

Advancing Muscle-Computer Interfaces with High-Density Electromyography

Christoph Amma, Thomas Krings, Jonas Böer, Tanja Schultz

Karlsruhe Institute of Technology (KIT)

{christoph.amma, tanja.schultz}@kit.edu, {thomas.krings, jonas.boeर}@student.kit.edu

ABSTRACT

In this paper we present our results on using electromyographic (EMG) sensor arrays for finger gesture recognition. Sensing muscle activity allows to capture finger motion without placing sensors directly at the hand or fingers and thus may be used to build unobtrusive body-worn interfaces. We use an electrode array with 192 electrodes to record a high-density EMG of the upper forearm muscles. We present in detail a baseline system for gesture recognition on our dataset, using a naive Bayes classifier to discriminate the 27 gestures. We recorded 25 sessions from 5 subjects. We report an average accuracy of 90% for the within-session scenario, showing the feasibility of the EMG approach to discriminate a large number of subtle gestures. We analyze the effect of the number of used electrodes on the recognition performance and show the benefit of using high numbers of electrodes. Cross-session recognition typically suffers from electrode position changes from session to session. We present two methods to estimate the electrode shift between sessions based on a small amount of calibration data and compare it to a baseline system with no shift compensation. The presented methods raise the accuracy from 59% baseline accuracy to 75% accuracy after shift compensation. The dataset is publicly available.

Author Keywords

EMG; Muscle-Computer-Interfaces; Gesture recognition; electrode arrays.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI); Miscellaneous

INTRODUCTION

While body-worn computer systems become more and more popular, the question of the ideal input devices is still open. Since output devices are often decoupled from the input device, keyboard and touchscreen based solutions are inappropriate. A combination of speech and gesture interfaces seems most promising. While speech is the most intuitive way of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

CHI 2015, April 18 - 23 2015, Seoul, Republic of Korea
 Copyright is held by the owner/author(s). Publication rights licensed to ACM.
 ACM 978-1-4503-3145-6/15/04\$15.00
<http://dx.doi.org/10.1145/2702123.2702501>

entering information in situations in which the computer is perceived as communication partner, gestures are especially suited for spatial or silent and inconspicuous interaction. Gestures could be performed with the whole arm, the hand or the fingers: small sized gestures are generally more appropriate for wearable computing, since large scale gestures are tiring for the user and humans are specialized in fine-grained manipulation with their fingers. Therefore, interfaces allowing for fine-grained gestures and manipulation of virtual objects seem promising. Possible application scenarios are controlling smart glasses or watches with subtle finger motions and gestures.

The application of sensors directly at the fingers itself e.g., with data gloves, is inappropriate in daily-life situations. As a result, multiple approaches have been proposed to sense finger motion distantly. This includes body-worn cameras [12, 19], wrist-worn depth cameras [9], measuring the movement of the tendons [14] or directly measuring the muscle activity via Electromyography (EMG) [2, 10, 16, 17, 22]. The EMG approach and also the related tendon approach make use of the fact that most of the major finger muscles are located at the forearm and thus allow for sensing finger motion by attaching sensors around the forearm instead of the fingers. However, inferring the finger motion from the EMG is a challenging task. The positioning of the sensors is crucial for acquiring the right signals. Furthermore, there are interpersonal differences in the anatomy of the muscles and tissue layers. Wearable EMG based interfaces should not require exact positioning of electrodes and ideally autocalibrate themselves [18].

In contrast to earlier work we explore the use of high-density electrode arrays for EMG based gesture recognition. We analyze the impact of the number of electrodes on the recognition performance and show that such arrays can improve recognition accuracy and thus, an integration in future wearable devices would be beneficial. Additionally we introduce two methods for calibration across recording sessions of the same user and evaluate them on cross-session data. Furthermore the dataset is publicly available and can serve as a common ground for future research.

We do not present a readily wearable interface in this paper but rather want to lay the basis for follow-up work on high-density EMG based interfaces. Although technically challenging, it is possible to miniaturize all components and build devices with a high number of electrodes in the future.

Related Work

There have been numerous works on using EMG for Human Computer Interaction, in particular for recognizing hand and finger movement for gesture control and input. One can discriminate approaches which use electrodes that are precisely attached over the muscle [4, 10] and approaches that use array-like arrangements of electrodes to capture the activity under a certain area of the forearm [2, 16, 17, 18, 22]. Exact positioning of electrodes is in our opinion not a feasible method for wearable devices that need to be taken on and off quickly but is often used for controlling prostheses. Therefore we restrict our review mainly on related work that uses array-like electrode arrangements that do not require exact positioning of single electrodes.

Wheeler et al. [22] introduced a wearable EMG based controller which they used to recognize digits typed on an imaginary number keypad. Eight electrode pairs were arranged in two rings. One was placed distal (close to the wrist) and one proximal (close to the elbow) around the forearm in a regular grid. They applied hidden Markov models and performed cross-session evaluation of the keypad recognition reaching accuracies of 70% to 100% for the individual classes (overall mean accuracy is not reported). The differences of electrode placement between sessions lead to a degradation of performance. They propose the use of more electrodes for spatial oversampling. Saponas et al. [17] used an armband with 8 EMG channels at the proximal part of the forearm. They define 18 gestures which are divided in 4 smaller subsets (containing 4 to 5 classes each) and classified the gestures within each set using Support Vector Machines (SVM). The gestures include subtle motion like tapping of individual fingers. Accuracies from 78% to 95% were reached on the different sets. A preliminary evaluation for a person-independent setting revealed a strong degradation of the accuracy due to inter-individual differences and differences in the positioning of electrodes. In a follow-up work Saponas et al. [18] use a wireless armband and investigate performance under realistic conditions including cross-session evaluation on different days. Similarly to our work they only ask the study participants to try to attach the armband in approximately the same position than in the first session. For two gesture sets each containing three finger gestures they reach an accuracy of 86% and 76% for the within-session case and 86% and 66% for the cross-session case respectively using a calibration gesture to normalize the features. Assad et al. [2] also propose the use of a wearable EMG cuff for various control tasks. The cuff is equipped with 16 dry electrodes and an inertial measurement unit (IMU). In a small user study ($N=3$), SVMs are used for session-dependent classification of 17 static hand and finger gestures solely on EMG data, for which an accuracy of 96.6% was reached. A set of 9 dynamic gestures was classified with Dynamic Time Warping (DTW) using EMG and IMU data reaching an accuracy of 99%. Samadani and Kulić [16] use a prototype commercial wearable EMG device from Thalmic Labs to analyze person-independent recognition of hand gestures. On data acquired from 25 subjects performing 21 gestures they reach a person-independent accuracy of 49% (chance 4.8%). They use Hidden

Markov Models to model the sequential patterns occurring during performing the gesture.

High-density EMG has mainly been used for research on how muscle activation patterns are composed and distributed [11] e.g., the identification and analysis of single motor unit action potentials (MUAP). Recently such electrode arrays have also been used to classify gestures and decode finger and hand motion. Rojas et al. [15] use electrode arrays at the upper arm and forearm to discriminate 12 motion classes, also involving large muscles like the biceps brachii. They reach accuracies of over 90%. Muceli et al. [13] decoded hand (not finger) kinematics from high-density EMG recordings by performing a regression on joint angles with artificial neural networks. They also evaluated the impact of using only a subset of the many EMG channels which revealed that a high number of electrodes increases performance. The evaluations were performed for the within-session case. Wand et. al [21] use EMG arrays attached to the face for silent-speech recognition. They analyze cross-session performance and successfully apply shift compensation by correlating all possible shifts within certain boundary conditions and choosing the shift with highest correlation.

Recently the open NinaPro database was established for myoelectric movement classification [3]. It aims primarily at researchers in the field of prostheses but includes EMG recordings of hand gestures which are of interest for HCI researchers. Besides individually placed electrodes, an armband with eight electrodes is attached around the proximal side of the forearm as it is common in HCI related EMG research. A total of 52 gestures and grasp types were recorded, on which a within-session accuracy of 76% is reached.

Generally, the performance of an EMG based recognition system strongly depends on multiple factors, the type and number of gestures, the type, number and placement of electrodes, and the conditions under which the subjects performed the recordings. Authors defined their own gesture sets and recorded data under varying conditions. Therefore, it is hard to compare different approaches to the problem based on numbers given in different publications. However, related work clearly indicates that competitive accuracies can be reached for the within-session case and a strong degradation of performance is encountered for the session and person independent case. How to deal with major changes in electrode positions across sessions or a generalization of the classifier across subjects are still open research questions.

Structure of the paper

In the remainder of the paper we will 1) give an introduction to forearm EMG, 2) introduce the used methodology, including EMG measurement, classification technique and the algorithms for estimation and correction of electrode displacement, 3) describe the conducted experiment, 4) discuss the results and 5) finally conclude the paper.

BACKGROUND

Figure 1 shows an illustration of the forearm muscles with the relevant finger muscles indicated by numbers. The muscles for the thumb are all located distal and the muscles which

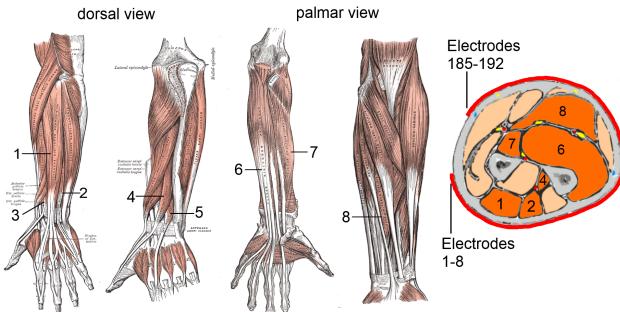


Figure 1. Illustration of the relevant forearm muscles. The dorsal view shows the extensors, the palmar view shows the flexors. The right most picture shows a cross section of the forearm in the middle between the distal and the proximal end. The red line around it indicates the approximate position of the electrode array. The muscles are 1: M. extensor digitorum communis, 2: M. extensor digiti minimi, 3: M. extensor pollicis brevis, 4: M. extensor pollicis longus 5: M. extensor indicis, 6: M. flexor digitorum profundus, 7: M. flexor pollicis longus, 8: M. flexor digitorum superficialis (pictures from Wikimedia Commons)

are in parts responsible for the pinkie and index finger are located distal as well. The main muscles for the fingers except the thumb are located proximal. In our experiment, we apply electrodes at the proximal part of the forearm, therefore we are able to capture at least parts of all finger's muscle activities except the thumb.

Electromyography

Surface Electromyography (EMG) denotes a technique for recording the electrical activity produced by muscles on the surface of the skin. Since the electrical activity correlates with the amount of force the muscle produces, it can be used to measure force and resulting motion.

Motion recognition based on EMG signals incorporates several challenges. One is the positioning of the sensors. In a real world day-to-day scenario, one does not want to narrow down the exact position for recording a certain muscle. Furthermore, there are inter- and intrapersonal differences in the signals. Interpersonal differences arise mainly from differences in anatomy, namely the actual size and position of the muscles and the properties and thickness of the skin and fat layer. In [3] Atzori et al., identify a negative correlation between EMG based gesture recognition accuracy and Body-Mass-Index. Intrapersonal differences arise from the dependence of the signal on parameters like skin conductance which vary with personal and environmental conditions.

METHODOLOGY

Within this section we describe the experimental setup, the applied classification pipeline, and the two methods for electrode shift compensation.

Experimental setup

We use an electrode array with 192 single electrodes arranged in a regular 8x24 grid with an inter-electrode distance of 10 mm. Both electrodes and the amplifier are from OT-Bioelettronica¹. Figure 2 shows a picture of the array.

¹<http://www.otbioelettronica.it/>

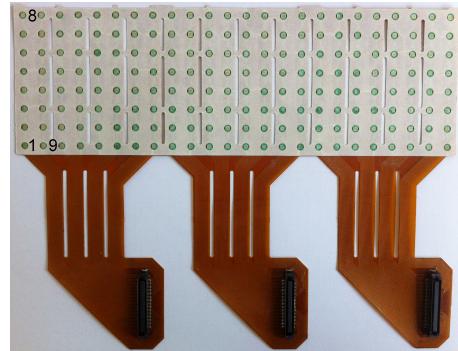


Figure 2. Picture of the electrode array used. The inter-electrode-distance is 10 mm. The electrodes are arranged in a 8x24 grid. Electrode numbering starts at the lower left corner and works columnwise.

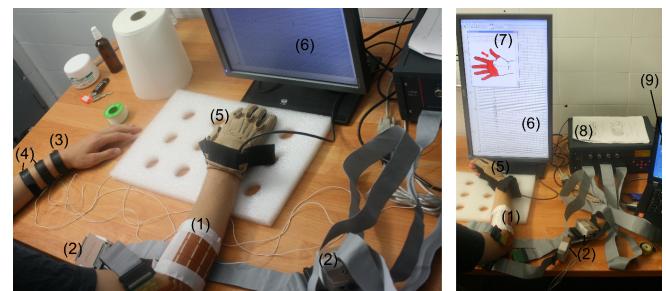


Figure 3. Picture of the actual setup used: (1) electrode array, (2) preamplifier, (3) Reference electrode, (4) DRL circuit, (5) data glove (not used), (6) real-time signal visualization, (7) guiding software, (8) amplifier, (9) recording computer

array is connected via three flat cables, each one with 64 wires. A pre-amplifier is attached to every cable closely to the electrodes to minimize noise caused by cable length. We made bipolar recordings, which means the amplifiers takes the difference between consecutive channels resulting in 192 channels in total. This means, every eighth channel does not contain meaningful data, because it contains the difference of the last channel in one column and the first channel in the next column. These channels are therefore ignored for the data analysis, leading to a total of 168 channels of usable data.

The hardware setup we used is actually very bulky, however the works by Assad et al. show for example a wearable system with 16 electrodes. With further miniaturization and integration of components it should be straightforward to increase the number of electrodes. Furthermore our analysis shows that there is no need to use all 168 channels but that a number between 20 and 80 already gives a good trade-off between accuracy and complexity of the technical system.

All data was recorded with a sampling rate of 2048 Hz. The amplifier has an adjustable gain factor, which was set to 1000. The data is delivered via USB to a computer. We used BiosignalsStudio [7] to record the signals, which is a signal recording framework for multimodal data sources that offers synchronization facilities with other programs via timestamps. The data recordings were synchronized with the guiding software that presented the stimuli for the gestures to the user.

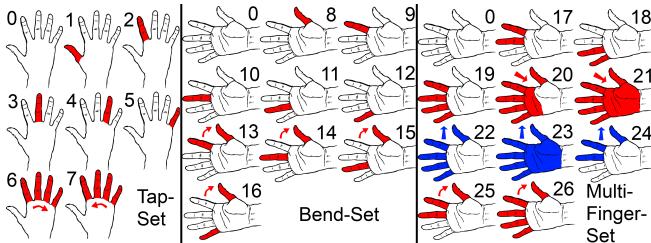


Figure 4. Iconic illustrations of the performed gestures. The gestures were divided into three sets: the tap-set, the bend-set and the multi-finger-set. The idle gesture (0) is contained in all three sets.

Figure 3 shows a picture of our actual recording setup. The subject was sitting on a chair in front of a screen. The screen showed the guiding software and the EMG signals in real-time. The screen was rotated to portrait format so that all 192 EMG channels could be visualized at once. This was done to be able to check the signals continuously for any anomalies e.g., if parts of the array were peeled off from the skin. The subject also wore a Cyberglove dataglove, which was intended to deliver accurate ground truth data for the finger motion. However, due to hardware synchronization issues, we unfortunately weren't able to use the acquired data.

Gesture set

We recorded a pre-defined set of 27 gestures, including an idle gesture. The set of gestures was chosen to cover extension and flexion of each individual finger and also incorporate some typical gestures that might be used in a HCI context. The gestures were organized in three sets, a set of tapping gestures, a set of bending gestures and a set of gestures including more complex multi-finger movements. Figure 4 shows iconic illustrations of the gestures that we recorded, which will be explained below.

Tapping gestures

The tap-set contains gestures which were performed with the hand lying flat on the desk in front of the subject (see left-hand picture in figure 3). For each of the gestures in the set, the subjects were asked to tap once with the finger(s) shown in red (gestures 1 to 5). In case of gestures 6 and 7, all four fingers should be tapped sequentially in the direction of the arrow. The tapping gesture set was designed to contain the muscle activity of the extensors. The actual tap motion after lifting the finger does not necessarily contain actual muscle activation, since it could be performed by simply relaxing the muscles involved in stretching. However this behavior was observed and not enforced.

Bending gestures

The bend-set was designed to contain activity of the flexor muscles. It contains bending of each of the fingers (gestures 8 to 12) and also pinch grips with the thumb (gestures 13 to 16) as these might be appropriate for interaction systems. Subjects were told to rest their hand with the ulnar side on the table (see right-hand picture in figure 3).

Multi-finger gestures

The multi-finger-set contains gestures that involve movement of multiple or all fingers in one gesture. These gestures were

performed in the same hand position as in the bend-set. Gestures 17-19 correspond to a bending of two or four fingers similar to the gestures 9-12 in the bend-set. Gestures 20 and 21 correspond to making a fist, where gesture 20 means making a fist without applying force and gesture 21 means making a fist with force. Gestures 22 to 24 correspond to stretching out the fingers marked in blue. Gesture 22 means a light stretch of all fingers and gesture 23 means a stretch with force of all fingers. In gesture 24, thumb, index and middle finger are stretched while ring finger and pinkie are bent. Gestures 25 and 26 correspond to pressing the marked fingers against the thumb and are thus similar to the pinch grip gestures in the bend-set.

Classification Pipeline

We apply a standard classification scheme for our gesture recognition system. It consists of a segmentation step, in which we identify the actual segment of muscle activity within the recorded 3 s of data for each gesture. We then extract features on the segmented data and apply a naive Bayes classifier to discriminate all 27 classes. Before describing the individual steps in detail, we will shortly explain Root-Mean-Square (RMS) and baseline normalization, since both appear multiple times within the section. Root-Mean-Square (RMS) is a common measure of muscle activity. The RMS of an electromyogram has a high correlation to the force generated by the muscle and is therefore an adequate and widely used feature [6] for quantifying muscle activity.

By baseline normalization we mean the subtraction of the average activity measured during idle gestures from all gestures recorded. This allows for compensation of long-term artifacts leading to constant offsets in the activity patterns. Such offsets typically occur, if electrodes are partly or completely detached over multiple repetitions of gestures. Baseline normalization is done after computing the RMS for the data. We subtract the average RMS values acquired for the idle gesture from the RMS values of the desired signal. For each session and gesture set (tapping, bending, multi-finger) we compute the average RMS value over all repetitions of the idle gesture for each of the 168 used channels. The resulting vector is used for the baseline normalization of the data in this particular session and set.

We will now describe the individual steps of the classification pipeline:

Signal Preprocessing

EMG recordings typically suffer from different kind of signal artifacts e.g., powerline noise or skin motion. When recording such a high number of channels in parallel, it is hard to avoid any of these artifacts. Consequently, the presented dataset also contains different kinds of artifacts. Besides powerline and cable motion artifacts, the movement of the skin under the electrodes can lead to partial or complete detachment of single electrodes. The resulting artifacts are usually high amplitude, low frequency signal distortions. The signals were observed in real-time during experiments, to make sure artifacts appear only in a small amount of channels. The number of artifacts is typically higher in the multi-finger-set, since the forceful movement of all fingers leads to more skin

deformation than gestures in the tap-set and bend-set. According to [8] and [13], we applied a fourth order butterworth band-pass filter with a pass-band of 20-400 Hz to attenuate dc offset, motion artifacts, and high-frequency noise.

Segmentation

In the segmentation step, we determine the segment containing the actual muscle activity within the 3 s window for each gesture. We use a thresholding approach on a sliding window and search the longest contiguous sequence of windows containing activity. We use non-overlapping windows of length 150 samples, which equates 73.2 ms. The last incomplete window is ignored. On each window, we compute the RMS for each channel. We then perform baseline normalization. Each window is therefore represented by a 168-dimensional vector of RMS values. We apply a spatial order 3 1-dimensional median filter on the vector to compensate local artifacts. As threshold for the segmentation, we compute the average of the summarized RMS values per window. If the sum of RMS vector elements of one window is greater than the threshold, we denote it as active. We then fill gaps in the sequence by setting every window to active whose direct predecessor and successor is active. Finally, the longest contiguous sequence of active windows is selected as the gesture segment.

Feature Extraction

The found segment is partitioned in N windows of equal length. We compute the RMS as feature for each channel on all windows and normalize the mean RMS over all channels. This results in a 168 dimensional feature vector per window. We concatenate the obtained RMS vectors resulting in a $168 \cdot N$ dimensional feature vector. Since the RMS is length normalized, it is irrelevant if the found activity segments do not have the same length.

Naive Bayes classifier

We use a naive Bayes classifier for classification. The feature distributions are modeled by kernel density estimation with Gaussian kernel function. The independence assumption of the naive Bayes classifier is clearly violated in our case since multiple electrodes span across the same muscle and different muscles do not act independently. However, Naive Bayes classifiers are known to work well under such conditions [23].

Estimation of Electrode Displacement

We introduce two methods to estimate the displacement of the electrodes between two sessions based on parts of the sessions data. Both methods are based on the idea to identify characteristic regions within two sessions and determine the shift between these regions. The first method tries to identify an area of low muscle activity over the ulna, the second tries to identify two regions of high muscle activity over the main flexor and extensor muscles. For increased robustness the coordinates of the regions are averaged over all gesture repetitions of one session.

Once the shift between two sessions is determined, data will be shifted accordingly. Since shifts usually are not aligned with the 10 mm inter-electrode distance, the shifted data is interpolated by bicubic interpolation. At the border of the

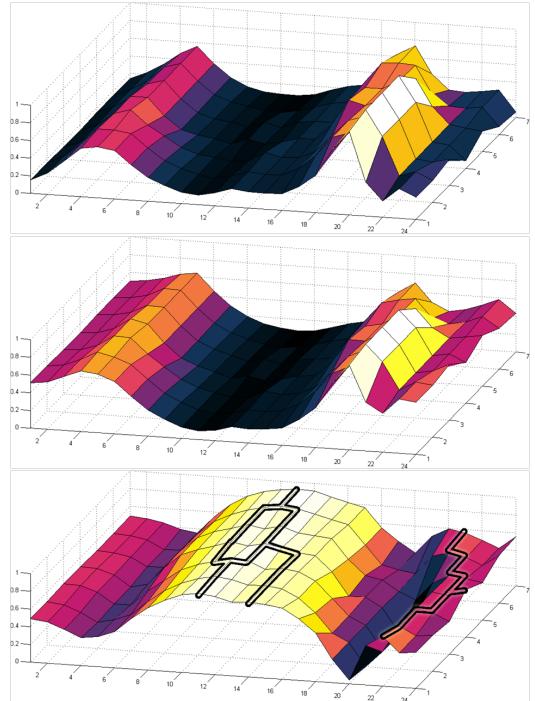


Figure 5. Example steps of identifying the ulna: 1) preprocessed normalized data, 2) data after application of the penalty function, low and high x values receive penalty, 3) found watersheds on the inverted data.

array, there will be a region of unknown data after shifting occurs. This region is filled with the marginal values.

Estimation of ulna position

Directly over the ulna is an area with no muscles and thus very low activity in the EMG. We try to identify the x position of the ulna and shift the array accordingly (only in x position). Since the electrode array is wrapped around the arm, we assume the ulna to be adjusted parallel with the y-axis. Therefore, estimation of the x position is sufficient. It should be noted that this method technically just identifies a region of low activity and we do not verify or claim that this is the actual position of the ulna.

The approach includes the following steps, of which 1-3 are illustrated in figure 5: 1) preprocessing and data normalization, 2) application of a penalty function to favor the region in the middle of the array's x range, 3) finding possible paths by applying the Watershed algorithm [20], 4) choose path with lowest cost using Dijkstra's algorithm [5].

We perform the same preprocessing as for classification. Before application of the penalty function, the data is scaled linearly to the interval $[0, 1]$. We base the penalty function on the density function $\varphi(x)$ of the normal distribution $N(\mu, \sigma^2)$ with $\mu = 12.5$ and $\sigma = 10$. The penalty function p is then computed as:

$$p(x) = 1 - \frac{\varphi(x)}{\max(\varphi(x))}, x \in \mathbb{N}$$

It has the properties $p(1) = p(24) \approx 0.5$ and $p(12) = p(13) = 0$ resulting in no penalty in the middle of the array

| subject | A | B | C | D | E |
|---------------------|------|----|----|----|----|
| circumference in cm | 25.5 | 26 | 26 | 30 | 27 |
| length in cm | 26 | 23 | 27 | 30 | 27 |

Table 1. Anatomical properties of the subject's forearms.

and a penalty of 0.5 at the borders. After applying the penalty function data is scaled linearly to [0, 1] again and afterwards inverted to comply with the preconditions of the Watershed algorithm. The Watershed algorithm [20] interprets the data as a topographic relief and identifies watersheds (see figure 5, step 3). Since the algorithm identifies all watersheds, we need to find the one with lowest overall EMG activity connecting the distal with the proximal edge of the array. To find the watershed with lowest activity we use the Dijkstra path finding algorithm to find the shortest path in a weighted graph (see [5]). The arrays electrodes are interpreted as nodes and an edge between two nodes exists, if they are directly adjacent (not diagonally). The weight of an edge is the intensity value associated with the node it points to. An additional row of nodes and edges with cost zero is inserted at the top (distal edge) and bottom (proximal edge) of the array and the start and end nodes are chosen as the upper left and lower left corner of the resulting graph. This enforces equal costs for all entry or exit positions at the distal or proximal edges of the array on the x-axis. The mean x value of all nodes in the found path are taken as the position of the ulna, which allows also rational numbers.

Estimation of center of main muscle activity

We estimate the center and shape of the two main areas of muscle activity over the flexors and extensors by fitting a Gaussian Mixture Model (GMM) with two components to the data. That means, each activity area is modeled by one normal distribution. The initial mean values μ_1, μ_2 were set to both ends of the array, $\mu_1 = (1, 4)$ and $\mu_2 = (24, 4)$ to support convergence towards a left and right component of the GMM. Model fitting is done by Maximum-Likelihood estimation with the Expectation-Maximization algorithm. The mean values of the distributions are used to estimate the electrode shift. We compute the average mean values for the flexor and extensor models, compute the two shift values and take the mean as the final resulting shift.

EXPERIMENT

We performed an experiment with five subjects. All were in the age of 20 to 23, one of them female and four male. Every subject contributed five recording sessions on different days. Since every gesture was recorded 10 times (excluding idle gestures which were recorded 30 times) per session, a number of 6500 gestures plus 750 idle gestures were recorded, resulting in 7250 in total.

Experimental Procedure

The array was attached to the forearm at 2 cm to 4 cm distance to the bend of the elbow. The length of the array was less than the circumference of the arm of the subjects. The forearm circumferences ranged from 26 cm to 30 cm (table 1). Therefore it was not possible to completely wrap the array around the forearm. The array was attached in a way that the resulting

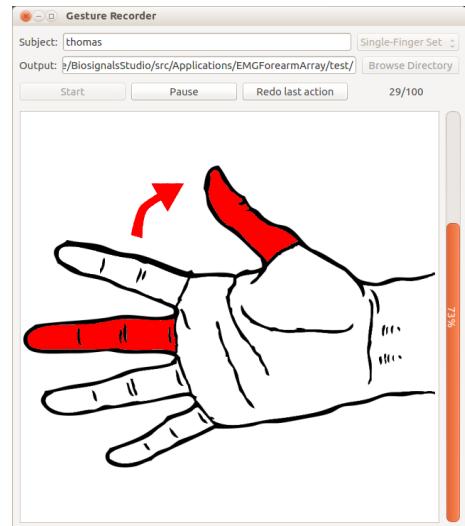


Figure 7. Screen shot of the program guiding the subject through a recording session.

gap was at the ventral side of the arm where no relevant finger muscles are located. Figure 1 illustrates the typical array position as red thick line around the cross-section of the arm. The different sessions per subject were recorded on different days. As a result, the array was attached at slightly different positions each time. We did this on purpose in order to get a realistic dataset. Invariance against positioning of the electrodes will be of major importance for every possible future wearable EMG system. The dataset allows for examining and evaluating methods to deal with this problem.

The subjects were guided through the recording session by the guiding software. Figure 7 shows a screen shot of the program. The graphical representations of the gestures from figure 4 were shown to the subject. The subject was asked to perform the presented gesture within a time interval of 3 s. A vertical progress bar at the right indicated the remaining time to perform the gesture. In case of a gesture performance error, it was possible to redo the gesture multiple times. Typical errors were performing the wrong gesture, performing the shown gesture in an awkward way or not performing a gesture in the 3 s time frame. Usually subjects realized performance errors by themselves. Pausing of the procedure was also possible and subjects could pause at their will. The subject was also informed how many gestures he or she already had performed and how many are left (the two numbers in the upper right corner). The three different sets of gestures were recorded in sequence starting with the tap-set, then the bend-set and finally the multi-finger-set with short pauses in between. The main reason for not randomizing the order of the sets was that the multi-finger-set contains gestures with high activity of many muscles. This leads to a stronger skin deformation than the other sets. As these deformations sometimes lead to disconnection of one or more electrodes and robust reattachment is hard once it was disconnected, we chose to fix the order of the sets. Additionally the sets have growing complexity of gestures, so the subject could start with easy gestures and gradually get to more complex ones. Subjects

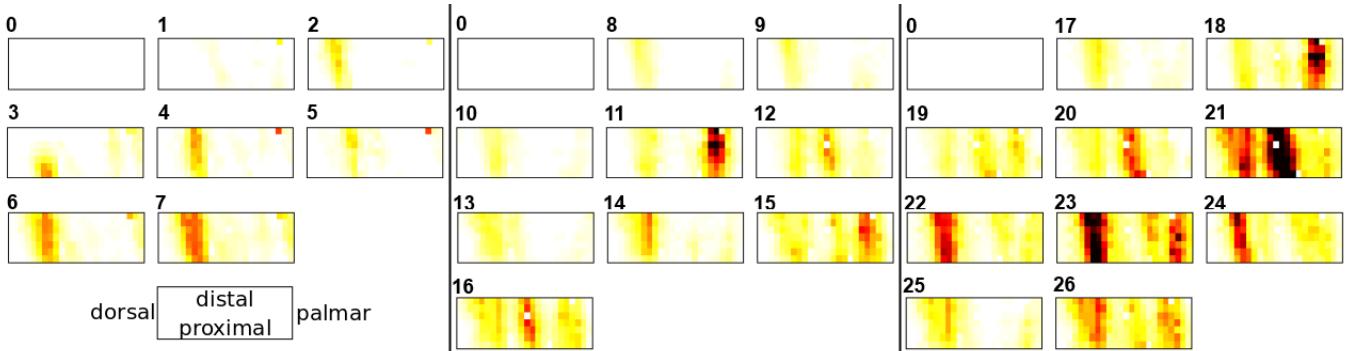


Figure 6. Average muscle activity per gesture for one selected session. The position of the graphs resembles the order in Figure 4. The color map encodes the average RMS value over the ten repetitions per gesture after segmentation and baseline normalization. Each pixel represents on channel. Single isolated channels with very high or low RMS values are defective channels which were not compensated by filtering and baseline normalization.

had to perform every gesture ten times, except the idle gesture, which is present in all three sets and therefore had to be performed 30 times in one session. Within the gesture sets, the order of gestures was randomized with a fixed seed value, resulting in the same order for every session and every subject. The complete *csl-hdemg* corpus is publicly available².

The synchronization works with timestamps on a software basis. The recording software and the guiding software run on the same machine and write timestamps on certain events (receive new data, present new stimulus). Afterwards, the recorded data is assigned to the corresponding gesture by comparing the timestamps. Due to buffering and latency in program execution, there is a small imprecision in this assignment. However the delay between stimulus presentation and start of motion by the user is much longer and therefore, the software synchronization is adequate.

Figure 6 shows averaged RMS values for each gesture of one selected recording session. Each of the 27 graphs corresponds with one gesture. The layout is the same as in figure 4. Each pixel represents the averaged RMS value over the ten repetitions of one gesture for one channel. The graphs exactly represent the electrode array layout i.e., every pixel represents the bipolar EMG between two consecutive electrodes and each graph can be read as activity map where white means no activity and black means high activity. The extensor activities can be seen in the left side of the graphs, whereas the flexor activities can be seen in the right side.

RESULTS AND ANALYSIS

Within-session classification

We report the results of within-session classification on the full gesture set of 27 gestures. Since the *idle* gesture is part of all 3 gesture sets, it was recorded 30 times per session. We selected 10 samples (numbers 2,4,7,8,11,13,19,25,26,30) out of the 30 repetitions to end up with a balanced dataset of 10 repetitions for each of the 27 gestures. We perform a leave-one-out cross-validation (LOOCV) on each session and report the average recognition performance. We did this for the case of one window ($N = 1$) and three windows ($N = 3$) per gesture segment. Table 2 shows the corresponding accuracies

²<http://csl.anthropomatik.kit.edu/cslhdemg.php>

| Sub | S1 | S2 | S3 | S4 | S5 | Avg | Std |
|-----|------|------|------|------|------|------|-----|
| A | 83.7 | 90.0 | 95.9 | 94.4 | 84.4 | 89.7 | 5.6 |
| B | 89.6 | 94.4 | 90.0 | 88.5 | 88.2 | 90.1 | 2.5 |
| C | 87.4 | 91.5 | 90.4 | 88.5 | 83.3 | 88.2 | 3.2 |
| D | 90.4 | 95.6 | 82.6 | 90.4 | 88.2 | 89.4 | 4.7 |
| E | 93.3 | 88.9 | 94.1 | 92.6 | 88.2 | 91.4 | 2.7 |
| | | | | | | 89.8 | 3.7 |

Table 2. Accuracies of the session-wise cross-validations with $N = 1$. Each row corresponds to one subject and the columns denoted by S1 to S5 correspond with session 1 to session 5.

| Sub | S1 | S2 | S3 | S4 | S5 | Avg | Std |
|-----|------|------|------|------|------|------|-----|
| A | 85.2 | 90.7 | 95.2 | 94.8 | 90.7 | 91.3 | 4.0 |
| B | 83.7 | 92.2 | 94.1 | 88.9 | 90.0 | 89.8 | 3.9 |
| C | 88.9 | 93.3 | 92.6 | 92.2 | 88.9 | 91.2 | 2.1 |
| D | 87.8 | 92.2 | 87.0 | 85.4 | 84.8 | 87.4 | 2.9 |
| E | 91.5 | 89.6 | 96.3 | 94.1 | 90.4 | 92.4 | 2.8 |
| | | | | | | 90.4 | 3.2 |

Table 3. Accuracies of the session-wise cross-validations with $N = 3$. Each row corresponds to one subject and the columns denoted by S1 to S5 correspond with session 1 to session 5.

of each session together with the mean and standard deviation for each subject and over the complete corpus for $N = 1$. Table 3 shows the results for $N = 3$. Overall, an accuracy of approximately 90% is reached, independent if only one window or three windows were used. We included the $N = 3$ case since this gave the best results in the cross-session evaluation reported below. The relatively low overall standard deviation indicates that the results are stable across subjects and sessions.

Figure 8 shows the accumulated confusion matrix over all subjects and sessions. It can be seen that the *idle* gesture is one of the most confused gestures. This is due to frequent confusion with gestures 1, 2 and 8. Gestures 1 and 8 are tapping and bending with the thumb, for which the flexors and extensors are not located under the area we recorded. A similar reason applies for gesture 2, tapping of index finger. Here the extension is performed by two separate muscles, of which one, the *Musculus extensor indicis*, is also not located under the area that was recorded. Gestures 22 and 23 are also often confused. They only differ in the amount of force

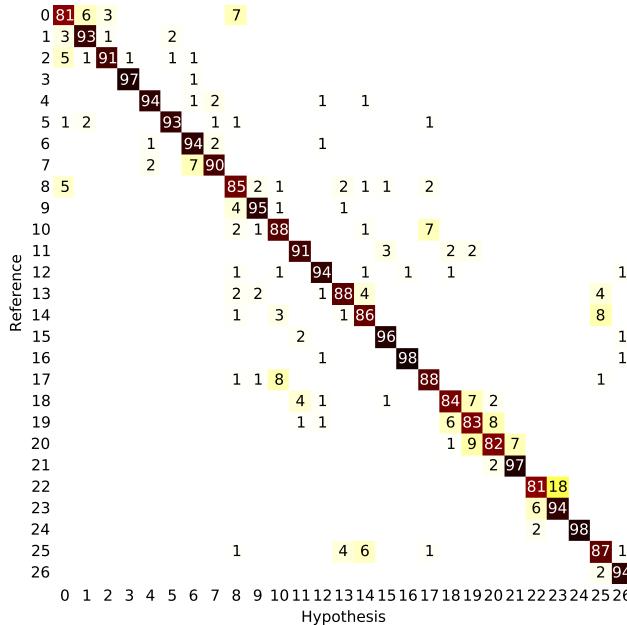


Figure 8. Accumulated confusion matrix for the within-session evaluation over all subjects and sessions. The numbers give the percentage of how often the pair of reference and hypothesis appeared in the classification results. Numbers were rounded and zeros were omitted.

applied and not in the movement itself. The same applies for gestures 19 and 20, which are also frequently confused. Further inspection of the confusion matrix shows, that most of the more frequent confusions relate to similar movements like e.g., gestures 13, 14 and 25 as well as gestures 18 and 19 and gestures 10 and 17. However, it should be noted that all gestures were clearly classified correctly in the majority of cases.

To analyze the impact of the number of electrodes on the recognition performance we evaluated the accuracy depending on the number of used electrodes. The actual electrodes that we used were chosen by maximizing mutual information with the labels on a per session basis. As feature selection method we compute the mutual information of each of the channels with the label vector. Mutual information is computed using kernel density estimation. A detailed description of this algorithm can be found in [1]. This results in different feature sets for each session. However since our aim here is to assess the relation between the number of electrodes and the accuracy and not to find the best overall reduced feature set, this is legitimate. Figure 9 shows the average accuracy for the number of used electrodes. Accuracy is improving when adding new electrodes up to about the number of 100, from where improvements get very small. The results indicate that classification accuracy benefits from a higher number of electrodes than usually used in state-of-the-art EMG based gesture recognition systems.

Shift compensation

We perform a leave-one-out cross-validation on the sessions of each subject to assess the cross-session classification accuracy. This means four sessions were taken as training set and

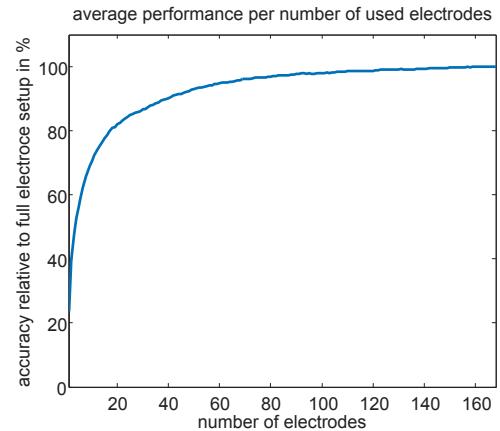


Figure 9. Average classification accuracy by number of used electrodes relative to full setup of 168 channels. Electrodes were selected by highest mutual information with the labels on a per session basis.

| N | Baseline | | GMM | | Ulna | |
|-----|----------|------|------|------|------|------|
| | 1 | 3 | 1 | 3 | 1 | 3 |
| A | 43.5 | 60.6 | 73.9 | 83.9 | 69.9 | 82.9 |
| B | 58.1 | 73.0 | 65.4 | 77.2 | 66.2 | 76.4 |
| C | 55.4 | 62.3 | 58.3 | 71.0 | 56.8 | 69.1 |
| D | 40.5 | 53.3 | 53.9 | 66.5 | 59.0 | 71.3 |
| E | 27.9 | 45.3 | 67.4 | 78.4 | 63.5 | 76.4 |
| Avg | 45.1 | 58.9 | 63.8 | 75.4 | 63.1 | 75.2 |

Table 4. Accuracies of the cross-session evaluation. Each row corresponds to one subject and the columns correspond to the different shift compensation methods including the baseline with no shift compensation and the number of windows used (parameter N).

one session as test set. Calibration was always done against the first session i.e., all data was shifted to the position of the first session before cross validation. Since the calibration gestures were also classified, the reported accuracy might be biased in favor of a higher accuracy compared to an independent set of calibration gestures from the same session. Since this effects only the calibration gesture, an upper bound of 10 biased classifications can be given. We believe the effect on overall results is negligible but want to point out that there might be a small bias.

As baseline system we report the results when performing no shift compensation at all. This resembles the use case of simply putting on a sensor sleeve and using the system without any calibration gestures. We compare the baseline to the two methods we introduced in the paper. Table 4 lists the results. For the GMM based system, we chose gesture 11 and for the ulna method gesture 23 for calibration. Results for all possible calibration gestures are given in the next paragraph. We set the number of windows to $N = 3$ since this gave best results and additionally report the results for $N = 1$ as a reference. Both shift compensation methods reach mean accuracies of around 75% versus 60% of the baseline system. Obviously the shift estimation methods work equally well and lead to a considerable improvement of accuracy although the within-session accuracy cannot be reached. Since all sessions have been taken on different days, it can be as-

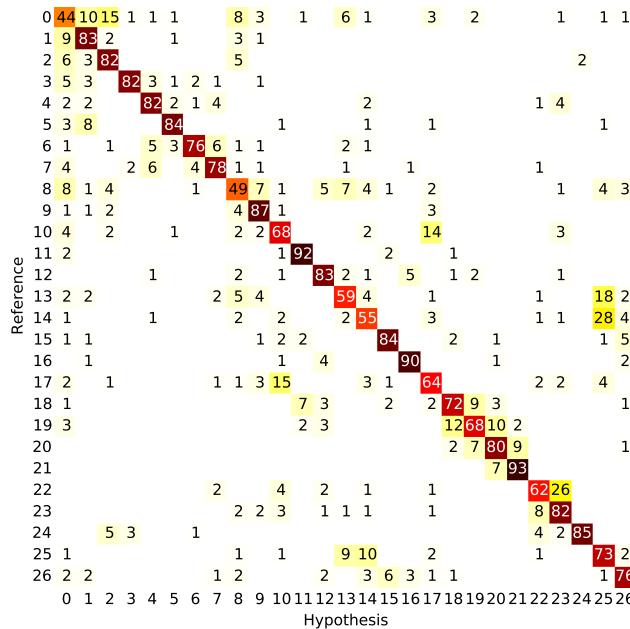


Figure 10. Accumulated confusion matrix for the cross-session evaluation over all subjects and sessions. The numbers give the percentage of how often the pair of reference and hypothesis appeared in the classification results. Numbers were rounded and zeros were omitted.

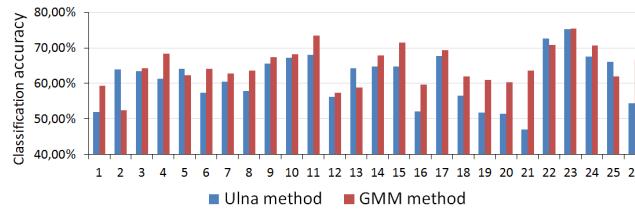


Figure 11. Classification accuracy for different calibration gestures.

sumed that additional factors degrade classification accuracy. These might include differences in skin conductance due to sweat, quality of electrode adhesion and gesture performance differences. Therefore, the within-session classification accuracy can serve as a gold standard though it seems unlikely to come very close to it. The accumulated confusion matrix for the ulna method is shown in figure 10. Pairs of gestures that were already often confused in the within-session evaluation get even more often confused for the cross-session evaluation. Additionally we observe an overall increase of smaller amounts of confusion for a variety of gestures.

Calibration gesture

We investigated which gestures are best suited to estimate the shift of the electrode array. We took each of the 26 non idle gestures as calibration gestures and determined the classification accuracy. Figure 11 shows the results for both methods of shift estimation. The ulna method works best for gestures 22 and 23 working best. Both gestures involve stretching of all fingers once with low force (gesture 22) and once with high force (gesture 23). Both gestures seem to lead to a distinct zone of inactivity along the y-axis. We didn't validate if the found path is actually consistent with the anatomical

| Gestures | WS | CS |
|---|------|------|
| 11,17,21,23,24 | 98.8 | 94.6 |
| 1,2,3,11,17,18,21,23,24,25 | 97.0 | 92.3 |
| 1,2,3,7,9,10,11,13,14,17,18,21,23,24,25 | 94.2 | 83.9 |

Table 5. Results for the restricted gesture sets given as mean accuracy for the within-session (WS) and cross-session (CS) case.

position of the ulna. For the GMM method, gestures 11, 14, 23, and 24 lead to highest accuracies with 11 performing best. Gestures 11 and 14 both consist of bending of the middle finger, once without any other finger involved (gesture 11) and once against the thumb (gesture 14). Both gestures lead to distinguished areas of activity that can be well modeled by two normal distributions.

Restricted gesture sets

We evaluated the classification accuracy on subsets of the complete gesture set to assess performance in more practical scenarios. Using the full set in a real HCI application is not feasible since a) the burden for the user to learn so many gestures is too high and b) we now know that some gestures are hard to recognize. We therefore chose subsets of size 5, 10, and 15 gestures, each containing the two calibration gestures and gestures that we chose based on the accuracy and possible suitability in a HCI context. This was done on a best guess base i.e., we did not evaluate the usefulness of our chosen subset in a real application. Table 5 lists the chosen gesture sets and the accuracies achieved. Restricting the gesture set increases the accuracy for both the within- and between-session case above 90% in all cases except one.

CONCLUSION AND FUTURE WORK

In this paper we investigate the use of EMG electrode-arrays for gesture recognition. We recorded a set of 27 gestures from five subjects in five different sessions. We introduce a baseline recognition system using a Naive Bayes Classifier that reaches an accuracy of 90% on the 27 classes, showing the feasibility of our approach to classify a large number of partly subtle finger gestures. We show that in our setup, best performance is reached for a number of 100 and above used electrodes. This shows that increasing the number of electrodes leads to performance gains. We also investigate the cross-session performance. Since sessions of one subject were recorded on different days with slightly varying electrode positions, this is more challenging. The accuracy for the cross-session case drops to 59%. We introduce two methods for calibration by estimating the shift in electrode positions across sessions by making use of anatomical characteristics. With both methods we achieve an accuracy of approximately 75%.

Currently the equipment to make such high-density recordings is still very bulky and not made for mobile use. However such devices might become ubiquitous in a few years and we believe that our results motivate that this is a direction which is worth to pursue. The dataset itself will allow for further analysis of the spatio-temporal distribution of the EMG activity during the recorded finger movements. Such high-density recordings enable for example the use of source-separation

methods which we plan to investigate in the future. It might also serve as a common ground for future research and performance comparison of different approaches on EMG based wearable interfaces.

ACKNOWLEDGMENTS

This research was partly funded by Google through a Google Faculty Research Award.

REFERENCES

1. Ang, K. K., Chin, Z. Y., Zhang, H., and Guan, C. Mutual information-based selection of optimal spatial-temporal patterns for single-trial eeg-based bcis. *Pattern Recognition* 45, 6 (2012), 2137–2144.
2. Assad, C., Wolf, M., Theodoridis, T., Glette, K., and Stoica, A. Biosleeve: a natural emg-based interface for hri. In *Proc. HRI 2013* (2013), 69–70.
3. Atzori, M., Gijsberts, A., Kuzborskij, I., Heynen, S., Mittaz Hager, A., Deriaz, O., Castellini, C., Muller, H., and Caputo, B. Characterization of a benchmark database for myoelectric movement classification. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on PP*, 99 (2014), 1–1.
4. Castellini, C., Fiorilla, A. E., and Sandini, G. Multi-subject/daily-life activity EMG-based control of mechanical hands. *Journal of neuroengineering and rehabilitation* 6 (Jan. 2009), 41.
5. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., et al. *Introduction to algorithms*, vol. 2. MIT press Cambridge, 2001.
6. De Luca, C. J. The use of surface electromyography in biomechanics. *Journal of applied biomechanics* 13 (1997), 135–163.
7. Heger, D., Putze, F., Amma, C., Wand, M., Plotkin, I., Wielatt, T., and Schultz, T. Biosignalsstudio: a flexible framework for biosignal capturing and processing. In *Proc. KI 2010* (2010), 33–39.
8. Hermens, H. J., Freriks, B., Merletti, R., Stegeman, D., Blok, J., Rau, G., Disselhorst-Klug, C., and Hägg, G. *European recommendations for surface electromyography*. Roessingh Research and Development The Netherlands, 1999.
9. Kim, D., Hilliges, O., Izadi, S., Butler, A. D., Chen, J., Oikonomidis, I., and Olivier, P. Digits: freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *Proc. UIST 2012* (2012), 167–176.
10. Kim, J., Mastnik, S., and André, E. Emg-based hand gesture recognition for realtime biosignal interfacing. In *Proc. IUI 2008* (2008), 30–39.
11. Merletti, R., Holobar, A., and Farina, D. Analysis of motor units with high-density surface electromyography. *Journal of electromyography and kinesiology* 18, 6 (2008), 879–890.
12. Mistry, P., Maes, P., and Chang, L. Wuw-wear ur world: a wearable gestural interface. In *CHI EA 2009*, ACM (2009), 4111–4116.
13. Muceli, S., and Farina, D. Simultaneous and proportional estimation of hand kinematics from emg during mirrored movements at multiple degrees-of-freedom. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 20, 3 (2012), 371–378.
14. Rekimoto, J. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Proc. ISWC 2001* (2001), 21–27.
15. Rojas-Martínez, M., Mañanas, M. A., Alonso, J. F., et al. High-density surface emg maps from upper-arm and forearm muscles. *J. Neuroeng. Rehabil.* 9, 85.10 (2012), 1186.
16. Samadani, A.-A., and Kulić, D. Hand gesture recognition based on surface electromyography. In *Proc. EMBC 2014*, IEEE (2014), 4196–4199.
17. Saponas, T. S., Tan, D. S., Morris, D., and Balakrishnan, R. Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In *Proc. SIGCHI 2008* (2008), 515–524.
18. Saponas, T. S., Tan, D. S., Morris, D., Turner, J., and Landay, J. A. Making muscle-computer interfaces more practical. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2010), 851–854.
19. Starner, T., Auxier, J., Ashbrook, D., and Gandy, M. The gesture pendant: A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *Wearable computers, the fourth international symposium on*, IEEE (2000), 87–94.
20. Vincent, L., and Soille, P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE transactions on pattern analysis and machine intelligence* 13, 6 (1991), 583–598.
21. Wand, M., Schulte, C., Janke, M., and Schultz, T. Compensation of recording position shifts for a myoelectric silent speech recognizer. In *Proc. ICASSP 2014* (2014), 2094–2098.
22. Wheeler, K. R., Chang, M. H., and Knuth, K. H. Gesture-based control and emg decomposition. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 36, 4 (2006), 503–514.
23. Zhang, H. The optimality of naive bayes. *AA* 1, 2 (2004), 3.