Detecting Localised Muscle Fatigue during Isometric Contraction using Genetic Programming

Ahmed Kattan, Mohammed Al-Mulla, Francisco Sepulveda and Riccardo Poli

School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe, United Kingdom
akatta@essex.ac.uk, mrhlam@essex.ac.uk, fsepulv@essex.ac.uk and rpoli@essex.ac.uk

Keywords: Muscles fatigue, Transition-to-Fatigue, Genetic Programming, EMG, K-means.

Abstract: We propose the use of Genetic Programming (GP) to generate new features to predict localised muscles fatigue from pre-filtered surface EMG signals. In a training phase, GP evolves programs with multiple components. One component analyses statistical features extracted from EMG to divide the signals into blocks. The blocks' labels are decided based on the number of zero crossings. These blocks are then projected onto a two-dimensional Euclidean space via two further (evolved) program components. K-means clustering is applied to group similar data blocks. Each cluster is then labelled into one of three types (*Fatigue, Transition-to-Fatigue and Non-Fatigue*) according to the dominant label among its members. Once a program is evolved that achieves good classification, it can be used on unseen signals without requiring any further evolution. During normal operation the data are again divided into blocks by the first component of the program. The blocks are again projected onto a two-dimensional Euclidean space by the two other components of the program. Finally blocks are labelled according to the k-nearest neighbours. The system alerts the user of possible approaching fatigue once it detects a Transition-to-Fatigue. In experimentation with the proposed technique, the system provides very encouraging results.

1 INTRODUCTION

The electro-myo-gram (EMG) is a test used to record the electrical activity of muscles (Lieber 2002). Muscles produce an electrical potential that is non-linearly related to the amount of force produced in a muscle. Analyzing these signals and associating them with muscle state has been an area of active research in the biomedical community for many decades. For example, Sony (Dubost & Tanaka 2002) has presented a hardware system for musical applications that is controlled by EMG signals. The system is able to recognise different gestures and associate them with particular commands to the machine. Detecting muscle fatigue, however, is still an extremely challenging task.

Atieh *et al.* (Atieh 2005) tried to design more comfortable car seats by trying to identify and classify EMG signals using data mining techniques and statistical analysis to determine localised muscle fatigue. Artificial neural networks have been used to detect muscle activity by Moshoua *et al.* (Moshoua, Hostensa & Papaioanno 2005). In that work, wavelet coefficients were proposed as features for identifying muscle fatigue. Song and collaborators (Song, Jung & Zeungnam 2006) proposed an EMG pattern classifier of muscular fatigue. The adaptation process of hyperboxes of fuzzy Min-Max neural networks was shown to significantly improve recognition performance.

In this paper we investigate the idea of predicting localised muscle fatigue by identifying a *transition state* which resides between the non-fatigue and the fatigue stages within the EMG signal. Genetic Programming (GP) has been used to automate this process (Poli, Langdon & McPhee 2008).

As we will illustrate in the following sections, the proposed approach was able to give an early warning before the onset of fatigue in most of the experimental cases we looked at. Therefore, this approach shows some potential for application domains such as ergonomics, sports physiology, and physiotherapy.

2 THE METHODOLOGY

We try to spot regularities within the EMG data and to associate them to one of three classes: i) *Non-Fatigue*, ii) *Transition-to-Fatigue*, and iii) *Fatigue*. Each class indicates the state of the muscle at a particulate time.

The system works in two main stages: i) *Training,* where the system learns to match different signals' characteristics with different classes, and ii) *Testing*, where the system applies what it has learnt to classify unseen data.

In the training phase, the system processes filtered EMG signals and performs two major functions: i)

Segmentation of the signals based on their statistical features, and ii) *Classification* of the identified segments based on their types (i.e., Non-Fatigue, Transition-to-Fatigue, or Fatigue). For these tasks, GP has been supplied with a language that allows it to extract statistical features from EMG. Table 1 reports the primitive set of the system.

Table 1. Primitive Set

Primitive Set	Input
Median, Mean, Avg_dev, Std, Variance, Signal size, Skew, Kurtosis, Entropy, Zero crossings	Vector of real number
+, -, /, *, Sin, Cos, Sqrt	Real Number

The system starts by randomly initialising a population of individuals using the ramped half-and-half method (Poli, Langdon & McPhee 2008). Each individual has a multi-tree representation. In particular, each individual is composed of one splitter tree, and two feature-extraction trees. (Multi-tree representations of this kind are common in GP, and have been used, for example, for data classification in (Estevez & Pablo 2007)).

2.1. Splitter Tree

The main job of splitter trees is to split the EMG signals in the training set into meaningful segments, where by "meaningful" we mean that each segment indicates the state of a muscle at a particular time.

The system moves a sliding window of size *L* over the given EMG signal with steps of *S* samples. At each step the splitter tree is evaluated. This corresponds to applying a function, $f_{splitter}$, to the data within the window. The output of the program is a single number, λ, which is an abstract representation of the features of the signal in the window. The system splits the signal at a particular position if the difference between the λ's in two consecutive windows is more than a predefined threshold θ. The threshold θ has been selected arbitrarily (θ = 10).

Once the data have been divided into blocks, the system labels each block with one of the three identified classes based on the number of zero crossings in the raw EMG signal, i.e., the number of times the signal crosses the zero-amplitude line (details are in section 2.3). A good splitter tree should be able to place block boundaries at the transitions between three types of muscle states: *i) Non-Fatigue, ii) Transition-to-Fatigue, and iii) Fatigue*.

Preliminary tests showed that an average EMG signal in our set has 50% of non-fatigue, 10% transition-to-fatigue and the remaining 40% is fatigue. Thus, the splitter tree can be considered to be good if it divides the signal into the three types of blocks with both meaningful proportions (i.e., fatigue > non-fatigue > transition-to-fatigue) and meaningful sequence (non-fatigue should appear before transition-to-fatigue and fatigue). Splitter trees that violate these conditions are discouraged by penalizing their fitness value (see section 2.4).

2.2. Feature-Extraction Tree

The main job of the two feature-extraction trees in our GP representation is to extract features using the primitives in Table 1 from the blocks identified by the splitter tree and to project them onto a two dimensional Euclidian space, where their classification can later take place.

We used a standard pattern classification approach on the outputs produced by the two feature-extraction trees to discover regularities in the training signals. In principle, any classification method can be used with our approach. Here, we decided to use K-means clustering to organise blocks (as represented by their two composite features) into groups. With this algorithm, objects within a cluster are similar to each other but dissimilar from objects in other clusters. The advantage with this approach is that the experimenter doesn't need to label the training set. Also, the approach does not impose any constrains on the shape of the clusters. Once the training set is clustered, we can use the clusters found by K-means to perform classification of unseen data.

In the testing phase, unseen data go through the three components of the evolved solution. Blocks are produced by the splitter tree and then projected onto a two-dimensional Euclidean space by the two feature-extraction trees. Then, they are classified based on the majority class labels of their k-nearest neighbours. We use a weighted majority voting, where each nearest neighbour is weighted based on its distance from the newly projected data point. More specifically the weight is $w = 1 / distance\ (x_i, z_i)$, where x_i is the nearest neighbour and z_i is the newly projected data point.

Once the system detects a transition-to-fatigue, it alerts the user about a possible approaching fatigue.

2.3. Labelling the training set

The approach described above is based on an unsupervised learning model. In our case the given EMG signals in the training set are unlabelled. Here, we used the zero crossings to recognise the state of the muscle.

There are several ways to recognise muscle state from the EMG signal. Some Authors (Mannion et al. 1997; Finsterer 2001) argue in favour of the idea of counting the number of times the amplitude of the signal crosses the zero line based on the fact that a more active

muscle will generate more action potentials, which overall causes more zero crossings in the signal. However, at the onset of fatigue the zero crossings will drop drastically due to the reduced conduction of electrical current in the muscle.

Hence, our system decides the labels of the blocks found by the splitter tree in the training set based on the number of zero crossings of the EMG signal. Before the system starts the evolution process, it scans the training set signals and divides them into blocks of predefined length. The number of zero crossings from each block is stored in a sorted vector that does not allow duplicate elements. Then, the system divides this vector into three parts. The lower 40% is taken to represent fatigue, the middle 10% represents transition-to-fatigue, and the higher 50% is non-fatigue. These three propositions (10%, 40% and 50%) were selected based on the preliminary tests with the EMG signals (as we mentioned in Section 2.1). The numbers of zero-crossings of the blocks in these three groups are then used to identify three intervals. Later, the output of the splitter trees is classified into one of these three groups based on the interval in which the number of zero crossings in it falls.

Despite the simplicity of this labelling technique, the labels that it assigned to the output of the splitter trees were consistent and were judged by an expert to indicate the muscle state in most of the cases. However, noise in some parts of the EMG signal can result in wrong labels. Sometimes when the EMG signal is noisy, the labels tend to not be in a meaningful sequence and do not reflect the actual state of the muscle. To solve this problem, we carefully selected a training set with the least noisy signals. Also, each fragment of signal in the training set was manually screened to ensure that it had the correct label.

2.4. Fitness Measurement

The calculation of the fitness is divided into two parts. Each part contributes with equal weight to the total fitness. The fitness contribution of the splitter tree is measured as follows.

Splitter tree fitness is measured by assessing the amount of help provided to the feature extraction trees in projecting the segments into tightly grouped and well separated clusters, plus a penalty if required. More formally, the quality of the splitter tree can be expressed as follows. Let $f_{Feature\text{-}extraction}$ be the fitness of the feature-extraction trees, and μ a penalty values:

$f_{Splitter} = f_{feature\text{-}extraction} + \mu$.

where μ is added if, for example, the splitter does not respect the non-fatigue/transition-to-fatigue/fatigue sequence. The value of μ is fixed.

The second part of the individual's fitness is the classification accuracy (with K-means) provided by the feature-extraction trees. After performing the clustering using K-means we evaluate the accuracy of the clustering by measuring cluster homogeneity and separation. The homogeneity of the clusters is calculated as follows.

The system counts the members of each cluster. Note that, each data point in the cluster represents a block of the signal. Since we already know (from the previous step) the label for each block, we label the clusters according to the dominant members. The fitness function rates the homogeneity of clusters in terms of the proportion of data points – blocks – that are labelled as the muscle's state that labels the cluster. The system prevents the labelling of different clusters with the same label even in cases where the proportions in two or more clusters are equal.

The Davis Bouldin Index (DBI) (Bezdek & Pal 1998) was used to measure clusters separation. DBI is a measure of the nearness of the clusters' members to their centroids and the distance between clusters' centroids. A small DBI index indicates well separated and grouped clusters. Therefore, we add the negation of the DBI index to the total feature extraction fitness in order to encourage evolution to separate clusters (i.e., minimise the DBI). It should be noted that the DBI here is treated as a penalty value, the lower the DBI the lower penalty applied to the fitness.

Thus, the fitness of feature extraction trees is as follows. Let H be a function that calculates the homogeneity of a cluster and let CL_i be the i^{th} cluster. Furthermore, let K be the total number of clusters (three clusters in our case: fatigue, transition-fatigue and non-fatigue). Then,

$$f_{\text{Homogeneity}} = \frac{\sum_{i=1}^{K} H(CL_i)}{K} - DBI$$

The total fitness of the individual is:

$$f = (f_{feature\text{-}extraction}/2) + (f_{Splitter}/2)$$

3 EXPERIMENTS

Experiments have been conducted in order to investigate the performance of the proposed technique. The aim of these experiments is to measure the prediction accuracy with different EMG signals.

3.1. EMG Recording

The data was collected from three healthy subjects (aged 23-25, non-smoker, athletic background). The local ethical committee approved the experiment's design. The three participants were willing to reach

physical fatigue state but not psychological fatigue. Selection criteria were used to minimise the differences between the subjects, which would facilitate the analysis and comparison of the readings. The participants had comparable physical muscle strength. It was also preferred that the subjects had an athletic background with a similar muscle mass, which would facilitate the correlation of the results. It was also important that the volunteers were non-smokers, as it is known that smoking affects physical abilities, which again could lead to inaccurate readings of muscle fatigue in a participant who smokes.

The participants were seated on a chair. Each participant was asked to hold a weight training bar - dumbbell – until their muscle fatigued. The steps in our test bed set up are the following: 1) A bipolar pair of electrodes was placed on the right arm's biceps muscle for EMG recording. 2) The force gauge was perpendicular to the dumbbell to ensure that the force gauge was taking the correct reading. 3) A strap wass handed to participants to enable the force gauge reading. 4) A protractor was used to ensure a 90 degree angle of the elbow for the initial setup. 5) A dumbbell was handed to the participant. 6) A laser was embedded in the dumbbell to give visual guidance of the elbow angle. 7) The elbow position was padded, so the participant was comfortable.

The myoelectric signal was recorded using two channels; Double Differential (DD) recording equipment at 1000Hz sampling rate with active electrodes on the biceps branchii during two isometric dumbbell exercises, with 30% Maximum Voluntary Contraction "MVC" and 80% "MVC" respectively. For each of the three participants 6 trials were carried out, providing 18 trials in total.

3.2. GP Setup

Of the 18 EMG signals (trials) acquired, we used 3 trials for the training set, one trail from each subject. In this way we hoped we would allow GP to find common features for the three different participants and build a general prediction model. The experiments that are presented here were done using the following parameters: population of size 100, maximum number of generations 30, crossover with probability of 90%, mutation with probability 5%, reproduction with probability 5%, tournament selection of size 5 and maximum tree depth of 10.

The performance of our approach has been measured through 18 independent runs, each of which trains the system and uses the output of the training to predict the muscle state of 15 EMG signals (5 signals for each participant). The aim is to obtain a good prediction for each participant and a reasonably general prediction algorithm that performs well on average for all participants. Each GP run results in one splitter tree and two feature extraction trees. Since during offline tests it is possible to know the actual label for each block by counting its number of zero crossings, we compared the GP predictions against the actual labels and counted the proportion of times this was correct (hit rate).

Table 2 reports the best achieved hit rate for each test signal, as well the average hit rate in all runs. Also, the worst hit rates are presented to show the algorithm performance in its worst case. Moreover, standard deviation for each signal is presented to illustrate the system's robustness. The last row of Table 2 reports the approximate average processing time for both training and testing.

Table 2: Summary of performance of 18 different GP runs

Signal	Avg. Hit	Best Hit	Worst Hit	Std.
A1	60.36%	77.27%	37.84%	8.99
A2	57.54%	77.36%	38.89%	9.03
A3	60.69%	85.71%	39.39%	11.24
A4	62.40%	100%	25.00%	19.31
A5	57.96%	81.82%	9.09%	17.92
B1	56.62%	90%	39.29%	13.10
B2	62.73%	82.35%	41.18%	12.98
B3	41.67%	100%	0.00%	28.58
B4	65.08%	100%	0.00%	23.78
B5	58.24%	100%	25.00%	20.00
C1	67.08%	100%	42.31%	14.73
C2	62.28%	78.79%	43.59%	8.27
C3	66.05%	85.71%	40.00%	11.76
C4	56.75%	100%	0.0%	16.56
C5	65.34%	100%	16.56%	42.84
	Average Testing Time			**1 min/test signal**
	Average Training Time			**18 Hours**

Figure 1 summarise the runs' information. We measured the quality of each run by calculating the average hit rate for all test signals (test set). This reflects the accuracy of the evolved programs when dealing with signals from different participants. More specifically, the first column in the histogram in Figure 1 shows the average of the runs' qualities (i.e., the average of the averages). The second column in the figure shows the quality (i.e., average hit rate for all test signals) of the

best evolved predictor. The third column shows the quality of the worst evolved predictor. Finally, the last column shows the average standard deviation for all runs. The low standard deviation with the reasonable average prediction accuracy indicates that our system is likely to produce accurate models within a few runs.

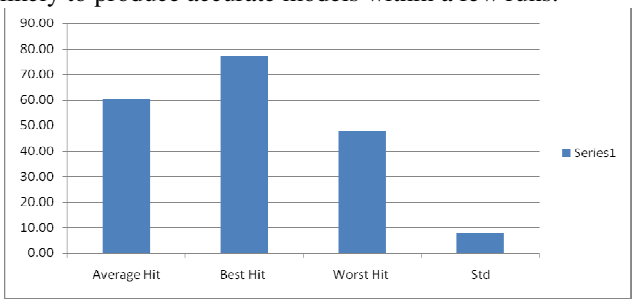

Figure 1: Summary of 18 runs

Table 3 reports the performance of the best evolved predictor as well as the performance of the worst, signal by signal. It should be noticed that the achieved results are promising, especially considering that the system is predicting muscle fatigue for three different individuals.

Table 3: Best GP test run vs. worst GP test run

Signal/ results	Best run	Worst run
A1	66.67%	37.84%
A2	66.66%	53.12%
A3	75.47%	48.91%
A4	25%	50%
A5	66.67%	40%
B1	90%	40.74%
B2	81.25%	65.52%
B3	50%	33.33%
B4	100%	50%
B5	100%	42.86%
C1	100%	42.31%
C2	78.79%	43.84%
C3	75%	56.25%
C4	80%	69.23%
C5	100%	42.857%

4 CONCLUSION

Our results are encouraging, in the sense that a good prediction has been achieved and further significant improvements could be obtained.

Although, this approach has achieved good prediction rates (100% in some cases), it suffers from a major disadvantage. The proposed labeling mechanism – being based on zero crossings – is not very reliable (as described in section 2.3). This is due to the fact that the number zero crossings is affected by noise. Thus, the labels of the signal's blocks might not be accurate. This has the potential to hamper or prevent learning in the system. We avoided this problem altogether by carefully selecting the least noisy signals for the training set and suitably preprocessing the signals. Nevertheless, there remains the possibility that some errors in the labeling still occur.

There are many directions for us to further improve the performance of this technique. For example, a simple extension in the set of statistical functions available in the primitive set might improve the classification accuracy. Also, the use of more sophisticated technique to label the EMG blocks (e.g., fuzzy classification) might improve the system's reliability and flexibility.

REFERENCES

Atieh, M, Younes, R, Khalil, M & Akdag, H December 2005, 'Classification of the Car Seats by Detecting the Muscular Fatigue in the EMG Signal', *International Journal of Computational Cognition*.

Bezdek, JC & Pal, NR Jun 1998, 'Some new indexes of cluster validity', *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol 28, no. 3, pp. 301-315.

Dubost, G & Tanaka, A September 2002, 'A Wireless NetWork based Biosensor Interface for Music', *Proc. of Intern. Computer Music Conf.*

Estevez, NB & Pablo, A 25-28 September 2007, 'Genetic Programming-Based Clustering Using an Information Theoretic Fitness Measure', *IEEE Congress on Evolutionary Computation*, IEEE Press.

Finsterer, J August 2001, 'EMG-interference pattern analysis J', *Journal of Electromyography and Kinesiology*, vol 11, pp. 231-246(16).

Lieber, RL 2002, *Skeletal muscle structure, function & plasticity: the physiological basis of rehabilitation*, 2nd edn, Lippincott Williams & Wilkins.

Mannion, AF, Connolly, B, Wood, K & Dolan, P 1997, 'The use of surface EMG power spectral analysis in the evaluation of back muscle function', *Journal of Rehabilitation*, vol 34, pp. 427-439.

Moshoua, D, Hostensa, I & Papaioanno, G July 2005, 'Dynamic muscle fatigue detection using self-organizing maps', *Applied Soft Computing*, Elsevier.

Poli, R, Langdon, WB & McPhee, N 2008, *A field guide to genetic programming*, http://lulu.com.

Song, J-H, Jung, J-W & Zeungnam, B 2006, 'Robust EMG Pattern Recognition to Muscular Fatigue Effect for Human-Machine Interaction', Springer.