataset and configuration of the methods, is presented in Section IV, followed by experimental results in Section V. Finally, Section VI draws conclusions from the results and contains possible directions for future work.

## II. Related Work

There is a vast body of work related to sEMG-based control of prostheses. Central to these approaches is a common process that can be subdivided in data acquisition and preprocessing, feature extraction, and finally grasp, posture or movement classification. Even though data acquisition and preprocessing may have a profound impact on final performance (e.g., number and position of electrodes, filtering), we will restrict ourselves to an overview of the representative literature on feature extraction and classification methods. The interested reader is referred to a general treatment of myoelectric prosthesis control [4], [5].

Selection of appropriate features for sEMG signals has been driven by expert knowledge of the application domain, such as features capturing a relation to force exerted at joints [6] or action potentials of motor units participating in the observed signal [7]. The extraction methods that have been employed successfully consider spectral and amplitude properties of the signal and can thus be categorized into those operating in the time domain (e.g., Mean Absolute Value, Variance, or Cepstral Coefficients) or frequency domain (e.g., Frequency Ratio or Mean Frequency). More sophisticated types of features may also consider the time and frequency domains simultaneously, such as in Short-Time Fourier Transform, Wavelet Transform, or Wavelet

978-1-4577-1787-1/12/$26.00 ©2012 IEEE 

Packet Transform. Typically, time-frequency domain features contain richer information about the signal at the cost of increased computation. Detailed overviews of feature extraction methods used for sEMG signals can be found in the work by Zecca *et al.* [8] and Micera *et al.* [5].

A number of studies have compared various extraction methods in terms of clustering criteria [9], [10] or final classification performance [11], [10], [12], [13]. Unfortunately, these comparisons most often identified different methods as best performing. A possible cause for these conflicting results is the considerable discrepancy in the respective acquisition protocols and experimental setups. Moreover, extraction methods were typically used with a single classifier, thereby failing to investigate whether extraction methods may perform better with some classifiers than with others.

The use of classification methods has mostly been restricted to relatively standard methods, such as Linear Discriminant Analysis [14], [15], [11], *k*-Nearest Neighbors [16], Gaussian Mixture Models [13], or Multi-Layer Perceptrons [15], [11], [12]. More recent work also reports the use of Support Vector Machines [11], [17], [18]. This apparent disinterest in exploring a broader variety of classifiers is partially explained by the belief that feature representations contribute more significantly to the overall performance than classifiers. This belief is supported by Hargrove *et al.* [13], who reported nearly identical results when comparing five different classifiers [13]. Lorrain *et al.* found similar performance between Support Vector Machines (SVMs) and Linear Discriminant Analysis (LDA) when using time-domain and autoregressive features [18]. It should be noted, however, that other comparisons have in fact found considerable differences in performance between classifiers [19], [11]. In this case, conflicting results might be caused by variability in the experimental setup, since the performance gain of powerful non-linear classifiers with respect to linear methods can be expected to increase with problem complexity (e.g., number of movements considered).

## III. Methods

The selection of features and classification methods for the present evaluation is based primarily on popularity in existing literature, with computational constraints being a secondary consideration. In total, we have selected seven feature extraction methods and four classification methods, which are motivated and described in the following.

### A. Feature Extraction

Our choice of feature extraction methods stems from several assumptions on sEMG: (1) There is a quasi-linear relation between Root Mean Square (RMS) amplitude of sEMG signal and force exerted by a muscle [6] subject to number of conditions, such as thickness of tissue, Motor Unit (MU) recruitment strategy and so on [20]; (2) sEMG can be modeled as a summation of Motor Unit Action Potential (MUAP) trains [7]; (3) sEMG spectral characteristics might be related to conduction velocity of muscle fibers, subject to number of conditions [20]. This relation can be indicative of recruitment of certain MU, which can help in discrimination of certain MU activation patterns.

TABLE I

Per-channel definition and order of dimensionality over all $C$ channels of the feature types used in the evaluation. The features $\hat{x}$ are computed from signal $x$ of length $T$ and subindexed by $t$. $B$ denotes number of HIST bins. For STFT, we consider $M$ frequency bins indexed with $k$ and computed over blocks obtained by a sliding window function $g$ of length $R$. For mDWT, we use $\psi_{l,\tau}$ to denote the mother wavelet with translation $l$ and dilation $\tau$, while the total number of considered translations is referred to as $L$.

| Feature | Definition (per channel) | Dim. |
|---|---|---|
| Mean Absolute Value (MAV) | $\hat{x} = \frac{1}{T}\sum_{t=1}^{T} \lvert x_t \rvert$ | $C$ |
| Variance (VAR) | $\hat{x} = \frac{1}{T}\sum_{t=1}^{T} (x_t - \bar{x})^2$ | $C$ |
| Waveform Length (WL) | $\hat{x} = \sum_{t=1}^{T-1} \lvert x_t - x_{t+1} \rvert$ | $C$ |
| sEMG Histogram (HIST) | $\hat{x}_{1:B} = \mathrm{hist}\,(x_{1:t}, B)$ | $CB$ |
| Cepstral Coefficients (CC) | $\hat{x}_k = \mathcal{F}^{-1}(\log \lvert \mathcal{F}(x_{1:t}) \rvert)_k$ | $CT$ |
| Short-Time Fourier Transform (STFT) | $\hat{x}_{k,t} = \sum_{m=0}^{R-1} x_{m-t} g_m e^{-i\frac{2\pi}{M}km}$ | $CMT$ |
| marginal Discrete Wavelet Transform (mDWT) | $\hat{x}_l = \sum_{\tau=0}^{T/2^l-1} \left\lvert \sum_{t=1}^{T} x_t \psi_{l,\tau}(t) \right\rvert$ <br> $\psi_{l,\tau}(t) = 2^{-\frac{m}{2}} \psi(2^{-l}t - \tau)$ | $CL$ |

If the first assumption holds, time domain features, such as Mean Absolute Value (MAV) when treated in multi-channel setting, could potentially encode profile of movement through force-related measurements. The second and third assumptions are related to time-frequency domain feature representations. As a more sensible counterpart of MAV, we use sEMG Histogram (HIST) [10]. For spectral analysis, we chose Short-Time Fourier Transform (STFT) since it is more robust when dealing with non-stationary signal compared to Fourier transform. As a more elaborate alternative Wavelet Transform (WT) was employed. MUAP is often modeled as a waveform with similar shape over time, which might alter under variable conditions such as electrode size or its position [20]. WT has been used extensively to capture localization and energy allocation of MUAP under condition that MUAP shape match the shape of used wavelet function as much as possible [14], [21], [18]. As a representative of WT we chose marginal Discrete Wavelet Transform (mDWT), since there is no particular interest in wavelet time instants. Instead, a cumulative energy allocated to wavelets within some signal segment suggests better discriminative criterion [21]. On the basis of these considerations, seven feature representations are considered in this work, five time-domain features and two time-frequency features. A short summary for those is given in Table I.

### B. Classification

In contrast to feature extraction methods, only a relatively small set of general purpose classification methods have been employed for myoelectric movement classification. The four well-known classifiers considered here have all been used

previously in related literature and span from simple statistical methods to more advanced machine learning techniques.

*1) Linear Discriminant Analysis:* LDA is a well-known statistical method to find a linear discriminant that maximizes the ratio of between-class scatter to within-class scatter [22]. The applicability of this method on a given dataset relies strongly on the assumption that the conditional probabilities of the features given the class labels are normally distributed.

*2) k-Nearest Neighbors:* The *k*-Nearest Neighbors (*k*-NN) algorithm classifies samples based on a majority vote among the $k$ closest training samples [22]. Despite its conceptual and computational simplicity, excellent performance can be achieved if sufficient training data is available. A possible disadvantage in time-critical applications is that all computation is deferred to the testing phase. Furthermore, its performance is critically dependent on the selection of $k$ and a suitable distance measure.

*3) Multi-Layer Perceptron:* A Multi-Layer Perceptron (MLP) is arguably the most popular type of feedforward Artificial Neural Network (ANN) [22]. The network is composed of at least three fully interconnected layers, namely an input and output layer, as well as one or more hidden layers in between these. Both hidden and output layers consist of a number of perceptrons (i.e., the "neurons"), which feed a weighted sum of inputs through a non-linear activation function. The weights are randomly initialized and subsequently optimized during supervised training by means of back-propagation. Preventing overfitting with MLPs requires careful selection of the stopping criteria of the optimization procedure and the number of neurons in each hidden layer.

*4) Support Vector Machine:* SVMs are linear binary classifiers that attempt to maximize the margin between the two classes [23]. Their widespread popularity is due to large extent to the possibility to use kernel functions. These functions allow SVMs (and many other algorithms) to be used on non-linear problems by implicitly mapping the data into a high or even infinite dimensional feature space. Though the standard SVM is defined in binary form, it can be used on multi-class problems by converting these to multiple binary classification problems. Advantageous properties of either linear or kernel SVMs include determinism and convexity of the optimization problem, which effectively guarantees convergence to a unique global optimum (cf. random initialization and local optima in MLPs). Furthermore, the balance between overfitting and underfitting can be regulated using a single trade-off hyperparameter $C$, which limits the classifier to a desired capacity.

## IV. Experimental Setup

This section contains detailed descriptions of the dataset, preprocessing, and configuration of the extraction and classification methods.

### A. Dataset

The NinaPro database has been acquired recently with the aim to advance the state of sEMG controlled hand prosthetics by forming a widely accepted benchmark dataset [2]. Aside from being publicly available[1], a primary advantage with respect to earlier datasets is the comparatively rich set of $52$ movements collected from $27$ intact subjects. This set of movements can be decomposed in four different categories, which are shown graphically in Figure 1. Movement classification is challenging with this large number and variety of movements, which makes this dataset ideal for a comparison of feature extraction and classification methods. Moreover, the large number of considered subjects allows for a reliable estimation of classification performance.

During the data acquisition, subjects were explicitly instructed to perform ten repetitions of each movement by imitating a video, while all movements were alternated with an intermediate rest movement. For the entire duration, data was recorded at $100\,\mathrm{Hz}$ from ten active sEMG electrodes, which already provides an amplified, bandpass-filtered, and an RMS rectified version of the raw sEMG signal. Eight of the electrodes were placed uniformly just beneath the elbow at a fixed distance from the radio-humeral joint, while two more were placed on the flexor and extensor muscles.

### B. Relabeling

The actual time window of the performed movement does not necessarily correspond perfectly to video duration, since subjects require time to react to a new video being played and may finish the movement prior to the end of the video. In order to reduce this label "noise", we employ an offline relabeling algorithm that constrains movement labels to those samples in which there is increased sEMG activity.

Similar to the onset detection approach by Staude [24], we remove irrelevant autoregressive components by whitening the signals using a multivariate VAR($p$) model [25]. In our case, an order of $p = 20$ was found to perform adequately. Detection of sEMG activity is restricted to the original video window extended with an additional $100$ samples at the end, as to allow subjects to finish a movement with $1\,\mathrm{s}$ of delay. The resulting feasible movement window of length $T$ is then divided in rest-movement-rest segments marked by change points $t_0$ and $t_1$. The optimal change points are found by maximizing the log-likelihood of a rest model $\boldsymbol{\theta}_0$ and movement model $\boldsymbol{\theta}_1$, corresponding to the objective function

$$\underset{1\le t_0\le T}{\arg\max}\ \underset{t_0\le t_1\le T}{\arg\max}\ \sup_{\boldsymbol{\theta}_0\in\Theta_0}\ \sup_{\boldsymbol{\theta}_1\in\Theta_1} \left[\sum_{i=1}^{t_0-1}\ln p_{\boldsymbol{\theta}_0}(\boldsymbol{y}_i)+\sum_{j=t_0}^{t_1-1}\ln p_{\boldsymbol{\theta}_1}(\boldsymbol{y}_j)+\sum_{k=t_1}^{T}\ln p_{\boldsymbol{\theta}_0}(\boldsymbol{y}_k)\right]. \tag{1}$$

Simple exhaustive search is adequate for finding optimal $t_0$ and $t_1$, while $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ are optimized by a maximum likelihood estimate of a multivariate Gaussian distribution over the corresponding window segments.

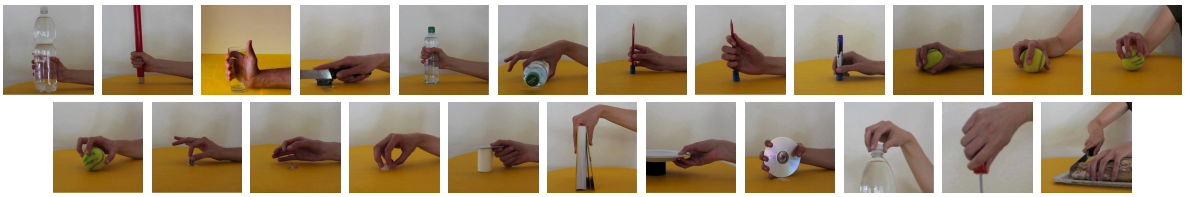
In order to (subjectively) improve the results in practice, we impose a minimum duration for both the rest (i.e., $t_0 \geq 10$) and movement window segments (i.e., $t_1 - t_0 \geq 0.3T$).

[1] http://www.idiap.ch/project/ninapro/

(a) 12 basic flexions and extensions of the fingers

(b) 8 isometric and isotonic hand configurations

(c) 9 basic wrist movements



(d) 23 grasp and functional movements

Fig. 1. The 52 movements considered in the NinaPro dataset.

Moreover, the straightforward assumption that sEMG activity is higher during movement is explicitly enforced by requiring the sample variance $s^2$ to be lower during rest (i.e., $s_0^2 \leq s_1^2$). This simple condition is effective at preventing erroneous outcomes in cases where a feasible window is lacking a clear initial rest. Finally, we impose a prior distribution on any sample belonging either to rest or movement (i.e., random variables $R_i$ and $M_i$). This prior is chosen uniformly as $p(R_i) = 0.1$ for $1 \leq i \leq T$, and due to mutual exclusivity $p(M_i) = p(\neg R_i) = 1 - p(R_i)$. The effect of this prior is that the algorithm will identify slightly larger movement windows, which helps to ensure that the entire sEMG activity is captured in the movement segment.

### C. Preprocessing and Data Split

Since collected sEMG signals are RMS pre-filtered during data acquisition [2], valuable information is contained within the low frequency spectrum band. To remove high-frequency equipment noise components we applied 2nd order $5\,\mathrm{Hz}$ low-pass zero-phase Butterworth digital filtering at each channel. Filtering of similar configuration was carried out by Castellini *et al.* [17], where signals were recorded by a similar acquisition setup. After filtering, each signal channel is segmented into windows. Windows of length $100\,\mathrm{ms}$, $200\,\mathrm{ms}$ and $400\,\mathrm{ms}$ with $N{-}10\,\mathrm{ms}$ overlap are considered, where $N$ is the window length in milliseconds[2]. Note, that $N{-}10\,\mathrm{ms}$ is simply one-sample sliding window, taking into account signal sampling frequency.

Subsequently, the dataset is split equally into training and testing set at the 50% ratio. Splitting is performed at the level of rest-repetition pairs, meaning that 5 repetitions with preceding rest segments are included in the training set while another 5 are kept for the testing set. Among multiple random splits considered by Atzori *et al.* [2], we chose a split yielding an accuracy as close as possible to the average one. Namely, training indices are $\{1, 3, 4, 5, 9\}$, while testing are $\{2, 6, 7, 8, 10\}$. After splitting, the training set is reduced by keeping every 10th sample to achieve a computationally feasible training set. Note, that from the standpoint of windowing this equates to sampling windows with $N{-}100\,\mathrm{ms}$ overlap. On the other hand, testing set overlap exploits all available testing data by selecting windows in a one-sample sliding window manner. Eventually different parameterizations yield $\approx 2.5 \cdot 10^4$ and $\approx 2.5 \cdot 10^5$ samples in training and testing set per subject.

### D. Method Configuration and Implementation

The features described in subsection III-A are extracted from each window independently for each electrode channel. Based on preliminary evaluation runs, we selected a 4-sample rectangular window for STFT. Furthermore, for mDWT we used a Symlet wavelet function of order four at the first three levels and first five coefficients for Cepstral Coefficients (CC). Finally, we used a 10 bin sEMG Histogram computed over a logarithmic scale, where a small constant is added to avoid $-\infty$ sample values. Apart from these, other features do not require any explicit parametrization.

Each feature representation is combined with the four classifiers described in section III, although an exception is made for LDA. Due to computational issues (i.e., a singular covariance matrix), LDA could not be evaluated with the high dimensional feature representations (i.e., STFT, CC, and HIST). For the remaining classification methods, the extracted features are standardized to have zero mean and

[2]In addition, we tried to conduct experiments with 800 ms window with no noticeable increase of performance.

unit standard deviation. During a preliminary evaluation standardized data resulted in better accuracy. The exception is LDA, for which standardization was not applied, due to singular covariance matrix issue.

We use SVM with Radial Basis Function (RBF) and linear kernels in one-vs-one multiclass classification setting, MLP with one hidden layer and sigmoid activation function, *k*-NN with Euclidean distance measure and LDA with no specific configuration. We chose linear SVM kernel to investigate the robustness of linear discriminative model on extracted feature set, while RBF kernel is dedicated to non-linear classification boundary modeling. MLP, *k*-NN and LDA are presented in most canonical configurations.

The mentioned classifier configurations span hyperparameters, which are tuned for each subject by grid search. At each grid point, 5-fold cross-validation is performed on the 10% random sample of the training set. During grid search we consider non-linear SVM $C \in \{2^i : i \in \{0, 2, ..., 14, 16\}\}$ and RBF $\gamma \in \{2^i : i \in \{-16, -14, ..., -4, -2\}\}$, linear kernel SVM $C \in \{2^i : i \in \{-10, -8, ..., 14, 16\}\}$, MLP hidden unit number in $\{4, 8, 16, 32, 64, 100, 150, 200, 250, 300, 400\}$ and *k*-NN with $k \in \{1..7\}$. The exception is MLP, for which, instead of cross-validation, the original training set is split into 90%/10% training/validation sets, and optimized with early stopping to avoid overfitting. The early stopping criterion is set as 20 consequent decreases of accuracy on the validation set. The maximum number of MLP optimization iterations are set to 1000 and optimized with error back-propagation by scaled conjugate gradient descend [26].

To solve the SVM optimization problem with RBF kernel we have used LibSVM [27], and LibLINEAR [28] for linear kernel, since LibLINEAR is more efficient in solving linear model optimization. The MLP optimization problem was solved by NetLab toolbox [26]. For LDA we have used Discriminant Analysis Toolbox [3]. *k*-NN was evaluated using MATLAB Statistics Toolbox implementation.

## V. Results

We present evaluation results in three perspectives: classification accuracies for a given method and feature representation, movement classification accuracy over normalized time and misclassification analysis by confusion matrices. Finally we make a comment on evaluation timings.

Classification accuracies for the methods considered in Section III-B with respect to the feature representations described in Section III-A under configuration outlined in Section IV are summarized in Figure 2. Given this, the best performing combinations are SVM with RBF kernel, MLP, and *k*-NN in conjunction with MAV, HIST, mDWT, and STFT features. The good performance of non-linear classifiers might be due to the non-linearity of the problem; this seems confirmed by the poor performance obtained by linear models (linear kernel SVM and LDA). In some cases, the performance obtained using the MAV and mDWT features is found to be statistical significant ($p \leq 0.05$, sign test) with respect to some classifiers and windows lengths, but in most cases those yield insignificantly different accuracies.

To give a general view on the misclassification profile with respect to individual movements, we present in Figure 4 the confusion matrix for the mDWT features combined with the non linear SVM. Clearly, the diagonal components suggest that the classifier is mostly consistent in giving correct predictions. On the other hand, the non-clear first column strongly points out that a considerable number of movements are misclassified as rest (absence of movement). To our understanding, this might occur for possible reasons:

- Due to the windowing of a signal, non-rest labeled windows can include both rest and non-rest samples. This might aggravate misclassification rate specifically during the initial period of the movement. Feature-wise this means that, although a sequence of windows might be tagged with the same movement label, in reality these windows can contain portions of rest signal.
- In this work we consider movements rather than stand-alone postures or grasps. This introduces the problem of label semantics, since rest-to-movement segments can be attributed neither to rest, nor to movement. Relabeling of the dataset, explained in Section IV-B, reduces the amount of clearly mislabeled samples, but cannot resolve the problem of ambiguous labels. This issue of transitional segments was also mentioned in [14], [11], [12], and usually "solved" by ignoring transitional segments between movements.

Apart from that, an off-diagonal scatter can be observed. Although magnitudes of individual off-diagonal scatter elements are small ($\leq 10\%$), when marginalized over rows, they can account for reasonable misclassifications in general. One of the reasons can be an inability of the feature representation to capture sufficient discriminative information. Another is the lack of relevant physiological information acquired during recording sessions.

Having in mind a transient signal, it is of interest to investigate whether different features react to transitional windows in different ways. To explore these discrepancies, a movement time-normalized accuracy plot is presented in Figure 3. An increase of performance can be observed as movement windows contain progressively less transitional samples. As the number of transitional samples increases towards the end of the movement, performance deteriorates. In some cases, longer windows result in better performance at the central part of the movement and onwards. This suggests that the window length has little effect on the discrimination of transitional segments and the following movement. It also should be noted that at the start of each movement the considered features suffer from rest and rest-to-movement transitional segment history. This is evident from the steepness at the beginning of each curve. MAV showed to be more robust to this kind of misclassifications as compared to

[3] http://www.mathworks.com/matlabcentral/fileexchange/189-discrim

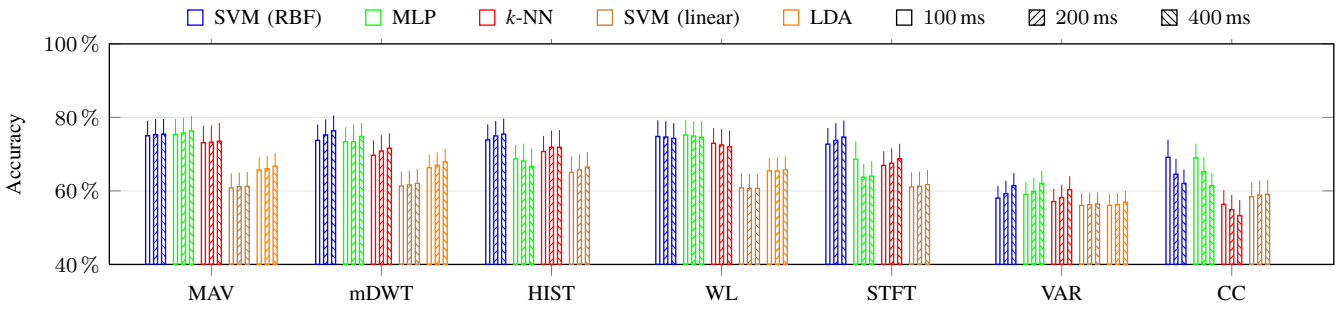



Fig. 2. Classification accuracies. Each bar represents method classification accuracy with respect to feature representation and window length, while line atop the bar is one standard deviation of accuracy. Classifiers are grouped by feature representations and labeled by different colors. Window lengths are represented in increasing order, namely 100ms, 200ms and 400ms and are tagged with different textures. Note, that LDA results are missing in case of STFT, CC and HIST due to reasons described in Section IV.

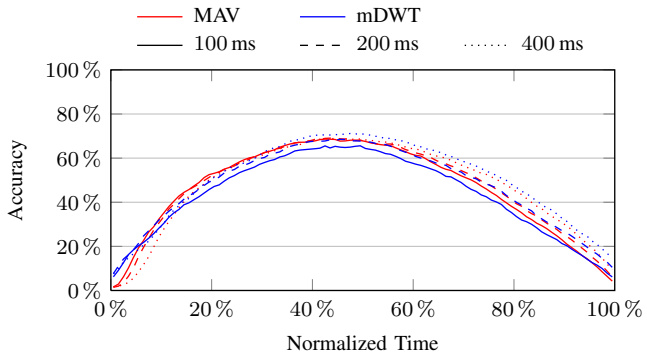



Fig. 3. Classification accuracy with respect to normalized movement duration. Each curve follows a histogram, where each bin represents ratio of correct classification counts to total number of samples within considered bin. Ratios are averaged over all subjects and movements, while keeping duration normalized, meaning that number of bins is equal for all movements. Each curve corresponds to evaluation by SVM with RBF kernel with respect to Mean Absolute Value and marginal Discrete Wavelet Transform features at windows of length 100ms, 200ms, and 400ms.

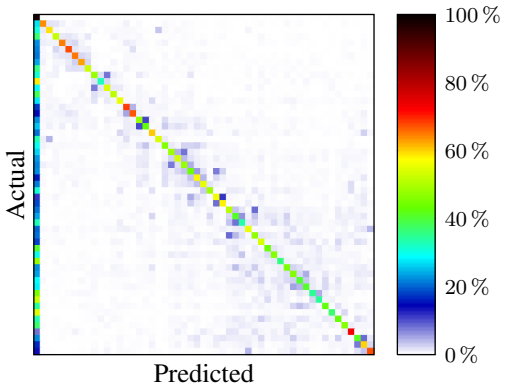



Fig. 4. Confusion matrix for the SVM classifier with mDWT feature representation and 400 ms window length. Each cell represents prediction accuracy of row-indexed class. First class is a rest, i.e. absence of movement. Thus, correct predictions would result in clear left-top to right-bottom diagonal, while off-diagonal cells are indicative of misclassifications.

mDWT. On the other hand, mDWT demonstrates an overall accuracy improvement at the center of a movement, when the number of relevant samples within a window increases.

Finally, we would like to give an approximate estimate of the runtime of some algorithms in conjunction with feature extraction methods. It is impossible to give precise timings, since the tasks were distributed over a computing grid, spanning a variety of machine configurations. Therefore the average of those should give an idea of the computational requirements. SVM gives the best trade-off between effectiveness and efficiency: computing results for one subject with feature extraction and hyperparameter tuning took on average 15 minutes for MAV, 50 minutes for mDWT, and from 40 to 160 minutes for STFT, depending on the window length. MLP is more demanding, which resulted in 240 minutes in case of MAV and mDWT, and up to 350 minutes in case STFT. Finally, *k*-NN in standard implementation suffers from extreme computational effort growth with the increasing dimensionality of the data. For MAV it took *k*-NN on average 10 minutes to complete, while for mDWT this increases to 40 minutes and, finally, up to 500 minutes for STFT. Highest prediction time for SVM/MAV pair amounted to 600 s for complete testing set.

## VI. Conclusions

In this work we have conducted a benchmark evaluation of a large-scale surface electromyography dataset, containing 52 different movements collected from 27 subjects. Prior to evaluation, we introduced a relabeled version of the dataset by correcting rest and movement transition time points. Four classifiers previously used in the sEMG community were evaluated for each subject with respect to seven well-known sEMG features and three different window lengths. None of the classifier-feature-window combinations exceeded an 80% classification accuracy on average, but approached it sufficiently close.

The evaluation reveals a considerable superiority of non-linear classifiers (e.g., Support Vector Machine) over linear classifiers (e.g., Linear Discriminant Analysis). On the other hand, in most cases no considerable difference has been noticed between time domain and time-frequency domain features given the best performing classifiers. Accuracy analysis over the movement duration indicated poor performance

during transitional movement segments at the beginning and the end of each movement, with only slight differences with respect to different features. Class-wise misclassification analysis pointed out that classification is mostly consistent in predictions, although priority in improvement should be given to rest and movement misclassifications, as these are mostly confused.

Evaluation timings on testing set suggest, that SVM might be considered as a predictor for use in real-time systems. The hint comes from prediction time per sample of $\approx 2.4$ ms, which is lower than acquisition rate of sample in 10 ms.

Summarizing effectiveness, efficiency, and method tuning experience, we conclude that SVM is the most suitable classifier for the task at hand. In comparison, MLP requires intricate parameter tuning to achieve comparable performance, such as setting the number of iterations, stopping condition, or network configuration. *k*-NN suffers from the "curse of dimensionality", furthermore it requires all training data during testing, which might be impractical in real-life settings. Linear models such as SVM with linear kernel and LDA yield unsatisfactory accuracy.

### A. Future Work

The obtained results are admittedly far from what can be considered usable in real-life settings, but on the other hand they clearly suggest challenging tasks in both myoelectric control and machine learning perspectives. It is likely that to tackle the challenges at hand, novel machine learning or feature extraction methods will have to be introduced. That said, the presented work gives a good baseline for future comparisons on the NinaPro dataset.

A number of directions can be addressed in future work, namely: (1) improving performance of per-subject classification by combining features or introducing new features; (2) assessing the possibility of inter-subject models, meaning, the same model capable of classifying movements from multiple subjects; (3) adaptive control, allowing rapid introduction of new movements and new subjects by exploiting information from already trained models; (4) collection of an extended dataset, possibly with an altered acquisition setup given conclusions based on the current dataset.

## REFERENCES


[1] D. J. Atkins, "Epidemiologic overview of individuals with upper-limb loss and their reported research priorities," *Journal of Prosthetics & Orthotics*, vol. 8, no. 1, pp. 2–11, 1996.

[2] M. Atzori, A. Gijsberts, S. Heynen, A.-G. M. Hager, O. Deriaz, P. van der Smagt, C. Castellini, B. Caputo, and H. Müller, "Building the Ninapro database: A resource for the biorobotics community (submitted)," in *Proceedings of IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob 2012)*, 2012.

[3] M. Atzori and H. Müller, "NINAPRO project first milestone: Set up of the data base," Institute of Business Information Systems, University of Applied Sciences Western Switzerland, Sierre, Switzerland, Tech. Rep., 2012, available at http://publications.hevs.ch/index.php/publications/show/1165.

[4] P. Parker, K. Englehart, and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *Journal of Electromyography and Kinesiology*, vol. 16, no. 6, pp. 541–548, 2006.

[5] S. Micera, J. Carpaneto, and S. Raspopovic, "Control of hand prostheses using peripheral information," *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 48–68, 2010.

[6] C. De Luca, "The use of surface electromyography in biomechanics," *Journal of applied biomechanics*, vol. 13, pp. 135–163, 1997.

[7] R. Merletti and P. Parker, *Electromyography: Physiology, engineering, and noninvasive applications*. Wiley-IEEE Press, 2004, vol. 11.

[8] M. Zecca, S. Micera, M. Carrozza, P. Dario *et al.*, "Control of multifunctional prosthetic hands by processing the electromyographic signal," *Critical Reviews in Biomedical Engineering*, vol. 30, no. 4-6, p. 459, 2002.

[9] R. Boostani and M. H. Moradi, "Evaluation of the forearm emg signal features for the control of a prosthetic hand," *Physiological Measurement*, vol. 24, no. 2, p. 309, 2003.

[10] M. Zardoshti-Kermani, B. Wheeler, K. Badie, and R. Hashemi, "EMG feature evaluation for movement control of upper extremity prostheses," *IEEE Transactions on Rehabilitation Engineering*, vol. 3, no. 4, pp. 324–333, 1995.

[11] M. Oskoei and H. Hu, "Support vector machine-based classification scheme for myoelectric control applied to upper limb," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 8, pp. 1956–1965, 2008.

[12] F. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. Thakor, "Decoding of individuated finger movements using surface electromyography," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1427–1434, 2009.

[13] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE Transactions on Biomedical Engineering*, vol. 54, pp. 847–853, 2007.

[14] K. Englehart, B. Hudgin, and P. Parker, "A wavelet-based continuous classification scheme for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 3, pp. 302–311, 2001.

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

[16] P. C. Doerschuk, D. E. Gustafon, and A. S. Willsky, "Upper extremity limb function discrimination using emg signal analysis," *IEEE Transactions on Biomedical Engineering*, vol. 30, no. 1, pp. 18–29, 1983.

[17] C. Castellini, E. Gruppioni, A. Davalli, and G. Sandini, "Fine detection of grasp force and posture by amputees via surface electromyography," *Journal of Physiology-Paris*, vol. 103, no. 35, pp. 255–262, 2009.

[18] T. Lorrain, N. Jiang, and D. Farina, "Influence of the training set on the accuracy of surface EMG classification in dynamic contractions for the control of multifunction prostheses," *Journal of NeuroEngineering and Rehabilitation*, vol. 8, no. 1, p. 25, 2011.

[19] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson, "Classification of the myoelectric signal using time-frequency based representations," *Medical Engineering & Physics*, vol. 21, no. 6-7, pp. 431–438, 1999.

[20] D. Farina, R. Merletti, and R. M. Enoka, "The extraction of neural strategies from the surface EMG," *Journal of Applied Physiology*, vol. 96, no. 4, pp. 1486–1495, 2004.

[21] M.-F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli, and D. Farina, "Multi-channel surface EMG classification using support vector machines and signal-based wavelet optimization," *Biomedical Signal Processing and Control*, vol. 3, no. 2, pp. 169–174, 2008.

[22] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2001.

[23] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[24] G. Staude, "Objective motor response onset detection in surface myoelectric signals," *Medical Engineering & Physics*, vol. 21, no. 6-7, pp. 449–467, 1999.

[25] J. D. Hamilton, *Time Series Analysis*. Princeton University Press, 1994.

[26] I. Nabney, *NETLAB: Algorithms for Pattern Recognition*. Springer-Verlag New York, Inc., 2002, software available at http://www1.aston.ac.uk/eas/research/groups/ncrg/resources/netlab.

[27] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1–27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[28] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.