A MATLAB Toolbox for Upper Limb Movement Classification

Andreas Attenberger and Klaus Buchenrieder

Institut für Technische Informatik
Universität der Bundeswehr München
Neubiberg, Germany
{andreas.attenberger,klaus.buchenrieder}@unibw.de

Abstract. Modern myoeletric prostheses necessitate more powerful control algorithms to derive hand movement information from sensor data. Common approaches utilize classifiers for recognizing motion or grip patterns from input data like myoeletric signals (MES). The selection of features for the classification process and the classifiers themselves impact the detection accuracy of the control schemes. In this contribution, we present a MATLAB™ movement classification toolbox for sensor data recorded during hand movements. By covering different feature calculation- and classification-algorithms, the toolbox supports the modeling of select prostheses control schemes. In addition to MES, novel near-infrared (NIR) sensor input is equally supported by the toolbox. A modular development approach allows the integration of new features, classifiers as well as the extension to other types of sensor data.

Keywords: Feature Extraction, Hand Prostheses, MATLAB Toolbox, Movement Classification, Myoelectric Signals, Near Infrared Sensor.

1 Introduction

While basic myoelectric prostheses merely rely on the detection of muscle activity and threshold-based gripper actuation [1], new models of prosthetic hands like the Michelangelo™ Hand from Otto Bock[1] or the bebionic3™ hand manufactured by RSL Steeper[2] offer additional grip patterns and necessitate more complex control algorithms. In the classification model brought forward by Englehart, Hudgins, Parker and Stevenson [2], features are extracted from the source signals acquired through sensors measuring muscle activity. The resulting data is employed for the training of a selected classifier. Once the classification model has been created, it can be used to detect known hand movement classes from new sensor data. Different feature sets as well as the choice of the classification method can impact the correct detection of movements significantly [3] [4]. In order to choose the optimal combination of features and a classification

[1] http://www.leben-mit-michelangelo.de/home/
[2] http://bebionic.com/the_hand

R. Moreno-Díaz et al. (Eds.): EUROCAST 2013, Part II, LNCS 8112, pp. 191–198, 2013.

method, a systematic comparison of feature-extraction- and classifier-algorithms is necessary. While existing toolboxes like gaitCAD only focus on general time series processing [5], the toolbox brought forward in this contribution has specifically been developed for hand movement classification.

2 Features and Classification

Before training a classifier, features have to be calculated in order to extract salient characteristics from the signal. The MATLAB™ toolbox offers a range of feature extraction algorithms, including the commonly utilized RMS values. The following list gives an overview of all currently implemented MES features as well as the novel NIR signal feature with a short description of their properties.

Root Mean Square (RMS): The root mean square value denotes the average signal strength [6].

Zero Crossings (ZC): This feature is extracted by counting the number of times the signal value will pass zero, thus giving a basic estimation of the incitation frequency [7]. A threshold is commonly applied to reduce noise.

Mean Absolute Value (MAV): Next to the RMS feature, the mean absolute value (MAV) can also be used to represent the intensity of a muscle contraction [8].

Mean Frequency (MF): This feature calculates the mean frequency contained in the corresponding sample window [9].

Slope Sign Changes (SSC): The slope sign changes value measures frequency characteristics of the signal by counting how many times the signal slope changes. Similiar to the ZC feature, a threshold is applied for noise attenuation [7].

Variance (VAR): The quadatric variance or standard deviation of an EMG signal, assuming a mean of zero, equally gives an estimate of the signal's power [6] [10].

Willison Amplitude (WAMP): The Willison Amplitude Feature is another way to gauge the strength of a muscle exertion [7]. To calculate the feature, the number of times the difference between two consecutive sample values exceeds a preset threshold in the selected time window, is counted.

Waveform length (WFL): The Waveform Length Feature (WFL) extracts the length of the waveform in a sample window, thus giving information about the amplitude and frequency variation of a myoeletric signal [11].

Near Infrared Signal (NIRS): This feature extraction method calculates the actual near infrared signal derived from the original, pulsed sensor data [4] [12].

The feature data serves as input both training and application of the classifier algorithms. The following classifiers have been implemented in the toolbox:

Decision Trees: Decisions tree classifiers are simple classifiers that do not mandate complicated parameter selection. During classification the nodes of a tree are traversed, hence yielding a sequence of decisions resulting in class attribution [13].

Naive Bayes Classification: The naive Bayes classifier works under the assumption that all variables are statistically independent. The probability distribution of features is calculated according to a training set [13] [14].

k-Nearest Neighbor Classification (kNN): The k-nearest neighbor classifier assigns class labels according to the class of the nearest neighbor sample points from the training set. The number of neighbors k and the employed distance metric can be selected [15] [13].

Linear Discriminant Analysis (LDA): For Linear Discriminant Analysis, a normal multivariate distribution of the feature data is assumed. Classes are separated by linear functions. [16] [14].

Support Vector Machines (SVM): Another popular approach to classification are Support Vector Machines. Here, classes are separated by hyperplanes. Different kernel functions can be applied for mapping problems into higher dimensional vector spaces [13].

3 Toolbox

The toolbox can process source data in MATLAB™ files with EMG and NIR data contained in variables named accordingly. Each column of equal length holds the data of one sensor. Tabs in the GUI allow to select parameters for feature extraction, classifier training and validation as shown in Fig. 1. Parameter, filename and path settings for all individual tabs can be saved to disk for later reference. Apart from the SVM algorithm, all other classifier implementations have been taken from the MATLAB™ Statistics Toolbox [3]. LIBSVM was chosen over the integrated SVM as it supports multiclass classification [17]. All application functionality has been encapsulated in modules [18].

3.1 Feature Selection

The feature calculation offers options to calculate selected features from a preset number of source hand movement recordings. Filenames and paths for in- and output files can be chosen by the user. General parameters like window size and increment can be set for all implemented features. Optional DC correction for EMG data is also available. Further options like window function or thresholds for the individual feature algorithms are highlighted when required. Any number of sensors and sensor technology type combination supported by the toolbox is possible. The extracted values can then be utilized for classifier training.

3.2 Classifier Training

A user-selected percentage of the movement recordings is input to the classification algorithm yielding a classification model. During this step, a noise threshold for feature values can be selected for each sensor. The path containing the

[3] `http://www.mathworks.de/products/statistics/`

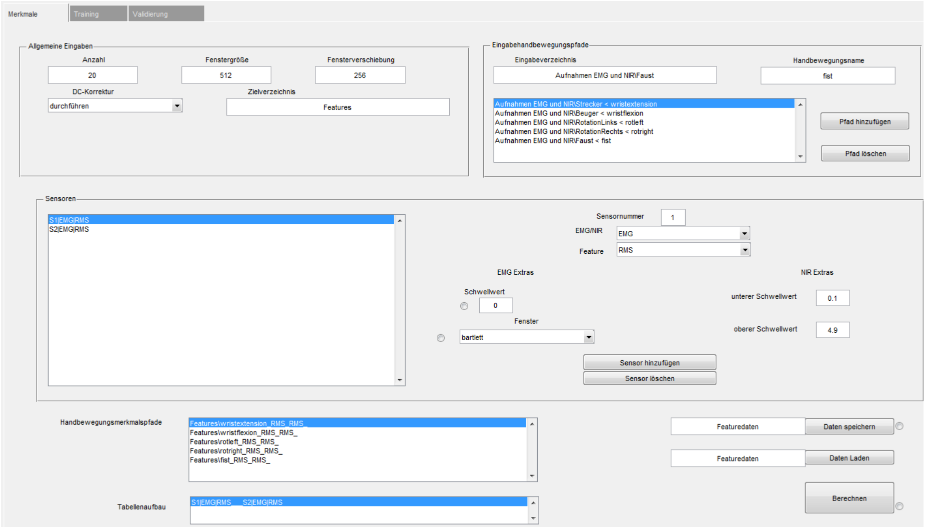


Fig. 1. The main window of the MATLAB™ toolbox with tabs for feature extraction, classifier training and validation.

calculated feature values is set as the default source input for this processing step. The toolbox offers the selection of additional parameters depending on the choice of a classifier. Required settings, which include kernel functions for SVMs and distance metrics for nearest neighbor classification, are highlighted. Optional plotting of training data with class boundaries is available to select feature vectors.

3.3 Classifier Evaluation and Selection

Classification accuracy is verified with the features of the remaining hand movement recordings. The toolbox automatically generates a list of classification results including overall accuracy, recognition rate for individual movement classes and attribution percentage for the feature values of each recording. Noise thresholds can be employed to filter all recordings prior to the classification process. It is possible to adjust the required recognition rate for simulating classifier robustness.

4 Example Application

The toolbox introduced here, allows for a systematic comparison of a wealth of parameters and selections involved in creating a classification-based prosthesis control scheme. In this section, we give examples for comparison which include examples for the choice of sensor technology or classification scheme. The source data, that was employed for generating the results, was recorded with two combined NIR-EMG-sensors. These were placed on opposing sides of the forearm over extensor and flexor muscles of a 27-year old male proband. Further information about the acquisition setup is given in [4].

4.1 Comparison of Sensor Technologies

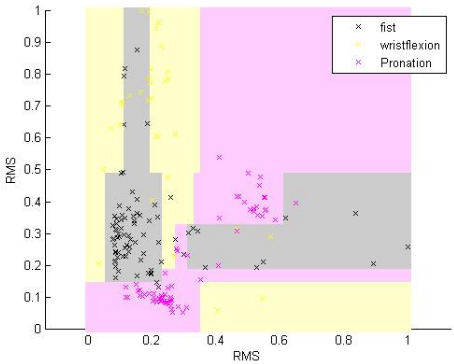
It has previously been shown, that classification accuracy can be significantly improved by employing novel sensor technology like NIR sensors. Table 1 shows the results of a kNN classifier with the cityblock distance metric. It validates the improvement of classifier outcomes through the inclusion of NIR sensor technology into muscle activity detection. Additionally, the data show that the best choice of sensor technology can also differ with placement. For calculating the features, a window size of 512 samples and an increment of 256 samples was chosen including DC correction before calculation. Noise thresholds of 0.2 and 0.1 were applied to the normalized RMS feature values. The classifier was trained with 15 recordings of each of the following movement patterns: fist, pronation, supination, wrist-flexion and wrist-extension. The remaining 5 recordings were utilized for the evaluation.

Table 1. Impact of selected sensor technology on classification results

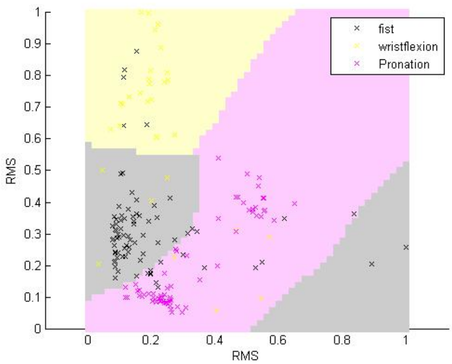
Rating	Sensor Type 1/2 (Feature)	Classification Accuracy
1	NIR (NIRS)/EMG (RMS)	96,00 %
2	EMG (RMS)/EMG (RMS)	88,00 %
3	EMG (RMS)/NIR (NIRS)	84,00 %
4	NIR (NIRS)/NIR (NIRS)	80,00 %

Table 2. Rating of classifier accuracy for 3 movements with the same feature set

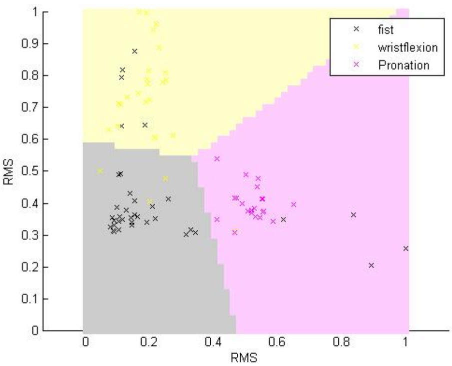
Rating	Classifier	Results
1	k-Nearest-Neighbor: City Block Distance	93.33 %
1	Decision Tree	93.33 %
2	k-Nearest-Neighbor: Euclidian Distance	86,66 %
2	SVM: Radial Kernel	86,66 %
3	SVM: Linear Kernel	80,00 %
3	SVM: Sigmoid Kernel	80,00 %
4	LDA	73,33 %
5	Naive Bayes Classifier	66,66 %
6	k-Nearest-Neighbor: Cosine Distance	53,33 %
7	SVM: Polynomial Kernel	40,00 %
8	k-Nearest-Neighbor: Correlation Distance	33,33 %



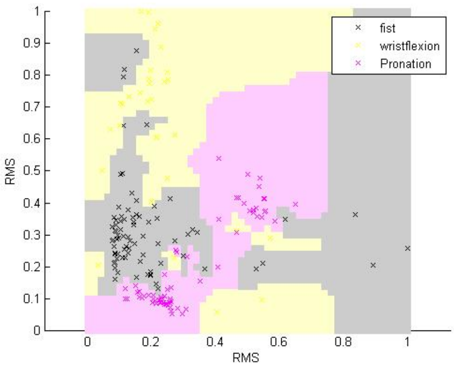
(a) Decision Tree



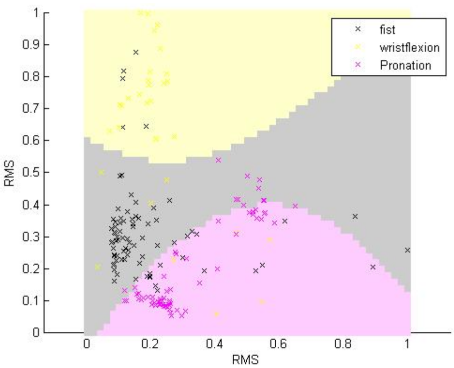
(b) SVM (Radial Kernel)



(c) LDA



(d) kNN (City Block)



(e) Naive Bayes Classifier

Fig. 2. Plots for different classifier algorithms with exemplar parameter selections

4.2 Comparison of Classifiers

With fist, pronation and wrist-extension movement patterns, the accuracy of the different classifiers implemented in the toolbox was compared with EMG data. The RMS feature for both sensors was calculated with the same window and threshold settings. Training and validation was also realized with the same number of recordings. Table 2 shows, that the kNN classifier utilizing city block distance and the decision tree algorithm yield best results. Fig. 2 gives an overview of the class boundary plots for the different types of classifiers utilized. Note, that the SVM classifier underperforms in comparison to previous work [4]. This is to the parameter selection necessary for SVMs. For reasons of processing speed, automatic parameter search had been restricted to smaller intervals for this test. Future versions of the toolbox will include additional options to take this into account.

5 Conclusion and Future Research

Our example data show the impact of sensor technology, feature selection as well as classifier algorithms on hand movement recognition accuracy. Systematic comparisons are necessary for designing classification-based control schemes mandated by modern hand prostheses. The toolbox developed at the Universität der Bundeswehr München constitutes a valid tool for supporting selection choices regarding the algorithms and parameters involved in this process. Extensibility and modularity make integration of future technologies, novel features and classifiers possible. Future research will target the inclusion of capture and editing tools for hand movement sensor data. Real-time classification with visualization of the result through the 3D hand model, developed at our department [3] [4], is planned. Furthermore, a data base consisting of hand movements by six probands comprising more than a thousand recordings is set to be analyzed with our toolbox. Future steps are the creation of an English-language interface.

Acknowledgment. The authors thank Marcus Eckert, M.Sc. for his efforts toward implementing the MATLAB™ movement classification toolbox. Furthermore, we are grateful to Michael Burg for executing the classifier comparison tests.

References

1. Muzumdar, A.: Powered Upper Limb Prostheses: Control, Implementation and Clinical Application. Springer (2004)
2. Englehart, K., Hudgins, B., Parker, P., Stevenson, M.: Classification of the Myoelectric Signal using Time-Frequency Based Representations. Medical Engineering & Physics 21(6-7), 431–438 (1999)
3. Attenberger, A., Buchenrieder, K.: Modeling and Visualization of Classification-Based Control Schemes for Upper Limb Prostheses. In: Popovic, M., Schätz, B., Voss, S. (eds.) ECBS, pp. 188–194. IEEE (2012)

4. Attenberger, A., Buchenrieder, K.: Modeling and Visualization of Classification-Based Control Schemes for Upper Limb Prostheses. Comput. Sci. Inf. Syst. 10(1), 349–367 (2013)
5. Burmeister, O., Reischl, M., Bretthauer, G., Mikut, R.: Data-Mining-Analysen mit der Matlab-Toolbox Gait-CAD (Data Mining Analyses with the Matlab Toolbox Gait-CAD). Automatisierungstechnik 56(7), 381–389 (2008)
6. Buchenrieder, K.: Dimensionality Reduction for the Control of Powered Upper Limb Prostheses. In: Proc. 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS 2007), pp. 327–333 (March 2007)
7. Hudgins, B., Parker, P., Scott, R.N.: A New Strategy for Multifunction Myoelectric Control (1993)
8. Clancy, E.A., Hogan, N.: Theoretic and Experimental Comparison of Root-Mean-Square and Mean-Absolute-Value Electromyogram Amplitude Detectors (1997)
9. Herrmann, S., Buchenrieder, K.: Advanced Control Schemes for Upper Limb Prosthesis. Advances in Medicine and Biology 15, 377–408 (2011)
10. Zecca, M., Micera, S., Carrozza, M.C., Dario, P.: Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal. In: Critical Reviews in Biomedical Engineering, pp. 459–485 (2002)
11. Englehart, K.: Signal Representation for Classification of the Transient Myoelectric Signal. PhD thesis, University of New Brunswick (October 1998)
12. Herrmann, S., Buchenrieder, K.: Fusion of Myoelectric and Near-Infrared Signals for Prostheses Control. In: Proc. 4th International Convention on Rehabilitation Engineering & Assistive Technology, iCREATe 2010, Kaki Bukit TechPark II, Singapore, Singapore Therapeutic, Assistive & Rehabilitative Technologies (START) Centre, pp. 54:1–54:4 (2010)
13. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 4th edn. Academic Press (August 2008)
14. The MathWorks, Inc.: Supervised Learning - MATLAB & Simulink - MathWorks Deutschland (May 2013), `http://www.mathworks.de/de/help/stats/supervised-learning.html`
15. Cover, T., Hart, P.: Nearest Neighbor Pattern Classification. IEEE Transactions on Information Theory 13(1), 21–27 (1967)
16. McLachlan, G.: Discriminant Analysis and Statistical Pattern Recognition. Wiley Series in Probability and Statistics. Wiley (2005)
17. Chang, C.-C., Lin, C.-J.: LIBSVM: A Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011), `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
18. Eckert, M.: Simulation und Validierung von Klassifikationsmethoden für die Steuerung von Handprothesen. Master's thesis, Universität der Bundeswehr München (2012)