



# Article Classification Strategies for P300-Based BCI-Spellers Adopting the Row Column Paradigm

Sofien Gannouni <sup>1,\*</sup>, Kais Belwafi <sup>1,2</sup>, Nourah Alangari <sup>1</sup>, Hatim AboAlsamh <sup>1</sup>, and Abdelfettah Belghith <sup>1</sup>

- <sup>1</sup> Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
- <sup>2</sup> Electrical Engineering and Computer Science Department, Khalifa University, Abu Dhabi P.O. Box 127788, United Arab Emirates
- \* Correspondence: gnnosf@ksu.edu.sa

Abstract: Acknowledging the importance of the ability to communicate with other people, the researcher community has developed a series of BCI-spellers, with the goal of regaining communication and interaction capabilities with the environment for people with disabilities. In order to bridge the gap in the digital divide between the disabled and the non-disabled people, we believe that the development of efficient signal processing algorithms and strategies will go a long way towards achieving novel assistive technologies using new human–computer interfaces. In this paper, we present various classification strategies that would be adopted by P300 spellers adopting the row/column paradigm. The presented strategies have obtained high accuracy rates compared with existent similar research works.

**Keywords:** brain-computer interface; P300 spellers; P300 row/column paradigm; n-class classification; ensemble classifiers



Citation: Gannouni, S.; Belwafi, K.; AboAlsamh, H.; Alangari, N.: Belghith, A. Classification Strategies for P300-Based BCI-Spellers Adopting the Row Column Paradigm. *Sensors* **2022**, *22*, 9159. https://doi.org/10.3390/s22239159

Academic Editors: Sidong Liu, Roneel V. Sharan and Hao Xiong

Received: 19 September 2022 Accepted: 22 November 2022 Published: 25 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). 1. Introduction

Severe motor disabilities strongly threaten patients' quality of life by deteriorating their ability to communicate with other people and making them lose their independence. The brain–computer interface (BCI) has emerged over the past few decades as the most important technology that aims to assist severely disabled people in regular everyday activities by offering them an efficient and easy-to-use muscle-independent pathway to control, communicate and interact with their environment [1]. BCI systems acquire and measure brain activity and translate it into control commands. Electroencephalography (EEG) is one of the most known non-invasive methods that is used by BCI systems to measure cerebral activity.

BCI-spellers were the first BCI applications that enabled people with severe motor disabilities to regain the ability to communicate with their environment. Various BCI-spellers have been proposed over the past few decades [2]. BCI-spellers are usually controlled either by motor imagery (MI) [3–6] or by event-related potentials (ERP) [2]. An MI-based BCI-speller derives its output from brain activity that is directly and consciously controlled by the user, independently from external events, by focusing on a specific mental task, i.e., imagining the movement of a muscle. An ERP-based BCI-speller derives its output from brain activity changes arising in reaction to external stimuli. ERPs are defined as changes in EEG signals during or after presenting external events (stimuli). ERP-based BCI-spellers are categorized to P300 [7–10] or steady-state visual evoked potential (SSVEP) spellers [11–16]. A P300 ERP is a positive peak in EEG that is elicited by visual or auditory stimuli. It appears about 300 ms after the stimuli start. SSVEP is characterized by positive and negative deflection in EEG signals arising in response to a visual stimulus that is flickering at a specific constant frequency. P300-based spellers have a relatively high information transfer rate and require minimal user training (compared to MI). Contrary to MI and SSVEP-based spellers, P300-based spellers enable BCI systems to spell a wide range of words and support a high number of commands. Unfortunately, the use of a limited number of imagined movements has considerably reduced the ability of MI-based spellers to support a wide range of symbols. Indeed, to the best of our knowledge, very few imagined movements such as leftor right-hand movement, foot movement, tongue movement and eye gazing have been considered by MI-based spellers in the literature. On the other hand, SSVEP-based spellers do not require subject training or system calibration. Unfortunately, it is very difficult to discriminate between SSVEP responses when the number of presented stimuli is increased. Indeed, in such a case, it is very difficult to discriminate between various flickering visual stimuli with close frequencies. Moreover, it has been observed that some people have low SSVEP responses which reduced drastically the accuracy of the BCI system to detect such responses. All these reasons made P300-based spellers the most commonly used spellers by BCI systems. This paper overviews P300 paradigms and P300-based spellers paradigms.

BCI-spellers usually correspond to graphical representations of letters, numbers, and symbols which are controlled using MI or ERPs for spelling and typing. Most of the research papers dealing with BCI-spellers focus mainly on the design and implementation of the Graphical User Interface (GUI) of the BCI-speller which is the front-end of the BCI system. However, more attention should be given to the back-end of the BCI system which consists of EEG signal processing algorithms to increase the performance of the BCI system. It is very attractive to design and develop strategies to boost signal processing algorithms. Some research studies are continuously searching to improve the accuracy of the P300 speller. For example, in [17], an adaptive channel selection method is proposed to enhance the classification accuracy of the P300 potential. It efficiently permits channel selection by introducing multiple kernel learning (MKL) to select the model, mapping the EEG signals in different acquisition channels into other feature spaces through different kernel functions. So, it constructs MKL by linear weighting and uses many training sessions to learn weight coefficients to select the optimal sampling set channel combination adaptively. Other research studies focus on the development of deep learning algorithms to maximize the prediction rate of the P300 potential. In [18], a capsule network algorithm called ERP-CapsNet is proposed to perform ERP detection in a BCI-speller application. The experimental results on BCI Competition datasets and the Akimpech dataset show that ERP-CapsNet achieves a classification accuracy of about 65% for three subjects.

The current work has been conducted within a funded research project which aims to promote the accessibility of people with severe disabilities by developing a series of integrated Brain controlled tools such as mail–client, Web browser and command–line interface [19]. The developed prototypes adopted a P300-based BCI-speller. In this regard, several strategies have been explored during the project to classify P300 responses and to predict users' desired symbols and/or commands. This manuscript describes all classification strategies we have explored during this project including a novel classification strategy that adopted a new partitioning approach to spread trials of the training dataset over an ensemble of classifiers. The results of the novel strategy outperformed all previous approaches including the winner of the BCI competition. To the best of our knowledge, there is no previous work that has adopted the proposed partitioning approach and the proposed classification strategy. Moreover, this work shows how a multi-class classification problem could be solved using two-phase classification strategies.

The remainder of this paper is organized as follows: Section 2 describes the different P300 paradigms and highlights the most commonly used P300 speller paradigms. Section 3 introduces the different annotations used in this paper. Section 4 describes different classification strategies for P300 spellers. Section 5 presents the results obtained by the discussed classification strategies and compares these results with those obtained by similar research works. Section 6 summarizes the current research work.

## 2. P300 Paradigms

The P300 is arising in response to an external event that appears as a positive deflection in voltage at the brain's parietal lobe. It can be measured 300 milliseconds after the stimulus starts [20]. Thus, to detect and record these deflections, the EEG electrodes should cover the brain's parietal lobe. Many prototypes have adopted this paradigm [19,21–28]. They are all characterized by a high degree of accuracy and require a short time of training. In our review of the literature, we identify three main P300 paradigms. The first one is the single-stimulus paradigm which includes one type of stimuli [29]. The second one is called the oddball paradigm, where the system shows a random sequence of two types of stimuli 'the target stimulus' that infrequently appears in a sequence of stimuli, and the 'non-target' stimulus, which appears more frequently [30,31]. The user focuses on a specific stimulus, "target", that represents the user's desire. The target stimulus elicits a P300 response while the other stimuli do not [30]. The last paradigm is the three-stimulus paradigm which contains three types of stimulus: target, standard, and distracter (also referred to as probes or novels). In the three-stimulus paradigm, the novel stimuli are presented infrequently and produce a P300 response that is different from the typical P300, which represents the response to the target (P3b), so this "novelty" P300 is called the (P3a) [29].

# 2.1. P300 Speller Paradigms

Different P300 Speller paradigms have been proposed in the literature and have developed distinct protocols. We hereby describe the most widely known P300 paradigms.

# 2.2. Row/Column Paradigm (RC)

Farwell and Donchin introduced the P300 row/column paradigm by presenting for subjects a  $6 \times 6$  character matrix [7]. The subject is invited to concentrate on the character he desires to spell. For the spelling of a single character, rows and columns of the matrix are periodically flashed in random order. A P300 response is elicited 300 ms after the row or the column that contains the character the user wants to spell is flashed. After analyzing the user's brain activity after each flash, P300 responses are detected, and consequently, the target character is identified as the intersection of the column and row that have elicited P300 responses.

#### 2.3. Single Character Paradigm (SC)

In this paradigm, only one character is flashed at a time. This paradigm has a low accuracy rate. Moreover, SC can be prone to the crowding effect [29].

## 2.4. Checkerboard Paradigm (CB)

A novel P300-based stimulus presentation paradigm is presented in [8] to handle the adjacency problems and double flash in RC. A matrix with dimension  $8 \times 9$  containing 72 elements is defined. The matrix is superimposed on a checkerboard. The checkerboard is split into two matrices, each with a dimension of  $6 \times 6$ . The first matrix contains the white cells from the checkerboard, and the second contains the black cells. Before each flash sequence, the checkerboard items will randomly populate the two matrices. So, the user sees a random group of six items flashing. The checkerboard layout controls the adjacency-distraction error because the adjacent cells will not be included in the same group. The rows of the matrix flash first randomly, then the columns flash. Any item will not flash again, at least for six intervening flashes; this will eliminate the double flash problem and avoid overlapping target epochs.

## 2.5. Region-Based Paradigm (RB)

In the RB paradigm [29,32], the screen was partitioned into seven regions. With 49 symbols (26 alphabet, 10 numeric, and 13 special characters). The 49 symbols are distributed among seven regions that are flashing randomly. The user focuses on the region that contains the target, then the system detects the region that contains the target by detecting the P300

response. Next, the seven symbols in the target region are distributed among the seven regions. Again the seven regions flash and then the system will identify the target symbol. This method provides more input characters. Moreover, avoid crowding effect and adjacency problem.

# 2.6. Comparison between RC, SC, CB, RB Paradigms

The authors in [33] have compared SC with RC P300 paradigms and state that only 55.3% of the trained subjects can spell with 100% accuracy in SC, whereas 72% of these subjects can spell with 100% in RC. Another study [29] has conducted experiments on six subjects (males 20-25 years old). Each participant completed the test with the SC, RC, and RB P300 paradigms. The participants were asked to spell two words. The total accuracy was 95% for RB, 85% for RC, and 72% for SC. In [8], experiments have been conducted on 18 participants (11 men, 7 women-2 women, and one man with ALS). Each participant completed two sessions of the experiments, one for RC and a second for CB. The CB accuracy was 91.52%, whereas the RC accuracy was 77.34. Table 1 compares between the different P300 speller paradigms.

Paradigm	RC	SC	СВ	RB	
Accuracy	Medium	Low	Very High	High	
Adjacency problem	$\checkmark$	$\checkmark$	×	X	
Crowding Effect	$\checkmark$	$\checkmark$	$\checkmark$	X	
Double flash	$\checkmark$	X	×	X	

Table 1. Comparison between the P300 Speller Paradigms.

✓ exist X not exist.

## 3. Terminology and Annotations

Let us consider a matrix of symbols, called commands, denoted M, of dimension  $n \times m$ ; *n* and *m* correspond to the numbers of rows and columns of the matrix *M*, respectively.

	$[C_1^1]$	•••	$C_1^j$	•••	$C_1^m$	
	:		÷		:	
M =	$C_i^1$	•••	$C_i^j$	• • •	$C_i^m$	(1)
	:		÷		:	
	$\lfloor C_n^1 \rfloor$		$C_n^j$		$C_n^m$	

The intensification denoted *I* is the process of flashing (intensifying the luminosity) all symbols of a given row or column of the matrix M. Figure 1 shows an illustration of the intensifications of the third column and of the second row of a matrix of symbols M, respectively. An intensification *I* elicits a single post-stimulus signal denoted  $\xi(I)$ .

¶	A G	B	C	D	E K	¶ F	A G	ВH	C I	DJ	ЕK
L	Μ	Ν	0	Ρ	Q	L	Μ	Ν	0	Ρ	Q
R	S	Т	U	٧	W	R	S	Т	U	V	W
Х	Υ	Ζ		1	2	X	Y	Z		1	2
3	4	5	6	7	8	3	4	5	6	7	8

Figure 1. Intensification of the second row and the third column of a matrix of symbols.

A sequence of intensification denoted by *S* represents an ordered collection of intensifications, where all rows and columns of *M* are intensified randomly once. Thus, *S* is formed by (n + m) distinct intensifications, denoted  $S = \langle I_1^i, I_2^j, \ldots, I_{n+m}^k \rangle$ . Each component *I* corresponds to an intensification of a row/column of *M*. In fact,  $I_i^j$  means that the *i*th intensification of the sequence *S* occurs in the *j*th row/column of the matrix *M*. We notice that rows and columns of *M* are indexed using unique serial numbers, as shown in the following matrix (Equation (2)).

$$1 \cdots j \cdots m$$

$$m+1 \Rightarrow \begin{pmatrix} \Downarrow & \Downarrow & \Downarrow & \Downarrow \\ C_1^1 \cdots & C_1^j & \cdots & C_1^m \\ \vdots & \vdots & \vdots & \vdots \\ C_i^1 \cdots & C_i^j & \cdots & C_i^m \\ \vdots & \vdots & \vdots & \vdots \\ C_n^1 \cdots & C_n^j & \cdots & C_n^m \end{pmatrix}$$

$$(2)$$

A single sequence of intensification *S* elicits (n + m) post-stimulus signals denoted by  $\xi(S)$  corresponding to the following ordered collection of post-stimulus signals:

• 
$$\xi(S) = \langle \xi(I_1^i), \xi(I_2^i), \cdots, \xi(I_{n+m}^k) \rangle$$

•  $|\xi(S)| = (n+m)$ 

For the selection, denoted by  $\sigma$ , of a single symbol/command  $c_i^l$  of M,  $\alpha$  sequences of intensifications are performed successively. As such, every row/column of M is intensified  $\alpha$  times during the same selection  $\sigma$  but in random orders.

$$\sigma = \bigcup_{i=1}^{\alpha} S_i \tag{3}$$

 $S_i$  is the *i*th sequence of the selection  $\sigma$ . Thus, a selection  $\sigma$  of a single command elicits a set of post-stimulus signals denoted  $\xi(\sigma)$ :

$$\xi(\sigma) = \bigcup_{i=1}^{\alpha} \xi(S_i) = \bigcup_{i=1}^{\alpha} \langle \xi(I_{1,i}^x), \xi(I_{2,i}^y), \cdots, \xi(I_{(n+m),i}^z) \rangle$$
(4)

 $I_{i,j}^k$  corresponds to the *i*th intensification of the *j*th sequence  $S_j$  and which occurs in the *k*th row/column of the matrix of symbols *M*.  $\xi(\sigma)$  is composed of  $\alpha \times (n + m)$  poststimulus signals.

$$|\xi(\sigma)| = \sum_{i=1}^{\alpha} |\xi(S_i)| = \sum_{i=1}^{\alpha} (n+m) = \alpha \times (n+m)$$
(5)

#### 4. Classification Strategies

The classification problem addressed in this paper is a multi-class classification problem and it is solved in two steps. In the first step, a binary classification problem is addressed and it consists of predicting if an EEG post-stimulus signal corresponds to a P300 response or not. In the second phase, a multi-class classification problem is addressed and it consists of identifying the symbol that the user wants to spell.

Various design choices could be considered to develop a reliable and efficient solution. We describe hereafter the most relevant choices to build an efficient classification strategy:

 A single classifier or an ensemble of classifiers: A classifier is a machine learning model that allows us to identify, based on a set of attributes called features, which group, called a class, an object belongs to. Advances in machine learning have shown that an ensemble of classifiers effectively improves the classification accuracy of a single classifier. An ensemble of classifiers applies fusion techniques to combine predictions of single classifiers called base learners. However, the size of the training dataset is a critical issue in designing a classification strategy based on an ensemble of classifiers rather than on a single classifier. Indeed, the size of the training dataset impacts the performance of a machine-learning model. Large datasets lead to higher accuracy classification while small datasets degrade the system performance due to over-fitting.

- A heterogeneous or a homogeneous ensemble of classifiers: If the training dataset is large enough to adopt an ensemble of classifiers, the most important question that should be raised at that stage is: should we adopt distinct classifiers or distinct instances of the same classifier. Homogeneous ensembles develop the models of the base learners using the same classification algorithm. In heterogeneous ensembles, the base learners adopt distinct classification algorithms.
- Voting or stacking: In the case of an ensemble of classifiers, the final decision is obtained by applying a voting or a stacking fusion technique on the different labels predicted by the various base learners. Voting allows us to identify the class that has been predicted (voted) by the majority of base learners. Stacking uses the outputs of the base learners as features to train another classifier which is called a meta-learner, which will make the final decision.
- Replication or fragmentation of the training dataset: In the case of an ensemble of classifiers being adopted, there are two approaches to spread the training dataset over the base learners. Fragmentation allows us to decompose the training dataset into distinct fragments each of which will be used to train a single base learner. Replication allows us to train every base learner using the whole dataset. Replication is applicable only in case the base learners implement distinct classification algorithms, i.e., in the case of heterogeneous ensembles of classifiers.

Figure 2 presents the different classification strategies investigated across this research work to predict a symbol from a matrix of symbols. These strategies are presented and discussed in the following sub-sections.



Figure 2. Classification strategies.

## 4.1. Single Classifier Strategy

The whole training dataset is used to train a 2-class classifier in the first phase. As such, the classifier is trained to predict if a post-stimulus EEG signal corresponds to a P300 response or not. For example, given a post-stimulus signal corresponding to an

intensification  $I_{i,j}^k$  of a given row or column of M, the prediction method denoted  $\rho$  returns a value ranging between 0 and 1, which corresponds to the probability that the signal  $\xi(I)$  is a P300 response or not.

$$\rho(\xi(I_{i,i}^k)) = v \text{ such that } v \in [0,1]$$
(6)

Given a sequence of intensifications  $S_i$ , the parsing method denoted  $\tau$  returns a row vector, denoted by  $U_i$ , the values of which are obtained using the prediction method  $\rho$ .

$$\tau(\xi(S_i)) = U_i \tag{7}$$

 $U_i$  is a row vector composed of the elements  $\langle u_i^1, u_i^2, \cdots, u_i^{n+m} \rangle$  such that:

$$u_i^j = \rho(\xi(I_{x,i}^j)) \tag{8}$$

 $u_i^j$  is the probability that the intensification, whatever its order/rank, of the *j*th row/column of *M* that happens during the *i*th sequence of intensifications  $S_i$  has elicited a P300 response or not. Thus,  $\tau(\xi(S_i))$  is computed as follows:

$$\tau(\xi(S_i)) = <\rho(\xi(I_{x,i}^1)), \rho(\xi(I_{y,i}^2)), \cdots, \rho(\xi(I_{z,i}^{n+m})) >$$
(9)

Every row and column of M is intensified once during the same sequence. Hence, the parsing method  $\tau$  identifies which intensifications of a given sequence  $S_i$  have elicited a P300 response and which have not. It will determine for every row/column of M whether its corresponding intensification that occurs during the given sequence  $S_i$  has elicited a P300 response or not.

We remind that during the same selection  $\sigma$ , every row and column of M is intensified  $\alpha$  times. So, given a selection  $\sigma$  of a single character, the corresponding  $\alpha$  sequences of intensifications are processed sequentially by the parsing method  $\tau$  leading to  $\alpha$  row vectors each of which corresponds to  $\tau(\xi(S_i))_{1 \le i \le \alpha}$ . The  $\alpha$  row vectors  $U_i = \tau(\xi(S_i))_{1 \le i \le \alpha}$  are then used to average the probability that the intensifications of rows and columns of M happening during the selection  $\sigma$  have elicited P300 responses. These probabilities are computed using the following prediction function  $\Psi$ :

$$\Psi(\xi(\sigma)) = \frac{1}{\alpha} < \sum_{i=1}^{\alpha} \tau(\xi(S_i)) >$$

$$= \frac{1}{\alpha} < \sum_{i=1}^{\alpha} U_i >$$

$$= \frac{1}{\alpha} < \sum_{i=1}^{\alpha} u_i^1, \sum_{i=1}^{\alpha} u_i^2, \cdots, \sum_{i=1}^{\alpha} u_i^{n+m} >$$

$$= \frac{1}{\alpha} < \sum_{i=1}^{\alpha} \rho(\xi(I_{x,i}^1)), \sum_{i=1}^{\alpha} \rho(\xi(I_{y,i}^2)), \cdots, \sum_{i=1}^{\alpha} \rho(\xi(I_{z,i}^{n+m})) >$$
(10)

As such  $\Psi(\xi(\sigma))$  returns a row vector denoted by *V* as follows:

$$\Psi(\xi(\sigma)) = \langle v^1, v^2, \cdots, v^m, v^{m+1}, \cdots, v^{n+m} \rangle$$
(11)

such that

$$v^{j} = \frac{1}{\alpha} \times \sum_{i=1}^{\alpha} \rho(\xi(I_{x,i}^{j}))$$
(12)

Given a selection  $\sigma$ , we can identify the user's desired symbol by maximizing the results of the prediction function  $\Psi(\xi(\sigma))$ . We select the row *x* and the column *y* of *M*' that

have most probably elicited P300 responses. We notify that y corresponds to the column number maximizing the score  $v^y$ .

$$v^{y} = Max_{i=1}^{m}(v^{j}) \text{ where } v^{j} = \frac{1}{\alpha} \times \sum_{i=1}^{\alpha} \rho(\xi(I_{x,i}^{j}))$$
 (13)

*x* is the number of the row that maximizes the score  $v^{m+x}$ .

$$v^{m+x} = Max_{i=1}^{n}(v^{m+i})$$
(14)

Thus, we consider that the symbol  $c_x^y$  of *M* is most probably the user's desired symbol.

## 4.2. Ensemble Classifiers Strategy

In the case of ensemble classifiers, two options are possible. The first option is called Heterogeneous Ensemble Classifiers, which consists of training different classifiers using the same training dataset. The second option is called Homogeneous Ensemble Classifiers, which consists of training different instances of the same classifier using different partitions of the training dataset.

Let us consider a training dataset, denoted *D*, comprising post-stimulus signals corresponding to the selection of  $\beta$  symbols (commands)  $c_i^j$  of *M*.

$$D = \bigcup_{j=1}^{\beta} \xi(\sigma^j) = \bigcup_{j=1}^{\beta} \bigcup_{i=1}^{\alpha} \xi(S_i^j)$$
(15)

where  $S_i^j$  is the *i*th sequence  $S_i$  that occurs during the *j*th selection. So, the dataset *D* consists of  $\beta \times \alpha \times (n + m)$  post-stimulus training signals.

$$|D| = \sum_{i=1}^{\beta} |\xi(\sigma_i)| = \sum_{i=1}^{\beta} \alpha \times (n+m) = \beta \times \alpha \times (n+m)$$
(16)

# 4.2.1. Heterogeneous Ensemble Classifiers Strategy

This step is often known as decision-level fusion, where different modalities are utilized for separate training models. An aggregation function is used at the end to determine the final decision by combining the different prediction results of distinct models. In such a case, the whole training dataset *D* is used to train *N* different 2-class classifiers, leading to an ensemble of distinct classifiers. Every classifier  $C_{i:1\cdots N}$  is trained to predict if a signal contains a P300 response or not. Every classifier  $C_{i:1\cdots N}$  builds its prediction model. Thus every classifier  $C_{i:1\cdots N}$  will customize the prediction method  $\rho$ , denoted  $\rho_{i:1\cdots N}$ , according to its prediction model. The total number of trials used to train and build the model of every classifier is defined using the following expression:

$$Total number Of trials = \beta \times \alpha \times (n+m)$$
(17)

There are two different approaches to building the final decision of this classification strategy: non-weighted voting and weighted voting.

# a. Non Weighted Voting

In this case, the different classifiers play the same role to predict the selected symbol/command. Given a selection  $\sigma$ , the corresponding  $\alpha$  sequences of intensifications are parsed simultaneously by the *N* distinct classifiers. Every classifier  $C_{i:1\cdots N}$  calculates  $\Psi_i(\xi(\sigma))$  and returns a row vector denoted  $V_i$  as follows:

$$\Psi_{i}(\xi(\sigma)) = \langle v_{i}^{1}, v_{i}^{2}, \cdots, v_{i}^{m}, v_{i}^{m+1}, \cdots, v_{i}^{n+m} \rangle$$
(18)

such that

$$v_i^j = \frac{1}{\alpha} \times \sum_{k=1}^{\alpha} \rho_i(\xi(I_{x,k}^j))$$
(19)

The results  $V_i = \Psi_i(\xi(\sigma))_{i:1\cdots N}$  obtained by the different classifiers  $C_{i:1\cdots N}$  are then combined to calculate the global decision as follows:

$$\Psi_{global}(\xi(\sigma)) = \frac{1}{N} \times \sum_{i=1}^{N} \Psi_i(\xi(\sigma))$$
(20)

 $\Psi_{global}(\xi(\sigma))$  returns a row vector denoted  $R = \langle r^1, r^2, \cdots, r^{n+m} \rangle$  such that

$$r^{j} = \frac{1}{N} \times \sum_{i=1}^{N} \frac{1}{\alpha} \times \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j})) = \frac{1}{N} \times \frac{1}{\alpha} \times \sum_{i=1}^{N} \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j}))$$
(21)

 $r^{j}$  is the probability that the intensification which occurs in the *j*th row/column of *M* has elicited a P300 response.

Given a selection  $\sigma$ , we can determine the user's desired symbol by maximizing the results of the prediction function  $\Psi_{global}(\xi(\sigma))$ . Using the values of the row vector  $W = \Psi_{global}(\xi(\sigma))$  we identify the column and the row that have most probably elicited P300 responses. Considering *y* is the number of the column of *M* that has most probably elicited P300 responses. Thus, *y* is the column number that maximizes the score  $r^y$ .

$$r^{y} = Max_{i=1}^{m}(r^{j}) \tag{22}$$

Let *x* be the number of the row of *M* that has most probably elicited P300 responses. *x* is the number of the row that maximizes the score  $r^{m+x}$ .

$$^{m+x} = Max_{i=1}^{n}(r^{m+i})$$
 (23)

Thus, we consider that the symbol  $c_x^y$  of M is the most probably user's desired symbol. b. Weighted Voting

In this case, we apply the same strategy as the non-weighted voting approach except that the decisions of the involved classifiers are waved to generate the global decision. Every classifier  $C_{i:1\cdots N}$  is assigned a weight denoted  $w_i$  that corresponds to the accuracy of  $C_{i:1\cdots N}$  during the training phase. As such the results obtained by the different classifiers  $C_{i:1\cdots N}$  are combined to predict the global decision using the following prediction function  $\Psi_{global}(\xi(\sigma))$  to calculate the global decision as follows:

$$\Psi_{global}(\xi(\sigma)) = \frac{1}{N} \times \sum_{i=1}^{N} w_i \times \Psi_i(\xi(\sigma))$$
(24)

 $\Psi_{global}(\xi(\sigma))$  returns a row vector denoted  $R = \langle r^1, r^2, \cdots, r^{n+m} \rangle$  such that

$$r^{j} = \frac{1}{N} \times \sum_{i=1}^{N} w_{i} \times \frac{1}{\alpha} \times \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j})) = \frac{1}{N} \times \frac{1}{\alpha} \times \sum_{i=1}^{N} w_{i} \times \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j}))$$
(25)

 $r^{j}$  is the probability that the intensification that occurs in the *j*th row/column of *M* has elicited a P300 response.

Given a selection  $\sigma$ , we can determine the user's desired symbol by maximizing the results of the prediction function  $\Psi_{global}(\xi(\sigma))$ . Using the values of the row vector  $R = \Psi_{global}(\xi(\sigma))$  we identify the column and the row that have most probably elicited P300 responses. Let *y* and *x* be the numbers of the column and the row of *M* that has most probably elicited P300 responses, respectively. *y* and *x* are obtained by applying Equations (22) and (23), respectively.

Thus, we consider that the symbol  $c_x^y$  of *M* is the most probably user's desired symbol.

## 4.2.2. Homogeneous Ensemble Classifiers Strategy

In this case, the training dataset *D* is split into disjoint partitions (portions). Every partition is used to train an instance of the same 2-class classifier, leading to an ensemble of homogeneous classifiers. Every classifier is as such trained to predict if a signal contains a P300 response or not. The training dataset *D* could be split into two different approaches: Vertical or Horizontal partitioning.

# a. Horizontal Partitioning

We remind that the dataset contains  $\beta$  selections  $\sigma_i$ . Every selection  $\sigma_i$  is composed of  $\alpha$  sequences of intensifications denoted by  $S_i^j$ .  $S_i^j$  represents the *j*th sequence of the *i*th selection.

In the horizontal partitioning strategy, the signals of the training dataset *D* are spread over  $\alpha$  partitions denoted  $\pi_{i:1\cdots\alpha}$  each of which is defined as follows:

$$\pi_{i:1\cdots\alpha} = \bigcup_{j=1}^{\beta} \xi(S_j^i) \tag{26}$$

A partition  $\pi_{i:1\cdots\alpha}$  contains all *i*th sequences of intensifications of the  $\beta$  selections. As such, the partition  $\pi_1$  contains all first sequences of the  $\beta$  selections, the partition  $\pi_2$  contains all second sequences of the  $\beta$  selections and so on. The obtained partitions have to satisfy the following properties:

$$\bigcup_{i=1}^{\alpha} \pi_i = D \text{ and } \bigcap_{i=1}^{\alpha} \pi_i = \emptyset$$
(27)

Every partition  $\pi_i$  is composed of  $\beta$  sequences of intensifications. Hence, a partition  $\pi_i$  contains  $\beta \times (n + m)$  post-stimulus signals.

$$|\pi_i| = \sum_{j=1}^{\beta} |\xi(S_j^i)| = \sum_{j=1}^{\beta} n + m = \beta \times (n+m)$$
(28)

Every partition  $\pi_i$  is used to train a single 2-class classifier  $C_{i:1\cdots\alpha}$ , leading to an ensemble of  $\alpha$  classifiers. Every classifier  $C_{i:1\cdots\alpha}$  is as such trained to predict if a signal corresponds to a P300 response or not. Thus, the total number of trials used to train every classifier is defined using the following expression:

$$Total number Of trials = \beta \times (n+m)$$
(29)

Given a selection  $\sigma$ , the corresponding  $\alpha$  sequences of intensifications are parsed simultaneously by the  $\alpha$  classifiers. Every classifier processes a single sequence of the  $\alpha$  sequences of  $\sigma$ . Thus, a classifier  $C_i$  will parse the  $i_{th}$  sequence  $S_i$  of  $\sigma$ . Every classifier  $C_i$  calculates  $\tau_i(\xi(S_i))$  as follows:

$$\pi_i(\xi(S_i)) = <\rho_i(\xi(I_{x,i}^1)), \rho(\xi(I_{y,i}^2)), \cdots, \rho(\xi(I_{z,i}^{n+m})) >$$
(30)

Every classifier  $C_i$  returns a row vector denoted  $V_i$  whose values are calculated as follows:

$$v_i^j = \rho_i(\xi(I_{x,i}^j)) \tag{31}$$

 $v_i^j$  is the probability that the intensification, whatever its order/rank, of the *j*th row/column of *M* that happens during the *i*th sequence of intensifications  $S_i$  has elicited a P300 response or not. The different results (row vectors)  $V_i = \tau_i(\xi(S_i))_{i:1\cdots\alpha}$  calculated by the different classifiers  $C_{i:1\cdots\alpha}$  are then combined to calculate the global decision as follows:

$$\Psi_{global}(\xi(\sigma)) = \frac{1}{\alpha} \times \sum_{i=1}^{\alpha} \tau_i(\xi((S_i)))$$
(32)

 $\Psi_{global}(\xi(\sigma))$  returns a row vector denoted  $R = \langle r^1, r^2, \cdots, r^{n+m} \rangle$  such that

$$r^{j} = \frac{1}{\alpha} \times \sum_{i=1}^{\alpha} \rho_{i}(\xi(I_{x,i}^{j}))$$
(33)

 $r^{j}$  is the probability that the intensifications that occur in the *j*th row/column of *M* have elicited P300 responses.

Given a selection  $\sigma$ , we can identify the column and the row that have most probably elicited P300 responses by maximizing the results of the prediction function  $\Psi_{global}(\xi(\sigma))$ . Let *y* and *x* be the numbers of the column and the row of *M* that has most probably elicited P300 responses, respectively. *y* and *x* are obtained by applying Equations (22) and (23), respectively. Thus, we consider that the symbol  $c_x^y$  of *M* is the most probable user's desired symbol.

# b. Vertical Partitioning

In the vertical partitioning strategy, the selections are spread over a set of classifiers. So, we defined a collection composed of N different instances of the same classifier. Then, we split equally the  $\beta$  selections over the different instances  $C_{i:1\cdots N}$ . Thus, every classifier  $C_{i:1\cdots N}$  will be trained on a subset of the training dataset composed of  $\frac{\beta}{N}$  selections. As such, every instance  $C_{i:1\cdots N}$  of the classifiers will be assigned a partition defined as follows:

$$\pi_{i:1\cdots N} = \bigcup_{j=(i-1)\times\frac{\beta}{N}+1}^{i\times\frac{\beta}{N}} \xi(\sigma_j)$$
(34)

A partition  $\pi_{i:1\cdots N}$  contains  $\frac{\beta}{N}$  successive selections and satisfies the properties described by expression (33).

Every partition  $\pi_i$  contains  $\frac{\beta}{N} \times \alpha \times (n+m)$  post-stimulus signals.

$$|\pi_i| = \sum_{j=(i-1)\times\frac{\beta}{N}+1}^{i\times\frac{\beta}{N}} |\xi(\sigma_j)| = \sum_{j=(i-1)\times\frac{\beta}{N}+1}^{i\times\frac{\beta}{N}} \sum_{k=1}^{\alpha} |\xi(S_j^k)| = \frac{\beta}{N} \times \alpha \times (n+m)$$
(35)

Every partition  $\pi_i$  is used to train a 2-class classifier  $C_{i:1...N}$ , leading to an ensemble of N classifiers. Every classifier  $C_{i:1...N}$  is as such trained to predict if a signal contains a P300 response or not. Thus, the total number of trials used to train every classifier is defined using the following expression:

Total number Of trials 
$$= \frac{\beta}{N} \times \alpha \times (n+m)$$
 (36)

Given a selection  $\sigma$ , the corresponding  $\alpha$  sequences of intensifications are parsed simultaneously by the *N* classifiers. Every classifier  $C_{i:1...N}$  calculates  $\Psi_i(\xi(\sigma))$  and returns a row vector denoted  $V_i$  (Equation (18)) whose values are calculated using Equation (19).

The different results (row vectors)  $V_i = \Psi_i(\xi(\sigma))_{i:1\cdots N}$  calculated by the different classifiers  $C_{i:1\cdots N}$  are then combined to calculate the global decision as follows:

$$\Psi_{global}(\xi(\sigma)) = \frac{1}{N} \times \sum_{i=1}^{N} \Psi_i(\xi(\sigma))$$
(37)

 $\Psi_{global}(\xi(\sigma))$  returns a row vector denoted  $R = \langle r^1, r^2, \cdots, r^{n+m} \rangle$  such that

$$r^{j} = \frac{1}{N} \times \sum_{i=1}^{N} \frac{1}{\alpha} \times \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j})) = \frac{1}{N} \times \frac{1}{\alpha} \times \sum_{i=1}^{N} \sum_{k=1}^{\alpha} \rho_{i}(\xi(I_{x,k}^{j}))$$
(38)

 $r^{j}$  is the probability that the intensification that occurs in the *j*th row/column of *M* has elicited a P300 response.

Given a selection  $\sigma$ , we can identify the column and the row that have most probably elicited P300 responses by maximizing the results of the prediction function  $\Psi_{global}(\xi(\sigma))$ . Let *y* and *x* be the numbers of the column and the row of *M* that has most probably elicited P300 responses, respectively. *y* and *x* are obtained by applying Equations (22) and (23), respectively. Thus, we consider that the symbol  $c_x^y$  of *M* is the most probable user's desired symbol.

# 5. Results and Discussion

# 5.1. Description of the Testing Dataset

A dataset provided by BCI competition, in [34], is used to test and check the effectiveness of the proposed approach and for comparison purposes. It contains the EEG signals of two subjects trying to select 85 characters (85 selections) during five different sessions. Each selection is recorded from the brain using 64 channels, most of which are put in the brain's parietal lobe. Every character spelling corresponds to  $180 = 12 \times 15$  post-stimulus labeled signals. So, each subject's dataset is composed of 15,300 post-stimulus in total.

The testing dataset is composed of 100 characters spelling, equivalent to  $18,000 = 180 \times 100$  post-stimulus signals. The diverse parameters of the selected dataset using the previous approach are summarized in Table 2.

Table 2. Description of the benchmarking dataset.

	Notation	Formula	Value
The dimensions of the command Matrix <i>M</i>	n,m		6,6
The number of post-stimulus signals during a single sequence of intensifications <i>S</i> .	$ \zeta(S) $	$ \zeta(S)  = n + m$	12
The number of sequences of intensifications <i>S</i> in a single selection $\sigma$ .	α		15
Total number of post-stimulus signals during a single selection.	$ \zeta(\sigma) $	$ \zeta(S)  = \alpha \times (n+m)$	180
The number of selections (per subject) of the training dataset.	β		85
Total number of post-stimulus signals (per subject) of the training dataset.	D	$ D  = \beta \times \alpha \times (n+m)$	15,300
The number of selections (per subject) of the testing dataset.	β′		100
Total number of post-stimulus signals (per subject) of the testing dataset.	D'	$ D'  = \beta' \alpha(n+m)$	18,000

Performing a benchmark test with only two subjects is insufficient in general. However, every subject was invited to select 100 characters which correspond to 18,000 trials (EEG signals) per subject. The number of trials is enough to make a significant benchmark test.

## 5.2. EEG Signals Pre-Processing and Features Extraction

We are interested in this work only in EEG signals that appeared after each intensification. As mentioned above, the evoked potentials appear 300 ms after the stimulus. Thus, samples between 0 to 667 ms posterior to the beginning of intensification are extracted for every channel. The sampling rate of the EEG signals was set to 240 Hz. To the best of our knowledge, this epoch is large enough to acquire efficient time features. The selected epoch is bandpass filtered using an infinite impulse response (IIR) filter. The order of the filter is set to eight as it is sufficient to keep the spectrum components between 0.1 and 10 Hz. Afterward, the filtered signals were decimated according to the high cut-off frequency, 10 Hz. The decimation process reduces the sampling frequency of a signal to a lower sampling frequency that differs from the original frequency. Decimation is also known as down-sampling. So, the extracted signal comprises only 14 samples at each electrode. Finally, each post-stimulus signal is transformed as a single vector composed of 896 =  $14 \times 64$  samples. As mentioned above, the EEG signal has been filtered with an 8-order bandpass filter the cut-off frequency. The decimation process reduces the sampling frequency of a signal to a lower sampling frequency of a signal to a lower sampling frequency of a signal to a lower sampling frequency that differs from the original frequency. The decimation process reduces the sampling frequency of a signal to a lower sampling frequency that differs from the original frequency. Decimation also is known as down-sampling. Therefore, the extracted signal from a single channel is composed of 14 samples.

## 5.3. Results

The various strategies analyzed in this work have been evaluated using the dataset mentioned above. For every classification strategy, Table 3 shows the total number of psst-stimulus signals used for training the classifiers.

The benchmark dataset contains two subsets of trials. The first subset is composed of 85 selections per subject which correspond to 15,300 post-stimulus signals. This subset is used to train the classifier(s). Table 3 summarizes how these trials are spread over the classifiers according to the classification strategy. The second subset is composed, as mentioned in Table 2, of 100 selections per subject which correspond to 18,000 poststimulus signals per subject too. The second subset is used for testing the accuracy of the classification strategies.

	Number of	Total Number of Post-Stimulus Signals per Classi		
	Classifiers	Formula	Value	
Ensemble Heterogeneous Classifiers Strategy	4	$ D  = \beta \times \alpha \times (n+m)$	15,300	
Ensemble Homogeneous Classification Strategy with Vertical Partitioning	17	$ D  = \frac{\beta}{N} \times \alpha \times (n+m)$	900	
Ensemble Homogeneous Classification Strategy with Horizontal Partitioning	15	$ D  = \beta \times (n+m)$	1020	

Table 3. Settings of the different classification strategies.

Given a selection  $\sigma$ , the  $\alpha$  sequences of intensifications are parsed by the classifier(s) of a given strategy. The distinct (n + m) intensifications of every sequence are processed separately by the classifier(s) to determine which ones correspond to a P300 response and which do not. For every sequence of intensifications, the classifier(s) aggregates the (n + m) predictions and returns a row vector. Every strategy consolidates the distinct  $\alpha$  row vectors, using appropriate formulas, to identify which character corresponds most probably to the selection  $\sigma$ .

Various classifiers are applied during the first phase of the classification problem to analyze and select the appropriate one for our application. For example, we have evaluated the linear discriminant analysis (LDA) [35], partial least squares regression (PLS) [36], logistic regression (REG) [37], and support vector machine (SVM) [19]. These algorithms were evaluated using the BCI competition dataset containing the recording of two subjects, A and B. Every classifier has been trained using 15,300 post-stimulus signals and tested using

18,000 post-stimulus signals. Table 4 summarizes the classification accuracy of each classifier, which represents the number of trials classified correctly. The maximum classification accuracy was reached using the SVM algorithm, where for subject A the accuracy achieved was 96% and 93% for subject B, respectively. Thus, the average classification accuracy using this technique is about 94.5%. Regarding subject A, the classification accuracy achieved was 93%, 94%, and 94% using LDA, PLS, and REG, respectively. The SVM fails to maintain high accuracy for subject B, where the highest accuracy was obtained using the PLS and REG classifiers.

Subject	LDA	SVM	PLS	REG
Subject A	93	96	94	94
Subject B	92	93	94	94

Table 4. Average accuracy of the single classification strategy.

SVM, LDA, PLS and REG classifiers have been fused to evaluate the performance of the ensemble heterogeneous classification strategy. Table 5 shows the accuracy rate of the ensemble classifiers for subjects A and B. The ensemble classifiers have been trained using 15,300 post-stimulus signals and have been tested using 18,000 post-stimulus signals. Using the non-weighted voting technique, the ensemble classifiers reach an accuracy rate of 88% and 91% for subjects A and B, respectively. The weighted voting techniques enhance the accuracy of the ensemble classifiers by achieving an accuracy rate of 89% for subject A. For subject B, the performance remains the same.

Table 5. Average accuracy of the ensemble heterogeneous classification strategy.

	Accuracy (%)				
Subject	Weighted Voting	Non-Weighted Voting			
Subject A	89	88			
Subject B	91	91			

The ensemble homogeneous classification strategy was evaluated by fusing the predictions of different instances of the SVM classifier. In the case of vertical partitioning, we decided to split the training dataset into partitions each of which was composed of post-stimulus signals of five successive selections, leading to 17 different partitions. As such, every partition corresponded to 900 post-stimulus signals. Every partition has been used to train a single instance of the SVM classifier. Table 6 shows the accuracy rate of the ensemble classifiers for subjects A and B. The accuracy rate of the ensemble classifiers using the vertical partitioning reached an accuracy rate of 96% and 93% for subjects A and B, respectively.

**Table 6.** Average accuracy of the ensemble homogeneous classification strategy using vertical or horizontal partitioning.

	Accuracy (%)				
Subject	Horizontal Partitioning	Vertical Partitioning			
Subject A	98	96			
Subject B	96	93			

In the case of horizontal partitioning, the training dataset was split into 15 partitions each of which was composed of post-stimulus signals that correspond to sequences that happen in the same order during the selections. Partition 1 contained post-stimulus signals of all first sequences. Partition 2 contained post-stimulus signals corresponding to all second sequences, etc. As such, every partition corresponded to 1020 post-stimulus signals. Every partition has been used to train a single instance of the SVM classifier. As shown in Table 6, the accuracy rate of the ensemble classifiers using the horizontal partitioning reached an accuracy rate of 98% and 96% for subjects A and B, respectively.

## 5.4. Benchmarks

For comparison purposes, Table 7 summarizes the accuracy rates obtained by the winners of the BCI competition.

	Winners of	Winners of the BCI Competition			
Subject	1st	2nd	3rd		
Subject A	96	90.5	80		
Subject B	95	90.5	80		

Table 7. Accuracy rates obtained by the winners of the BCI competition (%).

Unexpectedly, the efficiency of the ensemble heterogeneous classification strategy is low compared to those of the single classification strategy. However, results show that, using a weighted voting technique or not, the ensemble heterogeneous classification strategy has obtained better accuracy rates than the third-ranked algorithm of the BCI competition. In the meantime, the proposed single classification strategy outperformed the second and third-ranked algorithms.

Compared to the winner, the proposed ensemble homogeneous classification strategies have obtained competitive results. Adopting the vertical partitioning approach, the ensemble homogeneous classification strategy has reached the same accuracy as the winner for subject A, but the accuracy has decreased by 2% for subject B. Using the horizontal partitioning, the ensemble homogeneous classification strategy outperformed the winner for both subjects A and B.

#### 6. Conclusions and Future Work

P300 spellers offered serious opportunities for people with severe motor disabilities to interact efficiently with their environment. Different P300 speller paradigms have been proposed in the literature. This paper overviewed the most known P300 speller paradigms the Row/Column P300 speller paradigm still remains the most popular paradigm adopted by P300 spellers. This paper discussed various efficient classification strategies for the row/column P300 speller paradigm. These strategies were tested using a public dataset. The results were competitive compared to those obtained by the winner algorithms of the BCI competition using the same dataset. One of the presented strategies outperformed the winner of the competition.

The classification problem addressed in this paper was complex and it was solved in two steps. The first step aimed to predict if a post-stimulus EEG signal corresponded to a P300 response or not. Thus, this first step was a 2-class classification problem-solving. The second phase dealt with a multi-class classification problem since it aimed to predict the user's desired symbol/command from a matrix of symbols.

Various strategies were discussed in this paper including single classifier strategy and ensemble classification strategies. The single classifier strategy was tested using four different 2-class classifiers: LDA, PLS, REG, and SVM. The average accuracy rate of such a strategy ranged between 92% and 96%. These distinct classifiers were fused to validate the ensemble heterogeneous classification strategy by applying a weighted voting and a non-weighted voting algorithm. Contrary to all expectations, the performance decreased compared to the single classification strategy. This lack of performance was fixed by the ensemble homogeneous classification strategy by fusing different instances of the same classifier, in this case, the SVM classifier. The ensemble was built based on how the dataset was split among the different instances. Two approaches were tested: horizontal and vertical partitioning. The ensemble homogeneous classification strategy using the vertical partitioning technique obtained results too close to the winner algorithm of the BCI competition. However, the ensemble classification strategy using the horizontal partitioning technique outperformed the winner algorithm of the competition. The average

In the future, we intend to extend our classification strategies to include deep learning algorithms and study their efficiency. Moreover, we are planning to combine the two selective attention methods P300 and SSVEP, or P300 with ERS/ERD motor imagery BCI mental strategies knowing that recent research showed that the subject can simultaneously produce both SSVEP and ERS/ERD.

**Author Contributions:** Methodology, S.G. and K.B.; investigation, S.G., K.B., N.A., H.A. and A.B.; writing original draft preparation, S.G. and K.B.; review and editing, S.G., K.B., N.A., H.A. and A.B.; funding acquisition, S.G., K.B., H.A. and A.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Plan for Science, Technology and Innovation (MAARIFAH), King Abdulaziz City for Science and Technology, Kingdom of Saudi Arabia, grant number 14-INF3139-02.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

accuracy rate was improved by 2%.

Data Availability Statement: Not applicable.

Acknowledgments: This research work was funded by the National Plan for Science, Technology and Innovation (MAARIFAH), King Abdulaziz City for Science and Technology, Kingdom of Saudi Arabia, Award Number (14-INF3139-02).

Conflicts of Interest: The authors declare no conflict of interest.

# References

- 1. Wolpaw, J.R.; Birbaumer, N.; McFarland, D.J.; Pfurtscheller, G.; Vaughan, T.M. Brain–computer interfaces for communication and control. *Clin. Neurophysiol.* 2002, 113, 767–791. [CrossRef]
- Rezeika, A.; Benda, M.; Stawicki, P.; Gembler, F.; Saboor, A.; Volosyak, I. Brain–Computer Interface Spellers: A Review. *Brain Sci.* 2018, *8*, 57. [CrossRef]
- 3. Blankertz, B.; Dornhege, G.; Krauledat, M.; Tangermann, M.; Williamson, J.; Murray-Smith, R.; Müller, K.R. The Berlin brain-computer interface presents the novel mental typewriter Hex-O-Spell. *Clin. Neurophysiol.* **2006**, *113*, 108–109.
- Blankertz, B.; Krauledat, M.; Dornhege, G.; Williamson, J.; Murray-Smith, R.; Müller, K.R. A Note on Brain Actuated Spelling with the Berlin Brain-Computer Interface. In *Universal Access in Human-Computer Interaction. Ambient Interaction*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 759–768. [CrossRef]
- Cao, L.; Xia, B.; Maysam, O.; Li, J.; Xie, H.; Birbaumer, N. A Synchronous Motor Imagery Based Neural Physiological Paradigm for Brain Computer Interface Speller. *Front. Hum. Neurosci.* 2017, 11, 274. [CrossRef] [PubMed]
- 6. D'albis, T.; Blatt, R.; Tedesco, R.; Sbattella, L.; Matteucci, M. A predictive speller controlled by a brain-computer interface based on motor imagery. *ACM Trans. -Comput.-Hum. Interact.* **2012**, *19*, 1–25. [CrossRef]
- Farwell, L.; Donchin, E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalogr. Clin. Neurophysiol.* 1988, 70, 510–523. [CrossRef]
- Townsend, G.; LaPallo, B.; Boulay, C.; Krusienski, D.; Frye, G.; Hauser, C.; Schwartz, N.; Vaughan, T.; Wolpaw, J.; Sellers, E. A novel P300-based brain–computer interface stimulus presentation paradigm: Moving beyond rows and columns. *Clin. Neurophysiol.* 2010, 121, 1109–1120. [CrossRef]
- Postelnicu, C.C.; Talaba, D. P300-Based Brain-Neuronal Computer Interaction for Spelling Applications. *IEEE Trans. Biomed. Eng.* 2013, 60, 534–543. [CrossRef]
- 10. Ron-Angevin, R.; Varona-Moya, S.; da Silva-Sauer, L. Initial test of a T9-like P300-based speller by an ALS patient. *J. Neural Eng.* **2015**, *12*, 046023. [CrossRef]
- Volosyak, I.; Cecotti, H.; Valbuena, D.; Graser, A. Evaluation of the Bremen SSVEP based BCI in real world conditions. In Proceedings of the 2009 IEEE International Conference on Rehabilitation Robotics, Kyoto, Japan, 23–26 June 2009; pp. 322–331. [CrossRef]

- 12. Volosyak, I.; Moor, A.; Gräser, A. A Dictionary-Driven SSVEP Speller with a Modified Graphical User Interface. In *Advances in Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 353–361. [CrossRef]
- Volosyak, I. SSVEP-based Bremen–BCI interface—boosting information transfer rates. J. Neural Eng. 2011, 8, 036020. [CrossRef] [PubMed]
- Cecotti, H. A Self-Paced and Calibration-Less SSVEP-Based Brain–Computer Interface Speller. *IEEE Trans. Neural Syst. Rehabil.* Eng. 2010, 18, 127–133. [CrossRef] [PubMed]
- 15. Ansari, I.A.; Singla, R. BCI: an optimised speller using SSVEP. Int. J. Biomed. Eng. Technol. 2016, 22, 31. [CrossRef]
- Volosyak, I.; Gembler, F.; Stawicki, P. Age-related differences in SSVEP-based BCI performance. *Neurocomputing* 2017, 250, 57–64. [CrossRef]
- Guo, W.; Qin, W.; Zheng, D.; Song, T.; Zhang, P. P300 Detection Adaptive Channel Selection Method under the Multiple Kernel Learning. In Proceedings of the 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT), Chongqing, China, 22–24 November 2021; pp. 898–902. [CrossRef]
- 18. Ma, R.; Yu, T.; Zhong, X.; Yu, Z.L.; Li, Y.; Gu, Z. Capsule network for ERP detection in brain-computer interface. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2021**, 29, 718–730. [CrossRef]
- Gannouni, S.; Belwafi, K.; Al-Sulmi, M.R.; Al-Farhood, M.D.; Al-Obaid, O.A.; Al-Awadh, A.M.; Aboalsamh, H.; Belghith, A. A Brain Controlled Command-Line Interface to Enhance the Accessibility of Severe Motor Disabled People to Personnel Computer. *Brain Sci.* 2022, *12*, 926. [CrossRef] [PubMed]
- Sutton, S.; Braren, M.; Zubin, J.; John, E.R. Evoked-Potential Correlates of Stimulus Uncertainty. *Science* 1965, 150, 1187–1188. [CrossRef]
- Martínez-Cagigal, V.; Santamaría-Vázquez, E.; Hornero, R. A Portable P300-Based Brain–Computer Interface as an Alternative Communication Device. In *Converging Clinical and Engineering Research on Neurorehabilitation IV*; Torricelli, D., Akay, M., Pons, J.L., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 383–387.
- 22. Martinez-Cagigal, V.; Gomez-Pilar, J.; Alvarez, D.; Hornero, R. An Asynchronous P300-Based Brain-Computer Interface Web Browser for Severely Disabled People. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1332–1342. [CrossRef] [PubMed]
- 23. Gannouni, S.; Alangari, N.; Mathkour, H.; Aboalsamh, H.; Belwafi, K. BCWB: a P300 brain-controlled web browser. *Int. J. Semant. Web Inf. Syst.* **2017**, *13*, 55–73. [CrossRef]
- 24. Bai, L.; Yu, T.; Li, Y. A brain computer interface-based explorer. J. Neurosci. Methods 2015, 244, 2–7. [CrossRef] [PubMed]
- Yu, T.; Li, Y.; Long, J.; Wang, C. A brain-computer interface controlled mail client. In Proceedings of the 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Osaka, Japan, 3–7 July 2013. [CrossRef]
- Corralejo, R.; Álvarez, D.; Hornero, R. A P300-Based BCI Aimed at Managing Electronic Devices for People with Severe Disabilities. In *Proceedings of the Converging Clinical and Engineering Research on Neurorehabilitation*; Pons, J.L., Torricelli, D., Pajaro, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 641–645.
- Gannouni, S.; Alrayes, N.; Alameer, A.; Alsaudi, A.; Alabdulkareem, A. b-mail: A Brain-Controlled Mail Client. J. Digit. Inf. Manag. 2011, 9, 167–170.
- He, S.; Zhou, Y.; Yu, T.; Zhang, R.; Huang, Q.; Chuai, L.; Mustafa, M.U.; Gu, Z.; Yu, Z.L.; Tan, H.; et al. EEG- and EOG-Based Asynchronous Hybrid BCI: A System Integrating a Speller, a Web Browser, an E-Mail Client, and a File Explorer. *IEEE Trans. Neural Syst. Rehabil. Eng.* 2020, *28*, 519–530. [CrossRef] [PubMed]
- 29. Fazel-Rezai, R. Recent Advances in Brain-Computer Interface Systems; BoD–Books on Demand; IntechOpen: London, UK, 2011.
- Merzagora, A.; Butti, M.; Polikar, R.; Izzetoglu, M.; Bunce, S.; Cerutti, S.; Bianchi, A.; Onaral, B. Model comparison for automatic characterization and classification of average ERPs using visual oddball paradigm. *Clin. Neurophysiol.* 2009, 120, 264–274. [CrossRef]
- 31. Starr, A.; Sandroni, P.; Michalewski, H. Readiness to respond in a target detection task: Pre- and post-stimulus event-related potentials in normal subjects. *Electroencephalogr. Clin. Neurophysiol. Potentials Sect.* **1995**, *96*, 76–92. [CrossRef]
- 32. Naik, G. Recent Advances in Biomedical Engineering; IntechOpen: London, UK, 2009. [CrossRef]
- 33. Guger, C.; Daban, S.; Sellers, E.; Holzner, C.; Krausz, G.; Carabalona, R.; Gramatica, F.; Edlinger, G. How many people are able to control a P300-based brain–computer interface (BCI)? *Neurosci. Lett.* **2009**, *462*, 94–98. [CrossRef] [PubMed]
- Wolpaw, J.; Birbaumer, N.; Heetderks, W.; McFarland, D.; Peckham, P.; Schalk, G.; Donchin, E.; Quatrano, L.; Robinson, C.; Vaughan, T. Brain-computer interface technology: a review of the first international meeting. *IEEE Trans. Rehabil. Eng.* 2000, *8*, 164–173. [CrossRef] [PubMed]
- 35. Belwafi, K.; Romain, O.; Gannouni, S.; Ghaffari, F.; Djemal, R.; Ouni, B. An embedded implementation based on adaptive filter bank for brain–computer interface systems. *J. Neurosci. Methods* **2018**, *305*, 1–16. [CrossRef]
- Abdi, H. Partial least squares regression and projection on latent structure regression (PLS Regression). WIREs Comput. Stat. 2010, 2, 97–106. [CrossRef]
- 37. Niu, L. A review of the application of logistic regression in educational research: common issues, implications, and suggestions. *Educ. Rev.* **2018**, 72, 41–67. [CrossRef]