Algorithms for classification of a single channel EMG signal for human-computer interaction

Andrei Lukyanchikov [1], *Alexei* Melnikov[1] and *Oleg* Lukyanchikov [1,*]

[1] Moscow technological university, 119454 Moscow, Russia

Abstract. One of the most accurate and effective ways to control gestures is to control muscle activity, which occurs with any movement. Electromyography (EMG) is used to record such activity. This article compares SVM classification algorithms, perceptron, random trees and the method of density of probability in relation to the EMG signal. Arduino Leonardo with a single-channel Shield EMG is used to record the signal. The aim of this paper is to prove the possibility of creating a cheap and accessible biointerface based on EMG signal.

1 Introduction

Since the beginning of the development of computer technology, the main interface of human-computer interaction has been dialog boxes controlled by keyboard and mouse. However, with the development of hardware and software, for example, with the advent of mobile phones, augmented reality helmets, IoT, more intelligent interaction interfaces were required, which can be classified as follows: (1) Voice control [1, 2]; (2) Control gestures (with video cameras [3], special gloves [4] or special sensors, such as Kinect); (3) The brain-computer interface [5,6].

The use of these interfaces provides convenience in situations where there is no possibility or need to interact with external physical devices.

One of the most accurate and effective ways to control gestures is to control muscle activity, which occurs with any movement. Therefore, the use of signals of bioelectrical activity of muscles to control a device is an urgent task for today. Electromyography (EMG) is used to record this activity. In addition, the scope of EMG signals is very wide: they can be used in medicine to study muscle activity abnormalities; when evaluating the effectiveness of rehabilitation measures; for monitoring the human condition, etc.

When analyzing the bioelectrical activity of different muscles, one can understand which muscles were involved, and, therefore, with the best classification method, determine the movement that is made with their help.

2 Hardware

2.1 Signal nature

After the arrival of a nerve impulse from the spinal cord to the terminal plate of the nerve attached to the muscle fiber, the post-synaptic membrane of the muscle fiber depolarizes. The depolarization is caused by the movement of Ca++ ions inside. The depolarization process extends to both sides of the end plate. The electrodes located in this field register the voltage (electrical potential) of the muscle fibers. The range of amplitudes of the electromyographic signal lies in the range of one microvolt to several tens of millivolts. The frequency of the signals varies from 0.5 Hz to several hundred hertz, but the individual components of the EMG spectrum can have a frequency of 10-20 kHz.

2.2 Electrodes

Electrodes for removing muscle biopotentials are subdivided into surface and needle. In this paper, the surface electrodes shown in Figure 1 are used.

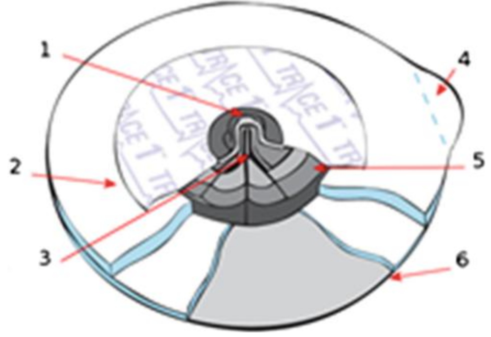


Fig. 1. Structure of the electrode. (1) Metal cone, (2) Foam lining, (3) Ag / AgCl sensor, (4) wing, (5) Electrolyte gel, (6) Adhesive protective film.

With the help of the surface electrode, only total muscle activity can be recorded, representing the interference of the action potentials of many hundreds and even thousands of fibers.

To record the action potentials of muscle fibers or their groups, needle electrodes inserted into the muscle

[*] Corresponding author: luk-it6@yandex.ru

are used. Such electrodes will make it possible to classify more qualitatively similar movements, but such electrodes cannot be used in everyday life, but only in the laboratory with the participation of medical specialists.

2.3 Equipment for EMG signal removal

In the research work for testing, the Arduino Leonardo microcontroller and the EMG shield from OLIMEX were required, the advantage of which is the general availability and cheapness of relatively expensive medical equipment.

Figure 2 shows the Arduino microcontroller and the shielded EMG shield.

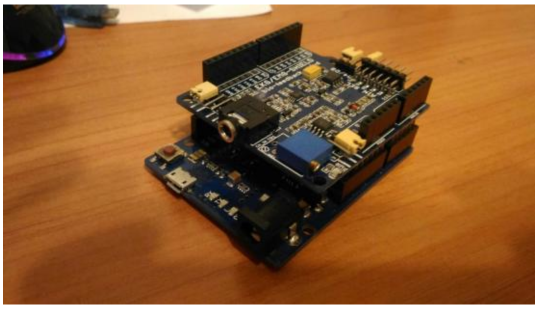


Fig. 2. Device for fixing EMG signal.

3 Classification of signals

In [7], four channels and the maximum likelihood method were used to classify eight gestures based on entropy probability densities. In this regard, it was decided to try to get the same result with only one channel and other hardware, but with the same method.

The location of the electrodes for one channel was chosen the same as the second channel in [7], since in this channel there is the most powerful useful signal. Figure 3 shows the location of the electrodes.

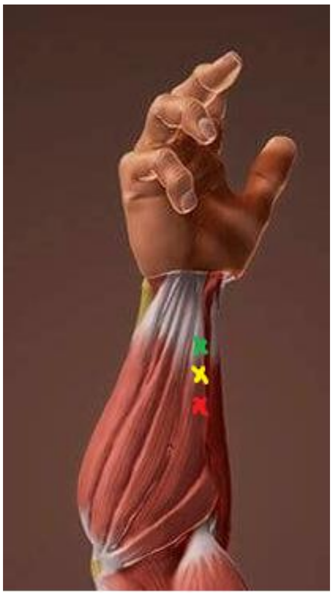


Fig. 3. Location of electrodes.

There are three gestures shown in Figure 4.

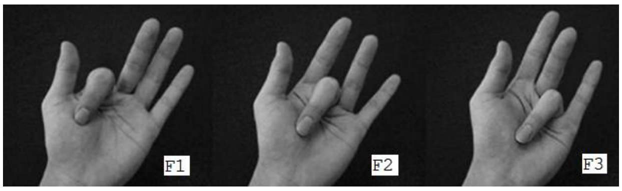


Fig. 4. Three finger gestures: F1– flexing of the index finger, F2 – bending of the middle finger, F3 – bending of the ring finger.

The signal recording lasted 5 seconds, of which 2 seconds – inactivity, then compression, and again inactivity.

Figure 5 shows the recorded signals of each of the three gestures.

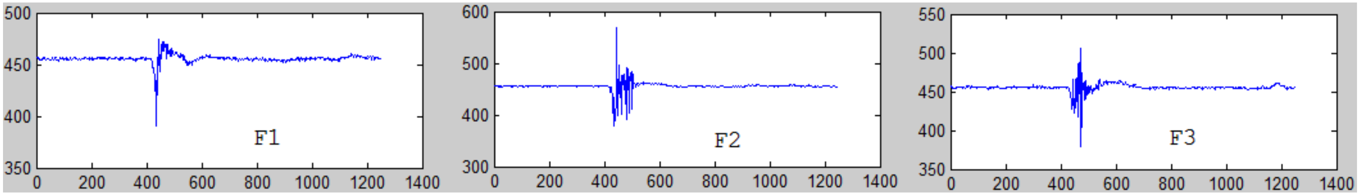


Fig. 5. Signals EMG of three gestures.

The essence of this classification method lies in the search for entropies of signals for each gesture. Based on the entropies obtained, the probability density for each gesture is calculated

$$H(\boldsymbol{x})=\sum_{m=1}^{M}p(\boldsymbol{x}_m)log_2\frac{1}{p(\boldsymbol{x}_m)} \quad , \tag{1}$$

where **x** is the row vector of the long M signal of the moment of flexion and extension of the finger selected by the window; H(**x**) is the entropy of the signal vector; $p(\boldsymbol{x}_m)$ is the probability of hits of counts **x** in the grid of amplitudes equally divided into n parts (the choice of n will be described later);

$$f(h)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2\sigma^2}(h-\mu)\right); \tag{2}$$

where f is the probability density; $\mu=E[H]$ is the expected value; $\sigma^2=E[(\mu-H)^2]$ is dispersion.

Consider an example: there are 60 experimental compression signals of the index finger, for each signal is calculated the entropy (1), resulting in a vector of 60 entropies. Further, based on these entropies the probability density (2) is calculated. This algorithm is applied to the rest of the gestures in the same order.

For training 45 measurements were taken for each gesture, and for testing 10 for each gesture and $n = 220$.

The result is the result of the probability densities shown in Figure 6 and the test result shown in Figure 7.

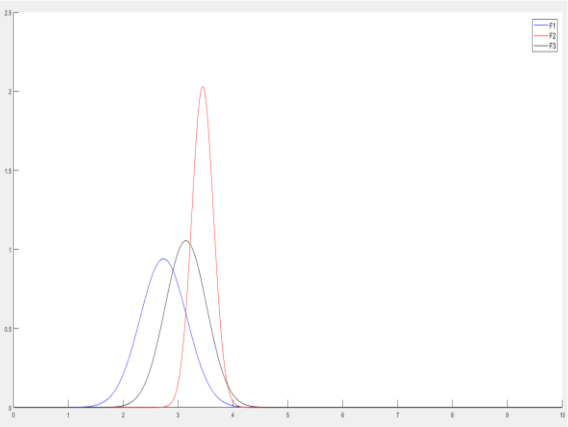


Fig. 6. Density of entropy probabilities on the training sample.

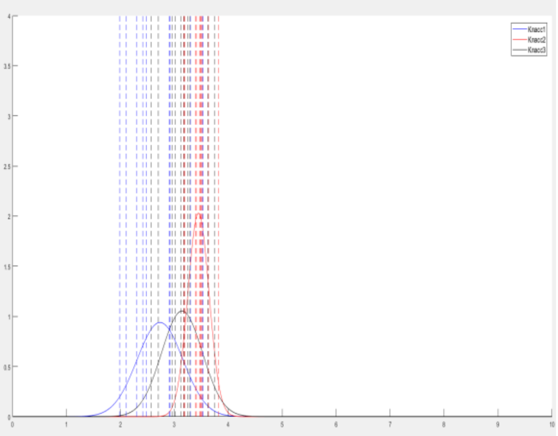


Fig. 7. Hits of the test sample.

Vertical lines show the locations of the test sample. Thus, the result of the misclassification:

– the probability of misclassification for F1 = 40%
– the probability of misclassification for F2 = 20%
– the probability of misclassification for F3 = 60%

In total, the probability of an misclassification for this method is 4 misclassifications of 10 for the first movement, 2 for the second movement, and 6 for the third. The total probability of incorrect classification is 12/30 = 40%.

4 Discussions

This method showed a poor result on one channel, and it was not possible to obtain such a good result as in [5], it is possible for this method that the number of channels for removing the EMG signal is important, as well as the high accuracy of the equipment.

Consider other classification methods: SVM, perceptron, and random tree method. For each of these methods, the signs indicated in [8] are distinguished from the signal:

– The algorithm for classifying three classes is the "one against all" method. Three classifiers are under construction.
– In training perceptron, 6 errors occurred (perceptron parameters: three hidden layers of 50 neurons in the first, 20 in the second and 5 in the third layer).
– Linear SVM was trained with one error (so that there was no retraining, SVM was allowed to err).
– The trees were trained without errors (100 random trees were used).

The results on the test data is given in Table 1.

Table 1. The results on the test data

Method:	SVM	Perceptron	Trees
Probability of misclassification:	3.3%	6.6%	3.3%

You can try to improve the result by selecting only significant features from the signal; this can be done using feature_importances_ method in Python.

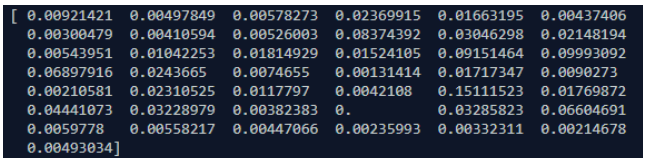
Order of features:

1) IEMG; 2) MV;
3) MAV; 4) MAV1;
5) MAV2; 6) TM4;
7) RMS; 8) V;
9) LOG; 10) DASDV;
11) MYOP; 12) WAMP;
13) MAVSLP; 14) SSI;
15) VAR; 16) STD;
17) WL; 18) AAC;
19) X; 20) ff;
21) PKF; 22) MNF;
23) MDF; 24) MNP;
25) FR; 26)SM1;
27) SM0; 28) VCF;
29) SSC; 30) FMD_vec;
31) FMN_vec; 32) MFMD_vec;
33) MFMN_vec; 34-44)LPC

The calculation of these characteristics can be seen in the open-source code [8].

In LPC is stored not one feature but a vector of 10.

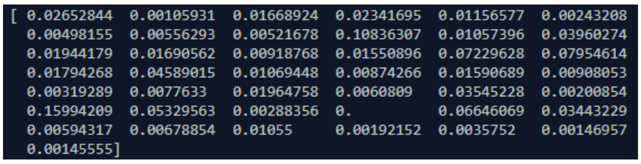
In Figures 8-10, important features for three gestures, respectively:



```
[ 0.00921421  0.00497849  0.00578273  0.02369915  0.01663195  0.00437406
  0.00300479  0.00410594  0.00526003  0.08374392  0.03046298  0.02148194
  0.00543951  0.01042253  0.01814929  0.01524105  0.09151464  0.09993092
  0.06897916  0.0243665   0.0074655   0.00131414  0.01717347  0.0090273
  0.00210581  0.02310525  0.0117797   0.0042108   0.15111523  0.01769872
  0.04441073  0.03228979  0.00382383  0.          0.03285823  0.06604691
  0.0059778   0.00558217  0.00447066  0.00235993  0.00332311  0.00214678
  0.00493034]
```

Fig. 8. Important features for the F1 movement.

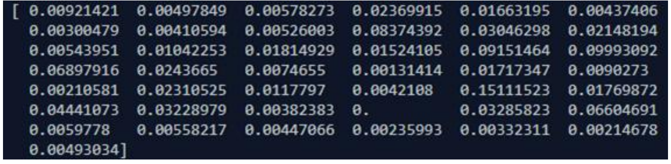
We chose a threshold equal to 0.02 and took the features above the preset threshold. Features: MAV1; DASDV; MYOP; WAMP; WL;AAC; X;ff; SM1; SSC; FMN_vec; MFMD_vec; LPC (2,3). Feature numbers: (4; 10; 11; 12; 17; 18; 19; 20; 26; 29; 31; 32; 35; 36)



```
[ 0.02652844  0.00105931  0.01668924  0.02341695  0.01156577  0.00243208
  0.00498155  0.00556293  0.00521678  0.10836307  0.01057396  0.03960274
  0.01944179  0.01690562  0.00918768  0.01550896  0.07229628  0.07954614
  0.01794268  0.04589015  0.01069448  0.00874266  0.01590689  0.00908053
  0.00319289  0.0077633   0.01964758  0.0060809   0.03545228  0.00200854
  0.15994209  0.05329563  0.00288356  0.          0.06646069  0.03443229
  0.00594317  0.00678854  0.01055     0.00192152  0.0035752   0.00146957
  0.00145555]
```

Fig. 9. Important features for the F2 movement.

Feature numbers: (1; 4; 10; 12; 13; 17; 18; 19; 20; 29; 31; 32; 35; 36)



```
[ 0.00921421  0.00497849  0.00578273  0.02369915  0.01663195  0.00437406
  0.00300479  0.00410594  0.00526003  0.08374392  0.03046298  0.02148194
  0.00543951  0.01042253  0.01814929  0.01524105  0.09151464  0.09993092
  0.06897916  0.0243665   0.0074655   0.00131414  0.01717347  0.0090273
  0.00210581  0.02310525  0.0117797   0.0042108   0.15111523  0.01769872
  0.04441073  0.03228979  0.00382383  0.          0.03285823  0.06604691
  0.0059778   0.00558217  0.00447066  0.00235993  0.00332311  0.00214678
  0.00493034]
```

Fig. 10. Important features for the F3 movement.

Feature numbers: (4; 10; 11; 12; 17; 18; 19; 20; 26; 29; 31; 32; 35; 36)

This set of features is tested on three classification methods. As a result, the probability of misclassification is shown in Table 2.

Table 2. The probability of misclassification

Method:	SVM	Perceptron	Trees
The probability of misclassification	3.3%	3.3%	0%

Having eliminated unnecessary signs, the probability of classification has improved, but it is necessary to allocate for each method its set of features. In the end, it is better to select the characteristics of the method and apply to other than not to provide the signs at all.

5 Conclusions

This study, using a single channel and cheap hardware, showed a good result on a test sample for three movements, thereby showing the ability to manufacture affordable, cheap and simple biointerfaces.

This simple and affordable hardware allows the expansion of the definition of the maximum number of well-classified gestures to increase the number of controlled impacts.

References

1. J. Yamagishi, T. Kobayashi, IEICE TRANSAC. on Information and Systems **90** 533 (2007)
2. J.A. Bilmes et al., The Vocal Joystick: A voice-based human-computer interface for individuals with motor impairments, *In Proceedings of the conference on human language technology and empirical methods in natural language processing. Association for Computational Linguistics* (2005)
3. A. Kapur, , A. Kapur, , N Virji-Babul et al., Gesture-based affective computing on motion capture data. *In International conference on affective computing and intelligent interaction* (Springer, Heidelberg. 2005)
4. V. Buchmann, S. Violich, M. Billinghurst, A. Cockburn, FingARtips: gesture based direct manipulation in Augmented Reality. *In Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia,* pp. 212-221 (ACM, 2004)
5. T.O. Zander, C. Kothe, Journal of neural engineering, **8** 025005 (2011)
6. J.R. Wolpaw, N. Birbaumer, D.J. McFarland, G. Pfurtscheller, T.M. Vaughan, Clinical neurophysiology, **113** 767 (2002)
7. K.J. You, K.W. Rhee, H.C. Shin, Experimental neurobiology, **19** 54 (2010)
8. A. Phinyomark, P. Phukpattaranont, C. Limsakul, Expert Systems with Applications, **39** 7420 (2012)