

PAPER • OPEN ACCESS

Virtual keyboard based on a brain-computer interface

To cite this article: O A Rusanu *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **514** 012020

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

Virtual keyboard based on a brain-computer interface

O A Rusanu¹, L Cristea¹, M C Luculescu¹, P A Coffas² and D T Coffas²

¹Product Design, Mechatronics and Environment Department, *Transilvania* University of Braşov, Braşov, Romania

²Electronics and Computers Department, *Transilvania* University of Braşov, Braşov, Romania

E-mail: oana.rusanu@unitbv.ro

Abstract. This paper proposes a novel LabVIEW based algorithm regarding the implementation of an eye-blinks strength controlled virtual keyboard aimed to help people with neuromotor disabilities. The application can be considered a brain-computer interface which is useful for the assistance of patients suffering from Locked-In syndrome or amyotrophic lateral sclerosis. A precise, repeatable and easily detectable control signal could be the eye-blink strength. The LabVIEW based virtual keyboard provides the following features: detection and counting the voluntary eye-blinks, switch command, select command, highlight action running in parallel with the previously mentioned processes, enabling cancel, delete and space commands. The working principle underlying the implementation of the proposed virtual keyboard is Divide and Conquer paradigm. Thus, it is possible the switching task across the rows, half rows and keys (characters) in order to select a row, half row, respectively key associated to the character that should be introduced in the text box.

1. Introduction

This paper proposes an application aimed for the assistance of people with disabilities. Unfortunately, the patients who suffered a brain stroke, spinal cord injuries or were diagnosed with Locked-In syndrome or amyotrophic lateral sclerosis, cope with severe speaking impairments and neuromotor diseases. Some significant examples could be the scientist Stephen Hawking and the journalist Jean Dominique Bauby. They lost their ability to communicate in a natural way (by spoken words or talking) and to interact with the outside environment via peripheral nerves and muscles, similar to the condition of healthy people.

The only available solution that has the potential to help them recover their lost independence and autonomy of doing simple and usual daily activities, without permanent support offered by a caregiver, is the development of an embedded system based on a brain-computer interface.

The brain-computer interface provides an alternative channel of communication and control. This can be achieved via advanced processing methods applied to the biopotentials recorded across the cerebral neurons using the electroencephalography technique. Thus, the mental activity detected during the execution of cognitive tasks (for instance: motor imagery based on the movement of a leg or an arm in the forward or backward direction, mental calculus solving, watching a flashing light stimulus) generates certain cerebral patterns which are firstly undergone to advanced techniques of artificial intelligence. Then, they are classified and translated into commands and signals aimed to control different biomechatronic devices: a motorized wheelchair, a robotic arm, a neuroprosthesis and the cursor position on the PC desktop.

The eye-blinking is an artefact of the electroencephalographic signal so that it can be easily



detectable and it is characterized by an increase followed by a decrease in the amplitude of the acquired biopotential. The intentional or voluntary eye-blinking could be considered a precise, repeatable and easy to quantify signal so that it proves its significance in the control application.

This paper proposes a virtual keyboard controlled by eye-blinking (Figure 1). The keyboard helps people with disabilities to write or type words. Taking into account the fact that it provides an unusual and interactive method of human-machine interaction, the virtual keyboard could also raise the interest of those who are passionate about futuristic technology.

According to the scientific literature, the current variants of virtual keyboard aimed to help people with disabilities are based on the following working principles: the selection of characters caused by triggering the SSVEP (Steady State Visual Evoked Potential) [1] and P300 signals [2], eye-tracking based on image processing methods [3], electrooculography and other hybrid techniques [4].



Figure 1. Front Panel (user-interface) of the LabVIEW based virtual keyboard controlled by eye-blinks strength via EEG signal acquired using NeuroSky Mindwave headset.

A rarely used method is related to providing only one control signal, respectively the amplitude or strength of the eye-blink detected via EEG signal analysis. Moreover, the application proposed in this paper is different from the previous research work [5][6][7][8] because it does not involve a given time interval in order to constrain the user to execute eye-blinking only during this time window. The currently existing applications were usually implemented using procedural programming languages, such as: C, C++, C#, Java, .net or Matlab development environment.

The software application proposed in this paper was developed in LabVIEW graphical programming environment because it provides a friendly user interface so that its progress could be improved via an interactive feedback. Another important benefit offered by LabVIEW is the efficient programming method, similar to the natural way of thinking. Therefore, it facilitates the implementation of the working principle underlying virtual keyboard design, which was based on the Divide and Conquer algorithm. It was also taken into consideration the state-machine paradigm, which is a decision-based algorithm. Clearly defined conditions should be fulfilled in order to enable the transition across the states in a proper manner.

2. Hardware system

The hardware system used for the proposed application is the NeuroSky Mindwave headset [9][10], which has an embedded sensor that should be placed on the user's forehead to the frontal cerebral lobe called FP1. The reference necessary to close the electrical circuit is a clip that should be attached to the ear's lobe. NeuroSky headset is based on the ThinkGear chip, which comprises different algorithms for the electroencephalographic signal processing in order to provide the user the possibility to read certain values corresponding to the following parameters: attention level, meditation level and the strength or the amplitude of the eye-blinking. Another feature is related to acquiring of

raw EEG signal, which can be classified in different rhythms, such as: delta (0-4 Hz), theta (4-8 Hz), alpha (8-12 Hz), beta (12-30 Hz) and gamma (30-50 Hz). The sampling frequency of the NeuroSky headset is equal to 512 Hz. National Instruments provides developers with a toolkit created for NeuroSky [11][12][12], which could be used to enable the functionality of ThinkGear chip.

3. Software system

Regarding the software system proposed in this paper, both the working principle and the way of thinking underlying the implementation of eye-blinks controlled virtual keyboard are based on Divide and Conquer paradigm. This approach was also the starting point for designing similar virtual keyboards based on a brain-computer interface [5][6]. According to the previous scientific literature [7][8], LabVIEW has been rarely used to create a logical algorithm aimed for these types of applications. Therefore, it should be emphasized that the novel LabVIEW based solution proposed in this section could be used by students as valuable material to reinforce the knowledge taught by their professors.

3.1. A general overview of the available features of the virtual keyboard

The algorithm was based on the state-machine (Table 1) paradigm providing the following features:

- Counting the total number of intentional (voluntary) eye-blinks. An intentional eye-blink is detected when its strength or amplitude is characterized by a value higher than the initially set threshold parameter;
- Implementation of an instruction set aimed for enabling the SWITCH command across rows, half rows or characters (keys) when a single voluntary eye-blink was executed; Therefore, Divide and Conquer principle enabled the following transitions:
 - i) SWITCH across the rows of keys;
 - ii) SWITCH across the two parts (first and second half) of every row of keys;
 - iii) SWITCH across the characters or keys placed on the two parts (first and second half) of every row of keys.
- Implementation of an instruction set aimed for enabling the SELECT command in order to choose a row, a half row or a character (key) when two voluntary eye-blinks were performed Therefore, Divide and Conquer principle allowed the following tasks:
 - i) The selection/activation of a single row of keys;
 - ii) The selection/activation of a certain part (first or second half) of every row of keys;
 - iii) Inserting in the text box a specific character from either the first or the second half row.
- Implementation of an instruction set aimed for enabling the HIGHLIGHT action necessary to point out the current row, half row or key (character) followed by the typing/inserting the corresponding letter in the text box;
- Enabling CANCEL command associated with the execution of three voluntary eye-blinks. The boolean variables would be reinitialized so that the transition across rows is enabled again. Further, it is possible to select a specific row of keys.
- Enabling DELETE command triggered by recording of four voluntary eye-blinks;
- Enabling SPACE command determined by the execution of five voluntary eye-blinks.

3.2. A general overview on the LabVIEW based algorithm underlying the virtual keyboard design

In the proposed algorithm, there were defined five Boolean variables. They were initialized to the following values: SelectRow \leftarrow True; SelectHalfRow \leftarrow False; SelectChar \leftarrow False; InsertChar \leftarrow False; Trigger \leftarrow False. The meaning of these assignments is described in the Table 2.

There were also defined a series of numeric variables in order to increment the indexes associated to rows, half rows and characters (keys). There were used the below notations:

IndexRow = {1, 2, 3, 4, 5}. It can be assigned with integer values included in the interval [1, 5].

IndexHalfRow = {1, 2}. It can be assigned with either value 1 or value 2.

IndexChar = {1, 2, 3, 4, 5}. The maximum value is 3, 4 or 5. It depends on how many characters

are placed on each part (first and second half) of every row of keys.

Table 1. The next running state based on the assessment of the main condition – Blink Strength > Threshold Parameter of the ‘Count Eye-Blinks’ algorithm.

Current State	Blink Strength > Threshold Parameter	
	TRUE	FALSE
INIT	SWITCH <One eye-blink detected> Switch Command <ArraySize ← 1>	READY No Command
SWITCH	SELECT <Two eye-blinks detected> Select Command <ArraySize ← 2>	READY Switch Command
SELECT	CANCEL <Three eye-blinks detected> Cancel Command <ArraySize ← 3>	READY Select Command
CANCEL	DEL <Four eye-blinks detected> Delete Command <ArraySize ← 4>	READY Cancel Command
DEL	SPACE <Five eye-blinks detected> Space Command <ArraySize ← 5>	READY Delete Command

Table 2. The meaning of the opposite states (true or false) of the Boolean variables.

Variable	Value – Description of the variable state	
	True	False
SelectRow	Transition across rows is enabled.	Transition across rows is disabled.
SelectHalfRow	Transition across half rows is disabled.	Transition across half rows is enabled.
SelectChar	Transition across characters is disabled.	Transition across characters is enabled. It is also a condition necessary when someone intends to repeatedly introduce a specific character in the text box.
InsertChar	The sequence responsible for typing a character in the text box is enabled.	The sequence responsible for typing a character in the text box is disabled.
Trigger	It indicates that a character was already introduced in the text box during the last action. It is necessary in order to avoid the continuous typing of a character.	It proves that no character was previously introduced in the text box during the last action.

The transitions between rows of keys, half rows, respectively all characters included by each part of every row are indicated by highlighting those elements with blue colour.

If SelectHalfRow==True, the transition between the two parts (first and second half) of each row of keys is inactive (it is not available at the current moment of time). The further step consists of checking the SelectChar variable. Another Case Structure with two states was used in order to fulfil the instruction set for the two conditions: SelectChar == True and SelectChar == False.

If SelectChar == True, then the next step consists in checking the InsertChar variable. Both possibilities (InsertChar == True and InsertChar == False) were considered by using another Case Structure with two states. A character is introduced in the text box when both two voluntary eye-blinks were executed and the following conditions were performed: SelectRow ← False; SelectHalfRow ← True; SelectChar ← True; InsertChar ← True. The next step consists in finding the position of the character via numeric variables: IndexRow (five states), IndexHalfRow (two states), IndexChar (three / four / five states) – Table 3.

If SelectChar == False, then it is available the transition across the keys/buttons of each part (first and second half) of every row. For the case SelectChar == False, it was used the previously presented sequence of imbricated Case Structures, taking into account that currently the active key or

highlighted character is coloured with blue. There are also used the Property Nodes. Variables are attached to them in order to choose the suitable colour: blue (active) or grey (inactive).

If SelectHalfRow == False, then it is available the transition across the two parts (first and second half) of each row of keys/characters (Table 3).

It is important to mention that the programming sequence aimed to highlight with blue colour the rows, half rows, each key and respectively, typing a specific character in the text box, is running in parallel with the alternative processes corresponding to recording a single voluntary eye-blink (associated with switch command) and performing two intentional eye-blinks (associated with select command for choosing the previously stated elements).

Table 3. The character introduced in the text box according to the values of Boolean variables.

IndexRow	IndexHalfRow	Visual highlighting	IndexChar				
			1	2	3	4	5
1	1		Q	W	E	R	T
	2		Y	U	I	O	P
2	1		A	S	D	F	G
	2		H	J	K	L	
3	1		Z	X	C	V	
	2		B	N	M		
4	1		0	1	2	3	4
	2		5	6	7	8	9
5	1		?	!	.	,	:
	2		”	/	*	+	-

After a character was introduced in the text box, there are two possibilities: either one eye-blink is detected or two eye-blinks are executed.

Firstly, if one eye-blink was detected, then the currently active/enabled row remains highlighted/selected. Further, if another one intentional eye-blink is recorded, then it is enabled the transition to the next row of keys. The two instructions: Trigger ← False and SelectRow ← True show that the transition across rows of keys is available again. Otherwise, if two intentional eye-blinks were performed, then it is enabled the transition to the next character. If another two voluntary eye-blinks are executed, it is possible to introduce the last active character in the text box.

Secondly, if two intentional eye-blinks were detected, then the same character remains active and it is typed in the text box one more time. Still, if another two voluntary eye-blinks are detected, then the same character is repeatedly introduced in the text box.

4. Conclusion

This paper proposed a LabVIEW application that is aimed for helping people with disabilities by providing them with an efficient method of typing via a virtual keyboard, which can be controlled using the eye-blinks. Its working principle is based on Divide and Conquer paradigm.

One eye-blink is associated to switch command and it is related to the transition (alternative highlighting) across rows, half rows or keys (characters).

Two eye-blinks are associated to select command and they are related to the alternative selection of previously mentioned elements and finally, it is possible to type a character in the text box.

Regarding the novelty of this paper, by comparing with the previous research work [6], it is

important to notice that the user is not constrained by a given time interval. He only needs to execute the eye-blinks with a strength strong enough to exceed the threshold parameter that was initially set. Thanks to the attractiveness of LabVIEW environment, the graphical user-interface is friendly and it provides an intuitive feedback. Some of the useful features are related to the possibility of repeatable insertion of the same character and the advantages offered by delete and space keys.

Otherwise, the LabVIEW based virtual keyboard is associated with various applications, taking into account that it provides a simple to use and easy to learn working principle. Moreover, it can even raise the interest of healthy people.

Future research efforts will be focused on the development of an equivalent mobile virtual keyboard controlled by eye-blinks strength, which should be available on smartphones. Further, the aim of the mobile application is instant messages sending. In addition, a mechanism based on words prediction should also be implemented. Then, thorough experiments are required in order to prove or validate the usefulness of the brain-controlled virtual keyboard. Moreover, it is necessary to figure out to what extent the accuracy, precision and transfer rate of information should be improved.

5. References

- [1] Gembler F, Stawicki P and Volosyak I 2017 Suitable Number of Visual Stimuli for SSVEP-Based BCI Spelling Applications *Proc. of the 2017 International Work-Conf. on Artificial Neural Networks Advances in Computational Intelligence IWANN 2017* pp 441 – 452
- [2] Martínez-Cagigal, Santamaría E and Hornero R 2018 Controlling a Smartphone with Brain-Computer Interfaces: A Preliminary Study *Proceedings of the 2018 International Conference on Articulated Motion and Deformable Objects AMDO 2018* pp 34-43
- [3] Garcia L, Ron-Angevin R, Loubière B, Renault L, Masson G L, Lespinet-Najib and André J M 2017 A Comparison of a Brain-Computer Interface and an Eye Tracker: Is There a More Appropriate Technology for Controlling a Virtual Keyboard in an ALS Patient? *Proceedings of the 2017 International Work-Conference on Artificial Neural Networks Advances in Computational Intelligence IWANN 2017* pp 464 – 473
- [4] Saif M B and Felzer T 2016 Replacement of the Standard Computer Keyboard and Mouse by Eye Blinks *Proceedings of the 2016 International Conference on Computers Helping People with Special Needs ICCHP 2016* pp 457 – 463
- [5] López A, Ferrero F, Yangüela D, Álvarez C and Postolache O 2017 Development of a Computer Writing System Based on EOG in the *Sensors* **17** pp 1 - 20
- [6] Dobosz K and Stawski K 2017 Touchless Virtual Keyboard Controlled by Eye Blinking and EEG Signals *Proceedings of the 2017 International Conference on Man-Machine Interactions 5 ICMMI 2017* pp 52-61
- [7] Rakesh R and Sasidharan P T 2016 Utilization of EEG Signal as Virtual Keyboard for Physically Disabled in the *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)* **5** pp 1486 - 1491
- [8] Chamayil B, Singla R and Jha R 2010 Virtual keyboard BCI using Eye blinks in EEG *Proceedings of the 2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications 2010* pp 466-470
- [9] NeuroSky Store. Mindwave Mobile. Brainwave Sensing Headset, <https://store.neurosky.com/pages/mindwave>, available on 26.07.2018
- [10] NeuroSky. EEG Biosensors., <http://neurosky.com/biosensors/eeg-sensor/>, available on 27.07.2018
- [11] NeuroSky LabVIEW Drivers, , <https://store.neurosky.com/products/labview-drivers>, available on 27.07.2018
- [12] NeuroSky LabVIEW Driver, <https://forums.ni.com/t5/NI-Labs-Toolkits/NeuroSky-LabVIEW-Driver/ta-p/3520085>, available on 27.07.2018