35th Annual International Conference of the IEEE EMBS
Osaka, Japan, 3 - 7 July, 2013

Real-Time Implementation of a Self-Recovery EMG Pattern Recognition Interface for Artificial Arms

Xiaorong Zhang, *Student Member, IEEE*, He Huang, *Member, IEEE*, and Qing Yang, *Fellow, IEEE*

Abstract—**EMG pattern classification has been widely studied for decoding user intent for intuitive prosthesis control. However, EMG signals can be easily contaminated by noise and disturbances, which may degrade the classification performance. This study aims to design a real-time self-recovery EMG pattern classification interface to provide reliable user intent recognition for multifunctional prosthetic arm control. A novel self-recovery module consisting of multiple sensor fault detectors and a fast LDA classifier retraining strategy has been developed to immediately recover the classification performance from signal disturbances. The self-recovery EMG pattern recognition (PR) system has been implemented on an embedded system as a working prototype. Experimental evaluation has been performed on an able-bodied subject in real-time to classify three arm movements while signal disturbances were manually introduced. The results of this study may propel the clinical use of EMG PR for multifunctional prosthetic arm control.**

I. Introduction

Electromyographic signal (EMG) pattern recognition (PR) is a widely used method for classifying user intent for neural control of artificial limbs [1-3]. However, unreliability of surface EMG recordings over time is a challenge for applying the EMG pattern recognition controlled prostheses for clinical practice. Motion artifacts, environmental noises, sensor location shifts, user fatigue, and other conditions may all cause changes in the EMG characteristics and thus lead to inaccurate identification of user intent and threaten the prosthesis control reliability and user safety[4-5].

Several strategies have been developed to address this challenge in order to make artificial limb control based on EMG PR clinically viable. Sensinger et al. [5] employed adaptive pattern classifier to cope with variations in EMG signals for reliable EMG PR. Tkach et al. [6] investigated different EMG features and suggested several time-domain features that were resilience to EMG signal change caused by muscle fatigue and exerted force levels. Hargrove et al. [7] suggested a new EMG PR training procedure in order to accommodate EMG electrode shift during prosthesis use.

Our research group developed a unique, reliable EMG pattern recognition interface, consisting of sensor fault detectors and a self-recovery mechanism. The sensor fault detectors monitor the recordings from individual EMG electrodes; the self-recovery mechanism will remove the faulty EMG signals from the PR algorithm to recover the classification accuracy [8-10]. It was observed that the EMG classification performance was not significantly affected by the removal of one or two EMG signals from redundant EMG recordings [2, 8]. Our new EMG-PR interface could salvage system performance by up to 20% increased classification accuracy when one or more EMG signals were disturbed [8].

While our previous study has demonstrated the concept of our design, the algorithm development and validation were tested offline. In order to implement this concept in real-time, especially in a wearable embedded system, several challenges still exist. *First*, the recovery strategy involves retraining of the pattern classifier. Currently this procedure involves reorganization of training feature matrix, computation of parameters in the pattern classifiers, and reorganization of testing feature vectors. Whether or not the embedded system can handle this procedure quickly for each decision-making is unknown. *Secondly*, since more components are included in the EMG PR algorithm, communication among components and precise timing control is crucial. *Finally*, a compact integration of all the components in an embedded computer is required. The system needs to provide necessary interfaces for data collection, adequate computing power for real-time decision making, efficient memory management, and low power consumption. All these challenges have never been explored.

This paper presents the first real-time self-recovery EMG pattern recognition interface for artificial arms. A novel self-recovery scheme with a fast and efficient retraining algorithm based on linear discriminant analysis (LDA) has been developed. The self-recovery EMG pattern recognition system was implemented on an embedded computer system as a working prototype. The prototype was preliminarily evaluated on an able-bodied subject in real-time in classifying three arm movements while motion artifacts were manually introduced by randomly tapping the EMG electrodes. The results of this study may propel the clinical use of EMG PR for multifunctional prosthetic arm control.

II. Methods

A. System Structure

The overall structure of the self-recovery EMG pattern recognition interface is shown in Fig. 1. The system seamlessly integrates EMG pattern recognition with the self-recovery module. Multiple channels of EMG signals

This work is partly supported by NIDRR#H133G120165, NSF/CPS #0931820, NIH #RHD064968A, NSF# 1149385, NSF/CCF #1017177, and NSF/CCF #0811333.

The authors are with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA.

Corresponding author: Xiaorong Zhang (email: zxiaorong@ele.uri.edu)

978-1-4577-0216-7/13/$26.00 ©2013 IEEE

segmented by overlapped sliding analysis windows are the system inputs. In each window, four time-domain (TD) features (mean absolute value, number of zero crossings, waveform length, and number of slope sign changes [11]) of the EMG signals are extracted from each input channel and fed to the self-recovery module. The sensor fault detectors closely monitor the key features of each EMG signal to detect disturbances. Based on the detection results, the EMG features extracted from 'normal' channels are concatenated into a feature vector as the input for pattern classification. If no disturbance is detected, the feature vector is directly sent to the classifier generated from the original training data. If one or more signals are determined as 'abnormal', the fast LDA retraining process is triggered and the reduced feature vector is fed to the new classifier for pattern recognition.

B. Fast LDA-based retraining algorithm

Previously the lack of a fast and efficient retraining algorithm was the most critical challenge to the design of a real-time self-recovery EMG PR interface. If the retraining process cannot be accomplished in a short period of time, the signal disturbances may impair the classification performance and even harm the prostheses users' safety. Linear discriminant analysis (LDA) is a widely used method for EMG pattern recognition [1, 10-11]. By examining the details of the LDA algorithm, we developed a fast and memory efficient LDA retraining algorithm by making the most efficient use of existing information.

The principle of the LDA-based PR strategy is to find a linear combination of features which separates multiple classes $C_g\,(g \in [1,G])$. Here G denotes the total number of studied classes. Suppose $\bar{f}$ is the feature vector in one analysis window, μ_g is the mean vector of class C_g and every class shares a common covariance matrix $\sum$, the LDA function is defined as $d_{C_g} = \bar{f}^T \Sigma^{-1} \mu_g - \frac{1}{2}\mu_g{}^T \Sigma^{-1} \mu_g$.

During the training procedure, $\sum$ and μ_g are estimated based on the feature matrix calculated from the training data. The estimations of $\sum$ and μ_g are expressed as

$$\tilde{\Sigma} = \frac{1}{G}\sum_{g=1}^{G}\frac{1}{K_g - 1}(F_g - Mi_g)(F_g - Mi_g)^T \text{ and } \tilde{\mu}_g = \frac{1}{K_g}\sum_{k=1}^{K_g}\bar{f}_{C_g,k}$$

where K_g is the number of analysis windows in class C_g; $\bar{f}_{C_g,k}$ is the k_{th} observed feature vector in class C_g; $F_g = [\bar{f}_{C_g,1}, \bar{f}_{C_g,2}, ..., \bar{f}_{C_g,k}, ..., \bar{f}_{C_g,K_g}]$ is the feature matrix of class C_g; $Mi_g = [\tilde{\mu}_g, \tilde{\mu}_g, ..., \tilde{\mu}_g]$ is the mean matrix which has the same dimension as F_g. In a feature vector $\bar{f}_{C_g,k} = [f_1{}^T, f_2{}^T, ..., f_n{}^T, ..., f_N{}^T]^T$, N is the total number of EMG input channels and f_n denotes the four EMG features extracted from the n_{th} channel.

In the previous retraining strategy [8], after the initial training process is done, the original EMG feature matrix is

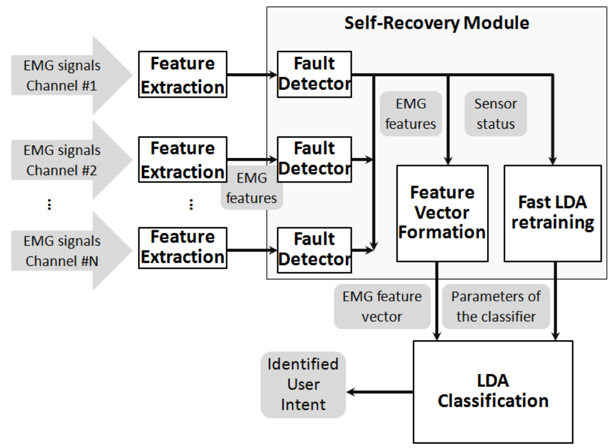

Fig. 1. System structure of the self-recovery EMG sensing interface for LDA-based pattern recognition.

stored in the memory for later use in the retraining process. During the retraining procedure, for each class, a new EMG feature matrix $F_g{}'$ is reorganized by removing the feature rows corresponding to the disturbed channels from F_g. The mean vector of each class $\tilde{\mu}_g{}'$ and the new common covariance matrix $\tilde{\Sigma}'$ are then recalculated based on $F_g{}'$. Our experimental analysis has shown that the calculation of $\tilde{\Sigma}'$ is the most computational intensive task in the retraining procedure, which accounts for more than 90% of the total processing time. This is because for each class, a large amount of analysis windows are collected as the training data. The number of columns in $F_g{}'$ may vary from several hundreds to a few thousands, which leads to intensive numerical operations in calculating $\tilde{\Sigma}'$.

Fortunately, after closely analyzing the details of the LDA training algorithm, we have found that the calculation of $\tilde{\Sigma}'$ and $\tilde{\mu}_g{}'$ can be avoided in a smart way. The trick is, instead of the large feature matrix F_g, only $\tilde{\mu}_g$ and $\tilde{\Sigma}$ are stored in the memory after the initial training process is finished. $\tilde{\Sigma}'$ and $\tilde{\mu}_g{}'$ can be easily retrieved from $\tilde{\Sigma}$ and $\tilde{\mu}_g$. Fig. 2 shows an example of the retrieving process if a single EMG channel is detected to be 'abnormal'. Assume there are totally 6 EMG channels. Each element in the mean vector is calculated by averaging one specific feature row in F_g. Therefore $\tilde{\mu}_g{}'$ can be obtained by taking off the four elements that are associated with the disturbed EMG channel from $\tilde{\mu}_g$. $\tilde{\Sigma}'$ is constructed by removing the corresponding rows and columns associated

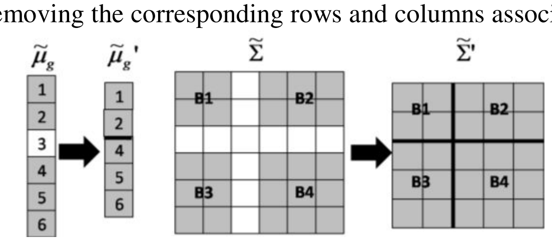

Fig. 2. An example of retrieving $\tilde{\Sigma}'$ and $\tilde{\mu}_g{}'$ from $\tilde{\Sigma}$ and $\tilde{\mu}_g$ when a single EMG channel is disturbed. The white blocks represent the elements associated with the disturbed channel.

with the disturbed channel from $\tilde{\Sigma}$ and then merging the remaining four small matrices ($B1$, $B2$, $B3$, and $B4$ in Fig. 2). If multiple EMG signals are disturbed, $\tilde{\Sigma}'$ and $\tilde{\mu}_g'$ can be obtained by doing the retrieving process repeatedly. Compared with the previous retraining algorithm which requires intensive numerical operations and a large memory space, the new strategy dramatically accelerates the retraining speed and is much more memory efficient.

C. *Sensor Falut Detection*

To detect individual EMG sensor abnormalities, various signal processing methods have been applied to sensor fault detection [8-10]. A detector based on Bayesian decision rule [8] has been proposed for accurately detecting three types of simulated distortions including EMG signal drift and saturation, additional noise in the signal, and variation of EMG magnitude. An abnormality detector using Cumulative Sum (CUSUM) algorithm [9] has been developed to closely monitor the changes of EMG features for detecting sudden changes or gradual changes in EMG signals.

In this study, the CUSUM detector is adopted in our implementation because of its computational efficiency for real-time processing, its high accuracy, and low false alarm rate in detecting motion artifacts [9-10]. Two EMG features including *mean absolute value* and *number of zero crossings* are monitored to recognize abnormal changes. Detailed algorithms of the CUSUM detector can be found in [9].

D. *Real-time Embedded System Implementation*

A preliminary prototype of the self-recovery EMG pattern recognition system was implemented on Gumstix Overo Air, an ARM Cortex-A8 OMAP3503 based computer-on-module (COM), and RoboVero, an expansion board with an ARM Cortex-M3 microcontroller and eight 12-bit analog-to-digital converters (Fig. 3). The Overo COM communicates with the RoboVero expansion board via two 70-pin connectors as shown in Fig. 3. The system implementation consists of two parts: the microcontroller on the RoboVero expansion board for data sampling and dispatching, and the Cortex-A8 processor on the Overo COM for EMG pattern recognition.

E. *Experimental Protocol*

This study was conducted with Institutional Review Board (IRB) approval at the University of Rhode Island and informed consent of subject. One male able-bodied subject was recruited. Four surface EMG electrodes (MA-420-002, Motion Lab System Inc.) were placed around the subject's right forearm. An MA-300 EMG system collected four channels of EMG signals. The analog EMG signals were digitally sampled at the rate of 1000 Hz by the Gumstix RoboVero expansion board. The sampled data were segmented into overlapped analysis windows with 160 ms length and 20 ms increment, resulting in a new decision every 20 ms. Three motion classes (Elbow Flexion, Elbow Extension, and No Movement) were investigated in this experiment. The experiment consisted of two sessions: training session, and testing session.

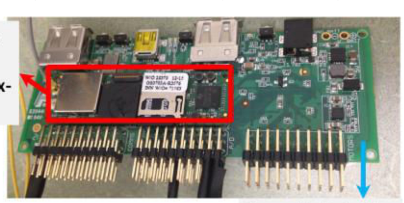

Fig. 3. The prototype based on Gumstix Overo Air COM and RoboVero expansion board.

The training session was conducted first to collect the training data and build the original classifier. The subject was instructed to perform one movement for about 4 seconds in one trial. For each movement task, three separate trials were collected. After the training process was done, the parameters of the generated classifier, as well as the mean vector for each class and the common covariance matrix were saved in the memory for later use in the testing session.

In the real-time testing session, for each movement task, the subject performed the movement for about 4 seconds in four separate trials. Totally 12 testing trials were conducted. In every trial, motion artifacts were manually introduced by randomly tapping the EMG electrodes with roughly equal strength. In the preliminary experiment, we only tapped one electrode at a time. To better evaluate the performance of our self-recovery module, two types of classification decisions with and without the self-recovery module were compared in every analysis window.

In addition, an offline evaluation was conducted to compare the performance between our fast LDA retraining algorithm and the previous retraining strategy [8, 10] by processing the same dataset collected in the real-time testing session.

III. Results & Discussion

A. *Performance of the Retraining Algorithm*

Table 1 summarizes the comparison between our new fast LDA retraining algorithm and the previous retraining algorithm. From the table we can see the new retraining algorithm was two orders of magnitude (118 times) faster than the previous retraining strategy and meanwhile only consumed less than 1% of the memory usage of the old strategy. Furthermore, our fast retraining algorithm only took less than 1 ms to generate the new classifier. This result makes it possible for the system to extract EMG features, detect signal disturbances, retrain the classifier, perform pattern recognition, and produce a decision seamlessly in a sequence within the duration of one window increment (i.e. 20 ms). This new design and implementation clearly demonstrated the feasibility of a self-recovery strategy that is truly 'imperceptible' to users.

B. *System Performance in Real-time*

In the 12 real-time testing trials, totally 48 motion artifacts were introduced, among which 43 were recognized by the CUSUM detector and 20 caused miss classifications if our self-recovery was not used. All the disturbances that led to classification errors were successfully detected. The undetected disturbances were those with either small amplitude or short duration, which did not affect the

Table 1. Comparison between the new retraining method and the previous retraining method

	New Fast Retraining	Previous Retraining
Processing time	0.55 ms (2307 windows, 3 classes, 4 channels)	65 ms (2307 windows, 3 classes, 4 channels)
Speedup	118	1
Memory Usage (2307 windows, 3 classes, 4 channels, 4 features per channel)	$\tilde{\mu}_g$:(4x4)x4 bytes=64 bytes; $\tilde{\Sigma}$: (4x4)x(4x4)x4 bytes =1024 bytes; Total: 64x3+1024 = 1216 bytes=1.2 Kbytes	Total size of the feature matrix: (4x4)x2307x4 bytes = 147648 bytes = 144.2 Kbytes
Meet real-time constraints?	Yes.	No.

classification performance. Without the self-recovery module, there were 277 miss classifications observed among 5993 decisions. All these errors were caused by motion artifacts. Our self-recovery module eliminated 259 of them, resulting in a 93.5% recovery rate.

Fig. 4 shows the real-time system performance of some representative testing trials. The blue line at the bottom demonstrates one channel of the EMG signals which was randomly disturbed by motion artifacts. The black line above is the detection results of the CUSUM detector. As seen in the figure, the CUSUM detector accurately recognized all five motion artifacts. The classification decisions without self-recovery are displayed by the red line. The green line denotes the recovered decisions. The three gray ellipses in the figure mark three typical cases in the experiment. Case A represents a situation in which the self-recovery module successfully eliminates the classification error caused by motion artifacts. This is also the most common case. B is a case in which the sensor fault detector identifies the disturbance but the retrained classifier still provides an incorrect decision. This may be because the disturbed EMG signal is critical to the recognition of this motion. Another case C is a situation where the disturbance does not affect the classification decision.

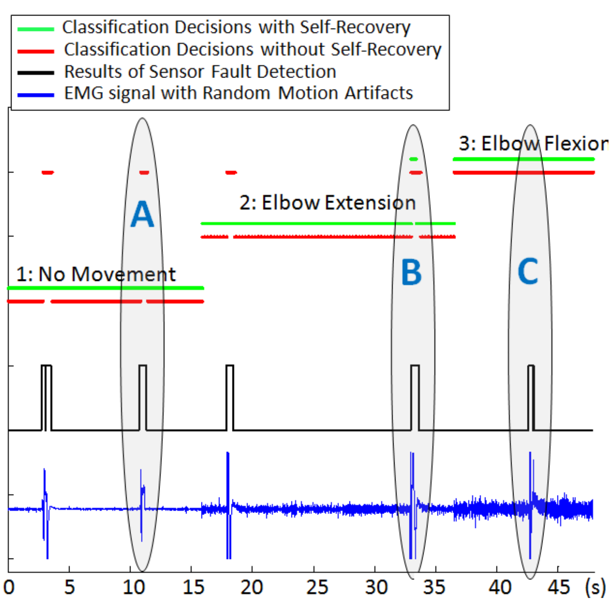

Fig. 4. Real-time system performance of some representative testing trials.

The results of the experiment have shown the promise of a robust, reliable, and efficient real-time EMG pattern recognition interface for artificial arms.

IV. Conclusions

This paper presented a real-time self-recovery EMG pattern recognition interface for artificial arms. The system seamlessly integrated EMG pattern recognition with a self-recovery module that could detect signal disturbances, retrain the classifier, and perform reliable pattern classification in real-time. A novel fast and efficient LDA-based retraining algorithm was developed and demonstrated the ability to immediately recover the classification performance from motion artifacts. The self-recovery EMG pattern recognition system was implemented on an embedded computer system as a working prototype. The preliminary experimental evaluation on an able-bodied subject showed that our system could maintain high accuracy in classifying three arm movements while motion artifacts were manually introduced. The self-recovery module was able to eliminate 93.5% of the miss classifications caused by motion artifacts. These results have demonstrated the feasibility of a clinically viable EMG PR interface for multifunctional prosthetic arm control.

References

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

[2] Huang, H., Kuiken, T.A., and Lipschutz, R.D., *A strategy for identifying locomotion modes using surface electromyography.* IEEE Trans Biomed Eng, 2009. **56**(1): p. 65-73.

[3] Zhang, X., Huang, H., and Yang, Q., *Implementing an FPGA System for Real-Time Intent Recognition for Prosthetic Legs*, in *Design Automation Conference*. 2012: San Fransico, CA. p. 169-75.

[4] Parker, P., Englehart, K., and Hudgins, B., *Myoelectric signal processing for control of powered limb prostheses.* Journal of electromyography and kinesiology: official journal of the International Society of Electrophysiological Kinesiology, 2006. **16**(6): p. 541.

[5] Sensinger, J.W., Lock, B.A., and Kuiken, T.A., *Adaptive pattern recognition of myoelectric signals: exploration of conceptual framework and practical algorithms.* Neural Systems and Rehabilitation Engineering, IEEE Transactions on, 2009. **17**(3): p. 270-278.

[6] Tkach, D., Huang, H., and Kuiken, T.A., *Study of stability of time-domain features for electromyographic pattern recognition.* J Neuroeng Rehabil, 2010. **7**: p. 21.

[7] Hargrove, L., Englehart, K., and Hudgins, B., *A training strategy to reduce classification degradation due to electrode displacements in pattern recognition based myoelectric control.* Biomedical Signal Processing and Control, 2008. **3**(2): p. 175-180.

[8] Huang, H., Zhang, F., Sun, Y.L., and He, H., *Design of a robust EMG sensing interface for pattern classification.* J Neural Eng, 2010. **7**(5): p. 056005.

[9] Liu, Y., Zhang, F., Sun, Y., and Huang, H. *Trust sensor interface for improving reliability of EMG-based user intent recognition*. in 33rd EMBC 2011, Boston, MA.

[10] Zhang, X., Liu, Y., Zhang, F., Ren, J., Sun, Y.L., Yang, Q., and Huang, H., *On Design and Implementation of Neural-Machine Interface for Artificial Legs.* Industrial Informatics, IEEE Transactions on, 2012. **8**(2): p. 418-429.

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