International Journal of Engineering & Technology, 7 (4) (2018) 2722-2728

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.16139
Research paper

SSVEP-based brain-computer interface for computer control application using SVM classifier

Raymond Sutjiadi [1] *, Timothy John Pattiasina [2], Resmana Lim [3]

[1] *Informatics Department, Faculty of Information Technology, Institut Informatika Indonesia, Surabaya, Indonesia*
[2] *Information System Department, Faculty of Information Technology, Institut Informatika Indonesia, Surabaya, Indonesia*
[3] *Electrical Engineering Department, Faculty of Industrial Technology, Petra Christian University, Surabaya, Indonesia*
**Corresponding author E-mail: raymond@ikado.ac.id*

Abstract

In this research, a Brain Computer Interface (BCI) based on Steady State Visually Evoked Potential (SSVEP) for computer control applications using Support Vector Machine (SVM) is presented. For many years, people have speculated that electroencephalographic activities or other electrophysiological measures of brain function might provide a new non-muscular channel that can be used for sending messages or commands to the external world. BCI is a fast-growing emergent technology in which researchers aim to build a direct channel between the human brain and the computer. BCI systems provide a new communication channel for disabled people. Among many different types of the BCI systems, the SSVEP based has attracted more attention due to its ease of use and signal processing. SSVEPs are usually detected from the occipital lobe of the brain when the subject is looking at a twinkling light source. In this paper, SVM is used to classify SSVEP based on electroencephalogram data with proper features. Based on the experiment utilizing a 14-channel Electroencephalography (EEG) device, 80 percent of accuracy can be reached by our SSVEP-based BCI system using Linear SVM Kernel as classification engine.

Keywords: *Brain Computer Interface; Brain Waves; Electroencephalography; Steady State Visually Evoked Potential; Support Vector Machine.*

1. Introduction

Currently many different disorders can disrupt the neuromuscular channels through which the brain communicates with and controls its external environment. Amyotrophic lateral sclerosis (ALS), brainstem stroke, brain or spinal cord injury, cerebral palsy, muscular dystrophies, multiple sclerosis, and numerous other diseases impair the neural pathways that control muscles or impair the muscles themselves. In the absence of methods for repairing the damage done by these disorders, there are 3 options for restoring function [1]. The first is to increase the capabilities of remaining pathways. The second option is to restore function by detouring around breaks in the neural pathways that control muscles. The third option is to provide the brain with a new, non-muscular communication and control channel, by using a direct Brain Computer Interface (BCI) for conveying messages and commands to the external world. A variety of methods for monitoring brain activity might serve as a BCI. At present, only Electroencephalography (EEG) and related methods, which have relatively short time constants, can function in most environments, and require relatively simple and inexpensive equipment, offer the possibility of a new non-muscular communication and control channel, a practical BCI. EEG is a non-invasive way of acquiring brain waves from the surface of human scalp, which is widely accepted due to its simple and safe approach [2].

Nowadays a lot of researches about BCI have been published. BCI technology represents a highly growing field of research with application systems. Its contributions in medical fields range from prevention to neuronal rehabilitation for serious injuries. Mind reading and remote communication have their unique fingerprint in numerous fields such as educational, self-regulation, production, marketing, security as well as games and entertainment. It creates a mutual understanding between users and the surrounding systems. The research community has initially developed BCIs with biomedical applications in mind, leading to the generation of assistive devices. They have facilitated restoring the movement ability for physically challenged or locked-in users and replacing lost motor functionality [3]. More recent studies have targeted normal individuals by exploring the use of BCIs as a novel input device and investigating the generation of hands-free applications [4].

On the other hand, some of BCI advantages for able-bodied users have been enlightened in. BCI could be helpful especially for applications where it is instantaneously difficult to move and the response time is crucial. Besides BCI can also be used to increase the accuracy of the HCI systems, resulting in BCI contribution in various fields such as educational, gaming industry, transportation, and entertainment. Despite its expected success, brain computer interfacing needs to overcome technical difficulties as well as challenges posed by user acceptance to deal with such newly discovered technology. Understanding the underlying mechanism of EEG within the thalamus in the brain, especially in the LGN, can also lead into a better implementation of SSVEP-based BCI [5].

2. Brain-computer interface

A Brain Computer Interface, sometimes called a Mind-Machine Interface (MMI), Direct Neural Interface (DNI), or Brain–Machine Interface (BMI), is a direct communication pathway between an enhanced or wired brain and an external device. BCI system is a communication channel that could assist and increase the performance of both disabled and normal people. In a BCI system, the brain activities are recorded from the scalp and coded to appropriate external control commands. Some modalities used for brain activity recording in BCI applications are Electroencephalography (EEG), magneto-encephalography, functional magnetic resonance imaging and near infrared spectroscopy [6]. However, due to its ease of utility and better temporal resolution the EEG is mostly used in BCI systems. BCI systems build a communication bridge between human brain and the external world eliminating the need for typical information delivery methods. They manage the sending of messages from human brains and decoding their silent thoughts. BCI systems can help handicapped people to tell and write down their opinions and ideas via variety of methods such as in spelling applications [7], semantic categorization [8], or silent speech communication [9].

In the 1970s, research on BCI's started at the University of California, which led to the emergence of the expression brain–computer interface. The focus of BCI research and development continues to be primarily on neuroprosthetics applications that can help to restore damaged sight, hearing, and movement. In 1990s marked the appearance of the first neuroprosthetic devices for humans. BCI does not read the mind accurately, but detects the smallest of changes in the energy radiated by the brain when you think in a certain way. A BCI recognizes specific energy/frequency patterns in the brain. The BCI can lead to many applications especially for disabled people such as: new ways for gamers to play games using their heads, social interactions; enabling social applications to capture feelings and emotions, helping partially or fully-disabled people to interact with different computational devices, and helping to understand more about brain activities and human neural networks. These applications depend on basic understanding of how the brain works.

The BCI framework is described in several building blocks that need to interact properly as shown in Fig. 1. Different BCIs rely on different mental activities and corresponding EEG patterns. BCI system components, regardless of its type, recording methods or applications, are: 1) signal acquisition, 2) pre-processing 3) feature extraction/selection, 4) classification, 5) application interface.

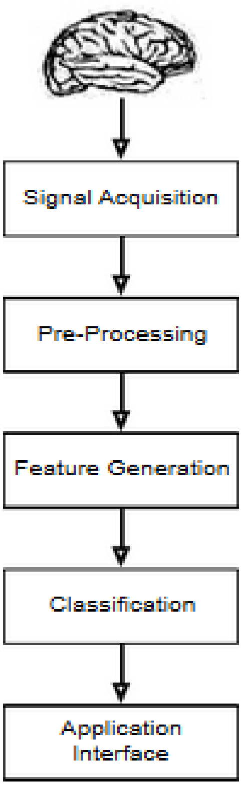

Fig. 1: BCI Framework.

There are three dominant approaches to BCI systems [10], categorized according to the type of mental activity and corresponding brain activity used for control. Three existing types of BCIs are Steady State Visually Evoked Potential (SSVEP), P300, and Event Related Desynchronization and Synchronization (ERD/S) or Motor Imagery (MI). Since each type of BCI relies on brain waves generated at different location of brain, they require electrodes placed on different areas of brain.

2.1. Steady state visually evoked potential

Steady State Visually Evoked Potential (SSVEP) BCIs [11]-[13] rely on attention to lights that flicker at specific frequencies, which elicit corresponding SSVEP signals over occipital areas. The lights might contain messages or commands like letters, words, or device commands. Hence, the BCI can identify target items by identifying which frequencies are apparent over occipital areas. This method will be elaborated further on Section 3.

2.2. P300

P300 BCIs [14]-[18] are based on the brain response to an event or stimulus considered as event-related potential (ERP) which can detect intention of the subject. After stimulus onset, positive and negative deflections occur in the EEG. The largest positive deflection that occurs around 300 milliseconds after the stimulus onset is called "P300" which is the most used ERP component in BCI systems. P300 BCIs rely on the flash of a rare event among frequent events. When users choose to count specific rare target items, each target flash produces P300 signals that are dominant over parieto-occipital areas. Moreover, several P300-based BCI studies found slight reduction in performance during the sessions, which might be due to a habituation effect. Because the P300 is largest for new, relevant, desired events, repeatedly presenting rare events results in decreased P300 amplitudes and thus reduced performance. using a P300-based BCI requires attention and concentration, the user should not be distracted. This could be difficult if the P300-based BCI is used in normal life.

2.3. Event related desynchronization and synchronization

ERD BCIs utilize changes in event related desynchronization and synchronization (ERD/S) that usually occur when people imagine specific movements [19]. They have also been called sensorimotor rhythm (SMR) or mu/beta BCIs. Imagining movement produced changes in 8-12 Hz activity over contralateral sensorimotor areas. This activity is in the mu band, and activity in the beta band sometimes changes with imagined movement as well. Nowadays, ERD/ERS is one of the most used brain wave patterns in BCIs. It allows the user to consciously manipulate his or her EEG patterns, and therefore it is suited for BCI applications that require the user to actively carry out control, as opposed to for instance P300, where the brain wave pattern is subconsciously altered. This also makes ERD and ERS suitable for identification purposes, since it can be reproduced at will at any time.

To design a practical BCI system needs to address several issues such as ease of use, a reliable system performance, and low-cost hardware and software. In recent years, with the biomedical sciences and electronics technology, mobile and online BCI's development has been proposed. Among them in this research SSVEPs are attracted due to its advantages of requiring less or no training, high Information Transfer Rate (ITR) and ease of use [20].

3. Steady state visually evoked potential

The Steady State Visually Evoked Potential (SSVEP) has recently become a popular paradigm in Brain Computer Interface (BCI) applications. Typically these applications offer the user a binary selection of targets that perform correspondingly discrete actions.

Such control systems are appropriate for applications that are inherently isolated in nature, such as selecting numbers from a keypad to be dialled or letters from an alphabet to be spelled. The mechanism behind SSVEPs is not yet well-understood, but recently SSVEP-based BCIs have been increasingly employed in research, as they have been demonstrated to be useful in different applications, especially for applications that require a large number of commands and self-paced performances. However motivation exists for users to employ proportional control methods in intrinsically analogue tasks such as the movement of a mouse pointer [21].

SSVEP BCI is normally constructed by presenting the user with multiple stimuli that are distinct in repetition rates. Stimuli can be presented through use of flashing Light Emitting Diodes (LEDS) or alternatively standard Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD) monitors can be employed to output a variety of stimuli through Computer Generated Imagery (CGI). However, this type of presentation is severely limited in the number of distinct stimuli that do not suffer from temporal aliasing due to the comparatively low refresh rate (60-120Hz) of typical standard visual display units. When using SSVEP, the monitor is used to show moving animation, for example a left arrow moving from right to left. User gazes this animation focally and as the result, brain will emit certain signal pattern to be classified into certain computer task. Another way, monitor is used to show some flickering objects in certain frequencies, usually in low frequency.

There are some issues about SSVEP BCIs, one is gaze dependence. Another issue is that in some users, the flickering stimulus is annoying and produces fatigue. Using higher frequencies for the flickering stimuli reduces the annoyance, but on the other hand, it is harder to detect the SSVEP. To improve the accuracy in an application such as in a BCI-based speller, the SSVEP-based BCI needs to be optimized [22].

A SSVEP BCI system contains the following modules: a) Stimulator module: is a LED panel or a monitor responsible to produce the visual stimuli at a specific frequency; b) Signal acquisition module: is responsible to acquire the EEG signals during the system operation; c) Signal processing module: is responsible for the analysis of EEG signals and the translation/transformation of them into meaningful "codewords"; and d) Device commands module: is appointed with the task to translate the "codewords" into interface commands according to the application setup, as shown in Fig. 2.

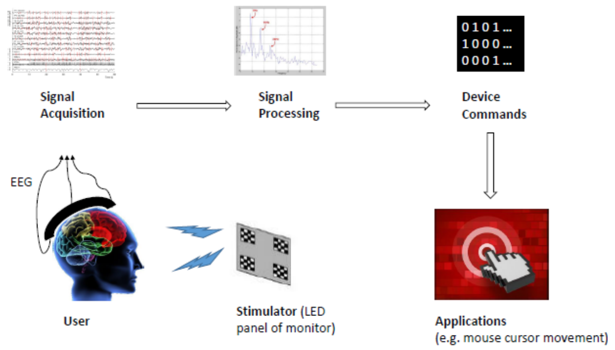

Fig. 2: SSVEP-BCI System.

The signal processing module consists of four submodules: a) preprocessing, b) feature extraction, c) feature selection and d) classification. The first three submodules have the goal to make the data suitable for the classification process, which will gives us the appropriate "codewords", as shown in Fig. 3.

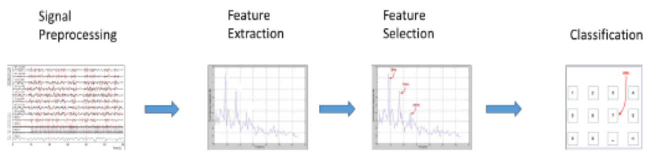

Fig. 3: Signal Processing Module of SSVEP BCI System.

Many methods have been applied in the pre-processing part of a SSVEP-BCI system. The most common of them is the filtering, and most specifically the band pass filtering. Various filters have been used at this point of analysis procedure depending of the particular needs of each SSVEP-BCI system, such as band pass IIR filter from 22-48Hz and more specifically, the Common Averaging Re-referencing (CAR) spatial filtering method is used in to spatially filter the multichannel EEG signals and remove unwanted components such as eye blinks. Finally, the decision step in SSVEP BCI system is performed by applying a classification procedure. More specifically, in classifiers [23 - 25] such as the Support Vector Machine (SVM), the Linear Discriminant Analysis (LDA) and Extreme Learning Machines (ELM) are used. SVM and LDA are the most popular classifiers among SSVEP community and have been used in numerous works.

4. Support vector machine

Support Vector Machine (SVM) was first heard in 1992, introduced by Boser, Guyon, and Vapnik in COLT-92. SVM are a set of related supervised learning methods used for classification and regression [26]. They belong to a family of generalized linear classifiers. In another terms, SVM [27] is a learning structure that can be used to solve classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over fit to the data. Support Vector Machines (SVM) can be defined as systems which use hypothesis space of a linear functions in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory. In the context of classification it can be understood as a maximal margin classifier whose linear or non-linear structure is defined by a kernel function. The design of a classifier of this kind gives rise to a quadratic constrained optimization task that can be solved using a number of efficient computational tools. In a classification system, the SVM follows two stages: training and classification. The advantage of SVM is that once a boundary is established, most of the training data is redundant. All it needs is a core set of points which can help identify and set the boundary. These data points are called support vectors because they "support" the boundary. In the training, labeled data are used in order to determine the hyperplane in a high-dimensional feature space that distinguish the classes with maximal margin. In practice, the training can be per-formed in the original data space using different kernel functions as linear, quadratic, polynomial, multilayer perceptron (MLP) or Gaussian radial basis (RBF) [28]. In this research a linear kernel function was selected after preliminary tests with several methods, in view of its stability for multiple trials. An example of SVM classification as shown in Fig. 4.

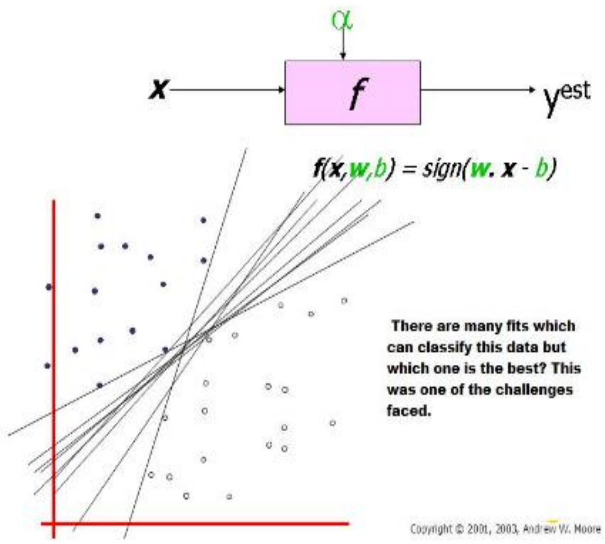

Fig. 4: Sample Hyper Planes Data Classifier [29].

As shown in Fig. 4, there are many linear classifiers (hyper planes) that separate the data. However only one of these achieves maximum separation. Next step is we try to give the maximum margin classifier which provides a solution to the above mentioned problem shown by Fig. 4

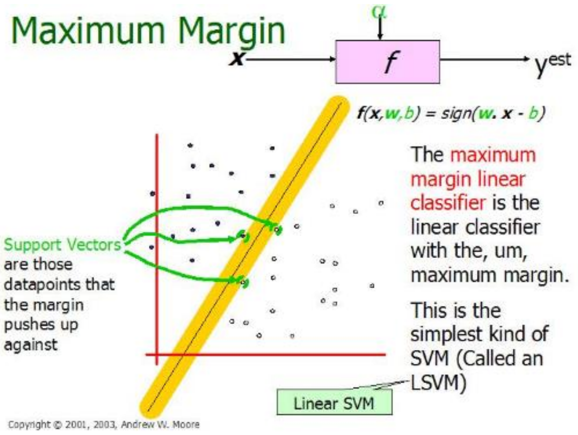

Fig. 5: Linear Support Vector Machines (LSVM).

As shown in Fig. 5, the maximum linear classifier with the maximum range using Linear Support Vector Machine (LSVM) can solved the problem. This example is simple linear SVM classifier. The goals of SVM are separating the data with hyper plane and extend this to non-linear boundaries using kernel trick [30].
Assume the training data of D with a set of n points of calculating by Eq. 1:

$$D = \{(x_i, y_i) \mid x_i \in \Re^p, y_i \in \{-1,1\}\}_{i=1}^{n} \tag{1}$$

Where the y_i is either 1 or −1, indicating the class to which the point x_i belongs. Each x_i is a P-dimensional real vector. We want to find the maximum-margin hyper plane that divides the points having y_i = 1 from those having y_i = −1. Any hyper plane can be written as the set of points X satisfying following Eq. 2:

$$W \cdot X - b = 0 \tag{2}$$

Where denotes the dot product and W denotes the normal vector of hyper plane. The parameter $\frac{b}{||W||}$ determines the offset of the hyper plane from the origin along the normal vector W.
For maximization distance in point for linearly separable data the margin is set. The set margin hyper plane is described in Eq. 3:

$$W \cdot X - b = 1 \text{ and } W \cdot X - b = -1 \tag{3}$$

Geometrically, the distance between these two hyper planes is $\frac{2}{||W||}$, so to maximize the distance between the planes we want to minimize ||W||. To prevent the data points to fall into the margin, the following constraint is added: for each i either, $W \cdot X - b \geq 1$ For x_i of first class; $W \cdot X - b \leq 1$ For x_i of second class. The final equation is written as Eq. 4:

$$yi\,(w \cdot xi - b) \geq 1, \text{for all } 1 \leq i \leq n \tag{4}$$

The extended extension of maximum-margin classifier which provides a solution to the above mentioned problem is given as Eq. 5:

$$\text{margin} \equiv \arg\min_{\mathbf{x}\in D} d(\mathbf{x}) = \arg\min_{\mathbf{x}\in D} \frac{|\mathbf{x}\cdot\mathbf{w}+b|}{\sqrt{\sum_{i=1}^{d} w_i^2}} \tag{5}$$

5. System design and implementation

For this research, BCI application is built using OpenViBE, an open source software for brain computer interfaces and real time neurosciences. OpenViBE is used to develop signal acquisition server that connects to Emotiv Epoc+ as EEG scientific contextual device via Bluetooth interface to gather raw brain waves from 14 different channels. Besides that, OpenViBE is also used to develop signal acquisition client which receive signal from port 1024 and do signal processing (tagging signal, signal filtering, and signal classification). Then, as an output of this signal processing is stimulation signal to be sent to VRPN (Virtual Reality Peripheral Network) server, as the demarcation point between BCI system to the operating system. To receive stimulation signal and translate it into cursor movement, a C++ application is developed as VRPN client. Fig. 6 showed complete architecture of BCI system.

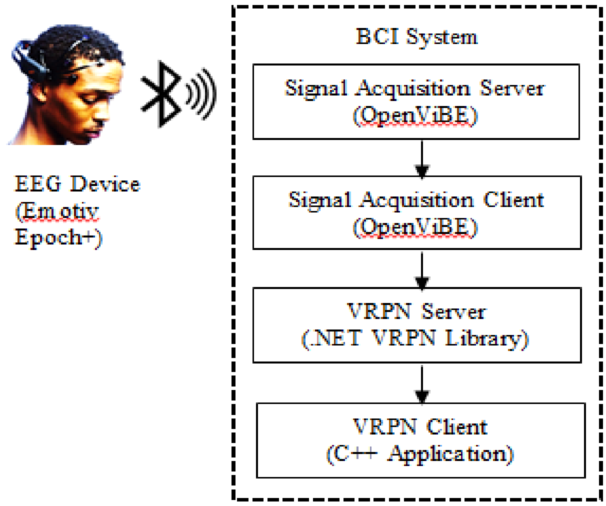

Fig. 6: BCI Architecture.

In part of signal acquisition client, there are 3 sections of signal pre-processing and processing to result final decision in form of particular cursor movement command:

1) Signal epoching and tagging.

In this process, raw signal received from EEG device is captured in short duration (1.5s) with interval 0.1s between epochs, as shown in Fig. 7. Then, each epoch is tagged starting and ending point as the boundary of processed signal for the next stage.

2) SSVEP generator.

SSVEP generator creates flickered object on screen as visual medium to stimulate brain waves. There are 4 white-square flickered objects with black background to give contrast view to user as shown in Fig. 8. Because LCD screen used in this research has refresh rate 60Hz, each flickered object must be set to flicker using frequency factor of 60, i.e. 15Hz (up direction), 12Hz (left direction), 10Hz (right direction), and 6Hz (down direction).

3) Common Spatial Pattern (CSP) training.

Some brain parts are more reactive to receive stimuli. CSP is used as feature extraction to search the best combination of electrodes. By using CSP filter is expected to get better precision for classifier training by removing unwanted signals.

4) Signal filtering.

Before entering SVM classifier, signal is filtered once again to remove any noise using band pass filter. Also applied simple Digital Signal Processing (DSP) and signal average to boost quality of signal to be processed as training model on the next step.

5) Classifier training.

This is the main section of the BCI system. In this section, model signals will be used to train SVM classifier. Tuning the SVM parameter is required to get better accuracy. The result of this section is SVM classifier model, which will be used to predict the online testing scenario.

6) Predicting.

Final process is doing online testing scenario by using SVM classifier model. The prediction output is received by VRPN server and then translate it into respective stimulation signal. This stimulation signal then sent to VRPN client.

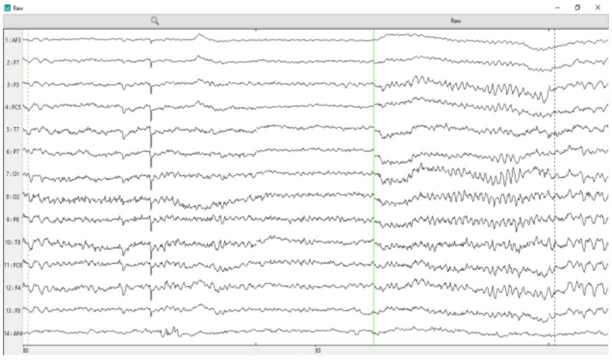

Fig. 7: 14-Channel Raw Brain Waves.

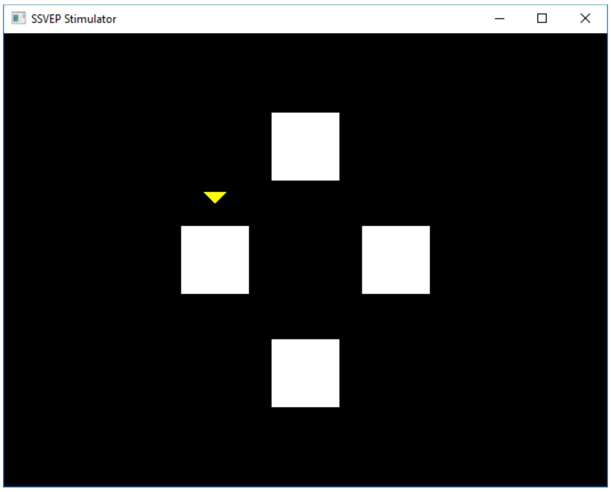

Fig. 8: SSVEP Stimulator.

After signal processing in acquisition client and VRPN server, then stimulation signal is delivered to VRPN client. In VRPN client, this stimulation signal is converted into command to move mouse cursor to respective direction according the SVM prediction. A C++ application is developed to trigger cursor movement accordance with received stimulation signal.

6. Result and discussion

In order to test the performance and accuracy of SSVEP-based BCI system using SVM classifier, several tests were implemented. All these experiments were used the same optimized parameters as listed below:

Epoch duration : 1.5s
Epoch interval : 0.1s
CSP Filter Dimension : 14
Support Vector Machine Parameters
Epsilon : 0.2
SVM Type : C-SVC
Degree : 3
Epsilon Tolerance : 0.001
Cost : 1
Cache Size : 100
Gamma : 0
Nu : 0.5
Shrinking : True
Coef 0 : 0
Number of Partition : 10
Balance Class : False

6.1. Frequency tolerance

In this section, the accuracy testing was held by tuning the frequency tolerance parameter. This frequency tolerance was used to set the point of band pass filter from each respective SSVEP frequency. For example: SSVEP frequency for 'Up Direction' = 15Hz and frequency tolerance parameter = 0.5. It meant band pass filter would be set 14.5Hz for High Pass Filter (HPF) and 15.5Hz for Low Pass Filter (LPF).

The lower frequency tolerance would eliminate noise better but tighter to detect brain waves range. On the other side, the higher frequency tolerance would increase the noise but wider to detect brain waves range. Table 1 below depicts the result of experiment by tuning frequency tolerance parameter using default SVM kernel type (Linear).

Table 1: System Accuracy by Tuning Frequency Tolerance Parameter

No	Actual	Prediction		
		Frequency Tolerance = 0.25	Frequency Tolerance = 0.5	Frequency Tolerance = 0.75
1	Neutral	Neutral	Neutral	Up
2	Down	Down	Down	Down
3	Right	Right	Up	Up
4	Left	Left	Left	Neutral
5	Up	Up	Up	Up
6	Neutral	Neutral	Neutral	Neutral
7	Down	Up	Down	Down
8	Left	Left	Left	Left
9	Up	Up	Up	Up
10	Right	Up	Right	Right
11	Neutral	Neutral	Neutral	Neutral
12	Left	Up	Left	Right
13	Right	Up	Neutral	Up
14	Up	Up	Right	Left
15	Down	Down	Down	Down
16	Neutral	Neutral	Neutral	Neutral
17	Left	Up	Left	Right
18	Up	Up	Up	Up
19	Right	Right	Right	Right
20	Down	Down	Down	Down
21	Neutral	Neutral	Up	Neutral
22	Right	Right	Right	Up
23	Down	Down	Down	Down
24	Left	Up	Up	Left
25	Up	Neutral	Neutral	Up
26	Neutral	Neutral	Neutral	Neutral
27	Up	Up	Up	Up
28	Left	Up	Right	Neutral
29	Down	Down	Down	Down
30	Right	Right	Right	Right
31	Neutral	Neutral	Neutral	Neutral
32	Down	Down	Down	Down
33	Right	Left	Up	Right
34	Up	Up	Up	Up
35	Left	Left	Left	Right
36	Neutral	Up	Neutral	Neutral
37	Down	Down	Down	Down
38	Up	Up	Up	Up
39	Right	Right	Right	Up
40	Left	Left	Left	Neutral
Accuracy		75%	80%	70%

As shown from Table 1, the best accuracy reached when using frequency tolerance 0.5 with accuracy percentage = 80%.

6.2. Kernel type

For second parameter, the experiment would focus on the choosing of SVM kernel type. SVM kernel is algorithm uses a set of mathematical functions that take data as input and transform it into the required form for pattern analysis. It comprise 4 types, i.e.: Linear, Polynomial, Radial Basis Function, and Sigmoid. Choosing the correct algorithm would increase the accuracy of SVM model classifier. This experiment uses the same configuration with the last section and for frequency tolerance uses 0.5 as the best parameter based on the previous experiment section. The result is shown in Table 2.

Table 2: System Accuracy by Tuning SVM Kernal Parameter

No	Actual	Prediction			
		Linear	Polinomial	Radial Basis Function	Sigmoid
1	Neutral	Neutral	Up	Up	Neutral
2	Down	Down	Down	Down	Down
3	Right	Up	Left	Left	Up
4	Left	Left	Up	Left	Up
5	Up	Up	Up	Up	Up
6	Neutral	Neutral	Neutral	Neutral	Neutral
7	Down	Down	Down	Down	Down
8	Left	Left	Left	Left	Up
9	Up	Up	Up	Up	Up
10	Right	Right	Right	Right	Up
11	Neutral	Neutral	Neutral	Neutral	Up
12	Left	Left	Right	Left	Right
13	Right	Neutral	Right	Up	Right
14	Up	Right	Up	Up	Neutral
15	Down	Down	Down	Down	Down
16	Neutral	Neutral	Neutral	Right	Neutral
17	Left	Left	Up	Right	Up
18	Up	Up	Up	Up	Up
19	Right	Right	Right	Right	Right
20	Down	Down	Down	Down	Down
21	Neutral	Up	Up	Up	Up
22	Right	Right	Right	Right	Right
23	Down	Down	Down	Down	Down
24	Left	Up	Up	Neutral	Left
25	Up	Neutral	Up	Up	Up
26	Neutral	Neutral	Neutral	Neutral	Neutral
27	Up	Up	Up	Up	Up
28	Left	Right	Neutral	Left	Left
29	Down	Down	Down	Down	Down
30	Right	Right	Right	Right	Right
31	Neutral	Neutral	Neutral	Neutral	Neutral
32	Down	Down	Down	Down	Down
33	Right	Up	Left	Left	Neutral
34	Up	Up	Left	Up	Up
35	Left	Left	Left	Neutral	Left
36	Neutral	Neutral	Neutral	Neutral	Neutral
37	Down	Down	Down	Down	Down
38	Up	Up	Up	Up	Up
39	Right	Right	Right	Right	Right
40	Left	Left	Up	Left	Left
Accuracy		80%	72.5%	77.5%	75%

As shown from Table 2, the best accuracy occurred when using Linear SVM Kernel with accuracy percentage = 80% and the worst accuracy occurred when using Polynomial SVM Kernel with 72.5% of accuracy.

7. Conclusion

BCI is an important system for helping motoric impaired people as well as normal people to access computers. However, the implementation of a BCI system needs a proper signal processing to make it accurate, because human brain waves are complex and susceptible to noise. Moreover, translating human brain waves pattern into actions that move mouse cursor is quite challenging. The abstract nature of human brain waves pattern should be stimulated using external stimulation. Such that the specific brain waves pattern can be read and translated by the computer.

This paper proposes an external visual stimulation for a SSVEP-based BCI using flickered object on LCD screen to stimulate the occurrence of similar brain waves frequency pattern. This method is effective and easy to use for motoric disabled people who still have good eyes as the stimuli receptor.

To make accurate BCI system, raw brain waves collected by EEG have to be filtered to minimize the noise or other unwanted signals. In this system, band pass filter is used to pass certain range of frequency. The lower frequency tolerance would eliminate noise better but tighter to detect brain waves range. On the other side, the higher frequency tolerance would increase the noise but wider to detect brain waves range. From the experiment, frequency tolerance = 0.5 would achieve 80 percent of accuracy.

On the other hand, the use of a correct classifier method is also important to make the BCI system more accurate. SVM is chosen because this method is quite efficient and robust to be used as a pattern recognition algorithm. This method is also widely used in bioinformatics applications. Based on the experiment utilizing a 14-channel EEG device, 80 percent of accuracy can be reached using our SSVEP-based BCI with Linear SVM to move a mouse cursor to four directions (up, down, left, and right).

For further refinement, our BCI system could also be improved to accommodate other mouse operations, such as click and double-click function. This will make the BCI system more functional and useful for helping disabled people to operate a computer.

Acknowledgement

The authors are thankful to the grant provided by Directorate of Research and Community Service, the Ministry of Research, Technology, and Higher Education, Republic of Indonesia under grant scheme of Junior Lecturer Research 2018, Contract Number: 004/SP2H/LT/P3M/II/2018.

References

[1] Wolpaw JR, Birbaumer N, McFarlanda DJ, Pfurtscheller G & Vaughan TM, "Braincomputer Interfaces for Communication and Control", *Clinical Neurophysiology*, Vol. 113, (2002), pp. 767-791. https://doi.org/10.1016/S1388-2457(02)00057-3.

[2] Yasin TV, Pasila F & Lim R, "A Study of Mobile Robot Control Using EEG Emotiv Epoc Sensor*", MATEC Web of Conferences,* 164, (2018), art. No. 01044.

[3] Rao R & Scherer R, "Brain-Computer Interfacing [in the Spotlight] Signal Process Mag", *IEEE*, Vol. 27, No. 4, (2010), pp. 152-150.

[4] Erp JV, Lotte F & Tangermann M, "Brain-Computer Interfaces: Beyond Medical Applications Computer", *IEEE*, Vol. 45, No. 4, (2012), pp. 26-34.

[5] Bhattacharya BS, Gotarredona TS, Balassa L, Bhattacharya A, Stokes AB, Rowley A, Sugiarto I, & Furber S, "A Spiking Neural Network Model of the Lateral Geniculate Nucleus on the SpiNNaker Machine", *Frontiers in Neuroscience* 11 (2017): 454. https://doi.org/10.3389/fnins.2017.00454.

[6] Resalat SN, et al, "High-Speed SSVEP-Based BCI: Study of Various Frequency Pairs and Inter-Sources Distances", *IEEE-EMBS International Conference Biomedical and Health Informatics*, (2012), pp. 220 – 223.

[7] Wang W, Degenhart AD, Sudre GP, Pomerleau DA &Tyler-Kabara EC. "Decoding Semantic Information from Human Electrocorticographic (Ecog) Signals", *Engineering in Medicine and Biology Society (EMBC)*, (2011), pp. 6294–98.

[8] Brumberg JS, Castanon AN, Kennedy PR & Guenther FH, "Brain-Computer Interfaces for Speech Communication", *Speech Commun*, Vol. 52, No. 4, (2010), pp. 367-379. https://doi.org/10.1016/j.specom.2010.01.001.

[9] Vourvopoulos A & Liarokapis F, "Robot Navigation Using Brain-Computer Interfaces", *11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, (2012), pp. 1785–1792.

[10] Alamdri N, Haider A, Arefin R, Verma AK & Tavakolian K, "A Review of Methods and Applications of Brain Computer Interface Systems", *IEEE International Conference on Electro Information Technology (EIT)*, (2016), pp. 345-350.

[11] Amiri S, Rabbi A, Azinfar L & Rezai RF, "A Review of P300, SSVEP, and Hybrid P300/SSVEP Brain-Computer Interface Systems", *Brain-Computer Interface Systems – Recent Progress and Future Prospects*, Rezai RF, Ed: In-Tech, (2013).

[12] Allison B, Luth T, Valbuena D, Teymourian A, Volosyak I, & Graser A, "BCI Demographics: How many (and what kinds of) people can use an SSVEP BCI?", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 18, (2010), pp. 107-116. https://doi.org/10.1109/TNSRE.2009.2039495.

[13] Setiono T, Handojo A, Intan R, Sutjiadi R & Lim R, "Brain Computer Interface for Controlling RC-Car Using Emotiv Epoc+", *Journal of Telecommunication, Electronic and Computer Engineering*, Vol. 10 (2-3), (2018), pp. 169-172.

[14] Rezai RF & Peters J, “P300 wave feature extraction: preliminary results”, *Canadian Conference on Electrical and Computer Engineering*, (2005), pp. 390-393.
[15] Rezai RF, "P300-based Speller Brain-Computer Interface”, *Recent Advances in Biomedical Engineering*, Naik GR, Ed., In-Tech, (2009), pp. 137-148.
[16] Rezai RF, Gavett S, Ahmad W, Rabbi A, & Schneider E, "A Comparison Among Several P300 Brain-Computer Interface Speller Paradigms", *Clin EEG Neurosci*, Vol. 42, (2011), pp. 209-213. https://doi.org/10.1177/155005941104200404.
[17] Rezai RF, "Human Error in P300 Speller Paradigm for BrainComputer Interface", *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, (2007), pp. 2516-2519.
[18] Sellers E & Donchin E, “A P300-Based Brain–Computer Interface: Initial Tests by ALS Patients”, *Clin Neurophysiol*, Vol. 117, No. 3, (2006), pp. 538–548. https://doi.org/10.1016/j.clinph.2005.06.027.
[19] Leocani L, Toro C, Zhuang P, Gerloff C & Hallett M, “Event-Related Desynchronization in Reaction Time Paradigms: a Comparison with Event-Related Potentials and Corticospinal Excitability”, *Clinical Neurophysiology*, Vol. 112, (2001), pp. 923–930. https://doi.org/10.1016/S1388-2457(01)00530-2.
[20] Cheng M, Gao XR, Gao SK & Xu D, “Design and Implementation of a Brain Computer Interface with High Transfer Rates”, *IEEE Trans Biomed Engineering*, Vol. 49, No. 10, (2002), pp. 1181-1186. https://doi.org/10.1109/TBME.2002.803536.
[21] Wilson J & Palaniappan R, Analogue Mouse Pointer Control via an Online Steady State Visual Evoked Potential (SSVEP) Brain–Computer Interface, IOP Publishing, (2011), pp 1-6.
[22] Sugiarto I, Allison B & Gräser A. “Optimization Strategy for SSVEP-based BCI in Spelling Program Application”, *International Conference on Computer Engineering and Technology 2009 (ICCET'09)*, Vol. 1, (2009), pp. 223-226.
[23] Jia C, Gao X, Hong B & Gao S, “Frequency and Phase Mixed Coding in SSVEP-Based Brain–Computer Interface”, IEEE Transactions on Biomedical Engineering, Vol. 58, (2011), pp. 200–206. https://doi.org/10.1109/TBME.2010.2068571.
[24] Guger C, Allison B, Grosswindhager B, Pruckl RR, Hintermuller C, Kapeller C, Bruckner M, Krausz G & Edlinger G, “How Many People Could Use an SSVEP BCI?”, *Frontiers in Neuroscience*, Vol. 6, No. 169, (2012). https://doi.org/10.3389/fnins.2012.00169.
[25] Rajesh S & Haseena B, “Comparison of SSVEP Signal Classification Techniques Using SVM and ANN Models for BCI Applications”, *International Journal of Information and Electronics Engineering*, Vol. 4, (2014).
[26] Cortes C & Vapnik V. “Support Vector Networks”, *Machine Learning*, Vol 20, (1995), pp. 273–297. https://doi.org/10.1007/BF00994018.
[27] Burges CJC, “A Tutorial on Support Vector Machines for Pattern Recognition”, *Data Mining Knowledge Discovery*, Vol. 2, No. 2, (1998), pp. 1–47. https://doi.org/10.1023/A:1009715923555.
[28] Cristianini N & Taylor JS, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, (2000). https://doi.org/10.1017/CBO9780511801389.
[29] Moore AW, Support Vector Machines, *Tutorial Slides*, (2003), pp. 2–10, https://www.autonlab.org/tutorials/svm.html.
[30] Mitchell T, *Machine Learning*, McGraw-Hill Computer Science Series, (1997), pp. 86-88.