ITM Web of Conferences **15**, 02003 (2017) DOI: 10.1051/itmconf/20171502003
CMES'17

Classifier testing for the brain-machine interface (BCI) based on Steady State Visually Evoked Potential (SSVEP)

Arkadiusz Kubacki[1,*], and *Arkadiusz* Jakubowski[1]

[1]Poznan University of Technology, Institute of Mechanical Technology, Skłodowska-Curie Square 5, 60-965 Poznań, Poland

Abstract. The paper describes the research on the classifiers for brain-computer interface (BCI) based on Steady State Visually Evoked Potential (SSVEP). Authors presented research on the checking the usability of classifiers for recognizing an EEG signal during the stimulus. Three classifiers have been checked: Support Vector Machine (SVM), Linear Discriminant Analysis (LDA) and one based on Artificial Neural Network (ANN). First part is concentrated on brain-computer interfaces and classification of them. The second part describes algorithms of all using classifiers. In the next part, authors present test stand and how the experiment is built. The last part consists of results of these tests. The best was the classifier based on Artificial Neural Network – up to 95% of correct identified. The worst results were obtained from Support Vector Machine – about 70%.

1 Introduction

Nowadays, people cannot imagine life without computers. So, it is normal that they are used to newer devices to control them. At present, two main trends are observed. These are vision system interfaces and interfaces based on human biosynthesis. In the first area, people use various cameras, such as Microsoft Kinect [1], to track and recognize some gestures or body movements. In the other area, the most popular are interfaces using electromyographic (EMG) sensors like the Myo armband [2], which can recognize hand and arm gestures using the electrical activity of forearm's muscles or brain-computer interfaces (BCI), which can recognize some activities directly from the brain [3]. There are many methods which can be used to monitor brain activity [4]:
- Electroencephalography (EEG),
- Positron emission tomography (PET),
- Functional magnetic resonance imaging (fMRI),
- Magnetoencephalography (MEG).

Owing to decreasing prices of headsets, BCIs based on EEG are developing rapidly [5]. EEG is a non-invasive method used to record the activity of brain from the examined person skull through the electrodes. Currently, the most popular and most commonly used system of electrodes placement is a "10-20" system [6]. This is an international system to describe and apply the location of scalp electrodes in the context of an EEG test or experiment. This method was created to guarantee standardized reproducibility. Thus, the EEG results of different people can be compared to each other. This system is based on a relationship between a location of an electrode and an underlying area of cerebral cortex. The name of this system refers to the fact that actual distances between adjacent electrodes are either 10% or 20% of the total front–back or right–left distance of the skull. The electrodes placement in this system is shown in Fig.1. All electrodes are labelled with a capital letter and a number. The letter refers to the area of brain e.g. F – Frontal lobe and T – Temporal lobe. Moreover, even numbers refer to the right side of the head and odd numbers the left side.

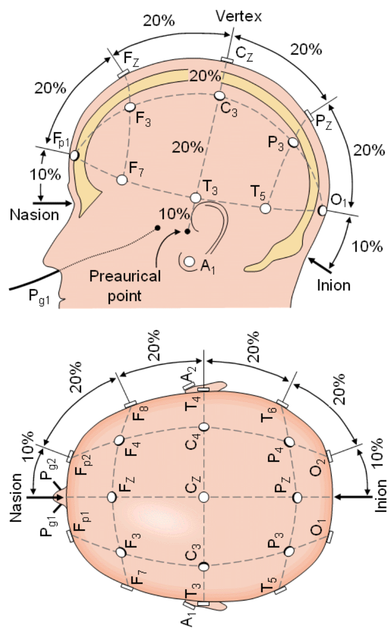

Fig. 1. Electrode placement in accordance with the "10-20" system [6].

---
[*] Corresponding author: arkadiusz.j.kubacki@doctorate.put.poznan.pl

DOI: 10.1051/itmconf/20171502003

1.1 Classification of BCIs

Currently, the basic method for brain-computer interfaces is electroencephalography, as it provides an opportunity to observe changes in brain activity in a short period of time. Unfortunately, EEG caps have poor spatial resolution and signal susceptibility to interference by artefacts from both, facial expressions and other activities of the body [7]. In the article [8], authors have shown how to find artefacts in the electroencephalogram using artificial neural network. Fig. 2 shows currently used methods to measure brain activity. Passive methods use external stimuli to induce brain activity. In active methods, human brain activity is caused through adequate imaginary [7].

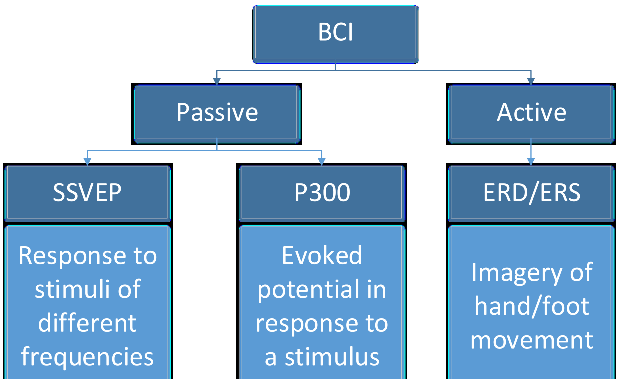

Fig. 2. Classification of BCIs [7].

Using only one method limits the user, especially in the number of possible commands and the effectiveness of their detection. It is possible to use used hybrid BCI. Their classification is shown in Fig. 3.

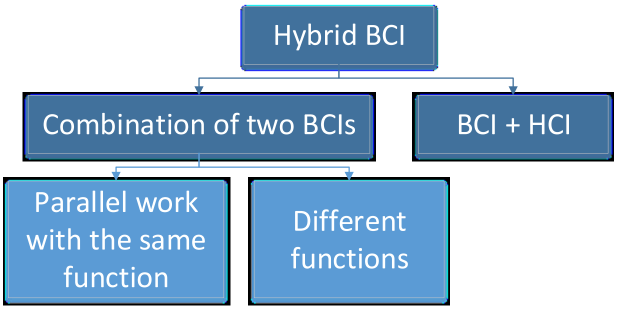

Fig. 3. Classification of hybrid BCIs [7].

Furthermore, more than only one type of interface can be used to increase the accuracy of the classification commands at parallel connection or increase the number of possible commands. BCI method can be also linked with other interfaces, such as speech recognition or eye movement observation, to make hybrid method. The aim is to have more accurate commands recognition and expanding their numbers [7].

1.2 Steady-State Visually Evoked Potential (SSVEP)

Steady-State Visually Evoked Potential (SSVEP) is a periodic call of evoked potentials through repeated visual stimulation. Usually, the frequency of blinking is more than 6Hz [9]. Flashing light is commonly used as a stimulating element. It is possible to use a LED diode or bulb or monitor with blinking point or chessboard [10]. Corresponding to the stimulation frequency and their harmonics, in an EEG signal appears the same frequency [11]. Therefore, a person without prior training can use this method for testing, which is the most important advantage of this method. Fig. 4 shows the Fast Fourier Transformation of signal taken from electrode called Oz, during initial testing of the SSVEP method. In Fig. 4a, the FFT is shown while the LED is not blinking and in Fig. 4b, the FFT of the same electrode output signal is shown when white LED is flashing with frequency 15 Hz.

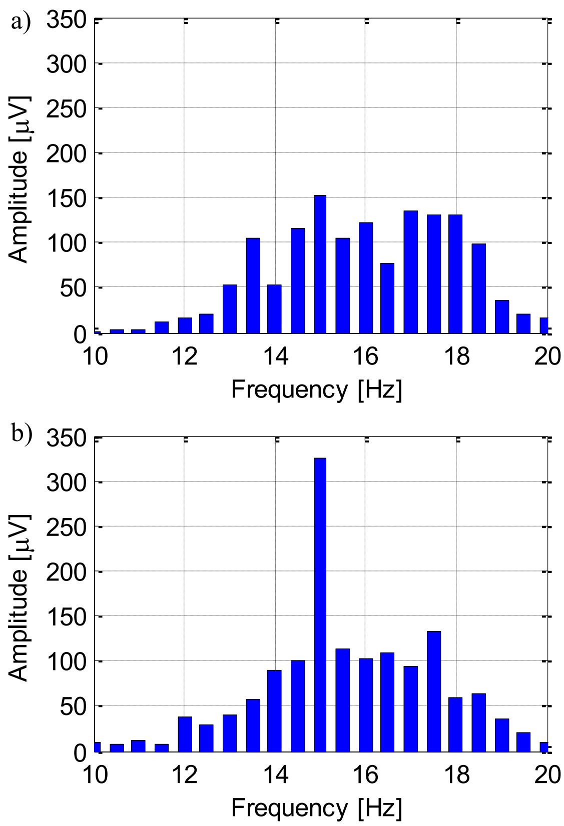

Fig. 4. Electrode Oz output signal FFT characteristics for: no blinking (a), LED blinking with 15 Hz (b).

1.3 State of art

Nowadays, brain-computer interfaces are built worldwide. In this state of art, authors present only this BCI which structure is based on Steady State Evoked Potential. Articles mentioning another two methods, shown in Fig. 2., are omitted.

In the article [11] authors created a speller based on Steady-State Visually Evoked Potential. Evoked potentials and stimulating subjects, by stimuli of frequencies from 6.66 to 8Hz, were indispensable for that purpose. The monitor with five squares was used. Three of them were used to divide the alphabet into 3 subgroups. On next level, a group of nine signs was divided into next 3 subgroups. On the last level, a user chose one letter. The other squares were used to delete letter and repeat the last action.

The authors of the article [12] were using the SSVEP to control prosthetic hand. Four kinds of commands can be invoked by the blinking field on the screen. Fields had a rate of 6Hz for the command "turn the hand to the left", 7Hz for the command "turn the hand to the right", 8Hz for the command "open the hand", 13Hz for the last command "close the hand".

In the article [5] authors presented the possibility of SSVEP to control a virtual car in virtual reality. Two HUD screens were used to display images of frequencies, 12 and 13Hz respectively for the command "turn left" or "turn right". Respondents got around previously designed route.

The authors of the article [13] used the SSVEP method to build hybrid BCI. They combined the SSVEP and MI task to play Tetris game. The SSVEP method was responsible for the left and right rotation of blocks and pause the game. The MI tasks were responsible for the left and right movement of blocks.

2 Classifiers

Three classifiers were considered in this paper: Linear Discriminant Analysis (LDA), Support Vector Machine (SVM) and Artificial Neural Network (ANN). The following subsections describe their algorithms.

2.1. Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a very important classifier, which can be used for wide variety of problems. For instance, in different machine learning problems, such as pattern or face recognition, feature extraction and data dimensionality reduction [14].

In this method, the ratio of between-class variance to the within-class variance in any data set is maximized, what guarantees maximal separability [15].

In this article, authors used LDA to two-class problem. In first step, the data sets were formulated (1).

$$set1 = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \cdots & \cdots \\ a_{m1} & a_{m2} \end{bmatrix} \quad set2 = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ \cdots & \cdots \\ b_{m1} & b_{m2} \end{bmatrix} \tag{1}$$

The equation (2) was used to compute the mean of each data set and mean of entire data set, where μ1 was mean of set1 and μ2 mean of set2, p1 and p2 were the a priori probabilities of the classes. In the case of this simple two-class problem, the probability factor was assumed to be 0.5.

$$\mu_3 = p_1 \times \mu_1 + p_2 \times \mu_2 \tag{2}$$

Within-class and between-class scatter in LDA method were used to formulate criteria for class separability. Covariance of each of the classes was the expected for within-class scatter. The scatter measures were computed using (3).

$$S_w = 0.5 \times cov_1 + 0.5 \times cov_2 \tag{3}$$

The following equation was used to compute covariance matrix (4).

$$cov_j = (x_j - \mu_j)(x_j - \mu_j)^T \tag{4}$$

The between-class scatter was computed using the equation (5).

$$S_b = \sum_j (\mu_j - \mu_3) \times (\mu_j - \mu_3)^T \tag{5}$$

As defined earlier, the optimizing criterion in LDA was the ratio of between-class scatter to the within-class scatter. Maximizing this criterion defined the axes of the transformed space. The class-dependent transform of the optimizing criterion was computed using equations (4) and (5). If the LDA is a class dependent type, all lasses should have separate optimizing criterion for each class. The equation (6) was used to compute the optimizing factors in case of class dependent type.

$$criterion_j = inv(cov_j) \times S_b \tag{6}$$

The optimizing criterion for the class independent transform was computed (7).

$$criterion = inv(S_w) \times S_b \tag{7}$$

Having obtained the transformation matrices, the data sets should be transformed using the single LDA transform or the class specific transforms whichever the case may be. The decision region in the transformed space was a solid line separating the transformed data sets. For the class dependent LDA – (8) and for the class independent LDA – (9).

$$transformed\_set\_j = transform_j{}^T \times set\_j \tag{8}$$

$$transformed\_set = transform\_spec^T \times data^T \tag{9}$$

When the transformations were completed, Euclidean distance or RMS distance was used to classify data points. Euclidean distance was computed using (10), where μ_{ntrans} was the mean of the transformed data set. Thus, for classes Euclidean distances were obtained for each test point.

$$dist_n = transform_{n_{spec}}{}^T \times x - \mu_{ntrans} \tag{10}$$

The smallest Euclidean distance among all distances classified the test vector as belonging to class n [14].

2.2 Support Vector Machine (SVM)

Support Vector Machines (SVMs) are a set of related methods for supervised learning, applicable to both classification and regression problems. SVM classifier was introduced by Vapnik in 1995. In this article, authors presented two types of SVM – C-SVM and Nu-SVM [17].

2.2.1 C-Support Vector Classification

In first step C-SVM solves the following (11) optimization problem.

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \tag{11a}$$

subject to

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, l, \tag{11b}$$

In this equation $\phi(x_i)$ maps x_i into a higher-dimensional space and $C > 0$ is the regularization parameter. It is possible that the vector variable w will be high dimensionality. Usually, this problem is solved by equation (12).

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \tag{12a}$$

subject to

$$y^T \alpha = 0,\ 0 \leq \alpha \leq C, i = 1, \dots, l \tag{12b}$$

In this equation $e = [1, \dots, l]^T$ is the vector of all ones, Q is an l by l positive semidefinite matrix, $Q_{ij} \equiv y_i y_j K(x_i x_j)$, and $K(x_i x_j) \equiv \phi(x_i)^T \phi(x_j)$ is the kernel function. After solving (12), the optimal w satisfies

$$w = \sum_{i=1}^{l} y_i \propto_i \phi(x_i) \tag{13}$$

Equation (14) is a decision function.

$$sgn(w^T \phi(x) + b),\ sgn\left(\sum_{i=1}^{l} y_i \propto_i K(x_i, x) + b\right) \tag{14}$$

2.2.2 Nu-Support Vector Classification

In first step Nu-SVM solves the following (15) optimization problem.

$$\min_{w,b,\xi} \frac{1}{2} w^T w - \nu\rho + \frac{1}{l} \sum_{i=1}^{l} \xi_i \tag{15a}$$

subject to

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \dots, l, \tag{15b}$$

Usually, this problem is solved by equation (16).

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha \tag{16a}$$

subject to

$$y^T \alpha = 0,\ 0 \leq \alpha \leq \frac{1}{l},\ \ i = 1, \dots, l, e^T \alpha \geq \nu \tag{16b}$$

In this equation $Q_{ij} \equiv y_i y_j K(x_i x_j)$. Problem (16) is feasible if and only if (17)

$$\nu \leq \frac{2 min(\#y_i = +1, \#y_i = -1)}{l} \leq 1 \tag{17}$$

Equation (18) is decision function.

$$sgn\left(\sum_{i=1}^{l} y_i \propto_i K(x_i, x) + b\right) \tag{18}$$

2.3 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) used in this classifier is Multilayer Perceptron (MLP). ANN consists of at least two layers, which means that in this method only one hidden layer is used. The number of neurons on this layer is changeable. Inside the hidden layer, the hyperbolic tangent is used as the activation function. Backpropagation algorithm is used to teach the ANNs. It is one of the most popular methods of teaching ANN [18]. The aim is to minimize the error E between the network outputs O and the target values T in step t, using the method of steepest descent in a network. In the equation (19) S stands for training samples and N for the number of neurons in the output layer [18, 19].

$$E(t) = \frac{1}{2} \sum_{s=1}^{S} \sum_{n=1}^{N} \left(T_s^n(t) - O_s^n(t)\right)^2 \tag{19}$$

In this method, the idea is to iterate to the moment when error E decreases. In the first step of teaching algorithm, all weights of network w are set on small random numbers and a low learning rate η is selected. In the next step, the input x is propagated forward through the ANN. The computed values for each output O are being saved. The errors are being propagated backwards through the network. For every output of the network, an error δ_k is being computed (20).

$$\delta_k = T_k - O_k \tag{20}$$

The equation (21) is used to calculate error δ_h for each hidden neuron, where i_h stands for input of the activation function from earlier units.

$$\delta_h = f'(i_h) \sum_{k \in outputs} w_{kh}\, \delta_k \tag{21}$$

The equation (22) is used to update the weights w_{ji} in all layers.

$$w_{ji}(t+1) = w_{ji}(t) + \eta \delta_j x_{ji} \tag{22}$$

Designing a neural network structure, consisting of several layers and the quantity of neurons in each layer, may be convoluted. At this moment, it is worth mentioning the Kolmogorov's theorem, in accordance with every continuous real function $f(x_1, \dots, x_n)$, which could be approximated using the function (23).

$$f(x_1, \dots, x_n) = \sum_{j=1}^{2n+1} g_j \left(\sum_{i=1}^{n} \phi_{ij}(x_i)\right) \tag{23}$$

Corresponding with the Kolmogorov's theorem, ANN structure has two layers with n inputs, 2n+1 neurons in the hidden layer and one output neuron with a linear activation function ϕ. There are very few problems requiring more than two hidden layers.

3 Equipment and experiment

In this paper, authors presented study conducted with the use of a neuroheadset called Emotiv EPOC+, which related to a computer via Bluetooth (Fig. 5). This device consisted of 14 EEG channels + 2 reference electrodes, which were placed in accordance with the "10-20" system (described before). Emotiv EPOC+ device uses one 16-bit ADC with a sampling rate of 256 samples per second (2048 Hz internal). This device could recognize all five main brain rhythms since it received brain waves in a bandwidth of 0.16 to 43 Hz [20].

Fig. 5. Emotiv EPOC+ headset [19].

Authors have built a special light with dimensions of 45x45x45 mm. Inside of it, there was a 3W Power LED (Fig. 6). Thanks to the RGB LED, there is possible to flash a light in different colours. A control module based on the Atmega 328P microcontroller has been developed, allowing to set the diode frequency between 5 and 30 Hz, and set the brightness of the diode in the range of 0 to 100%.

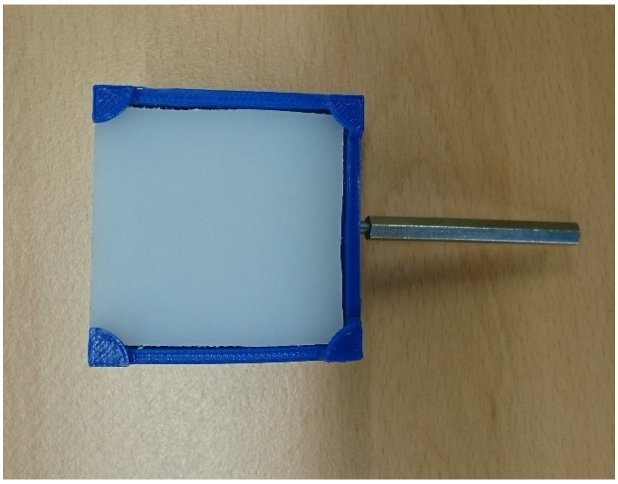

Fig. 6. Blinking module used in tests.

Authors created two scenarios in OpenVibe. The scheme of the program is shown in Fig. 7. In the first step, the input signal is filtered to the set frequency – 15Hz, 17Hz and 19Hz. Then, in the next block, an appropriate channel for the test is selected. For SSVEP, the part of the brain responsible for the sight is used; as it is placed on the back of the head, the Oz channel should be selected. Time based epoching (TBE) block was 1s.

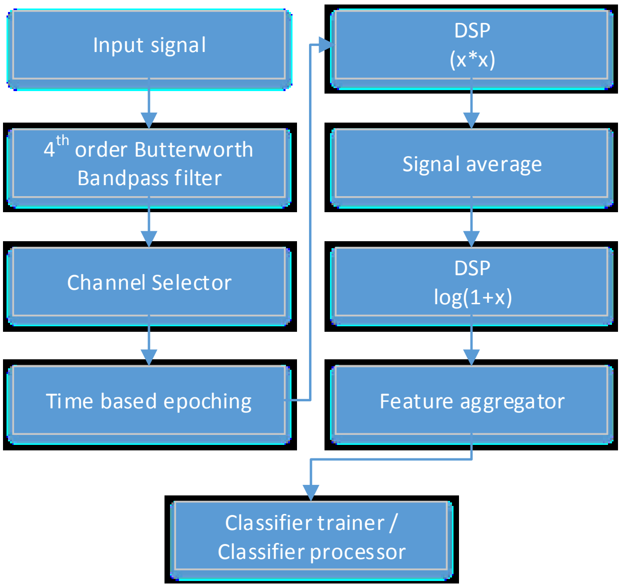

Fig. 7. Block diagram of signal processing during the experiments, made in OpenVibe system.

In the next step, the signal is raised to the power of 2, averaged, then logarithmically and outputted as Power of EEG signal. The last block before Classifier is a Feature aggregator. This block aggregates the features received on its inputs into a feature vector. Finally, signal go to the Classifier block. In the first one authors teach selected classifier with chosen settings. These settings are described in next chapter. The classifier has always been taught using the same input data. Data was collected in previous sessions of EEG test. Authors recorded data for frequency 15Hz, 17Hz and 19Hz. There was an additional scenario in OpenVibe for teaching classifiers. Authors recorded five different recorded data per one frequency per one person. Three people took part in the test. There were 3-minutes breaks between each frequency. In all tests data, which have been recorded before, were used.

4 Results

In the first test, authors checked LDA classifier. Authors taught three classifiers for 15 Hz, 17 Hz and 19Hz. To train k-fold cross validation test was set to 4 for all classifiers and frequency. In the next step, all classifiers were tested by five data per one frequency which was collected before. The results of this test are presented in Table 1.

Table 1. LDA classifier test results.

Frequency	Correctly identified
15 Hz	86 %
17 Hz	77 %
19 Hz	90 %

DOI: 10.1051/itmconf/20171502003

The results prove that the worst score was obtained for 17 Hz. The same regularity can be seen in next tests. Most likely, this is due to noisy teaching data for this frequency. Positive predictive value of the 19 Hz LDA test is presented in Table 2.

Table 2. Positive predictive value of the 19 Hz LDA test.

	Condition positive	Condition negative
Test positive	0.96	0.04
Test negative	0.06	0.94

In first test, authors checked C-SVM and Nu-SVM classifiers. Authors taught three classifiers for 15 Hz, 17 Hz and 19Hz. To train k-fold cross validation test was set to 4, ε to 0.1, cost to 1 and cache size was set to 100 for all classifiers and frequency. Authors checked four types of kernel: Linear, Polynomial, Radial basis function and Sigmoid. In the next step, all classifiers were tested by five data per one frequency which was collected before. The results of C-SVM test are presented in Table 3 and the results of Nu-SVM test are presented in Table 4.

Table 3. C-SVM classifier test results.

Frequency	**Correctly identified**			
	Linear	Polynomial	Radial	Sigmoid
15 Hz	90 %	82 %	86 %	1 %
17 Hz	66 %	50 %	70 %	33 %
19 Hz	95 %	95 %	95 %	81 %

Table 4. Nu-SVM classifier test results.

Frequency	**Correctly identified**			
	Linear	Polynomial	Radial	Sigmoid
15 Hz	70 %	77 %	56 %	0 %
17 Hz	70 %	62 %	82 %	62 %
19 Hz	95 %	90 %	95 %	86 %

The results show that the worst score was obtained for Sigmoid kernel. For all frequencies, that result is the lowest in C-SVM and Nu-SVM. The sigmoid kernel is a special type because it has only positive values. Due to this feature, is not suitable for all applications. Probably, the fitting turned out to be very difficult, as in case. The best score is obtained with C-SVM classifier using Linear kernel. Positive predictive value of the 19 Hz C-SVM linear test is presented in Table 5.

Table 5. Positive predictive value of the 19 Hz C-SVM linear test.

	Condition positive	Condition negative
Test positive	0.95	0.05
Test negative	0.00	1.00

In the last test, authors checked ANN classifiers. Authors taught three classifiers for 15 Hz, 17 Hz and 19Hz. To train k-fold cross validation test was set to 4, learning stop condition was set to 0.000001 and Learning coefficient was set to 0.01, for all classifiers and frequency. Authors changed number of neurons on hidden layer from 1 to 5. In the next step, all classifiers were tested by five data per one frequency which was collected before. The results of this test are presented in Table 6.

Table 6. ANN classifier test results.

Frequency	**Correctly identified**				
	1	2	3	4	5
15 Hz	86 %	86 %	86 %	86 %	86 %
17 Hz	53 %	82 %	82 %	82 %	82 %
19 Hz	95 %	95 %	95 %	95 %	95 %

The results show that the score for all frequencies and numbers of neuron on hidden layer are similar. Only in case of 17 Hz this classifier needs two neurons for correct recognition of input signal. The ANN classifier has got the best results from all tested in this article classifiers. Positive predictive value of the 19 Hz ANN test is presented in Table 7.

Table 7. Positive predictive value of the 19 Hz ANN test.

	Condition positive	Condition negative
Test positive	0.99	0.01
Test negative	0.04	0.96

5 Conclusion

The paper describes the research on the classifiers for brain-computer interface (BCI) based on Steady State Visually Evoked Potential (SSVEP). Authors presented research on checking the usability of classifiers for recognizing the EEG signal during the stimulus. The results show that it is possible to recognize Steady-State Visually Evoked Potential using Linear Discriminant Analysis (LDA), Support Vector Machine (SVM) and Artificial Neural Network (ANN). Achieved results were acceptable, but not perfect. The best performance presented the classifier based on Artificial Neural Network, while Support Vector Machine presented the worst. No articles were found on comparing these classifiers, in the context of BCI based on Steady State Visually Evoked Potential (SSVEP). Therefore, it is not possible to compare the results with other articles.

In further studies, the authors take as their aim an improvement of the classifier, so that the recognition results can cause minor errors. These classifiers will be very useful to build their hybrid brain-computer interface.

The work described in this paper was funded from 02/22/DSMK/1408 (Sterowanie urządzeniami mechatronicznymi z zastosowaniem technik nieliniowych oraz bioelektrycznej aktywności mózgu).

References

1. L. Shao, J. Han, D. Xu, J. Shotton, IEEE T Cybernetics, **43**, 1314-1317, (2013)
2. Z. Arief, I. A. Sulistijono, R. A. Ardiansyah, IES, **2015**, 11-14 (2015)
3. A. Kubacki, L. Sawicki, D. Rybarczyk, P. Owczarek, Adv Intell Syst, **550**, 433–440 (2017)
4. J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, T. M. Vaughan, Clin Neurophysiol, **113**, 767-791, (2002)
5. L. Bi, X. a Fan, K. Jie, T. Teng, H. Ding, Y. Liu, IEEE T Intell Transp, **15**, 959-966, (2014)
6. V. K. Varadan, S. Oh, H. Kwon, P. Hankins, J. Nanotechnol. Eng. Med, **1**, 3, (2010)
7. A. Cudo, E. Zabielska, B. Bałaj, KUL, (2011).
8. A. Kubacki, L. Sawicki, P. Owczarek, MMAR, **2016**, 783-787, (2016)
9. Z. Lin, C. Zhang, W. Wu, X. Gao, IEEE T Bio-Med Eng, **53**, 2610-2614, (2006)
10. D. Zhu, J. Bieger, G. G. Molina, R. M. Aarts, Intell. Neuroscience, **2010**, (2010)
11. H. Cecotti, IEEE T Neur Sys Reh, **18**, 127-133, (2010)
12. G. R. Muller-Putz and G. Pfurtscheller, IEEE T Bio-Med Eng, **55**, 361-364, (2008)
13. R.O. Duda, P.E. Hart, D.H. Stork., Pattern Classification (2nd ed.), (Wiley Interscience, 2000)
14. O. Dehzangi, Y. Zou, and R. Jafari, NER, **2013**, 1303–1306, (2013)
15. Balakrishnama, S., & Ganapathiraju, A. Linear discriminant analysis-a brief tutorial. Institute for Signal and information Processing, **18**, (1998)
16. A. Shmilovici, Springer, **2009**, 231–247, (2009)
17. C. Chih-Chung, L. Chih-Jen, (2001) http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf. [Accessed: 30-Jul-2017].
18. L. Rutkowski, Methods and techniques of artificial intelligence, (Wydawnictwo naukowe PWN, 2005)
19. T. Mitchell, Machine Learning, (McGraw-Hill, 1997)
20. http://emotiv.com/