UNIVERSITY OF LEEDS

This is a repository copy of *SSVEP-based Brain-Computer Interface Controlled Robotic Platform with Velocity Modulation*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/202741/

Version: Accepted Version

Article:
Zhang, Y., Qian, K., Xie, S.Q. et al. (3 more authors) (Accepted: 2023) SSVEP-based Brain-Computer Interface Controlled Robotic Platform with Velocity Modulation. IEEE Transactions on Neural Systems and Rehabilitation Engineering. ISSN 1558-0210 (In Press)

© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Reuse
Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

White Rose university consortium
Universities of Leeds, Sheffield & York

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

SSVEP-based Brain-Computer Interface Controlled Robotic Platform with Velocity Modulation

Yue Zhang, Kun Qian, Sheng Quan Xie, *Senior Member, IEEE,* Chaoyang Shi, *Member, IEEE,* Jun Li , *Member, IEEE,* and Zhi-Qiang Zhang, *Member, IEEE*

***Abstract*—Steady-state visual evoked potential (SSVEP)-based brain-computer interfaces (BCIs) have been extensively studied due to many benefits, such as non-invasiveness, high information transfer rate, and ease of use. SSVEP-based BCI has been investigated in various applications by projecting brain signals to robot control commands. However, the movement direction and speed are generally fixed and prescribed, neglecting the user's requirement for velocity changes during practical implementations. In this study, we proposed a velocity modulation method based on stimulus brightness for controlling the robotic arm in the SSVEP-based BCI system. A stimulation interface was designed, incorporating flickers, target and a cursor workspace. The synchronization of the cursor and robotic arm does not require the subject's eye switch between the stimuli and the robot. The feature vector consists of the characteristics of the signal and the classification result. Subsequently, the Gaussian mixture model (GMM) and Bayesian inference were used to calculate the posterior probabilities that the signal came from a high or low brightness flicker. A brain-actuated speed function was designed by incorporating the posterior probability difference. Finally, the historical velocity was considered to determine the final velocity. To demonstrate the effectiveness of the proposed method, online experiments, including single- and multi-target reaching tasks, were conducted. The extensive experimental results validated the feasibility of the proposed method in reducing reaching time and achieving proximity to the target.**

***Index Terms*—Brain-computer interface (BCI), electroencephalography (EEG), steady-state visual evoked potential (SSVEP), robotic arm, velocity modulation**

I. Introduction

Individuals who experience severe motor impairments encounter challenges in performing their daily tasks [1], [2]. Many studies attempted to help people achieve mobility with assistive robotic devices [3]. However, several traditional forms of assistance still require manual control ability from users. Brain-computer interfaces (BCIs) can establish a direct connection between brain signals and external devices, eliminating reliance on peripheral nerves and muscles [4]. In recent years, BCIs have been extensively used in various assistance and rehabilitation applications, such as wheelchair [5], speller [6], and robot arm [7], [8].

Electroencephalography (EEG) is broadly employed in BCI research due to many advantages, such as non-invasiveness, high temporal resolution, and simple operation [9]. Steady-state visual evoked potential (SSVEP) as one of EEG paradigms has attracted significant attention because of its minimal training requirements and high signal-to-noise ratio (SNR) [10]–[12]. The SSVEP-based BCI system maps the brain signals to robot commands and then transmits them to the corresponding manipulation [13]. Therefore, SSVEP-based BCIs have been widely explored in the field of assistive and rehabilitative applications in recent decades. For example, Guo *et.al* [14] designed an SSVEP-based BCI-controlled soft robotic glove for post-stroke hand function rehabilitation. Sakkalis *et.al* [15] implemented efficient electric wheelchair navigation by utilizing an SSVEP-controlled system. In a separate study, Wang *et.al* [16] built a portable SSVEP system specifically designed for rehabilitation exoskeletons. Chen *et.al* [17] integrated SSVEP with computer vision to fulfill robot pick-and-place tasks. Subsequently, Chen *et.al* [18] further employed the augmented reality (AR) technique that eliminates the need for subjects to switch attention between the stimulation and the robotic arm. These studies effectively established the connection between SSVEP signals and external robots with prescribed velocity control. However, modulating the velocity of the robots in response to the user's intentions can bring more benefits in human motion assistance and rehabilitative scenarios.

In recent years, several studies have considered velocity control in SSVEP-based BCI systems. For instance, Zhao *et.al* [19] mapped three flashing squares to the three motion modes, where different frequencies represent fast, medium, and slow speeds. However, by replacing the stimulus-command pair with the stimulus-velocity pair, the BCI achieves velocity modulation in a discrete manner. To realize continuous velocity mapping, Zhang *et.al* [20] introduced stimuli with different frequencies to control a navigation robot. A self-defined mapping function on the basis of correlation values was proposed to output continuous control commands. Additionally, Sharma *et.al* [21] demonstrated that the amplitude of brain signals can be modulated by varying the distances between the subject and the stimulus. However, only one stimulus was designed for unidirectional robot manipulation

This work was supported in part by Engineering and Physical Sciences Research Council (EPSRC) (Grant No. EP/S019219/1) and in part by China Scholarship Council (CSC) (Grant No. 201906460007). (Corresponding author: Zhi-Qiang Zhang, Chaoyang Shi and Jun Li.)

Yue Zhang, Sheng Quan Xie, and Zhi-Qiang Zhang are with the Institute of Robotics, Autonomous System and Sensing, School of Electrical and Electronic Engineering, University of Leeds, LS2 9JT, Leeds, U.K. (e-mail: elyzh@leeds.ac.uk; s.q.xie@leeds.ac.uk; z.zhang3@leeds.ac.uk). Kun Qian is with the School of Engineering and Physical Sciences, Heriot-Watt University, EH14 4AS, Edinburgh, UK (k.qian@hw.ac.uk). Chaoyang Shi is with the School of Mechanical Engineering, Tianjin University, 300072, Tianjin, China (chaoyang.shi@tju.edu.cn). Jun Li is with the College of Intelligent Systems Science and Engineering Hubei Minzu University, 445000, Enshi, China (1995007@hbmzu.edu.cn).

For the purpose of Open Access, the authors have applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

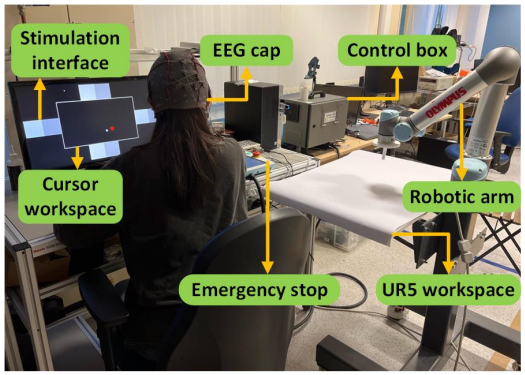


Fig. 1. The experimental environment of the proposed SSVEP-based BCI system for robotic arm velocity control.

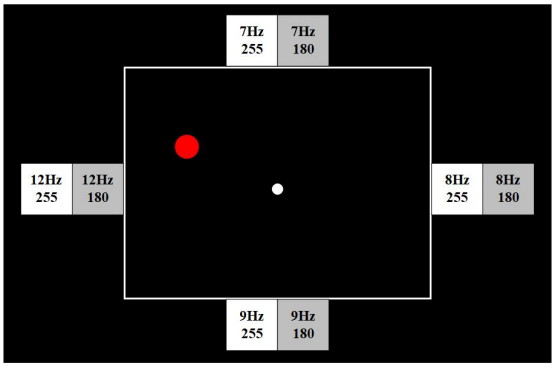


Fig. 2. The simulation interface of the eight-target SSVEP-based BCI system. The frequency and maximal brightness were displayed for each visual stimulus. The workspace for the random target and the cursor was represented by a rectangle with white edges. The red circle is the target and the white one is the cursor.

since orthoptic eye accommodation may bring difficulties in multi-class interface design. In addition to stimulus frequency and distance, voluntary attention increased stimulus-driven EEG activity [22]. Integrating the attention factor, Molina-Cantero *et.al* [23] utilized SSVEP responses obtained from a single-channel EEG headset known as NeuroSky Mindwave (NM). The NM was able to provide attention levels, which were then used to modulate the speed of a controlled cursor. However, the requirement for a specific headset limits its generalizability and scalability. On the other hand, flicker brightness is another important component that influences the characteristics of the SSVEP signal; thus, some studies have investigated the brain response to the modulation of stimulus brightness [24]. For instance, the findings in [25] revealed that SSVEP amplitude generally increases with higher brightness levels. Zhang *el.al* found that the power spectrum density (PSD) of the stimulus maintained a certain positive correlation with brightness [26]. Nevertheless, studies investigating the feasibility and effectiveness of employing brightness in robot velocity modulation are still limited to SSVEP-based BCIs.

In this study, we aimed to explore a velocity modulation method for controlling the robotic arm in the SSVEP-based BCI. The main contributions are: 1) Development of a stimulation interface including flickers, a target, and a cursor workspace. It enables synchronized movement of the cursor and robotic arm. Subjects are not required to switch their eyes between the stimulus and the robot. 2) Proposal of a stimulus brightness-based velocity modulation model. The feature vector is constructed using the correlation coefficient and power spectral density (PSD). 3) Devising a speed function that leverages the difference in posterior probabilities of being a high- or low-brightness flicker. The historical velocity was incorporated in the velocity determination to avoid occasional misclassification. To evaluate the performance of the proposed method, online experiments, including single- and multi-target reaching tasks, were conducted. Two additional control methods were used for comparison. Extensive evaluations showed that the proposed method enabled the cursor or robotic arm to complete tasks in a shorter time and get closer to the target.

This paper is organized as follows: Section II introduces the SSVEP experiment and the proposed method. Section III presents the experimental results. The discussion and conclusion are shown in Sections IV and V, respectively.

II. Methods and Materials

A. Experimental Environment

Fig. 1 showed the experimental setup of the SSVEP-based BCI system for robotic arm control. The computer screen displayed the stimulation interface, which includes flickers, the target, and the cursor workplace. The robotic arm and its workspace were located at the right hand of the subject. The experimental environment was detailed as follows:

1) Participants: Ten healthy subjects (five females and five males, mean age: twenty-eight years) volunteered for the experiment in this study. All subjects had normal or corrected-to-normal vision. The experiment has been approved by the Faculty Research Ethics Committee of the University of Leeds. Each participant read and signed an informed consent form.

2) Simulation Interface: The stimulation interface was shown in Fig. 2. There are eight stimuli on a 23.6-inch LCD monitor with a resolution of 1920×1080 pixels and a refresh rate of 60 Hz. Two adjacent stimuli have the same frequency but different brightnesses. For instance, the top two stimuli flashed at 7 Hz, corresponding to upward movements. The maximum brightnesses were set at 255 and 180, respectively. The right, bottom, and left stimuli flickered at 8, 9, and 12 Hz, respectively. The size of the stimulus is 210×210 pixels. The rectangle with white edges is 1060×620 pixels in size. The red circle (radius: 25 pixels) and the white one (radius: 15 pixels) are the target and the cursor. The interface was developed in MATLAB using the Psychophysics Toolbox Version 3 [27].

3) Signal Acquisition: The data was recorded using equipment from g.tec medical engineering GmbH, and it was sampled at a rate of 512 Hz using the g.USB amplifier. Nine electrodes, i.e., Pz, PO3, POz, PO4, PO7, O1, Oz, O2, and PO8, in the parietal and occipital areas were selected. The ground electrode was placed over FPz, and the reference electrode was positioned on the right earlobe.

4) Robotic Arm: The UR5 robotic arm (Universal Robots) was employed in this study. Its maximum reaching radius is 850 mm. The workspace for the UR5 robotic arm was set up

with a whiteboard positioned directly in front of the robot. It provides a designated horizontal area (803 mm × 470 mm) for the arm's actions and operations. The participants were asked to control the robotic arm to perform reaching tasks.

B. Experimental Protocol

1) Offline: For each subject, the experiment included four blocks, and each block contained eight trials corresponding to eight stimuli. Each trial began with a 2 s target cue (a red square). After the cue, all targets flickered for 7 s simultaneously. During the experiment, the subject sat in a comfortable chair in a dimly lit and quiet room. The viewing distance from the computer screen was set at 60 cm. The subject was asked to focus on the stimulus and avoid eye movement. The subject can take a rest between two blocks.

2) Online: The online experiments include single- and multi-target reaching tasks. In the single-target reaching experiment, each subject should complete eight trials. In each trial, the target was generated randomly within the cursor's workspace. There is a one-to-one mapping between the target position in the cursor's workspace and a specific location in the robotic arm's workspace. The subject should move the cursor to reach the target by focusing on different visual stimuli. Meanwhile, the cursor velocity was transformed and then sent to the robotic arm, so the robotic arm can map the cursor's movement in an enlarged workspace. Therefore, subjects were not required to switch their eyes between the stimulus and the robotic arm. Fig. 3 showed the diagram of the proposed robotic arm velocity modulation method in the SSVEP-based BCI.

The basic rule for governing reaching tasks is that the cursor/robotic arm should move quickly when it is far from the target and slow down as it approaches. It ensures that the cursor/robotic arm approaches the target with greater precision within a shorter duration. Hence, the subject should focus on the lighter flicker to move the cursor/robotic arm faster while focusing on the darker flicker to move it slower. When the distance between the centers of the cursor and the target is consistently below a certain value for four consecutive times, and the trial time is within 40 s, the target is considered to be hit successfully. Otherwise, the trial fails. The distance was set to 30 pixels. Two other control methods were employed for performance comparison. The comparing methods would be described in the next subsection. In each comparative experiment, the target position is the same. Therefore, each subject should finish 24 trials in the single-target reaching task.

In the multi-target reaching task, the subject should reach three randomly generated targets successively within 200 s. After successfully hitting a target, the next target occurs automatically. Each participant was required to complete three reaching tasks using the proposed method along with two comparative control methods, for a total of nine trials.

C. Data Pre-processing

Since the effect of visual latency in the human visual system, the data was extracted in $[0.14\ (0.14+d)]s$, where d was the data length for performance analysis. The data were filtered by a Butterworth band-pass filter (5–60 Hz) and a notch filter (50 Hz). The band-pass range is determined by the visual stimulus frequency and the number of harmonics in CCA.

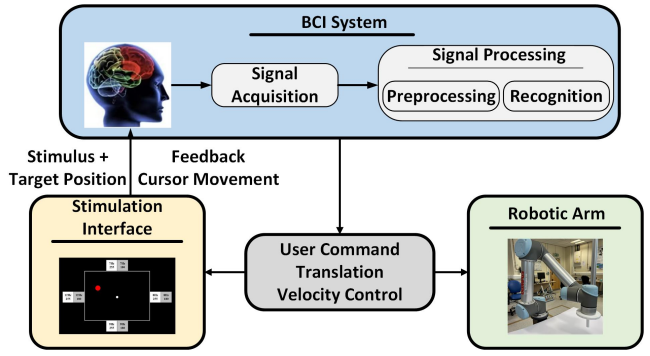


Fig. 3. Diagram of the proposed SSVEP-based BCI system for robotic arm velocity control.

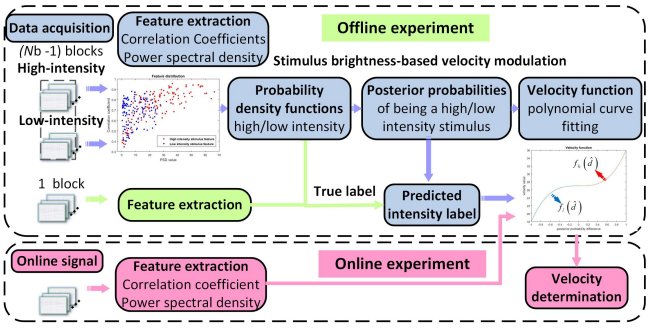


Fig. 4. The detailed framework of the proposed velocity modulation process for robotic arm control in the SSVEP-based BCI.

D. SSVEP-based BCI Controlled Robotic Platform with Velocity Modulation

A stimulus brightness-based velocity modulation method was presented for robotic arm control in the SSVEP-based BCI. The framework, as shown in Fig. 4, included two parts: offline and online experiments. The offline experiment includes data acquisition, feature extraction, and stimulus brightness-based speed modulation. In the online experiment, the velocity of the cursor/robotic arm was finally determined by the speed function trained in the offline and historical velocity. The data acquisition was described in previous sections. Subsequent content will explain the other modules in detail.

1) Feature Extraction: Suppose that $\boldsymbol{X} \in \mathbb{R}^{N_c \times N_s}$ represents the two-dimensional SSVEP signal from the offline experiment. N_c, and N_s are the number of channels and samples, respectively. SSVEP signals can also be described as sine-cosine waves, so the reference signal $\boldsymbol{Y}_i \in \mathbb{R}^{2N_h \times N_s}$ for the i-th stimulus can be artificially constructed as follows:

$$\boldsymbol{Y}_i = \begin{bmatrix} \sin(2\pi f_i t) \\ \cos(2\pi f_i t) \\ \vdots \\ \sin(2\pi N_h f_i t) \\ \cos(2\pi N_h f_i t) \end{bmatrix}, \ t = [1/F_s, 2/F_s, ..., N_s/F_s] \quad (1)$$

where f_i, N_h, and F_s refer to the frequency of the stimulus, the number of harmonics, and the sampling rate, respectively. In this study, the number of harmonics was set to five.

This study used canonical correlation analysis (CCA) for feature construction and recognition. CCA finds two spatial filters $\boldsymbol{w}_i \in \mathbb{R}^{N_c \times 1}$ and $\boldsymbol{v}_i \in \mathbb{R}^{2N_h \times 1}$ to maximize the correlation between linear projections $\boldsymbol{w}_i^T \boldsymbol{X}$ and $\boldsymbol{v}_i^T \boldsymbol{Y}_i$:

$$\rho_i = \max_{\boldsymbol{w}_i, \boldsymbol{v}_i} \frac{E[\boldsymbol{w}_i^T \boldsymbol{X} \boldsymbol{Y}_i^T \boldsymbol{v}_i]}{\sqrt{E[\boldsymbol{w}_i^T \boldsymbol{X} \boldsymbol{X}^T \boldsymbol{w}_i]} \sqrt{E[\boldsymbol{v}_i^T \boldsymbol{Y}_i \boldsymbol{Y}_i^T \boldsymbol{v}_i]}} \tag{2}$$

The correlation coefficient ρ_i can be calculated between $\boldsymbol{X}$ and each reference signal $\boldsymbol{Y}_i, i = 1, 2, ..., N_f$. The frequency of the reference signal with maximal correlation coefficient is determined as the frequency of $\boldsymbol{X}$. If the classification is correct, the maximal correlation coefficient ρ was recorded for the feature construction. Additionally, the PSD of $\boldsymbol{X}_j \in \mathbb{R}^{1 \times N_s}, j = 1, 2, ..., N_c$ was estimated by periodogram() in MATLAB. The PSD at the true frequency f for j-th channel was presented as $\text{PSD}_j(f)$. The average PSD value α across all channels was calculated as:

$$\alpha = \frac{1}{N_c} \sum_{j=1}^{N_c} \text{PSD}_j(f) \tag{3}$$

The signals from N_t blocks of the offline experiment were processed using the aforementioned procedure.

2) Stimulus Brightness-based Speed Function: As illustrated in Fig. 4, the offline signals were divided into two sets. Specifically, $(N_t - 1)$ data blocks were used to train conditional probability density functions, and one block served as an evaluation dataset to fine-tune the model's hyperparameters. A sliding time window with a step of 0.5 s was employed to divide the signals into multiple epochs. In the $(N_t - 1)$ blocks, ρ and α were further divided into two sets based on whether they were obtained from high- or low-brightness flickers:

$$\begin{aligned} \boldsymbol{\Phi}_h &= \begin{bmatrix} \rho_{h,1}, & \rho_{h,2}, ..., & \rho_{h,N_h} \\ \alpha_{h,1}, & \alpha_{h,2}, ..., & \alpha_{h,N_h} \end{bmatrix}^T \in \mathbb{R}^{N_h \times 2} \\ \boldsymbol{\Phi}_l &= \begin{bmatrix} \rho_{l,1}, & \rho_{l,2}, ..., & \rho_{l,N_l} \\ \alpha_{l,1}, & \alpha_{l,2}, ..., & \alpha_{l,N_l} \end{bmatrix}^T \in \mathbb{R}^{N_l \times 2} \end{aligned} \tag{4}$$

where N_h and N_l represent the number of trials where the subject directed their attention toward high- and low-brightness visual stimuli, respectively. The feature matrices $\boldsymbol{\Phi}_h$ and $\boldsymbol{\Phi}_l$ are derived from stimuli with high and low brightness. Suppose that when the subject observed the high and low brightness stimuli, the corresponding trials were labeled as T_h and T_l. The probability density functions of the feature matrices for high and low brightness stimuli are denoted as $p(\boldsymbol{\Phi}|T_h)$ and $p(\boldsymbol{\Phi}|T_l)$. GMM was used to fit $\boldsymbol{\Phi}_h$ and $\boldsymbol{\Phi}_l$. The GMM is an efficient probabilistic model capable of building complex probability distribution functions [28]. Therefore, two probability distribution functions are expressed as follows:

$$\begin{aligned} p(\boldsymbol{\Phi}|T_h) = p(\boldsymbol{\Phi}_h) &= \sum_{k=1}^{K} \lambda_k \mathcal{N}(\boldsymbol{\Phi}_h|\boldsymbol{\theta}_k) \\ p(\boldsymbol{\Phi}|T_l) = p(\boldsymbol{\Phi}_l) &= \sum_{k=1}^{K} \varphi_k \mathcal{N}(\boldsymbol{\Phi}_l|\boldsymbol{\xi}_k) \end{aligned} \tag{5}$$

where K is the number of mixture components, λ_k and φ_k are the mixture wights subject to the constraints $\sum_{k=1}^{K} \lambda_k = 1$ and $\sum_{k=1}^{K} \varphi_k = 1$. The Gaussian density functions $\mathcal{N}(\boldsymbol{\Phi}_h)$ and $\mathcal{N}(\boldsymbol{\Phi}_l)$ are defined by the parameters $\boldsymbol{\theta}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ and $\boldsymbol{\xi}_k = (\boldsymbol{\nu}_k, \boldsymbol{\Gamma}_k)$, where $\boldsymbol{\mu}_k$ and $\boldsymbol{\nu}_k$ are the mean, and $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\Gamma}_k$ are the covariance matrices. The GMM parameters, i.e., $\lambda_k, \eta_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\nu}_k$, and $\boldsymbol{\Gamma}_k$, were estimated by the Expectation-Maximization (EM) algorithm in MATLAB.

The left-one block was used to determine hyperparameters. Specifically, each trial from the left block was analyzed to form the feature vector $\hat{\boldsymbol{\phi}} = [\hat{\rho}\ \hat{\alpha}]$. Bayesian inference was used to compute the posterior probabilities, indicating the trial was obtained by observing a high and low brightness stimulus:

$$\begin{aligned} P(T_h|\hat{\boldsymbol{\phi}}) &= \frac{p(\hat{\boldsymbol{\phi}}|T_h)P(T_h)}{p(\hat{\boldsymbol{\phi}}|T_h)P(T_h) + p(\hat{\boldsymbol{\phi}}|T_l)P(T_l)} \\ P(T_l|\hat{\boldsymbol{\phi}}) &= \frac{p(\hat{\boldsymbol{\phi}}|T_l)P(T_l)}{p(\hat{\boldsymbol{\phi}}|T_h)P(T_h) + p(\hat{\boldsymbol{\phi}}|T_l)P(T_l)} \end{aligned} \tag{6}$$

where $P(T_h)$ and $P(T_l)$ are prior probabilities of high and low brightness stimulus. They are both set to 0.5. The difference between two posterior probabilities is calculated as follows:

$$d = P(T_h|\hat{\boldsymbol{\phi}}) - P(T_l|\hat{\boldsymbol{\phi}}) \tag{7}$$

Subsequently, d is compared with a threshold σ:

$$L(\hat{\boldsymbol{\phi}}) = \begin{cases} T_h, & d > \sigma \\ T_l, & \text{otherwise} \end{cases} \tag{8}$$

where $L(\hat{\boldsymbol{\phi}})$ is the predicted high/low brightness label of $\hat{\boldsymbol{\phi}}$. $L(\hat{\boldsymbol{\phi}})$ would be compared with the true brightness label of $\hat{\boldsymbol{\phi}}$. If they are equal, it indicates that the above process successfully recognized the brightness label of the SSVEP trial. The ratio of successfully predicted trials to total trials in the left block is defined as Acc. The grid-search method was used to determine optimal values for K and σ via calculation of Acc. The K and σ ranges are specified as [1, 4] and [-0.5, 0.5], respectively. An exhaustive search was conducted for the K with an interval of 1 and for σ with an interval of 0.1. The values that yielded the highest Acc were chosen as optimal values for K and σ.

The speed range of the cursor is set to $[v_l, v_h]$, so the middle speed is $v_m = \frac{v_l + v_h}{2}$. The brain-actuated speed function based on the brightness label is represented as follows:

$$v = \begin{cases} f_l(d), & d < \sigma \\ v_m, & d = \sigma \\ f_h(d), & d > \sigma \end{cases} \tag{9}$$

where $f_l(\cdot)$ and $f_h(\cdot)$ were obtained by polyfit() in MATLAB. For $f_l(\cdot)$, polyfit fits a polynomial of degree three to $[-1, v_l]$ and $[\sigma, v_m]$. For $f_h(\cdot)$, polyfit fits a same degree polynomial to $[\sigma, v_m]$ and $[1, v_h]$. $v_l = 18$, $v_h = 36$ in this study.

3) Velocity Determination: In the online experiment, the feature of signal $\boldsymbol{\chi} \in \mathbb{R}^{N_c \times N_s}$ was extracted as $[\hat{\rho}, \hat{\alpha}]$. Following the above process, $\boldsymbol{\chi}$ can be classified as the high or low brightness stimulus based on the posterior probability difference $\hat{d}$ in (8). Subsequently, the brain-actuated speed v

is obtained using (9). So the two-dimensional velocity vector $\boldsymbol{v}_d$ can be determined by the speed and recognition result:

$$\boldsymbol{v}_d = [v_x \ v_y] = \begin{cases} [0, \ -v], & f_d = 1 \\ [v, \ \ 0], & f_d = 2 \\ [0, \ \ v], & f_d = 3 \\ [-v, \ 0], & f_d = 4 \end{cases} \tag{10}$$

where f_d is the predicted class of $\boldsymbol{\chi}$. f_d would be compared with the recognition result of the last trial f'_d. The final velocity vector of cursor $\hat{\boldsymbol{v}}_c$ can be determined as follows:

$$\hat{\boldsymbol{v}}_c = \begin{cases} \boldsymbol{v_d}, & f_d = f'_d \\ \hat{\boldsymbol{v}}'_c - \frac{\hat{\boldsymbol{v}}'_c}{4} + \boldsymbol{v_d}, & f_d \neq f'_d \end{cases} \tag{11}$$

where $\hat{\boldsymbol{v}}'_c$ is the cursor's velocity at the last moment. The cursor moves at a velocity of $\hat{\boldsymbol{v}}_c$ for a duration of 0.5 s. Subsequently, the robotic arm's velocity $\hat{\boldsymbol{v}}_a$ can be expressed proportionally:

$$\begin{aligned} v_{ax} &= \frac{-W_h \times \hat{\boldsymbol{v}}_c(2)}{w_h} \\ v_{ay} &= \frac{-W_w \times \hat{\boldsymbol{v}}_c(1)}{w_w} \end{aligned} \tag{12}$$

where W_h and W_w are the height and weight of the robotic arm's workspace, respectively. w_h and w_w are the height and weight of the cursor's workspace.

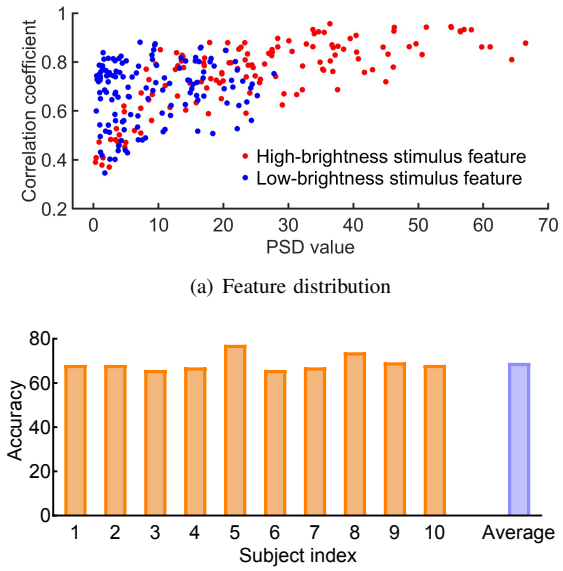
Based on the cursor's and target's positions, the subject adjusted the speed and direction via the stimulus's brightness and frequencies, respectively. The distance between their centers was calculated for each time duration. If the distance is less than 30 pixels for four consecutive instances within 40 s, it is considered a successful hit.

E. Velocity Control Strategies for Comparison

Two additional methods were compared with the proposed method. The first method is discrete velocity (DV) control, which is commonly utilized in SSVEP-based BCIs for controlling robots [14], [15]. The speed is a constant value v_{cons}, and its direction depends on the recognition result. The second method, named discrete attenuated velocity (DAV) control, incorporates velocity attenuation in (11) based on the configuration of the DV method. It shares the same setting as the proposed method. The only difference is that it does not include the proposed SSVEP-actuated velocity modulation. Given that DV and DAV methods do not consider the brightness factor, only high-brightness stimuli were used for a fair comparison. Therefore, v_{cons} is derived by the d obtained from high brightness stimuli. Specifically, the fitdist() function was used to fit these values, and consequently, the mean value d_m was obtained. Therefore, v_{cons} can be expressed as follows:

$$v_{cons} = f_h(\hat{d}_m) \tag{13}$$

To ensure a fair comparison, v_{cons} is not merely equivalent to v_m, but rather obtained through the above formula. Besides, the two comparison methods also employ the sliding window, the same as the proposed method.



(a) Feature distribution

(b) Brightness classification accuracy

Fig. 5. (a) Feature distribution of high- and low- brightness stimuli, and (b) brightness classification accuracy of each subject.

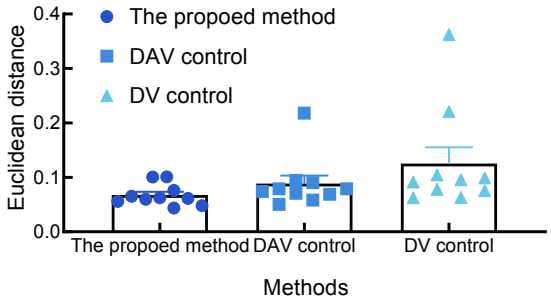
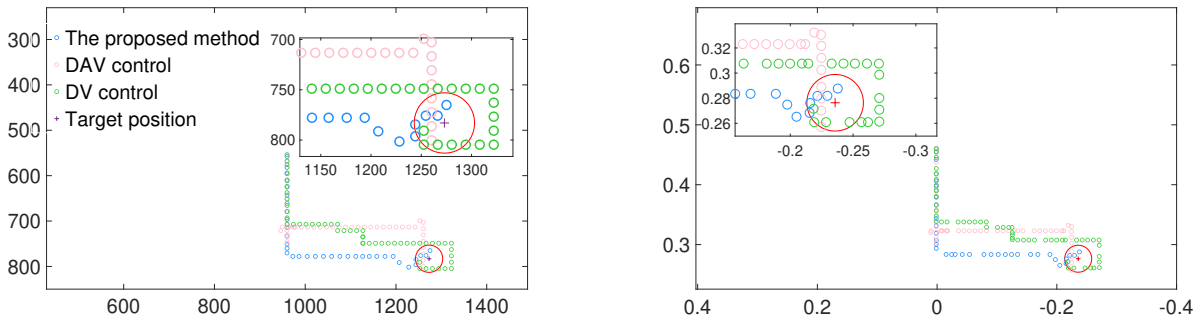


Fig. 6. The Euclidean distance between the robotic arms' positions projected by the cursor's movements and its actual arrival positions in single-target reaching tasks. The scatter points refer to the Euclidean distances provided by different subjects.

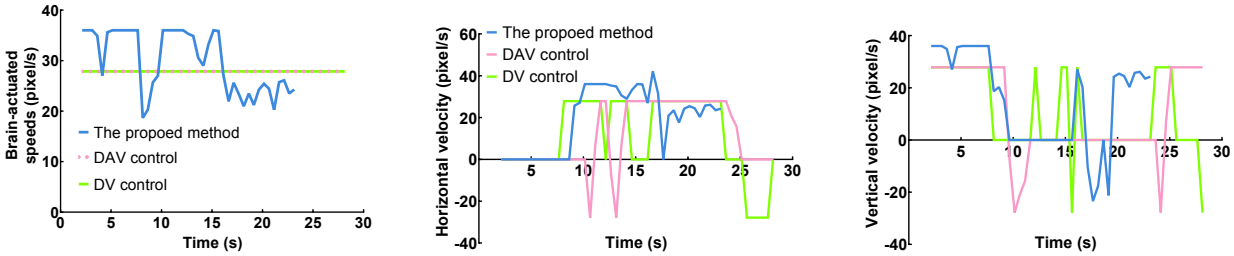
TABLE I
AVERAGE DISTANCE OF THE LAST FOUR POSITIONS AMONG THREE METHODS IN THE SINGLE-TARGET REACHING TASK

	Average distance across tasks (pixels)		
Subject index	**DV control**	**DAV control**	**The proposed method**
Subject 1	22.10	22.42	**15.52**
Subject 2	18.55	20.48	**15.95**
Subject 3	17.46	19.30	**15.77**
Subject 4	16.80	19.40	**16.36**
Subject 5	18.78	21.79	**16.72**
Subject 6	20.16	20.48	**16.36**
Subject 7	19.53	19.50	**16.33**
Subject 8	16.37	16.10	**14.42**
Subject 9	18.91	19.28	**16.93**
Subject 10	18.39	16.96	**14.33**
Average	18.70	19.57	**15.87**



(a) Cursor movement (b) Robotic arm movement

Fig. 7. (a) Cursor and (b) robotic arm movements generated by the three methods in a single-target reaching task. Each circle represents a movement. The interval between two circles represents the distance covered by two consecutive movements.



(a) (b) (c)

Fig. 8. (a) The brain-actuated speeds, (b) the horizontal velocity, and (c) the vertical velocity generated by the proposed method, DAV control, and DV control in a single-target reaching task.

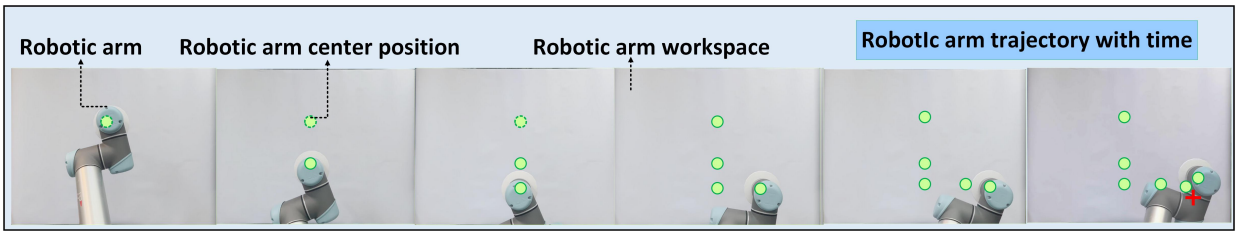


Fig. 9. Robotic arm movements in a single-target reaching task generated by the proposed method. The green circle represents the center of the robotic arm, and the red cross denotes the center of the target.

III. Results

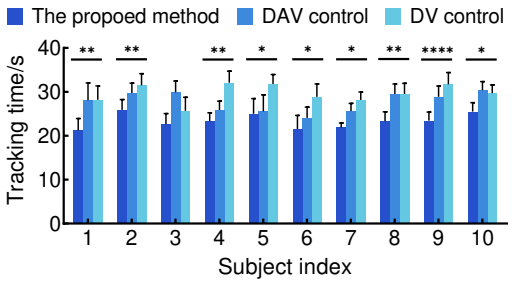
A. Offline Experiment Analysis

Fig. 5(a) displays the feature distribution of high- and low-brightness visual stimuli for the fifth subject. The x- and y-axes represent the PSD value and correlation coefficients, respectively. The scatter plots reveal distinct feature distributions for the two kinds of stimuli. Specifically, high-brightness stimuli generally exhibit higher PSD values and correlation coefficients. Fig. 5(b) presents the classification accuracy of high- and low-brightness stimuli. The graph displays the accuracy for each subject, along with the average value (i.e., 69.1%). Notably, the classification accuracy for each subject is within an acceptable range.

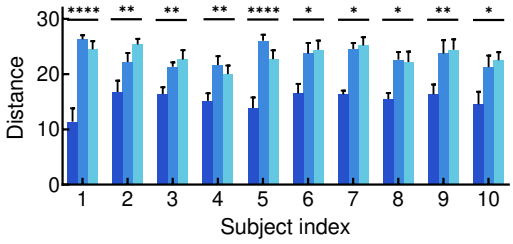
B. Single-target Reaching Task Performance Evaluation

Fig. 6 showed the Euclidean distance between the robotic arm's positions projected by the cursor and its actual arrival positions for the three methods. The scatters indicated the average distance across tasks for each subject. The Euclidean distance can also be interpreted as the tracking error of the robotic arm in following the cursor's movements. The numeric values were within an acceptable range, i.e., [0.044 0.10] for the proposed method, [0.050 0.22] for DAV control, and [0.063 0.36] for DV control. It indicated that the robotic arm largely followed the cursor's movements. Additionally, Fig. 6 demonstrates that the proposed method resulted in improved accuracy of the robotic arm in following the projected positions for most subjects. The average Euclidean distances across subjects were 0.067, 0.088, and 0.13 for the three methods, respectively.

Before the experiment, subjects were informed that when the cursor is far away from the target, they should concentrate on the high-brightness stimulus to accelerate the cursor/robotic arm. Therefore, it helps decrease the reaching time. When the cursor is close to the target, the user should focus on a low-brightness stimulus to initiate cursor/robotic arm deceleration.

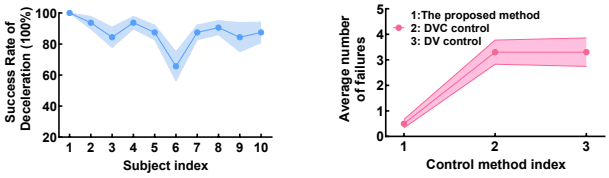


(a) Reaching time comparison in single-target tasks



(b) Distance comparison in single-target tasks

Fig. 10. Performance comparison in single-target reaching tasks. The (a) average reaching time and (b) average distance were compared between the three methods. The distance was calculated between the centers of the target and the last cursor. The error bars are the standard error of the mean (SEM). The asterisks indicate a significant difference between the three methods obtained by one-way repeated-measures ANOVA (∗: p<0.05, ∗∗: p<0.01, ∗ ∗ ∗: p<0.001, ∗ ∗ ∗∗: p<0.0001).



(a) Rate of successful deceleration (b) Average number of failures

Fig. 11. The (a) average rate of successful deceleration across tasks and (b) the average number of failures across subjects with different control methods. The SEM was shown as error envelopes.

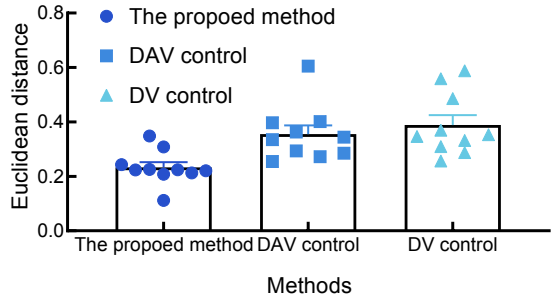


Fig. 12. The Euclidean distance between the robotic arms' positions projected by the cursor's movements and its actual arrival positions in multi-target reaching tasks. The scatter points refer to the Euclidean distances provided by different subjects.

This ensures that the cursor gets closer to the target center or does not deviate too much from its intended position. Consequently, the shorter distance between the center of the target position and the center of the cursor's final position validates the efficiency of the deceleration process.

Fig. 7 shows the cursor and robotic arm movements in a single-target reaching task performed by subject 8 using the three methods. Each circle represents a movement of the cursor/robotic arm. The interval between two circles represents the distance covered by two consecutive movements. Therefore, the density distribution of the dots also reflects velocity changes. Fig. 7(b) illustrates the actual position of the robotic arm, which was obtained through the robot sensor feedback. Fig. 8(a) showed the brain-actuated speeds of the three control methods. For the DAV and DV control, the constant value was determined by (13). The proposed method enabled the cursor to accelerate when it was distant from the target. When the cursor was tuning, there were a few speed reduction steps. However, users can achieve acceleration again by focusing on high-brightness stimuli. Fig. 8(b) and Fig. 8(c) show the vertical and horizontal velocity comparisons. The proposed method (blue line) controlled the cursor to initially move downward, leading to higher vertical velocities. As it turned to the horizontal, the velocity in this direction also exceeded that of other methods. Finally, it achieved deceleration in the last few moments. Overall, this method exhibited a shorter reaching time compared to other methods. Fig. 9 showed the robotic arm's actual movements in the proposed method.

The reaching time of the three methods with various subjects were shown in Fig. 10(a). The values were averaged across tasks. It illustrated that the proposed method consistently achieved the shortest time for each subject. The reaching time of the proposed method improved that of DAV control by 0.56 s-7.25 s and DV control by 3.05 s-8.76 s. The average reaching time across subjects of the three methods were 23.37 s, 27.75 s, and 29.72 s, respectively. A one-way repeated-measures ANOVA was conducted to explore the similarity of reaching time across methods. The statistical results revealed significant differences in reaching time among the methods for most subjects. Additionally, Fig. 10(b) illustrated the distance between the centers of the target and the last cursor position for the three methods. The results indicated that the cursor controlled by the proposed method demonstrated greater proximity to the target across tasks. For example, the distance of the proposed method increased DAV control by 4.84 to 13.10 pixels and DV control by 4.96 to 14.91 pixels. The results of the statistical analysis indicated significant differences among the methods for each subject. The difference is particularly pronounced for subject 1 and subject 5, with a highly significant p-value (P<0.0001).

The rate of successful deceleration from the proposed method was shown in Fig. 11(a). It was calculated based on whether its speeds in the last four steps were lower than those in the DV/DAV methods. The values were averaged across tasks. The results revealed that as the cursor/robotic arm approached the target, most subjects completed successful decelerations. For example, subjects 1, 2, and 8 achieved success rates of 100%, 93.75%, and 90.62%, respectively. The average

rate across subjects was 87.5%. It further demonstrated the efficacy of stimulus brightness-based velocity control. TABLE.I displayed the average distances across tasks for the last four positions. The average distances across subjects of the three methods were 15.87, 19.57, and 18.70 pixels, respectively. The compared methods achieved similar performance, while the proposed method generally provided the shortest distances. The results indicated that when utilizing the proposed method, the cursor tended to get closer or avoid excessive misses in the last four steps. The number of failures before the subject successfully conducted each task was counted in Fig. 11(b). The values were summed across tasks and averaged across subjects. The results demonstrated that the proposed method exhibits fewer failures, primarily due to its velocity control, which allows it to reach the target within a limited time.

C. Multi-target Reaching Task Performance Evaluation

Fig. 12 shows the Euclidean distance between the projected positions of the robotic arm based on cursor movements and its actual arrival positions for the three methods. It can represent the cumulative tracking error when the robotic arm moves with the cursor. The error ranges exhibited by different subjects are [0.11 0.35] for the proposed method, [0.25 0.60] for the DAV control, and [0.26 0.59] for the DV control, respectively. The proposed method provides relatively small tracking errors for most subjects. It can be attributed to the fact that the velocity direction of the robotic arm does not change suddenly like in DV control, resulting in a smoother trajectory. Besides, it provides a shorter tracking time compared to DAV control.

Fig. 13 shows the cursor and robotic arm movements generated by the three methods in a multi-target reaching task of subject 6. Three targets were generated randomly. The targets remain consistent across different methods for fair comparisons. Fig. 14(a) displayed the brain-actuated speeds of the methods. DAV and DV methods controlled the cursor or robotic arm with a constant speed regardless of distance from the target. The speed of the proposed method was subject-driven and relied on distance. For example, since the first target was near the cursor's origin, the cursor accelerated briefly and then approached the target at a slower speed. The speeds of the last four steps were marked in red. The third target was far away from the second target, so the cursor experienced a long acceleration period and then decelerated as it approached the target. The reaching time of the proposed method (74.82 s) was shorter than others, i.e., DAV control: 88.71 s; DV control: 99.45 s. Fig. 14(b) and Fig. 14(c) showed the horizontal and vertical velocities, which offer a more comprehensive view of velocity changes for each method.

The reaching time comparisons of multi-target reaching tasks among these methods were shown in TABLE. II. The proposed method consistently achieved the shortest reaching time. For each multi-target trial, three single tasks were included. To further evaluate its performance, Fig. 15 presented the reaching time and distance comparisons for each single task. The values were averaged across tasks. The results show that the proposed method provided the shortest reaching time and distance compared with other methods. The average reaching time across different subjects of the three methods were 28.34 s, 35.81 s, and 47.07 s, respectively. In addition, the average distances between the final position of the cursor and the target were 16.78, 21.64, and 23.60 pixels for the three methods, respectively. To investigate the similarity of the average reaching time and distance across different methods, a one-way repeated-measures ANOVA was conducted. The statistical analysis revealed significant differences in both performance indicators among these methods for most subjects.

TABLE II
AVERAGE REACHING TIME OF THREE METHODS IN MULTI-TARGET REACHING TASKS

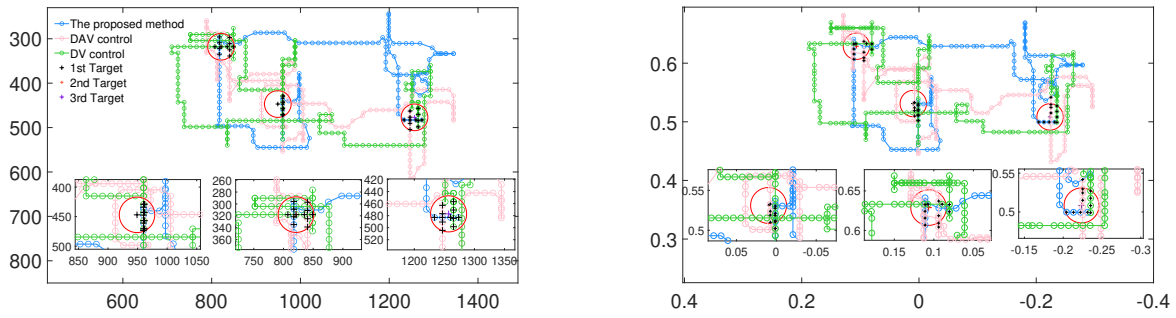
	Average reaching time across trials (s)		
Subject index	**DV control**	**DAV control**	**The proposed method**
Subject 1	143.34	81.69	**76.93**
Subject 2	148.33	110.98	**92.17**
Subject 3	160.49	104.78	**82.85**
Subject 4	150.80	111.42	**87.08**
Subject 5	159.27	124.75	**90.28**
Subject 6	102.94	126.07	**80.65**
Subject 7	135.90	91.68	**78.53**
Subject 8	149.32	121.92	**104.61**
Subject 9	135.78	115.29	**86.20**
Subject 10	128.51	88.35	**73.35**
Average	141.47	107.70	**85.27**

The average distances across tasks for the last four positions were given in TABLE.III. The results indicated that the proposed method generally achieved the shortest distance. The average distances across subjects were 19.20, 17.95, and 16.62 pixels, respectively. The average success rate of deceleration across tasks was shown in Fig. 16(a). Most subjects achieved a high rate of successful deceleration. The average rate across subjects was 76.39%. The numbers of failed trials of the three methods were shown in Fig. 16(b). The figure indicated that subjects encountered fewer failures when utilizing the proposed method to control the cursor/robotic arm.

IV. DISCUSSION

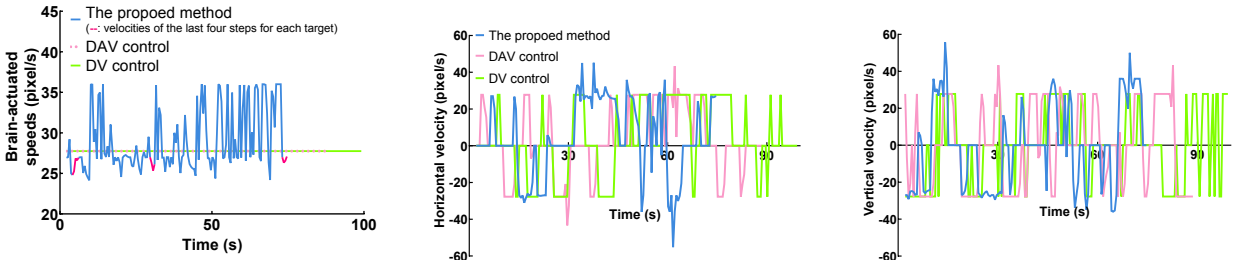
A. Model's Performance

Currently, most brain-controlled robotic platforms use discrete movement commands and constant velocity [29]. Studies involving velocity modulation explored the effects of factors, such as attention, distance, and frequency [20], [21], [23]. Differently, this study focused on stimulus brightness-based velocity modulation. The proposed method was used to control the cursor and robotic arm simultaneously to track the randomly generated target. The velocities in Fig. 8 and Fig. 14 both depicted that the proposed method achieved acceleration as the cursor was far away from the target and deceleration when their distance was small. The acceleration process helped reach the target more quickly. For example, the reaching time of the proposed method, DAV control, and DV control were 74.82 s, 88.71 s, and 99.45 s in Fig. 14. The



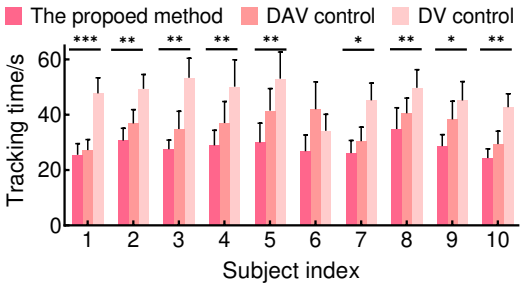
(a) Cursor movement (b) Robotic arm movement

Fig. 13. (a) Cursor and (b) robotic arm movements generated by the proposed method, DAV control, and DV control in a multi-target reaching task. Each circle represents a movement. The interval between two circles represents the distance covered by two consecutive movements.

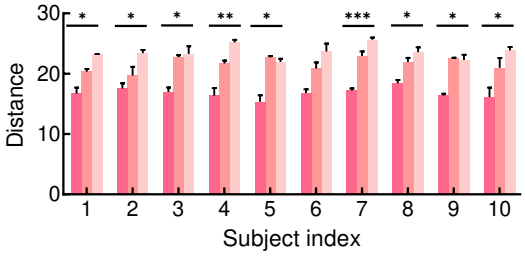


(a) (b) (c)

Fig. 14. (a) The brain-actuated speeds (b) the horizontal velocity, and (c) the vertical velocity generated by the proposed method, DAV control, and DV control in a multi-target reaching task. The red short lines in (a) refer to the speed of the last four steps for each target.



(a) Reaching time comparison in multi-target reaching tasks



(b) Distance comparison in multi-target reaching tasks

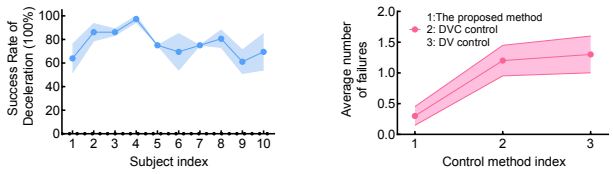
Fig. 15. Performance comparison in multi-target reaching tasks. The (a) average reaching time and (b) average distance were compared between the three methods. The distance was calculated between the centers of the target and the last cursor position. The error bars represent SEM. The asterisks indicate a significant difference between the three methods obtained by one-way repeated-measures ANOVA (∗: p<0.05, ∗∗: p<0.01, ∗ ∗ ∗: p<0.001).

TABLE III
AVERAGE DISTANCE OF THE LAST FOUR POSITIONS AMONG THREE METHODS IN MULTI-TARGET REACHING TASKS

	Average distance across tasks (pixels)		
Subject index	**DV control**	**DAV control**	**The proposed method**
Subject 1	19.71	**17.04**	18.64
Subject 2	20.15	17.63	**16.13**
Subject 3	20.17	19.203	**16.15**
Subject 4	19.64	16.78	**15.62**
Subject 5	18.31	17.80	**16.48**
Subject 6	18.38	19.36	**16.86**
Subject 7	20.19	17.32	**16.57**
Subject 8	18.07	**16.20**	18.25
Subject 9	18.42	20.05	**16.47**
Subject 10	18.91	18.15	**14.98**
Average	19.20	17.95	**16.62**

deceleration process assisted the robotic arm in getting closer to the target or decreasing overshooting. For instance, the DV method exhibited a shorter distance (21.84 versus 24.10 pixels) between the cursor and the target when it initially entered the third hit area (i.e., the red circle) in Fig. 13. However, the cursor controlled by the DV method gradually moved away from the target due to the fixed speed, increasing the distance over time (22.90 versus 17.65 pixels).

In the single-target task, the cursor's maximum distance to the target is 840 pixels. The slowest DV method would reach



(a) Success rate of deceleration (b) Average number of failures

Fig. 16. The (a) average success rate of deceleration across tasks and (b) average number of failures across various subjects with different control methods. The SEM was shown as error envelopes.

this position in about 32 s if there were no misclassifications. However, considering the occurrence of misclassifications, the task duration is set to 40 s to allow for extra correction time. For each multi-target reaching task, there are three single-reaching tasks. The first task is the same as the single-target reaching task, while in the other two tasks, the cursor starts away from the center, doubling the furthest distance. Consequently, the total duration for the multi-target reaching task is calculated to be 200 s.

B. Velocity Modulation Design

As described in (11), the velocity determination in the proposed method considered previous moment information when the classification result changed. The reasons are as follows. Firstly, occasional changes in velocity direction may be misclassified for various reasons, such as user fatigue and background noise. The proposed method preserved a part of the speed in the previous direction and then combined it with the speed in the new direction to determine the final movement of the cursor/robotic arm. If the classification was wrong, the proposed method can move toward the target while adjusting for the error. If the same direction was obtained again, it was considered the subject's intentional behavior, and the previous velocity would be overwritten without keeping the previous information. The DV method lengthens the experiment time since it is limited to adjusting erroneous directions individually and cannot simultaneously move toward the target. Secondly, discrete changes in direction have adverse effects on practical implementation. The robotic arm will experience mechanical stress and exhibit jerky movements. As shown in Fig. 6 and Fig. 12, the DV control method produced a larger tracking error in two kinds of reaching tasks with average values across subjects of 0.13 and 0.39. In comparison, the proposed method achieved lower tracking errors with average values of 0.067 and 0.23. Rapid, discrete direction changes cause discomfort when users conduct reaching tasks assisted by the robotic arm, especially for individuals with mobility limitations or during rehabilitation exercises. The proposed method allows smooth movements of the robotic arm.

Compared to the DAV method, the proposed stimulus brightness-based velocity modulation saves time in reaching tasks. The protocol also reflects the subjects' practical behaviors. While observing a distant object, individuals tend to increase their speed, whereas when nearing the object, they tend to slow down. It can also improve subjects' motivation and participation in conducting various activities, such as assistance and rehabilitation. Intentional participation is important to enhance brain plasticity, thereby increasing the chances of motor recovery [9].

In this study, a third-order polynomial was used to fit the speed function. The curve demonstrates an S-shaped behavior as the posterior probability difference changes. It is characterized by an initial increase, followed by a smooth transitional stage, and finally an increase. This indicates that the model outputs at higher speeds when the probability of the SSVEP signal being excited by a high-brightness stimulus significantly exceeds that of a low-brightness stimulus. Otherwise, the model outputs at a slower speed. As the probability difference approaches the training threshold, it becomes challenging to determine the signal evoked by which brightness stimulus. To address this uncertainty, a middle speed should be assigned, corresponding to a third-order polynomial that features a smooth transitional stage in the middle. Meanwhile, this setting aligns with practical scenarios. In Fig. 5(b), the average brightness classification accuracy was 69.1% across subjects. Therefore, assigning speeds around the median aims to strike a balance in cases where the model's uncertainty arises due to the proximity of probabilities to the threshold.

C. Feature Extraction

In this study, the correlation coefficient and PSD value were integrated into the feature vector. The correlation coefficient was provided by the CCA recognition method. One reason for choosing CCA is due to its simple implementation and low computational complexity [30]. Additionally, it offered sufficient features to train probability density functions and fine-tune hyperparameters in the offline experiment via the sliding window. It also showed satisfactory performance in the offline experiment's four-class problem. The average recognition accuracy across subjects is 90.5%. Thus, it can be a valid choice for the online experiment.

D. Future Work

Although the proposed method achieved stimulus brightness-based velocity control, there are certain directions for further improvement. This study scaled down the robotic arm's workspace and established the cursor's workspace on the computer screen. It may be inconvenient for the subject to focus on the stimulation interface while performing rehabilitation exercises. To address it, future work could involve integrating AR glasses [18] to provide a more user-friendly rehabilitation scenario. The AR glasses could simultaneously display visual stimulation, the cursor, and the target directly in the subject's field of view. Secondly, SSVEP-based BCI systems require the subject to maintain attention on the light source, which potentially leads to visual fatigue [13]. Some research has reported that high-frequency stimuli can reduce visual fatigue and discomfort [17]. Additionally, hybrid BCIs present promising opportunities to address this issue [31]. For instance, the combination of eye tracking-based assistive technologies with the SSVEP system has been explored for high-speed speller implementation, aiming to overcome fatigue and tiredness [32].

V. Conclusion

In this study, a stimulus brightness-based velocity modulation method was proposed for robotic arm control in the SSVEP-based BCI system. The flickers with different frequencies and brightnesses were employed to achieve velocity modulation. The feature vector was constructed from the correlation coefficient and PSD. The GMM model and Bayesian inference were then used to calculate the posterior probabilities that the signal came from a high- and low-brightness flicker. The speed function was designed using the posterior probability difference, and the velocity from the previous moment was incorporated to derive the final direction and speed. For performance comparison, two velocity control methods were included. The effectiveness and feasibility of the proposed method were demonstrated via online experiments involving single- and multi-target reaching tasks.

Author Contributions: Conceptualization, Z.Q.Z., and Y.Z.; Methodology, Z.Q.Z., K.Q., and Y.Z.; Validation, Y.Z., K.Q., and C.Y.S.; Writing-original draft preparation, Z.Q.Z., Y.Z., and C.Y.S.; Writing-review and editing, Z.Q.Z., Y.Z., and J.L.; Visualization, Z.Q.Z., Y.Z., and K.Q.; Supervision, Z.Q.Z., and S.Q.X.; Funding acquisition, Z.Q.Z., and S.Q.X. All authors have read and agreed to the published version of the manuscript.

Data Access Statement: The data presented in this study are available from the corresponding author upon request.

References

[1] N. Cheng, K. S. Phua, H. S. Lai, P. K. Tam, K. Y. Tang, K. K. Cheng, R. C.-H. Yeow, K. K. Ang, C. Guan, and J. H. Lim, "Brain-computer interface-based soft robotic glove rehabilitation for stroke," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 12, pp. 3339–3351, 2020.

[2] A. Kumar, L. Gao, E. Pirogova, and Q. Fang, "A review of error-related potential-based brain–computer interfaces for motor impaired people," *IEEE Access*, vol. 7, pp. 142 451–142 466, 2019.

[3] A. Kapsalyamov, P. K. Jamwal, S. Hussain, and M. H. Ghayesh, "State of the art lower limb robotic exoskeletons for elderly assistance," *IEEE Access*, vol. 7, pp. 95 075–95 086, 2019.

[4] Y. Zhang, S. Q. Xie, C. Shi, J. Li, and Z.-Q. Zhang, "Cross-subject transfer learning for boosting recognition performance in SSVEP-Based BCIs," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 1574–1583, 2023.

[5] M. Y. Naser and S. Bhattacharya, "Towards practical BCI-driven wheelchairs: a systematic review study," *IEEE Trans. Neural Syst. Rehabil. Eng.*, 2023.

[6] J. J. Podmore, T. P. Breckon, N. K. Aznan, and J. D. Connolly, "On the relative contribution of deep convolutional neural networks for SSVEP-based bio-signal decoding in BCI speller applications," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 611–618, 2019.

[7] Q. Ai, M. Zhao, K. Chen, X. Zhao, L. Ma, and Q. Liu, "Flexible coding scheme for robotic arm control driven by motor imagery decoding," *J. Neural Eng.*, vol. 19, no. 5, p. 056008, 2022.

[8] Y. Xu, H. Zhang, L. Cao, X. Shu, and D. Zhang, "A shared control strategy for reach and grasp of multiple objects using robot vision and noninvasive brain–computer interface," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 1, pp. 360–372, 2020.

[9] D. Liu, W. Chen, K. Lee, R. Chavarriaga, F. Iwane, M. Bouri, Z. Pei, and J. d. R. Millán, "EEG-based lower-limb movement onset decoding: Continuous classification and asynchronous detection," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 8, pp. 1626–1635, 2018.

[10] J. Huang, P. Yang, B. Xiong, Q. Wang, B. Wan, Z. Ruan, K. Yang, and Z.-Q. Zhang, "Incorporating neighboring stimuli data for enhanced SSVEP-Based BCIs," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 71, pp. 1–9, 2022.

[11] Y. Zhang, Z. Li, S. Q. Xie, H. Wang, Z. Yu, and Z.-Q. Zhang, "Multi-objective optimisation-based high-pass spatial filtering for SSVEP-based brain-computer interfaces," *IEEE Trans. Instrum. Meas.*, 2022.

[12] Y. Li, J. Xiang, and T. Kesavadas, "Convolutional correlation analysis for enhancing the performance of SSVEP-based brain-computer interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, 2020.

[13] Y. Zhang, S. Q. Xie, H. Wang, and Z. Zhang, "Data analytics in steady-state visual evoked potential-based brain–computer interface: A review," *IEEE Sensors J.*, vol. 21, no. 2, pp. 1124–1138, 2020.

[14] N. Guo, X. Wang, D. Duanmu, X. Huang, X. Li, Y. Fan, H. Li, Y. Liu, E. H. K. Yeung, M. K. T. To *et al.*, "SSVEP-based brain computer interface controlled soft robotic glove for post-stroke hand function rehabilitation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1737–1744, 2022.

[15] V. Sakkalis, M. Krana, C. Farmaki, C. Bourazanis, D. Gaitatzis, and M. Pediaditis, "Augmented reality driven steady-state visual evoked potentials for wheelchair navigation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2960–2969, 2022.

[16] F. Wang, Y. Wen, J. Bi, H. Li, and J. Sun, "A portable SSVEP-BCI system for rehabilitation exoskeleton in augmented reality environment," *Biomed. Signal. Proces.*, vol. 83, p. 104664, 2023.

[17] X. Chen, B. Zhao, Y. Wang, and X. Gao, "Combination of high-frequency SSVEP-based BCI and computer vision for controlling a robotic arm," *J. Neural Eng.*, vol. 16, no. 2, p. 026012, 2019.

[18] X. Chen, X. Huang, Y. Wang, and X. Gao, "Combination of augmented reality based brain-computer interface and computer vision for high-level control of a robotic arm," *IEEE Trans. Neural Syst. Rehabil. Eng.*, 2020.

[19] X. Zhao, Y. Chu, J. Han, and Z. Zhang, "SSVEP-based brain–computer interface controlled functional electrical stimulation system for upper extremity rehabilitation," *IEEE T. Syst. Man.Cy-S.*, vol. 46, no. 7, pp. 947–956, 2016.

[20] D. Zhang, S. Liu, J. Zhang, G. Li, D. Suo, T. Liu, J. Luo, Z. Ming, J. Wu, and T. Yan, "Brain-controlled 2d navigation robot based on a spatial gradient controller and predictive environmental coordinator," *IEEE J Biomed Health Inform.*, vol. 26, no. 12, pp. 6138–6149, 2022.

[21] K. Sharma and S. K. Maharaj, "Continuous and spontaneous speed control of a robotic arm using SSVEP," in *2021 9th International Winter Conference on Brain-Computer Interface (BCI)*. IEEE, 2021, pp. 1–5.

[22] Y. Joon Kim, M. Grabowecky, K. A. Paller, K. Muthu, and S. Suzuki, "Attention induces synchronization-based response gain in steady-state visual evoked potentials," *Nat. Neurosci*, vol. 10, no. 1, pp. 117–125, 2007.

[23] A. J. Molina-Cantero, J. A. Castro-García, F. Gómez-Bravo, R. López-Ahumada, R. Jiménez-Naharro, and S. Berrazueta-Alvarado, "Controlling a mouse pointer with a single-channel EEG sensor," *Sensors*, vol. 21, no. 16, p. 5481, 2021.

[24] X. Chen, Y. Wang, S. Zhang, S. Gao, Y. Hu, and X. Gao, "A novel stimulation method for multi-class SSVEP-BCI using intermodulation frequencies," *J. Neural Eng.*, vol. 14, no. 2, p. 026013, 2017.

[25] P. Autthasan, X. Du, J. Arnin, S. Lamyai, M. Perera, S. Itthipuripat, T. Yagi, P. Manoonpong, and T. Wilaiprasitporn, "A single-channel consumer-grade EEG device for brain–computer interface: enhancing detection of SSVEP and its amplitude modulation," *IEEE Sensors Journal*, vol. 20, no. 6, pp. 3366–3378, 2019.

[26] X. Zhang, G. Xu, J. Xie, and X. Zhang, "Brain response to luminance-based and motion-based stimulation using inter-modulation frequencies," *PLoS One*, vol. 12, no. 11, p. e0188073, 2017.

[27] D. H. Brainard and S. Vision, "The psychophysics toolbox," *Spatial vision*, vol. 10, no. 4, pp. 433–436, 1997.

[28] Y. Yang, W. Wu, B. Wang, and M. Li, "Analytical reformulation for stochastic unit commitment considering wind power uncertainty with gaussian mixture model," *IEEE Trans. Power Syst.*, vol. 35, no. 4, pp. 2769–2782, 2019.

[29] F. Peng, M. Li, S.-n. Zhao, Q. Xu, J. Xu, and H. Wu, "Control of a robotic arm With an optimized common template-based CCA method for SSVEP-Based BCI," *Front. Neurorobotics.*, vol. 16, 2022.

[30] X. Chen, Y. Wang, S. Gao, T.-P. Jung, and X. Gao, "Filter bank canonical correlation analysis for implementing a high-speed SSVEP-based brain–computer interface," *J. Neural Eng.*, vol. 12, no. 4, p. 046008, 2015.

[31] Z. Wang, Y. Yu, M. Xu, Y. Liu, E. Yin, and Z. Zhou, "Towards a hybrid BCI gaming paradigm based on motor imagery and SSVEP," *Int. J. Hum-Comput. Int.*, vol. 35, no. 3, pp. 197–205, 2019.

[32] M. M. N. Mannan, M. A. Kamran, S. Kang, H. S. Choi, and M. Y. Jeong, "A hybrid speller design using eye tracking and SSVEP brain–computer interface," *Sensors*, vol. 20, no. 3, p. 891, 2020.