Classification of localized muscle fatigue with Genetic Programming on sEMG during isometric contraction

M. R. Al-Mulla, F. Sepulveda, M. Colley and A. Kattan, *Member, IEEE*

***Abstract*—Genetic Programming is used to generate a solution that can classify localized muscle fatigue from filtered and rectified surface electromyography (sEMG). The GP has two classification phases, the GP training phase and a GP testing phase. In the training phase, the program evolved with multiple components. One component analyzes statistical features extracted from sEMG to chop the signal into blocks and label them using a fuzzy classifier into three classes: *Non-Fatigue, Transition-to-Fatigue* and *Fatigue.* The blocks are then projected onto a two-dimensional Euclidean space via two further (evolved) program components. K-means clustering is then applied to group similar data blocks. Each cluster is then labeled according to its dominant members. The programs that achieve good classification are evolved. In the testing phase, it tests the signal using the evolved components, however without the use of a fuzzy classifier. As the results show the evolved program achieves good classification and it can be used on any unseen isometric sEMG signals to classify fatigue without requiring any further evolution. The GP was able to classify the signal into a meaningful sequence of Non-Fatigue→Transition-to-Fatigue→Fatigue. By identifying a Transition-to Fatigue state the GP can give a prediction of an oncoming fatigue. The genetic classifier gave promising results 83.17% correct classification on average of all signals in the test set, especially considering that the GP is classifying muscle fatigue for ten different individuals.**

I. Introduction

This research investigates the ways in which Genetic Programming (GP) can be utilized to detect and predict localized muscle fatigue during isometric contractions in the bicep branchii. The study also explores the idea of classifying and hence predicting muscle fatigue by identifying a transition state. Genetic Programming (GP) has been used to automate this process.

Previous studies on muscle fatigue in isometric contraction have established typical sEMG readings when conducted in controlled settings. Changes in sEMG amplitude and centre frequency were studied [1]. That study found a decrease in the centre frequency of the spectrogram of all the muscle groups. Research in this field also shows that a development in muscle fatigue correlates with changes in amplitude and median frequency (MDF) [2].

Reference [3] tried to design more comfortable car seats by identifying and classifying sEMG signals using data mining techniques and statistical analysis to determine sEMG localized muscle fatigue. Reference [4] used artificial neural networks to detect muscle activity whereby wavelet coefficients are proposed as features for identifying muscle fatigue. Reference [5] proposed an sEMG pattern classifier of muscular fatigue. The adaptation process of hyperboxes of fuzzy Min-Max neural networks has shown a significant improvement in recognition performance.

There are three main aims of this study. Firstly, we want to understand the relationship between statistical features and the nature of different muscle states. This has potential to help researchers to better understand the nature of muscle fatigue and develop fatigue detection and prediction algorithms. Secondly, the study investigates offline processing of the surface electromyography (sEMG) signal to provide an early warning before the onset of fatigue. Finally, the limitations and the capabilities of using GP for the mentioned domain will be scrutinized in order to accelerate the evolution process and optimize the quality of the evolved solutions.

II. Methods and Materials

In the first part of this research an experimental study was conducted to record sEMG of localized muscle. The second part involved using the GP to classify fatigue of the localized muscle. The data recorded were utilized to investigate the performance of the proposed technique. The aim of the experiments was to measure the classification accuracy of the GP with different sEMG signals to give a generalized solution.

A sEMG Recording and Preprocessing

The data were collected from ten healthy subjects (mean age 27.5 +/- 3.6 yr, non-smoker, athletic background). The ten participants were willing to reach physical fatigue state but not psychological one. The participants were seated on a preacher curl machine to insure stability and biceps isolation.

Steps in the test bed set up:

- sEMG electrodes were placed on the participant's Biceps branchi belly to acquire sEMG reading.
- Goniometer was placed on the side of the arm to measure the elbow angle.

Manuscript received April 7, 2009.

M. R. Al-Mulla is with the University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester, CO4 3SQ, United Kingdom, (phone: +44 1206 874879; fax: +44 1206 872684; e-mail: mrhalm@essex.ac.uk).

Dr F. Sepulveda is with the University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester, CO4 3SQ, United Kingdom, (e-mail: fsepulv@essex.ac.uk).

Dr M. Colley is with the University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester, CO4 3SQ, United Kingdom, (e-mail: martin@essex.ac.uk).

A. Kattan is with the University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester, CO4 3SQ, United Kingdom, (e-mail: akatta@essex.ac.uk).

- The participant had a display placed in front of them which indicates the angle of the arm.
- The weight was handed to the participant at 90 deg elbow angle.
- Participants were asked to maintain the 90 deg angle.
- Participants stopped when they reach total biceps fatigue.
- All participants carried out isometric exercises with 40% Maximum Voluntary Contraction (MVC).

The myoelectric signal was recorded using two channels; Double Differential (DD) recording equipment at 2000Hz sampling rate. The sEMG signals acquired from the experiment went through a rectification and filtering process. The signals were filtered with a dual pass Butterworth filter, with the fifth order band positioned between 1 and 500Hz. The Goniometer readings were also recorded simultaneously. The reading of the Goniometer was then correlated with the sEMG signal to insure that fatigue resides within the sEMG. The Goniometer provided a reliable indication on the development of fatigue as it was difficult for the participants to maintain 90deg elbow angle throughout the trial .

For each of the ten participants three trials were carried out, providing 30 trials in total. There was a resting period of one week between each of the three trials.

B GP Set Up

Of the 30 sEMG signals (trials) recorded, we used 10 trials for the training set, one trail from each subject. Thus, we allowed the GP to find common features for the ten different participants and built a generalized classification model. The GP experiments that are presented here were completed using:

- Population of size 100.
- Maximum number of generations 30.
- One point crossover with probability of 90%.
- Mutation with probability 5%.
- Reproduction with probability 5%.
- Tournament selection of size 5.
- Maximum tree depth of 10.

Since GP search is stochastic, the performance of our approach has been measured through 20 independent runs, each of which trained the GP and used the output of the training to classify the muscle state of 20 sEMG signals (2 signals for each participant). The aim was to obtain a good classification ratio for each participant and a general classification algorithm that performs well on average for all participants. Each GP run resulted in one chopper tree and two feature extraction trees, which will be explained in the following section.

C GP algorithm

The approach to implement GP as a way to classify localized muscle fatigue based on sEMG's statistical features works as follows:

- Spotting regularities within the sEMG 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:
    - *Training:* the algorithm learns to match different signals' characteristics with different classes.
    - *Testing*: here the algorithm applies what it has learnt to classify unseen data.

In the training phase, the algorithm processed filtered sEMG signals and performed two major functions:

- Segmentation of the signals based on their statistical features.
- Classification of the identified segments based on their types (i.e. Non-Fatigue, Transition-to-Fatigue, or Fatigue).

For these tasks, the GP has been supplied with a language that allowed it to extract statistical features from sEMG. The selection of the primitive set, as shown in Table I, was carefully made to avoid unnecessary growth in the search space, while at the same time ensuring that it was rich enough to express the solution.

TABLE I

THE PRIMITIVE SET OF THE ALGORITHM

Primitive Set	Input
Median, Mean, Average deviation, Standard deviation, Variance, RMS, Skew, Kurtosis, Entropy	*Vector of real number*
+, -, /, *, *Sin, Cos, Sqrt*	*Real Number*

The algorithm started by randomly initializing a population of individuals using the ramped half-and-half method [6]. In particular, each individual was composed of one chopper tree (explained in section C.i), and two feature-extraction trees (explained in section C.ii). In the testing phase, the unseen data (test data set) goes through the three evolved components (Chopper tree and two feature extraction trees) without using the fuzzy label classifier (explained in section C.iii). The chopper tree segments the signal then passes the outcome to the feature extraction trees where they are classified based on the majority class labels of their k-nearest neighbors. In the majority voting, each nearest neighbor is weighted based on its distance from the newly projected data point. So, $w = 1 / distance\ (x_i, z_i,)$. Where x_i is the nearest neighbor and z_i is the newly projected data point.

C.i Chopper Tree

The chopper tree worked by chopping the sEMG signals in the training set into meaningful segments based on their statistical differences. This only occurred in the training phase of the algorithm so that it learned to match statistical characteristics with particular muscle status. The algorithm labeled each block with one of the three identified classes based on the Fuzzy labeling classifier explained in section C.iii. A good chopper tree should be able to detect three types of blocks: *Non- Fatigue*, *Transition-to-Fatigue*, and *Fatigue*. The algorithm detected these blocks by isolating the boundaries between the non-fatigue and the fatigue with a transition-to-fatigue block which usually resided just before the end of the signal.

The algorithm moved a sliding window of size L over the given sEMG with steps of S samples. At each step the chopper tree was evaluated. This corresponded to applying a function, $F_{chopper}$, to the data under the window. The output of the program was a single number, λ, which is an abstract representation of the features of the signal in the window. The algorithm then chopped the signal at a particular position if the difference between the λ's in two consecutive windows was more than a predefined threshold θ. Thus, we could formalize this stage with the following pseudo code:

```
Repeat
  if F_chopper (prevWindow) – F_chopper (currWindow) > θ then
     Chop
  else
     Move(S)  //slide the window by S steps.                (1)
```

Preliminary tests showed that an average sEMG signal in our set had 81.6% of non-fatigue, 4.26% transition-to-fatigue and the remaining 14.14% was fatigue. These numbers were varying from one individual to another. However, what was common among signals was that the smallest portion of the signal represented transition-to-fatigue while the largest portion was non-fatigue. Thus, the chopper tree should divide the signal into the three types of blocks with both meaningful sequence (i.e., Non-Fatigue → Transition-to-Fatigue→Fatigue). Chopper trees that violate these conditions are discouraged by penalizing their fitness value (see section C.iv).

C.ii Feature-Extraction Tree

The main job of the two feature-extraction trees in our GP representation was to extract features (see table 1) from the blocks identified by the chopper tree and to project them into a two dimensional Euclidian space, where their classification took place later. Each feature-extraction tree represented a transformation formula which mapped the original feature set into a single value output, which could be considered as a composite, higher-level feature.

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

Instead of forcing the clustering algorithm (the K-means in our case) to group segments based on their raw statistical features directly, we let evolution optimize two features-extractions trees and used them to project the training segments on a two-dimensional Euclidian space. If the trees were successfully evolved, K-means would then be able to group together blocks based on their raw statistical features, which would have otherwise been grouped separately. The GP might invent new features that a human finds impossible to discover.

C.iii Labeling the Training Set

There are several ways to recognize muscle state from the sEMG signal. In [7]-[9] the authors argued in favor 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 would 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.

In our case the given sEMG signals in the training set were unlabelled. Therefore, we needed a mechanism to label the outcome of the chopper tree to indicate the muscle state at a particular time (i.e., non-fatigue, transition-to-fatigue or fatigue). Here, we used the fuzzy classifier that had two inputs:

- Angular position of the arm provided by the Goniometer (0 to 180 degrees).
- Zero Crossings of the sEMG data.

The above fuzzy classifier inputs when used in conjunction was found to greatly reduce incorrect labeling of the training data sets. Both inputs were used to define a 6 rule zero order type-1 fuzzy classifier; using both triangular and trapezoidal antecedents and product inference. Table II below defines the rule base.

TALE II
RULE BASE

#	IF		THEN	
	in1	in2	DoS	out1
1	GOF	ZCF	1.00	FATIGUE
2	GOF	ZCTF	1.00	FATIGUE
3	GOF	ZCNF	1.00	FATIGUE
4	GONF	ZCF	1.00	TRANFATIGUE
5	GONF	ZCTF	1.00	TRANFATIGUE
6	GONF	ZCNF	1.00	NONFATIGUE

Rule Base: Goniometer Fatigue (GOF) ,(GONF) Goniometer Non-fatigue,(ZCF) zero crossing fatigue,(ZCTF) zero crossing transition-to-fatigue,(ZCNF) zero crossing non-fatigue.

As with all fuzzy classifiers only a single label was chosen as the final output; the rule with the greatest firing strength. In the current algorithm only the label and not the rule firing strength was used with training the GP and could be considered as a hard classifier. In the experiment the subjects were instructed to maintain a Goniometer angle of 90 degrees until complete fatigue. Thus the Goniometer angle of 85 degrees and below was used to generate the input fuzzy sets where the trapizoidal set is defined as a four point array, e.g. (GOF), and the triangle set is defined as three point array, e.g. (ZCTF), as follows:

GOF $\left[0,0,\frac{\sum_{i=0}^{n-1} G[i]}{n}, \frac{\sum_{i=0}^{n-1} G[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(G[i] - \frac{\sum_{i=0}^{n-1} G[i]}{n}\right)^2}\ \right]$

where $G[i]$ is less than 85. (2)

GONF $\left[\frac{\sum_{i=0}^{n-1} G[i]}{n}, \frac{\sum_{i=0}^{n-1} G[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(G[i] - \frac{\sum_{i=0}^{n-1} G[i]}{n}\right)^2}, 180, 180\right]$

where $G[i]$ is less than 85. (3)

ZCF $\left[0,0,\frac{\sum_{i=0}^{n-1} ZC[i]}{n}, \frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2}\ \right]$ (4)

ZCTF $\left[\frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \left(\sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2}\right)/2,\ \frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2},\ \frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2}\ /2\right]$ (5)

ZCNF $\left[\frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \left(\sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2}\right), \frac{\sum_{i=0}^{n-1} ZC[i]}{n} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2} + \sqrt{\sum_{i=0}^{n-1}\left(zC[i] - \frac{\sum_{i=0}^{n-1} ZC[i]}{n}\right)^2}\ /2, 200, 200\right]$

where $ZC[i]$ are all data points where the Goniometer is less than 85. (6)

Fig. 1 shows one of the signals after the fuzzy classification process. The fuzzy classifier correctly classified the signal into three different labels according to fatigue status.

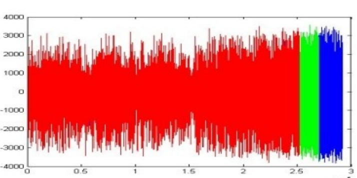

Fig. 1. Labels of a signal in the training set.
Fatigue = blue, Non-Fatigue = red, Transition-to-Fatigue = green

C.iv Fitness Measures

The fitness function is an important component of a GP algorithm. This function evaluated the quality of the individuals and guided the evolution to uncover progressively improved solutions during an algorithm run.

The calculation of the fitness was divided into two parts. Each part contributed with equal weight to the total fitness. Firstly, the fitness contribution of the chopper tree was measured. Any contravention to these conditions was penalized, and thus, the evolution process would discriminate against them in the following generations.

The fitness of the chopper tree was determined by the way it aided the feature-extraction trees to project segments into grouped and separated clusters and penalized when required. This gives the function:

$$f_{Chopper} = f_{feature\text{-}extraction} + \mu, \tag{7}$$

where $f_{Feature\text{-}extraction}$ is the fitness of the feature-extraction trees, and μ is the penalty values.

μ is a fixed value and applied whenever the chopper tree fails to divide the given sEMG signal into blocks (non-fatigue/transition-to-fatigue/fatigue sequence). Thus, the algorithm discriminate these trees in the following generations.

Secondly, the subject's fitness was classified using K-means. K-means aided in evaluating the accuracy of the clustering by measuring cluster homogeneity and separation. To calculate the homogeneity of the clusters the algorithm counted the members of each cluster (see Fig. 2), where each data point represented a block of signals. The labels from each block were known in the labeling stage, hence we labeled the clusters according to dominant members.

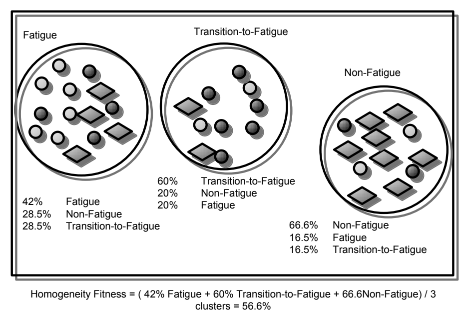

Fig. 2. Homogeneity of the clusters

To calculate the homogeneity of the clusters we used the following function:

$$f_{\text{Homogeneity}} = \frac{\sum_{i=1}^{K} H(CL_i)}{K} \tag{8}$$

where H is the function calculating the homogeneity and CL_i is the i^{th} cluster. Furthermore, K represents the total number of clusters (three clusters in our case: fatigue, transition-fatigue and non-fatiuge).

The Davis Bouldin Index (DBI) [10] was another mean used to decide cluster quality. DBI is a measure of the nearness of the clusters' members to their centroids and the distance between clusters' centroids. This measure helped where clusters with objects far apart extended the cluster's boundary and could lead to less accurate classification of unseen objects. Also, clusters that overlap each other were not suitable, as ideal clusters were separated from each other and densely grouped near their centroids. DBI can be expressed as follows:

C_{CLi} is the centroid of the CL_i^{th} cluster and d_{CLi}^{n} the n^{th} data member that belongs to the CL_i^{th} cluster. In addition, the

Euclidian distance between d_{CLi}^{n} and C_{CLi} is expressed by the function be *dis*(d_{CLi} , C_{CLi}). Furthermore, *K* is the total number of clusters. Finally, standard deviation is denoted as *std()*. Then,

$$\mathrm{DBI} = \frac{\sum_{i=1}^{K} Std\left[dis(C_{CLi,}d_{CLi}^{0}),\ldots,dis(C_{CLi,}d_{CLi}^{n})\right]}{dis(C_{CL0},C_{CL1},\ldots,C_{CLi})} \tag{9}$$

A small DBI index indicated well separated and grouped clusters. Therefore, we added the negation of the DBI index to the total feature extraction fitness in order to push evolution to separate clusters (i.e., minimize the DBI). It should be noted that the DBI here was treated as a penalty value, the lower the DBI the lower penalty applied to the fitness. Thus for, the feature extraction trees; fitness was as follows:

$$f_{feature\text{-}extraction} = f_{\mathrm{Homogeneity}} - \mathrm{DBI} \tag{10}$$

The total fitness of the individual was:

$$f = (f_{feature\text{-}extraction}/2) + (f_{chopper}/2) \tag{11}$$

Therefore, a GP individual's quality was defined by its ability to identify muscle states from the sEMG signal and classify them correctly.

C.v Search Operators

Search operators in any GP algorithm are important as they guide the search through the search space to discover new solutions. We used the standard genetic operators; crossover, mutation and reproduction. These operators took the multi-tree representation of the individuals into account.

There were several options for applying genetic operators to a multi-tree representation. Firstly, we could apply a particular operator that has been selected to all trees within an individual. Alternatively, we could iterate over the trees in an individual and select a potentially different operator for each. Also, we could constrain crossover to occur only between trees at the same position in the two parents or we could let evolution freely crossover different trees within the representation. In preliminary research we tried all of these approaches and found that a good way to guide the evolution is as follows:

the i^{th} individual of the population is denoted as I_i and T_c^i is the c^{th} tree of individual *i,* where $c \in$ {*chopper*, *feature-extractor*$_x$, *feature-extractor*$_y$}. The algorithm selects an operator with a predefined probability for each T_c^i .

In the crossover the operator was chosen and a restriction was applied so that chopper trees could only be crossed over with chopper trees. However, the algorithm was able to freely crossover feature-extractions trees at any position.

III. RESULTS

The genetic program achieved the best population around generation 28 in all of the 20 independent runs after which the average best fitness reaches a plateau. Table III reports the best achieved hit rate (correct classification) for each test signal (A1 refers to trial 1 for subject A and so forth), as well the average hit rate in all runs. Also, the worst hit rate are presented to show the algorithm performance in its worst case. Moreover, standard deviations are presented to show stability of the evolved program through the 20 different independent runs.

TABLE III
SUMMARY OF THE PERFORMANCE OF 20 DIFFERENT GP RUNS

Test Signal	Average Hit %	Best Hit %	Worst Hit %	Standard Dev. %
A1	77.71	90.20	65.20	8.34
A2	83.17	93.80	71.20	6.80
B1	73.28	83.57	65.71	6.05
B2	78.28	91.53	65.63	6.79
C1	74.87	86.31	61.04	7.89
C2	74.62	89.31	61.77	6.46
D1	68.41	83.08	59.79	5.16
D2	56.54	71.19	48.18	5.69
E1	36.57	45.00	28.60	3.67
E2	58.09	75.00	38.33	11.42
F1	72.72	82.50	63.06	5.99
F2	74.94	81.67	65.78	4.36
G1	81.58	93.41	73.02	6.44
G2	68.53	82.30	55.00	8.50
H1	70.74	81.60	62.14	5.26
H2	73.71	94.75	57.50	7.98
I1	40.45	60.50	30.60	8.36
I2	75.11	84.06	58.40	6.90
J1	76.09	81.40	69.50	5.48
J2	75.89	80.10	69.70	4.68

Table IV below defines the results for the best evolved individual when tested with A1 to J2 data sets.(Letters represent the subject and the number represent trials of that subject). This result highlights the promising potential of the GP when used for classifying sEMG signals.

TABLE IV
BEST GP RUN

Test Signal	Best Hit %
A1	73.42
A2	76.4
B1	88.11
B2	82.27
C1	79.675
C2	89.4
D1	87.8
D2	89.21
E1	77.05
E2	89.8
F1	93.8
F2	90.06
G1	89.5
G2	71.2
H1	75.77
H2	90.7
I1	79.02
I2	78.02
J1	78.3
J2	83.94

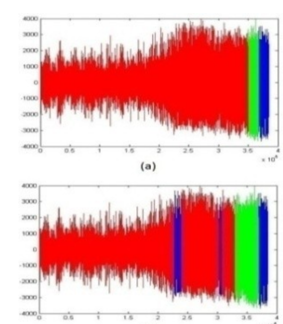

Fig. 3.: Visualized Illustration of GP Performance in One of the Runs for Test Signal D1.
Fatigue = blue, Non-Fatigue = red, Transition-to-Fatigue = green.

To simplify the results, Fig. 3 shows a visualized presentation of the algorithm performance in one of the test signals (D1) using the best evolved individual. The figure illustrates the difference between the actual fuzzy classifier output (a) and the prediction output of the GP (b). The figure shows the intervals where the algorithm failed and when it correctly classified the signal. Table IV shows that the hit percentage of this test was 87.8%. It can be noted that the detection of the transition-to-fatigue state can be used as a prediction mechanism for predicting fatigue.

The GP in all its 20 runs showed a repeated inclination to use some primitive functions than others (Table I showed all the primitive set that the GP was using). Fig. 4 below illustrates the average use of the primitive functions for the best evolved generation in all GP runs are shown.

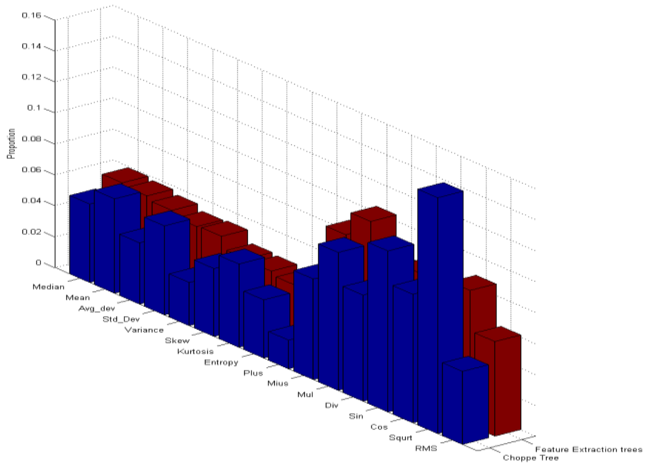

Fig. 4. Average Use of Primitive Functions for the best evolved generations.

Comparing the GP results with typical EMG classification techniques and their accuracy rates [11] showed that the overall accuracy of the GP is promising. Table V shows the correct classification of all these techniques including the average of the best evolved GP shown previously in Table IV.

Table V: Comparison of the GP with other EMG classification techniques

Classification Technique	Accuracy rate %
Coefficient of AR	99
Fuzzy Systems	85
Neural Network	84
Average of Best evolved GP	83.17

The performance of the GP shows great potential in this study and it proved that it can be used to classify the sEMG with comparable performance to other techniques mentioned in table V.

IV. Conclusion

The GP created a classifier which generally performs quite well in comparison to other techniques mentioned in Table V. Clearly the generation of a single classifier is a difficult task due to the stochastic nature of the sEMG signal. In some instances the GP classifier performs very well, as shown for the data set in Table IV.

Despite the low occurrence of the transition-to-fatigue in the training data (average of 6.44% of all the training sets) the GP was still able to identify transition-to- fatigue state with some accuracy using only a moderate primitive set. Additionally, the GP achieved one of our main goals which was to broadly classify the signal in the correct sequence of Non-Fatigue→Transition-to-Fatigue→Fatigue.

As we mentioned previously, our aim was to obtain a generic classifier that performs well on average of all signals in the test set. It should be noticed that the achieved results are promising, especially considering that the algorithm is classifying muscle fatigue for ten different individuals.

The combination of the primitive set that the GP uses corresponds to a solution that produces the ability to detect and predict muscle fatigue. This does not mean that they are the only reasons for producing good solutions. GP trees are known to be sensitive to its primitives (i.e. one node in the tree might have a large influence on the fitness). Thus, we can see the importance of these operations that are frequently used in detecting different muscle status. However, further investigation on each feature of the identified function set is required to measure their exact correlation.

The limitations of GP algorithms is that their evolved solutions are often difficult to interpret by humans, as they are complex and difficult to understand. There are many directions where we can further improve the performance of this technique. For example, a simple extension in the set of statistical function available in the primitive set. The fitness function could be tweaked to improve the classification accuracy and give better solutions.

V. References

[1] J. S. Petrofsky, R. M. Glaser, C.A. Phillips, A. R. Lind, and C. Williams, "Evaluation of the amplitude and frequency components of the surface EMG as an index of muscle fatigue," *Ergonomics*, 25, pp. 213-23, 1982.

[2] M. Hagberg, "Work load and fatigue in repetitive arm elevations," *Ergonomics*, 24, pp. 543-55, 1981.

[3] M. Atieh, R. Youn`es, M. Khalil and H. Akdag, "Classification of the car seats by detecting the muscular fatigue in the EMG signal," *Journal of Computational Cognition*, Vol. 3, No. 4, Dec 2005.

[4] D. Moshou, I. Hostens, G. Papaioannou, and H. Ramon, "Dynamic muscle fatigue detection using self-organizing map," *Applied Soft Computing,* Vol. 5(4): 391–8, Elsevier, July 2005.

[5] J. Son, J. Jung, and Z. Bien, "Robust EMG pattern recognition to muscular fatigue effect for human-machine interaction," *MICAI, Lecture Notes in Computer Science*, Vol. 4293, pp. 1190-1199, Springer, 2006.

[6] R. Poli, W. B. Langdon, and N. F. McPhee. *A field guide to genetic programming*, (2008).

[7] G. Ebenbichler, J. Kollmitzer, M. Quittan, F. Uhl, C. Kirtley, and V. Fialka, "EMG fatigue patterns accompanying etric fatiguing knee-extensions are different in monoand bi-articular muscles," *Electroencephalography and Clinical Neurophysiology*, 109, 256-262, 1998.

[8] A.F. Mannion, B. Connolly, K. Wood, P, Dolan, "The use of surface EMG power spectral analysis in the evaluation of back muscle function," *Journal of Rehabilitation*, 1997.

[9] J. Finsterer, "EMG-interference pattern analysis," *J. Electromyogr. Kinesiol,* Vol. 11, 231-46, 2001.

[10] J. C. Bezdek and N. R. Pal, " Some new indexes of cluster validity," *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, 28(3), pp. 301-315, 19, 1998.

[11] M. B. I. Reaz, M. S. Hussain, F. Mohd-Yasin, "Techniques of EMG Signal Analysis: Detection, Processing, Classification and Applications", Biological Procedures Online, vol. 8, issue 1, pp. 11-35, March 2006.