A Human-Computer Interface based on Electromyography Command-Proportional Control

Sergey Lobov[1], Nadia Krilova[1], Innokentiy Kastalskiy[1], Victor Kazantsev[1] and Valeri A. Makarov[1,2]

[1]*Lobachevsky State University of Nizhny Novgorod, Gagarin Ave. 23, 603950, Nizhny Novgorod, Russia*
[2]*Instituto de Matemática Interdisciplinar, Applied Mathematics Dept., Universidad Complutense de Madrid, Avda Complutense s/n, 28040, Madrid, Spain*

Keywords: Electromyography, Human-Computer Interface, Pattern Classification, Artificial Neural Networks.

Abstract: Surface electromyographic (sEMG) signals represent a superposition of the motor unit action potentials that can be recorded by electrodes placed on the skin. Here we explore the use of an easy wearable sEMG bracelet for a remote interaction with a computer by means of hand gestures. We propose a human-computer interface that allows simulating "mouse" clicks by separate gestures and provides proportional control with two degrees of freedom for flexible movement of a cursor on a computer screen. We use an artificial neural network (ANN) for processing sEMG signals and gesture recognition both for mouse clicks and gradual cursor movements. At the beginning the ANN goes through an optimized supervised learning using either rigid or fuzzy class separation. In both cases the learning is fast enough and requires neither special measurement devices nor specific knowledge from the end-user. Thus, the approach enables building of low-budget user-friendly sEMG solutions. The interface was tested on twelve healthy subjects. All of them were able to control the cursor and simulate mouse clicks. The collected data show that at the beginning users may have difficulties that are reduced with the experience and the cursor movement by hand gestures becomes smoother, similar to manipulations by a computer mouse.

1 INTRODUCTION

Recent years witness a rapidly growing interest to the development of devices controlled by electromyographic (EMG) signals through a human-machine interface. There have been proposed interfaces controlling personal computers (PC) (Chowdhury et al., 2013; "Myo™ Gesture Control Armband", 2013), mobile and humanoid robots (Wang et al., 2012; Lobov et al., 2015a; Lobov et al., 2015b), powered prostheses (Roche et al., 2014; Hahne et al., 2014), and exoskeletons (Kiguchi and Hayashi, 2012; Singh and Chatterji, 2013; Mironov et al., 2015), among others. Despite technical differences in the implementation, such devices in general exploit quite similar controlling strategies (see, e.g., Peerdeman et al., 2011; Roche et al., 2014).

The simplest approach uses a single-channel recording of the bioelectrical activity of a muscle and applies either proportional (gradual) (Bottomley and Cowell, 1964) or trigger-like (Kobrinskiy, 1960) transformation to generate the controlling output. Multi-channel setups allow for simultaneous treatment of the activity of several muscles and, in general, are more promising due to higher number of degrees of freedom involved in the analysis. Then, commands sent to an external device can be evaluated either by a regression over EMG signals or by a classification EMG in terms of the classical pattern recognition problem (Kiguchi and Hayashi, 2012; Roche et al., 2014).

Some of the proposed techniques have been implemented in commercially available devices. For example, the wearable bracelet MYO™ (Thalmic Labs Inc.) employs classification of five hand gestures for managing a personal computer (PC) ("Myo™ Gesture Control Armband", 2013). This bracelet, however, does not implement sEMG-based proportional control of a PC cursor. Instead, it uses measurements of spatial coordinates of a hand. This imposes restrictions for the use of the device by disabled people, e.g. by amputees.

The powered prostheses available on the market support either single channel or multichannel regression strategies generating controlling output (Roche et al., 2014). At the time being, pattern

Lobov, S., Krilova, N., Kastalskiy, I., Kazantsev, V. and Makarov, V.
A Human-Computer Interface based on Electromyography Command-Proportional Control.
DOI: 10.5220/0006033300570064
In *Proceedings of the 4th International Congress on Neurotechnology, Electronics and Informatics (NEUROTECHNIX 2016)*, pages 57-64
ISBN: 978-989-758-204-2

recognition methods have not been implemented yet due to a limited number of possible types of movements. Although the regression based methods showed important advances, there also have been revealed weak points. For example, the setup tuning procedure usually takes relatively long time and requires specific instruments for measuring the exact position of different parts of the arm (Fougner et al., 2012; Jiang et al., 2012; Hahne et al., 2014).

Artificial neural networks (ANNs) have also widely been used to solve both sEMG classification and regression problems. Their accuracy is rather high in comparison to other approaches (see, e.g., Peerdeman et al., 2011; Baspinar et al., 2013). However, the adaptation of ANNs to commercially-ready human-computer interfaces is still an open problem that also requires investigation of the user experience and potential restrictions.

In this work we describe a low-cost human-computer interface that uses multichannel sEMG signals processed by ANN. An output of the ANN is then used as a controlling signal for a PC. Thus, the gesture classification and regression are strongly overlapping processes made by the ANN. Moreover, the ANN learning can be accomplished relatively fast and requires no special measurement techniques. Using this interface a user can move a mouse cursor on a computer screen by hand movements (muscle contraction) and simulate mouse clicks. Then disabled people and amputees can use such an interface in their daily life.

2 METHODS

2.1 Subjects and Testing Task

For experimental purpose we recruited 12 healthy volunteers of either sex from 21 to 41 years old. The study complied with the Helsinki declaration adopted in June 1964 (Helsinki, Finland) and revised in October 2000 (Edinburg, Scotland). The Ethics Committee of the Lobachevsky State University of Nizhny Novgorod approved the experimental procedure. All participants gave their written consent.

After machine learning of the interface ANN (individual for each subject) all participants were asked to move remotely (using hand gestures) a PC pointer in Windows OS and to perform the following testing task: open calculator application, type "2 + 2 =", find the result and close the application. Each participant performed the testing task twice to examine two types of the learning procedures (see below). Then each participant described verbally his/her user experience.

All subjects had no previous experience with sEMG interfaces, except one who used MYO bracelet for two weeks. For this "experienced" user an additional test was developed. The user was asked to connect four points on the computer screen forming a diamond: a) by using a computer mouse and b) by the sEMG interface. Then the performance of "diagonal" movements was evaluated as

$$P = 1 - \frac{L_i - L_m}{L_m}, \qquad (1)$$

were L_i and L_m are the lengths of the curves drawn by means of the sEMG interface and computer mouse, respectively.

Figure 1: The hardware-software system MyoCursor. User with a MYO bracelet placed on forearm controls a PC. The bracelet transmits eight sEMG signals through a Bluetooth interface to the PC equipped with MyoCursor software.

2.2 Myocursor Hardware-Software Setup

Figures 1 and 2 show the developed hardware-software system, called MyoCursor. The system consists of a MYO bracelet worn on a forearm of the user and a PC with specially designed software (Fig. 1). The bracelet has eight equispaced sensors acquiring myographic signals at F = 1 kHz rate. However, our tests have shown that it cannot deliver data with the frequency above 300 Hz. Missing data are filled in by returning previously sampled values. The sEMG signals are sent through a Bluetooth interface to a PC. We use the MYO SDK to access raw eight-channel data, while the built-in software of the bracelet is disabled. Acquired signals are then processed by MyoCursor software in real-time (Fig. 2). The software performs the recognition of hand gestures and estimates the muscle efforts that finally control the cursor in Windows OS (Microsoft Inc.) in a way similar to that one can achieve with ordinary computer mouse.

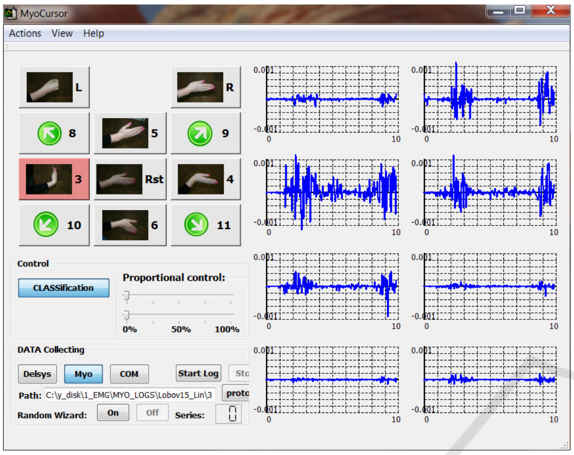

Figure 2: MyoCursor interface. The software processes in real time the sEMG signals and generates commands controlling the mouse cursor. Left-top: controlling hand gestures, red button marks the recognized gesture (wrist flexion). Right: Example of eight sEMG signals corresponding to the performed gesture (vertical and horizontal axes are in mV and s, respectively). Left-bottom: Controlling toolbars.

2.3 Basic Hand Gestures Imitating Mouse Manipulations

Natural hand gestures can be extremely rich. For the human-computer interface we have selected the following seven static hand gestures as basic motor patterns: 1) *hand at rest* is used for eliminating the cursor trend (see below); 2) *hand clenched in a fist* simulates mouse-left click; 3,4) *wrist flexion* and *extension* imitates the cursor movement to the left and to the right, respectively; 5,6) *radial* and *ulnar deviations* simulate up and down cursor movements, respectively; and 7) *extended palm* (fingers together or separately) is used for imitation of the mouse-right click. An artificial neural network (see below) should learn sEMG patterns associated with these basic motor patterns. For machine learning we adopted two procedures:

i) A user performs two series each consisting seven basic gestures.
ii) A user performs two series as in (i) and four additional gestures that are combination of pair gestures 3-6 (e.g. simultaneous wrist flexion (3) and radial deviation (5), which serves for diagonal left-up movement).

In either case the users performed each gesture during 2-3 seconds with a 2-3 seconds relaxing pause between gestures.

2.4 Signal Analysis and Neural Network

We divide in real-time the sEMG data flow, $x(t)$, into 100 ms time windows ($x(t) \in \mathbb{R}^8$). Windowing is performed every 50 ms. At this rate an artificial neural network performs calculations and provides the cursor controlling signal (Fig. 3).

At the first step the root mean square (RMS) of the EMG activity over 100 ms time window is evaluated:

$$z(t) = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} x(t-n)^2}, \qquad (2)$$

where $N = 0.1F$ is the number of samples in time window. The RMS data, as a composite feature of the current hand gesture, are fed into an ANN with one hidden layer containing eight neurons (Fig. 3, but see also Sect. 3.2). Each network neuron applies weighted sum over its inputs and uses sigmoidal activation function to generate the output, y:

$$y = F(w \cdot z),\ F(u) = \frac{1}{1+e^{-u}}, \qquad (3)$$

where $w \in \mathbb{R}^8$ is the vector of synaptic weights related to the given neuron and dot stands for inner product. The learning, i.e., adjustment of the neuronal weights $\{w\}$, is achieved by the standard back-propagation algorithm (Rumelhart et al., 1985).

For training and testing purposes we use sets containing 40-60 samples corresponding to each class. The classification error is calculated both for the training and for testing sets. It serves as a criterion for stopping the learning procedure as soon as the error starts increasing on test samples. In average the learning process on an optimized ANN requires about 5000 training epochs and takes less than 1 min on a standard Intel Core i5 PC. In case of examining different parameters of the ANN (e.g. the number of neurons in the hidden layer) the learning time changes accordingly (could be higher or lower).

For the first learning procedure, (i), each gesture corresponds to a single target class (Fig. 2). This facilitates the learning procedure since each output neuron should produce binary output: 1 for its own

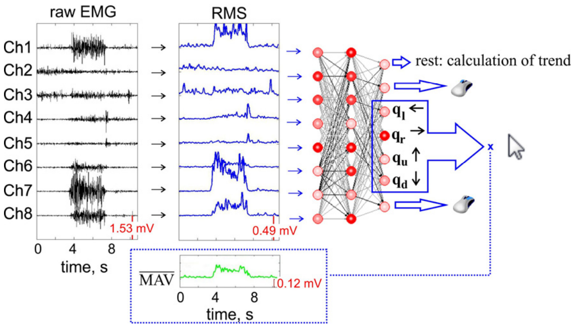

Figure 3: Data flux in the MyoCursor system. Raw sEMG activity is mapped into cursor movements and mouse clicks in Windows OS. First RMS and MAV activity is evaluated in a 100 ms time window. The RMS pattern is fed into the input layer of an artificial neural network with one hidden layer. Every 50 ms the network output from seven neurons provides two commands for mouse-like clicking and four commands for cursor movements. The latter are multiplied by the MAV to gain the cursor speed.

class and 0 for the others. To accommodate compound gestures added in the second learning procedure, (ii), we used the target value $1/\sqrt{2}$ for the two output neurons participating in the corresponding compound gesture. Such choice of the target value ensures the generation of a compound vector output with unitary length when both neurons are activated (i.e., adding two orthogonal vectors of this length results in a unitary length vector).

2.5 Proportional Control of Cursor

Once the learning is deemed finished, online controlling of the Windows interface can be enabled. The cursor movement along the *X*-axis (*Y*-axis) is proportional to the difference of the output neurons responsible for the gestures "left" and "right" ("up" and "down", Fig. 3). This difference is a step-like function, which is not optimal for the cursor manipulation. To introduce a proportional control we employ an approach similar to that described by Lobov et al. (2015b).

We estimate the muscle effort by evaluating the mean absolute value (MAV) averaged over all EMG sensors:

$$A(t) = \frac{1}{NK}\sum_{k=1}^{K}\sum_{n=0}^{N-1}|x_k(t-n)|, \qquad (4)$$

where K is the number of sEMG channels (in our case K = 8). Then the cursor speed can be set proportional to the MAV (Fig. 3). However, due to some intrinsic jitter in the muscle tone we usually have observed a slow involuntary cursor drift. To eliminate this artifact, the trend defined by relaxed hand state is subtracted from the cursor controlling signals. Thus, we define the cursor velocity by:

$$v(t) = (A(t) - A_{th})H(A(t) - A_{th}) \qquad (5)$$

where H is the Heaviside step function and A_{th} is the drift threshold, corresponding to $A(t)$ evaluated over time intervals with hand at rest.

Finally, the cursor displacement, Δ, along the *X*- and *Y*-axes is given by:

$$\Delta_x = \frac{v}{5}(q_r - q_l), \;\; \Delta_y = \frac{v}{5}(q_u - q_d), \qquad (6)$$

where q_r, q_l, q_u, and q_d are the network output (Fig. 3) corresponding to the gestures “move right”, “move left”, “move up”, and “move down”, respectively.

3 RESULTS

To perform the testing task described in Methods a user should be able to move the PC cursor on the screen and to simulate clicks of mouse buttons. We then implemented the hardware-software setup that replaced a physical computer mouse by a virtual pointer controlled by the human-computer interface based on sEMG signals (Fig. 1).

3.1 Mouse Clicks

We associated the left and right clicks of “mouse buttons” to two single hand gestures (see Methods). The gesture selection was not trivial. Indeed, the gestures assigned to the clicks must differ significantly from gestures for cursor movements. Otherwise, the ANN may confuse them, which can significantly diminish the user experience (usually clicks should be done at precise cursor positions).

At the beginning we performed experiments using nine gestures, including besides those described in Methods the *supination* and *pronation*. Then the latter two were refused since supination was badly recognized due to the electrode localization (around forearm), while pronation was confused sometimes with the forearm flexion.

To optimize the ANN performance we ran the process of gesture recognition on the same set of data varying the number of neurons in the hidden layer of the ANN and the learning rate. Figure 4 shows the results. The ANN error drops significantly between 4 and 8 neurons and then stays unchanged, while the learning time increases (Fig. 4A). Thus, we selected a network with eight hidden neurons for further experiments. We also observe that the learning rate 0.01 optimizes both the learning error and the learning time. Thus, this value has been used in all experimental tests of the interface.

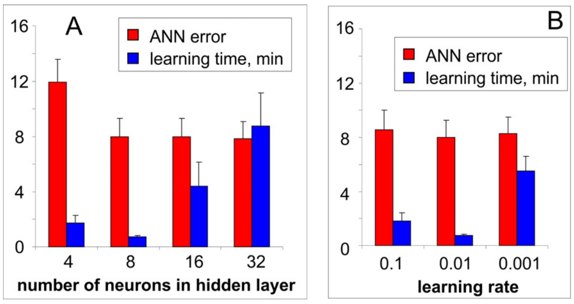

Figure 4: Performance of the artificial neural network (mean squared error and learning time) at classifying hand gestures with different number of neurons in the hidden layer (A) and different values of the learning rate (B). In case (A) the learning rate was set to 0.01. Error bars show the standard error.

3.2 Cursor Movement: Naïve Approach

A naïve approach to control the cursor movements can be implemented by the event coding similar to that used for the mouse clicks described above. We can set the speed of the cursor movement to a constant. Then the user will use gestures to start and stop the movement. The drawback of such a strategy resides in the inevitable trade off between the speed (responsiveness of the interface) and the accuracy of the cursor movement. Our experiments have shown that this strategy significantly downgrades the user experience. Nevertheless, we used these data to evaluate the classification accuracy that can be achieved in real tasks (Table 1, column "rigid classes"). Our results confirmed that the mean ANN error (2.6± 0.7%) is low enough for implementing the sEMG interface.

Table 1: Gesture classification error and time of execution of the user task (see Methods). User's body types: asthenic (a), normosthenic (n), and hypersthenic (h).

Subject number; sex; body type; age (years)	Classification error, %		Task execution time, s	
	rigid classes	fuzzy classes	rigid classes	fuzzy classes
1 male n 28	0.4	6.6	46	56
2 female n 28	2.1	7.3	83	110
3 male n 41	3.8	4.8	44	71
4 female n 40	0.8	6.7	64	300
5 male n 28	1.1	8.5	55	235
6 female n 21	1.2	8	76	207
7 male n 35	0.7	4.8	47	60
8 female h 21	7.8	12	169	257
9 female a 26	2.2	5.8	103	109
10 female n 28	1.4	3.1	113	179
11 male n 21	5.6	9.7	113	113
12 female h 23	4.6	5.8	120	100
mean ± s.e.	**2.6±0.7**	**6.9 ± 0.7**	**86 ± 11**	**150 ± 23**

3.3 Proportional Control of Cursor

As abovementioned, to achieve a flexible cursor movement we aim at a combined command-proportional control with two degrees of freedom. In this case the cursor movement direction is defined by gestures, while its speed is controlled by the degree of muscle contraction (MAV), which is almost equivalent to the palm angle. This may significantly improve the user experience.

Experiments conducted with twelve subjects showed that all users were able to move the cursor and successfully simulate left and right mouse clicks. Then we studied the performance in cases of using rigid and fuzzy classes (see also Sect. 2.3).

3.3.1 Rigid Classes

After the network training with rigid correspondence between hand gestures and cursor movements, all users managed to control the cursor. Performing the testing task (Sect. 2.1) took from 44 s to 169 s depending on the user with the mean 86 ± 11 s (Table 1, "rigid classes"). Nevertheless, after the test users reported a number of repetitive difficulties: i) Performing the task using the sEMG-interface was much harder than using a physical mouse. For comparison, the same test performed by using a hardware mouse was 10 times faster in average; ii) Diagonal cursor movements, requiring simultaneous displacement along the X and Y axes, were significantly more difficult than movements involving one axis only.

3.3.2 Fuzzy Classes

An important limiting factor of the "rigid classes" scheme resides in the intrinsic feature of the standard neural network approach, i.e., sharp boundaries among classes. It leads to the "winner takes all" phenomenon and difficulties in smooth controlling the cursor movement. The cursor usually follows a steps-like trajectory advancing in X or Y directions separately instead of a smooth diagonal movement (Fig. 5, green curve). To overcome this problem we introduced fuzzy class overlapping (see Methods).

The implementation of fuzzy classes indeed facilitated the diagonal movements of the cursor.

However, our tests showed that only 4 out of 12 users found this way better than using the rigid classes approach. In average the error of gesture identification increased to 6.9 ± 0.7% (Table 1). Moreover, the testing task execution time significantly increased to 150 ± 23 s. Subjectively this performance downgrade the users explained by the need of making unnatural gestures. For example, simultaneous wrist extension and ulnar deviation (required for the diagonal right-down movement) were reported as a pattern complex to perform. Then, an increase in wrong classification of compound gestures leaded to the cursor movement in wrong direction. This, in turn, increased the test time. Table 2 summarizes the subjective user experience and comparison of both schemes.

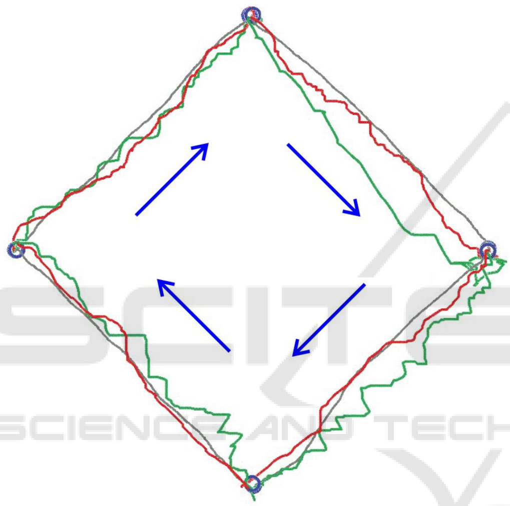

Figure 5: Representative example of line drawing by an experienced user. The task consists in connecting blue circles by a cursor by following directions shown by blue arrows. Grey, green, and red curves mark cursor traces corresponding to the use of a physical mouse, sEMG with ridged classes, and sEMG with fuzzy classes interfaces, respectively.

3.3.3 Performance of Experienced User

Since the results of experimental tests were quite unexpected, we hypothesized that the inconvenience of working with the sEMG-interface with fuzzy classes might be explained by the absence of the experience of dealing with such an interface. Indeed, all subjects were used to common mouse interface, while working with sEMG may require some preliminary practice. Thus, we selected one of the users and asked him to work with the sEMG-interface regularly during two weeks. Then we repeated the testing task.

Figure 5 shows the drawing made by this user employing three different interfaces: 1) Common mouse (grey curve); 2) sEMG with rigid classes (green curve); and 3) sEMG with fuzzy classes (red curve). As expected, the training significantly improved the sEMG performance. Taking the performance of the mouse interface as 100%, we obtained 75.1% for the "diagonal" performance (see Methods) by using MyoCursor with rigid classes and 92.5% for MyoCursor with fuzzy classes. Thus, training may improve significantly the user experience and the user may approach the performance close to the mouse interface.

Table 2: Subjective user experience with different types of interfaces.

N	User comments	
	Preferred method	Critical remarks
1	rigid	- Cursor drift - Direction of movement coincides badly with the desired direction - Mouse clicks are difficult because of the high threshold
2	fuzzy	Clicks provoke cursor jumps
3	fuzzy	Fuzzy method is easier to move the cursor diagonally
4	rigid	–
5	rigid	–
6	rigid	"Right-down" movement is confused with plain "down"
7	rigid	- "Right-down" is badly detected - 2nd test is done with a tired hand
8	fuzzy	- "Left" movement is confused with plain "down" and "rest" - If the hand is not relaxed before click, then the cursor goes down
9	rigid	"Right-up" is confused with plain "right"
10	fuzzy	- "Right-down" is confused with plain "down" - Clicks are complicated
11	rigid	- "Right-down" is confused with plain "down" - Clicks are complicated
12	rigid	- "Right-down" is confused with plain "down"

4 DISCUSSION

In this work we have proposed a human-computer interface based on a real-time recording and processing of the surface electromyographic signals. The interface allows controlling a PC with Windows OS by natural hand gestures. The signal acquisition has been implemented through an easy wearable commercially available sEMG bracelet. This, together with simplified software learning procedure, enables building low-budget and user-friendly sEMG solutions that may also be useful for disabled people and amputees.

The main difference in the algorithmic part of our approach with existing methods based on the regression techniques (Roche et al., 2014; Hahne et al., 2014; Fougner et al., 2014) is the use of an artificial neural network performing the gesture classification. The ANN is trained at the beginning by a relatively small set of simple hand gestures: seven or eleven gestures depending on the method type. This allows avoiding long lasting tuning process common for the regression approaches, which stems from gradual sampling of changes of muscle tension in different movements and their combinations. Once the ANN has been trained, it can detect commands for simulating the right and left mouse clicks, and for moving cursor on the PC screen. Using an estimate of the mean muscle effort we have implemented a proportional control of the cursor movement. Thus, the user can easily change the cursor velocity and hence the movement precision by “applying” more or less effort to the gesture.

We have tested the method on twelve healthy subjects of either sex. To do it we implemented two types of the cursor controlling strategies: “rigid” classes with four individual gestures for moving right, left, up, and down; and "fuzzy" classes with additional compound gestures for diagonal movements. In both cases all subjects were able to control cursor. Our experience suggests that the fuzzy approach is potential preferable (see Fig. 5). However, the experimental results have shown that in average the controlling performance decreases for this approach, despite a theoretically attractive possibility to move the cursor diagonally.

The subjective evaluation of the user experience has suggested that, on the one hand, the performance reduction can be linked with the requirement to perform unnatural gestures (for example simultaneous wrist extension and ulnar deviation). On the other hand, we can anticipate that in the fuzzy case there may exist a competition in the output layer of the ANN, which may have negative influence on the cursor controlling function. Thus, we can alert the reader on the necessity of future research involving optimization of the set of gestures and the ANN architecture.

In the present study specific features of the users (e.g. the degree of fitness) have been left out due to small size of the data set. However, the collected data allow us foreseeing that the type of constitution may play an important role in the success of the human-computer interface. For example, Table 1 suggests that hypersthenics may show worst results, though statistically significant conclusions require additional experiments.

Another point for discussion is the user readiness to a specific control of a PC by gestures. It is worth noting that all subjects had no previous experience in the use of such type of interfaces, while all of them used the common mouse interface in their daily life. Therefore, fair comparison between the mouse and gesture types of interfaces requires either special sampling over subjects (for example the use of elderly, with no experience with PC) or training subjects to use the MyoCursor system before testing. An experiment with one user has shown that the user training may improve significantly the ability to use the fuzzy sEMG-interface in such a way that its performance may approach the performance of the mouse interface (92% vs 100% performance reached in the test).

ACKNOWLEDGEMENTS

This work was supported by the Russian Ministry of Education and Science under the Federal Program (unique identification number RFMEFI58114 X0011).

REFERENCES

Baspinar, U., Varol, H. S., & Senyurek, V. Y. (2013). Performance comparison of artificial neural network and Gaussian mixture model in classifying hand motions by using sEMG signals. *Biocybernetics and Biomedical Engineering*, 33(1), 33-45.

Bottomley, A. & Cowell, T. (1964). An artificial hand controlled by the nerves. *New Scientist*, 21, 668-71.

Chowdhury, A., Ramadas, R., & Karmakar, S. (2013). Muscle computer interface: A review. In A. Chakrabarti & R. V. Prakash (Eds.), *ICoRD’13, Lecture Notes in Mechanical Engineering* (pp. 411-421). Springer India.

Fougner, A., Stavdahl, O., Kyberd, P. J., Losier, Y. G., & Parker, P. A. (2012). Control of upper limb prostheses: terminology and proportional myoelectric control – a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *20*(5), 663-667.

Hahne, J. M., Biessmann, F., Jiang, N., Rehbaum, H., Farina, D., Meinecke, F. C., Muller, K. R., & Parra, L. C. (2014). Linear and nonlinear regression techniques for simultaneous and proportional myoelectric control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *22*(2), 269-279.

Jiang, N., Vest-Nielsen, J. L., Muceli, S., & Farina, D. (2012). EMG-based simultaneous and proportional estimation of wrist/hand kinematics in uni-lateral trans-radial amputees. *Journal of Neuroengineering and Rehabilitation* (9).

Kiguchi, K. & Hayashi, Y. (2012). An EMG-based control for an upper-limb power-assist exoskeleton robot. *IEEE Transactions on Systems; Man, and Cybernetics, Part B: Cybernetics*, *42*(4), 1064-1071.

Kobrinskiy, A. E. (1960). Bioelectrical control of prosthetic devices. *Herald of the Academy of Sciences USSR*, 30, 58-61.

Lobov, S. A., Mironov, V. I., Kastalskiy, I. A., & Kazantsev, V. B. (2015a). A spiking neural network in sEMG feature extraction. *Sensors*, *15*(11), 27894-27904.

Lobov, S. A., Mironov, V. I., Kastalskiy, I. A., & Kazantsev, V. B. (2015b). Combined use of command-proportional control of external robotic devices based on electromyography signals. *Sovremennye tehnologii v medicine [Modern Technologies in Medicine]*, *7*(4), 30-38.

Mironov, V. I., Lobov, S. A., Kastalskiy, I. A., & Kazantsev, V. B. (2015). Myoelectric control system of lower limb exoskeleton for re-training motion deficiencies. *Lecture Notes in Computer Science*, *9492*, 428-435.

Myo^{TM} Gesture Control Armband - Wearable Technology by Thalmic Labs (2013). *Thalmic Labs Inc.* Retrieved May 26, 2016, from www.myo.com

Peerdeman, B., Boere, D., Witteveen, H., Huis in 't Veld, R. H., Hermens, H., Stramigioli, S., Rietman, H., Veltink, P., & Misra, S. (2011). Myoelectric forearm prostheses: State of the art from a user-centered perspective. *Journal of Rehabilitation Research and Development*, *48*(6), 719-738.

Roche, A. D., Rehbaum, H., Farina, D., & Aszmann, O. C. (2014). Prosthetic myoelectric control strategies: A clinical perspective. *Current Surgery Reports*, *2*(3), 1-11.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). Learning internal representations by error propagation. *Parallel Distributed Processing* (pp. 318-362). San Diego: La Jolla Institute for Cognitive Science, California University.

Singh, R. & Chatterji, S. (2012). Trends and challenges in EMG based control scheme of exoskeleton robots - a review. *International Journal of Scientific & Engineering Research*, 3(8), 1-8.

Wang, B., Li, Z., Ye, W., & Xie, Q. (2012). Development of human-machine interface for teleoperation of a mobile manipulator. *International Journal of Control, Automation and Systems*, 10(6), 1225-1231.