Correcting for ERP latency jitter improves gaze-independent BCI decoding

A. Van Den Kerchove,[1,2], H. Si-Mohammed[1], M.M. Van Hulle[2] and F. Cabestaing[1]

[1] Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France
[2] KU Leuven, Department of Neurosciences, Laboratory for Neuro- & Psychophysiology, Campus Gasthuisberg O&N2, Herestraat 49 bus 1021, BE-3000 Leuven, Belgium

E-mail: arne.vandenkerchove@univ-lille.fr, arne.vandenkerchove@kuleuven.be

Abstract. *Objective:* Patients suffering from heavy paralysis or Locked-in-Syndrome can regain communication using a Brain-Computer Interface (BCI). Visual event-related potential (ERP) based BCI paradigms exploit visuospatial attention (VSA) to targets laid out on a screen. However, performance drops if the user does not direct their eye gaze at the intended target, harming the utility of this class of BCIs for patients suffering from eye motor deficits. We aim to create an ERP decoder that is less dependent on eye gaze. *Methods:* ERP component latency jitter plays a role in covert visuospatial attention (VSA) decoding. We introduce a novel decoder which compensates for these latency effects, termed Woody Classifier-based Latency Estimation (WCBLE). We carried out a BCI experiment recording ERP data in overt and covert visuospatial attention (VSA), and introduce a novel special case of covert VSA termed split VSA, simulating the experience of patients with severely impaired eye motor control. We evaluate WCBLE on this dataset and the BNCI2014-009 dataset, within and across VSA conditions to study the dependency on eye gaze and the variation thereof during the experiment. *Results & discussion:* WCBLE outperforms state-of-the-art methods in the VSA conditions of interest in gaze-independent decoding, without reducing overt VSA performance. Results from across-condition evaluation show that WCBLE is more robust to varying VSA conditions throughout a BCI operation session. Together, these results point towards a pathway to achieving gaze independence through suited ERP decoding. Our proposed gaze-independent solution enhances decoding performance in those cases where performing overt VSA is not possible.

Keywords brain-computer interface, event-related potential, jitter, gaze-independence, covert attention, split attention

Submitted to: *J. Neural Eng.*

1. Introduction

Brain-Computer Interfaces (BCIs) decode brain activity to establish a communication channel that does not rely on speech or muscular activity (Naci et al., 2012; Chaudhary et al., 2016), which in turn can provide solutions to paralyzed individuals. The quest for performant and affordable solutions is most evident in visual BCIs using the electroencephalogram (EEG), where it has led to a gamut of increasingly sophisticated decoders, tailored to the needs of specific stimulation paradigms and usage contexts.

Traditional visual BCI scenarios require the user to overtly direct their visuospatial attention (VSA) and gaze toward the screen target they intend to select. In most settings, screen targets are overlaid with non-overlapping, transient stimuli that evoke event-related EEG potentials (ERPs). The selected target can be decoded from these ERPs, as is the case for the oddball paradigm where observing a rare but attended stimulus evokes a P3 ERP component. However, a critical challenge arises when users rely solely or in part on covert VSA, which involves directing visuospatial attention without corresponding eye gaze. In these cases, classical solutions often fall short of the widely accepted 80% target selection accuracy threshold deemed necessary for a comfortable user experience (Brunner et al., 2010; Frenzel et al., 2011; Treder and Blankertz, 2010; Ron-Angevin et al., 2019; de Neeling and Van Hulle, 2019), calling for alternative, gaze-independent solutions.

In this work, we will use the term *gaze-independent* meaning 'dealing explicitly with the fact that a user cannot control their gaze.' In the context of a visual BCI, this means that the user's visuospatial attention and their gaze do not necessarily coincide. Gaze-independent paradigms are particularly promising for individuals with impaired eye motor control, such as those suffering from specific types or stages of Amyotrophic Lateral Sclerosis (ALS), Multiple Sclerosis (MS), stroke, or brain stem stroke. For these patients, gazing directly at a screen target may be uncomfortable, impractical, or even impossible. Hence, assistive devices that rely on eye tracking are often inefficient for them. Consequently, while BCIs hold great promise for these individuals, conventional gaze-dependent BCI solutions do not meet their needs due to the absence of gaze control. Therefore, the development of decoding strategies that account for covert VSA becomes crucial in the pursuit of high-performance gaze independent BCIs.

Concept	**Definition**
VSA	Visuospatial attention, attention directed at a stimulus visible in the field of view
Overt VSA	VSA and gaze are directed at the same stimulus
Covert VSA	VSA is directed at a stimulus while the gaze is directed at an empty region
Split VSA	VSA is directed at a stimulus while the gaze is directed at a different stimulus
Distractor	The stimulus to which gaze is directed in split VSA
Gaze-independent	Coping with the dissociation between the locus of VSA and the gaze
Jitter	Variability of single-trial latencies relative to stimulus onset

Table 1: Definitions of some concepts presented in this work.

The novelty of our work consists of the following elements. Firstly, we introduce an ERP decoding algorithm designed to perform well in gaze-independent settings by explicitly addressing the increased variability of single-trial peak latencies (*jitter*) of the P3 ERP component relative to stimulus onset (Aricò et al., 2014). Secondly, we evaluate our algorithm in a range of settings that are of interest in gaze-independent decoding, including a special case of covert VSA termed *split* VSA. In split VSA, participants direct their visuospatial attention to one target while looking at a different one, called the *distractor*. Split VSA mimics the involuntary gaze drift of patients with limited eye motor control, as they might unintentionally fixate on an unrelated target. Despite its potential significance, split VSA remains an underexplored area in the visual BCI literature. With the proposed algorithm, we introduce a method to allow split attention in gaze-independent BCI operation.

2. Related work

2.1. Gaze-independent BCIs

Gaze-independent ERP-based BCIs (Riccio et al., 2012) can be realized in three ways. Firstly, non-visual stimulation paradigms such as auditory and somatosensory paradigms do not rely on gaze redirection but often result in lower information transfer rates, increased mental effort and user-dependent variability (Reichert et al., 2020b). Secondly, visual stimulation can be optimized, e.g. such that the stimuli are always present in the field of view (Treder and Blankertz, 2010; Pires et al., 2011; Lees et al., 2018). Non-spatial visual attention (feature attention) can also be exploited, such as attention to stimulus color, shape or symbol (Zhang et al., 2010; Treder et al., 2011; Hwang et al., 2015). Alternative visual stimulation paradigms can modulate specific ERP components that are more sensitive to stimulation in the visual periphery (Schaeff et al., 2012; Xu et al., 2022). However, they still rely to some extent on eye motor control, often necessitating central gaze fixation.

Thirdly, stimuli can be presented in a standard BCI paradigm, but visuospatial attention can be decoded separately from gaze direction. Aloise et al. (2012b) aimed to bridge the performance gap between covert and overt VSA decoding performance. They compared classical linear and non-linear ERP classifiers on a covert attention P3 ERP component dataset. The results revealed no significant performance improvement in covert VSA decoding for any of the investigated decoders. Aricò et al. (2014) observed higher variability in single-trial P3 peak latencies relative to stimulus onset during covert VSA compared to overt VSA. This latency variability contributes to reduced covert VSA decoding performance. While they proposed an analysis and performance prediction method, they did not provide a decoding solution. They suggested that compensating for latency jitter could enhance covert VSA decoding, but did not actually verify this hypothesis directly. Additionally, Hardiansyah et al. (2020) developed a classifier for covert VSA ERPs, exploiting single-trial latency features in

combination with amplitude features for classification with a support vector machine. They demonstrated the positive influence of single-trial ERP component latency features on covert VSA inference yet did not attempt to correct the amplitude features for these latencies. Frenzel et al. (2011) introduced a similar protocol to our split VSA setting. They showed that it is possible to perform split VSA and that, in this case, visuospatial attention and gaze direction can separately be decoded using classical ERP techniques. To the best of our knowledge, this is the sole study that investigated split attention in ERP-based BCIs. However, Frenzel et al. (2011) considered their interface only for the case where the user actively intends to select both targets determined by the gaze and the VSA. In the split VSA setting considered in our work, we rather instruct the participant to ignore the distractor and only attend the cued target, since we are interested in decoding the visuospatial attention only.

P3 latency generally falls between 350ms and 600ms (Luck, 2014), but this value is heavily dependent on the subject and the task, and can vary from trial to trial (Ouyang et al., 2017). The work of Aricò et al. (2014) illustrates that the variation in single-trial P3 latencies is important in gaze-independent decoding and has been hampering covert VSA decoding performance. In this work we aim to reprise their hypothesis stating that jitter compensation improves covert VSA performance and extend it by developing a decoder and evaluating it in a broader range of gaze-independent settings, including split VSA.

2.2. Latency estimation

Single-trial ERP latency jitter introduces a problem in ERP analysis: the *smearing effect* occurs when aggregating over multiple identical signals with different latencies. A familiar example of this happens when averaging over jittered epochs for ERP analysis. In the presence of jitter, the shape and amplitude of the average do not reflect the properties of the original signal. As the smearing effect reduces the amplitude of the average, it negatively impacts the signal-to-noise ratio (SNR). The smearing effect equally impacts the SNR of information captured by a classifier's parameters when training with a procedure that does not account for this jitter, and thus also affects its performance (Thompson et al., 2012). Hence, effective latency estimation and jitter compensation can contribute to higher decoding performance.

The most straightforward latency estimation method is *peak picking*. Peak picking determines the latency of an ERP as the time point of its maximum (or minimum for negative components) amplitude relative to stimulus onset. While easy to implement, this method does not perform well in low or negative SNR conditions because of the risk of picking a noise-related peak, unless combined with strong filtering in the space, time or frequency domains (D'Avanzo et al., 2011; Aricò et al., 2014; Treder et al., 2016; Ouyang et al., 2017). Due to the risk of picking noise peaks, template matching is generally preferred. Woody's algorithm (Woody, 1967) is a simple and performant latency estimation method that is considered to be among the most performant ones (Ouyang

et al., 2017). Woody's scheme starts with a template that approximates the expected response, e.g. the average target ERP, and iteratively enhances the SNR of this template by determining at each iteration the time point of maximum cross-correlation of the template with all single-trial ERPs, and aligning them based on these latencies to form the template for the next iteration. The final, enhanced template can then be cross-correlated again to retrieve an accurate estimation of single-trial latencies. Similar to peak-picking, cross-correlation template matching can be combined with a spatial filter such as XDAWN (Souloumiac and Rivet, 2013) or a spatiotemporal filter (Iturrate et al., 2014). More recent work achieves good performance in latency estimation using residue iteration (Ouyang et al., 2017), genetic algorithms (Pelo et al., 2018), a hidden process model (Kim et al., 2020) or graph-based network analysis (Dimitriadis et al., 2018).

Most of these methods suffer from a common drawback: they cannot be used in a decoding scheme to improve performance for incoming epochs with unknown class labels that contain either a target or non-target ERP response. They can be used offline on a set of labeled epochs for testing hypotheses concerning latency and jitter, for aligning templates, or for BCI performance prediction, but not for ERP classification. While some of the aforementioned methods could be adapted to perform classification tasks, few studies investigate how to exploit this latency estimation for jitter-resistant decoding. Hardiansyah et al. (2020) incorporated single-trial latencies in classification by peak-picking within a given ERP time window, unaware of the class of the epoch under investigation. The Classifier-Based Latency Estimation (CBLE) algorithm introduced by Thompson et al. (2012) also explicitly applies latency estimation in a decoding setting. Thompson et al. (2012) initially formulated CBLE as an off-line performance prediction method. Later, its output was successfully adapted to compensate for jitter to improve decoder performance (Mowla et al., 2017; Zisk et al., 2022).

Time series classification algorithms (Abanda et al., 2019) that are robust to jitter can be used in a decoding setting, but, in general, have scarcely been applied to ERP decoding. Data augmentation involving jittering the training data (Krell et al., 2018; Zisk et al., 2022) and Riemannian Geometry methods using spatial covariances as features (Aydarkhanov et al., 2020) have both been shown to perform well in the presence of ERP jitter. In this work, we opted to apply CBLE because it has successfully been applied to classify jittered ERPs. We adapt the CBLE to an iterative method akin to Woody's scheme that can better compensate for jitter, to improve covert VSA decoding performance.

3. Materials and Methods

3.1. Decoders

3.1.1. Classifier-based Latency Estimation Consider a training set of N EEG epochs with C channels of S samples $\{\mathbf{X}_n^{\text{train}} \in \mathbb{R}^{C\times S}\}_{n=1}^{N}$ with corresponding training labels

$\mathbf{l}^{\text{train}} \in \{\text{target}, \text{non-target}\}_{n=1}^{N}$, and a similar testing set of M epochs $\{\mathbf{X}_m^{\text{test}} \in \mathbb{R}^{C\times S}\}_{m=1}^{M}$. We assume that the sampling period is T, i.e. that the sample with index s is sampled at time sT. In the following, we use the matrix slicing notation to denote row or column intervals extracted from a matrix. For instance, $\mathbf{X}[:, s_1 : s_2]$ denotes all columns of $\mathbf{X}$ with indices between s_1 (included) and s_2 (excluded).

CBLE, summarized in Algorithm 1, works by training a *first-stage* classifier $\mathcal{C}(\theta, f)$ defined within a time period $[s_1 : s_2]$ with a set of parameters θ and a decision function $f(\mathbf{X}[:, s_1 : s_2], \theta) \to y$ outputting a classification score $y \in \mathbb{R}$ for a given epoch $\mathbf{X}$, such that

$$\theta = \text{train}_{\mathcal{C}}(\{\mathbf{X}_n^{\text{train}}[:, s_1 : s_2]\}_{n=1}^{N}, \mathbf{l}^{\text{train}}) \tag{1}$$

Then, f can be applied to all (possibly overlapping) slices of length $s_2 - s_1$ of an epoch $\mathbf{X}$, resulting in a vector of score values $\mathbf{y} = [y_1 \ \dots \ y_R]^T \in \mathbb{R}^R$ such that

$$y_s = f(\mathbf{X}[:, s : s + (s_2 - s_1)], \theta) \quad \forall s \in 1, \dots, R \tag{2}$$

with $R = S - (s_2 - s_1)$. To leverage CBLE for ERP classification, the score vectors $\mathbf{y}$ can be arranged in matrices $\mathbf{Y}^{\text{train}} \in \mathbb{R}^{N\times R}$ and $\mathbf{Y}^{\text{test}} \in \mathbb{R}^{M\times R}$. These can be further classified by training a *second-stage* classifier on $\mathbf{Y}^{\text{train}}$ and class labels $\mathbf{l}^{\text{train}}$. However, the resulting score-over-time vectors per epoch still suffer from jitter. For classification, we follow the approach of Mowla et al. (2017), using a maximum-level hierarchical Daubechies-4 wavelet transform to reduce dimensionality before classification with the second-stage classifier. In the CBLE-decoder, it is this wavelet transform that decreases the sensitivity to latency differences, actively compensating for ERP latency jitter.

When using a simple spatiotemporal linear classifier as first-stage classifier, CBLE is equivalent to the first iteration of Woody's algorithm with the spatiotemporal classifier weights as template. Thompson et al. (2012), Mowla et al. (2017) and Mowla et al. (2020) show that CBLE is relatively independent of the first-stage classifier for BCI accuracy prediction and for ERP classification. Therefore, we opt to use the variant of Linear Discriminant Analysis with block-Toeplitz regularized covariance matrix (tLDA) proposed by Sosulski and Tangermann (2022), as the first-stage classifier and logistic regression as second stage.

3.1.2. Robust CBLE latency features The ERP decoding method based on CBLE introduced by Mowla et al. (2017) does not make use of the extracted latencies, only passing score matrix $\mathbf{Y}$ on to the second-stage classifier. We noticed that, while CBLE performance was unaffected, the classification performance of our proposed method can be improved if the estimated latencies are also made available as features to the second-stage classifier, after a square transform for linear separability (Thompson et al., 2012).

These latency features give rise to the following issue when classifying unseen data. The CBLE latency estimate is defined only for target epochs as $s_{\text{target}} = \arg\max_s y_s$. This is the point in time where the target class reaches largest separation from the background noise and non-target class, indicating the target ERP is most likely to occur here. However, in a classifier test phase, it is not known a priori whether an

unseen epoch is a target or a non-target epoch. This problem is solved by defining an estimated latency per class s_{target} and $s_{\text{non-target}}$ for every epoch, regardless of its actual class. The estimated class latencies can then be used as features for training and testing the second-stage classifier. This way, the latencies of the testing data can be presented to the second-stage classifier without knowledge of the testing data class labels, making them useful in decoding.

In a similar manner to the target latency, the non-target latency could be defined as $s_{\text{non-target}} = \arg\min_s y_s$. However, this is problematic since it is not evident how to estimate latency of e.g. a P3 ERP component for a non-target epoch, since the non-target class is characterized by the absence of this component. In fact, y can have multiple local minima or entirely lack distinct peaks for non-targets, rendering the minimum estimate meaningless.

Instead, we opt for a more robust, probabilistic definition of class latencies. This robust estimation method yields latencies that (1) are more meaningful as input for the second-stage classifier, and (2) lead to smoother convergence in our proposed iterative alignment scheme for WCBLE, which heavily relies on exact latency estimation. Assume classifier $\mathcal{C}(\theta, f, \Pr)$ now can also output a probability per class $\Pr(l|\mathbf{X}[:, s_1 : s_2 :)], \theta)$ for a given epoch $\mathbf{X}$, a feature of many common classifiers. Analogous to equation 2, we can now write

$$\Pr(\mathbf{X}, \theta, l, s) = \frac{1}{R} \Pr(\mathbf{X}[:, s : s + (s_2 - s_1)], \theta, l) \quad \forall \quad s \in 1, \ldots, R \tag{3}$$

The latency features assuming the epoch belongs to a given class $l \in \{\text{target}, \text{non-target}\}$ are then defined as the median of the corresponding distributions

$$s_l = \text{median}\left[\Pr(s|\mathbf{X}, \theta, l)\right] \tag{4}$$

Note that $\Pr(s|\mathbf{X}, \theta, \text{non-target}) = 1 - \Pr(s|\mathbf{X}, \theta, \text{target})$. The median of the probability distribution over time is more robust to outliers and noise than the maximum or minimum score. For the non-target case, the median approach tends towards the center of a near-uniform distribution, resulting in a more consistent latency estimate over trials as compared to the minimum approach.

3.1.3. Classifier-based Latency Estimation with Woody Iterations To improve performance over CBLE, we propose a new algorithm inspired both by CBLE and the aforementioned Woody iteration scheme (WCBLE). Instead of using CBLE to estimate the features of a second-stage classifier directly, CBLE latency estimation is used as a step in a Woody iteration scheme. While the Woody algorithm iteratively enhances the SNR of an ERP template to cross-correlate with the data, WCBLE iteratively re-estimates the parameters of the first-stage classifier. To improve convergence and perform well in a classification setting, WCBLE aligns both targets and non-targets to their corresponding estimated latencies.

The WCBLE algorithm is presented in Algorithm 2. Its training phase is visualized in figure **1**. The initial training epochs $\{\mathbf{X}_n^{(1)}\}_{n=1}^N$ are set to $\{\mathbf{X}_n^{\text{train}}\}_{n=1}^N$. At every

iteration, classifier $\mathcal{C}$ is trained like in CBLE:

$$\theta^{(i)} = \text{train}_{\mathcal{C}}(\{\mathbf{X}_n^{(i-1)}[:, s_1 : s_2]\}_{n=1}^{N}, \mathbf{l}^{\text{train}}) \tag{5}$$

Next, latency $s_{l_n}^{(i)}$ is determined for every epoch $\mathbf{X}^{(i)}$ corresponding to its class label l_n using equation 4. Finally, the training epochs $\mathbf{X}^{(i+1)}$ for the next iteration are determined by aligning each original training epoch to the latency $s_{l_n}^{(i)}$ corresponding to its respective class label.

$$\mathbf{X}_n^{(i+1)} = \text{align}(\mathbf{X}_n^{\text{train}}, s_{l_n}^{(i)}) \quad \forall \quad n = 1, \ldots, N \tag{6}$$

Aligning is performed by shifting and zero-padding the signal to the right if the latency is negative relative to the time window onset, and to the left if positive, by the difference between the latency and the window onset. The process halts after a fixed amount of iterations or when the estimated set of latencies has been encountered before, indicating it ended up in a loop. In the end, the procedure should result in enhanced classifier parameters θ^*, closer to those when there would be no jitter between epochs. We can then apply the classifier with enhanced parameters θ^* in a CBLE manner to unseen epochs as illustrated in figure **2** to obtain a vector of scores over time as in section 3.1.1 and the estimated latencies as in section 3.1.2.

3.1.4. Riemannian Geometry We will compare WCBLE, CBLE and tLDA to Riemannian Geometry approaches that rely on spatial covariance as features. Together with tLDA, Riemannian Geometry generally reaches state-of-the-art decoding performance (Lotte et al., 2018). We implemented two Riemannian Geometry pipelines. The first one estimates shrunk covariances from the ERPs filtered with 6 XDAWN filters, projects these covariances to a tangent space and classifies the result using $L2$-regularized logistic regression (XDAWNCov-TS-LR) (Cecotti et al., 2017). Secondly, we adopt the pipeline from Aydarkhanov et al. (2020), since their work shows favorable performance in the presence of single-trial ERP latency jitter. Shrunk spatial covariance matrices are estimated from epochs that are augmented by concatenating the average target and average non-target ERP as extra channels, projected to tangent space and classified using $L2$-regularized logistic regression (ERPCov-TS-LR).

3.2. Experiments

We evaluate our approach on a publicly available dataset and one specifically recorded for this study, designed to probe different modalities of covert VSA.

3.2.1. CVSA-ERP dataset We recorded a dataset to validate our approach. The Covert Visuospatial Attention ERP (CVSA-ERP) dataset consists of 15 participants, mean age 26.34 ± 3.04 years. This study was approved by the Ethics Commission of University Hospital Leuven (S62547). Appendix A details the stimulation and recording procedure. Each subject performed different VSA conditions (overt, covert and split),

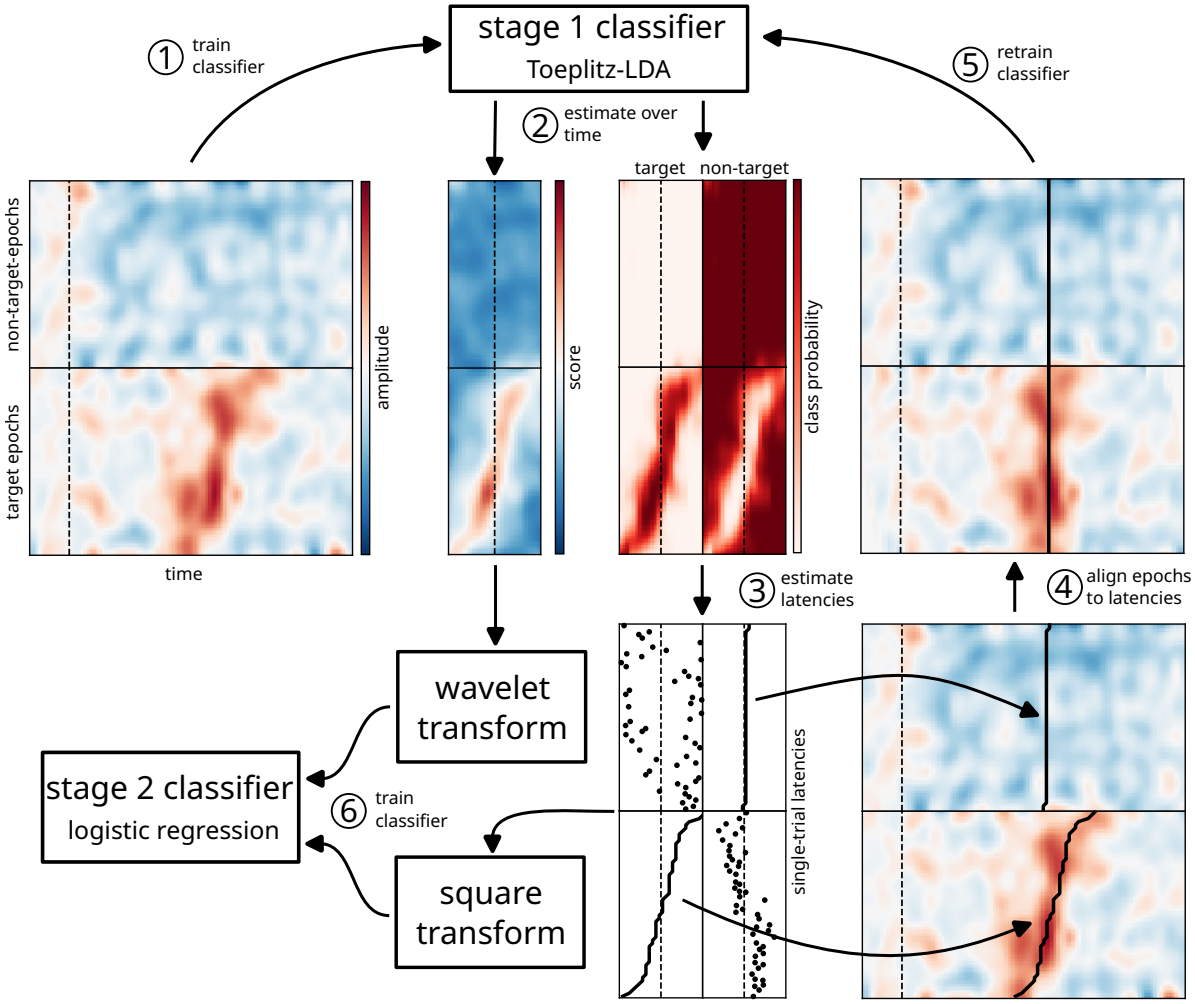

Figure **1**: Schematic representation of the Woody Classifier Based Latency Estimation training phase. (1) The first-stage spatiotemporal binary classifier is trained on a set of epochs. (2) It is then applied to time-shifted copies of the epochs to obtain scores and class probabilities over time. (3) The medians of these probability distributions are assumed the class latencies. (4) The epochs are aligned to their corresponding class latencies by shifting in time such that all latencies fall at the same moment. (5) The spatiotemporal classifier is then retrained on the aligned epochs for a next iteration. (6) After the iterative process halts, the scores and latencies obtained from the last iteration are used to train the second-stage classifier.

illustrated in figure **3A**. Using a hexagonal layout interface, similar to the visual Hex-o-Spell proposed by Treder and Blankertz (2010), we present six flashing targets (without letters or symbols) to the participant while the EEG, electrooculogram (EOG), and the participant's eye gaze using eye tracking were recorded. The VSA conditions described in the first row of figure **3A** are considered.

In contrast to the protocol proposed by Frenzel et al. (2011), split VSA was performed by instructing the participant to attend the intensifications of the cued target, and ignore the intensifications of the distractor target. Since we assume there will be an effect depending on the distance between attended target and the distractor, we discern

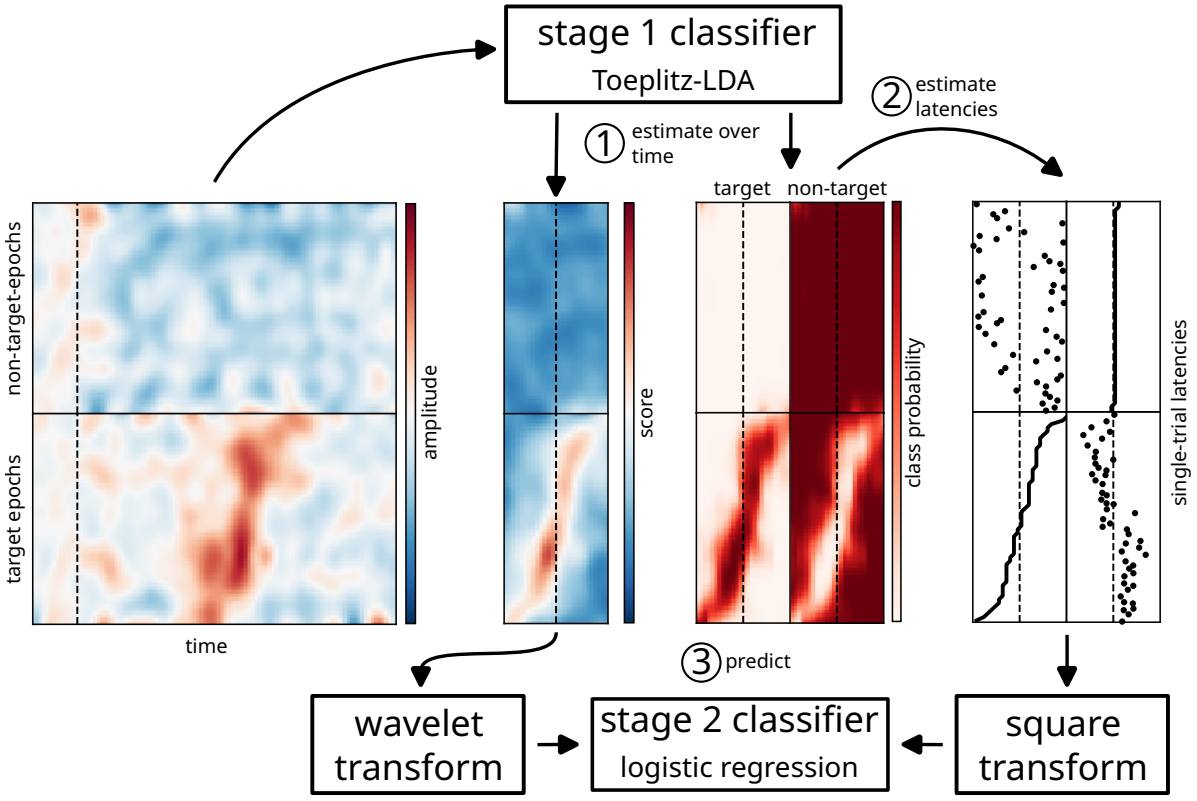

Figure **2**: Schematic representation of the Woody Classifier Based Latency Estimation test phase. (1) The first-stage spatiotemporal binary classifier obtained from the training phase is applied to time-shifted copies of the epochs to obtain scores and class probabilities over time. (2) The medians of these probability distributions are assumed as the new class latencies. (3) The scores and class latencies are input to the trained second-stage classifier, which predicts the label of the epochs.

three split VSA sub-conditions: the distractor is either clockwise or counterclockwise directly next to the attended target ($d = 1$), there is one other target between the attended target and the distractor ($d = 2$), or the distractor is opposite to the intended target ($d = 3$).

3.2.2. BNCI2014-009 dataset The BNCI2014-009‡ dataset (Aloise et al., 2012a) was used in the analysis performed in Aricò et al. (2014). It contains data from 10 subjects (median age 24.5 ± 1.9 years) that performed two spelling tasks illustrated in the second row of figure **3A**: using the P3 Matrix speller interface to exploit overt VSA, and the GeoSpell covert VSA interface. To use the GeoSpell interface, the participant gazes at the fixation point at the center of the screen, while groups of characters flash simultaneously in a circular layout around the fixation point. The user directs their visuospatial attention to the location where the intended letter is expected to appear, and when it does, a P3 ERP component is expected to be evoked. This results in a specific setting where both visuospatial attention and feature attention (the attended

‡ `https://bnci-horizon-2020.eu/database/data-sets`

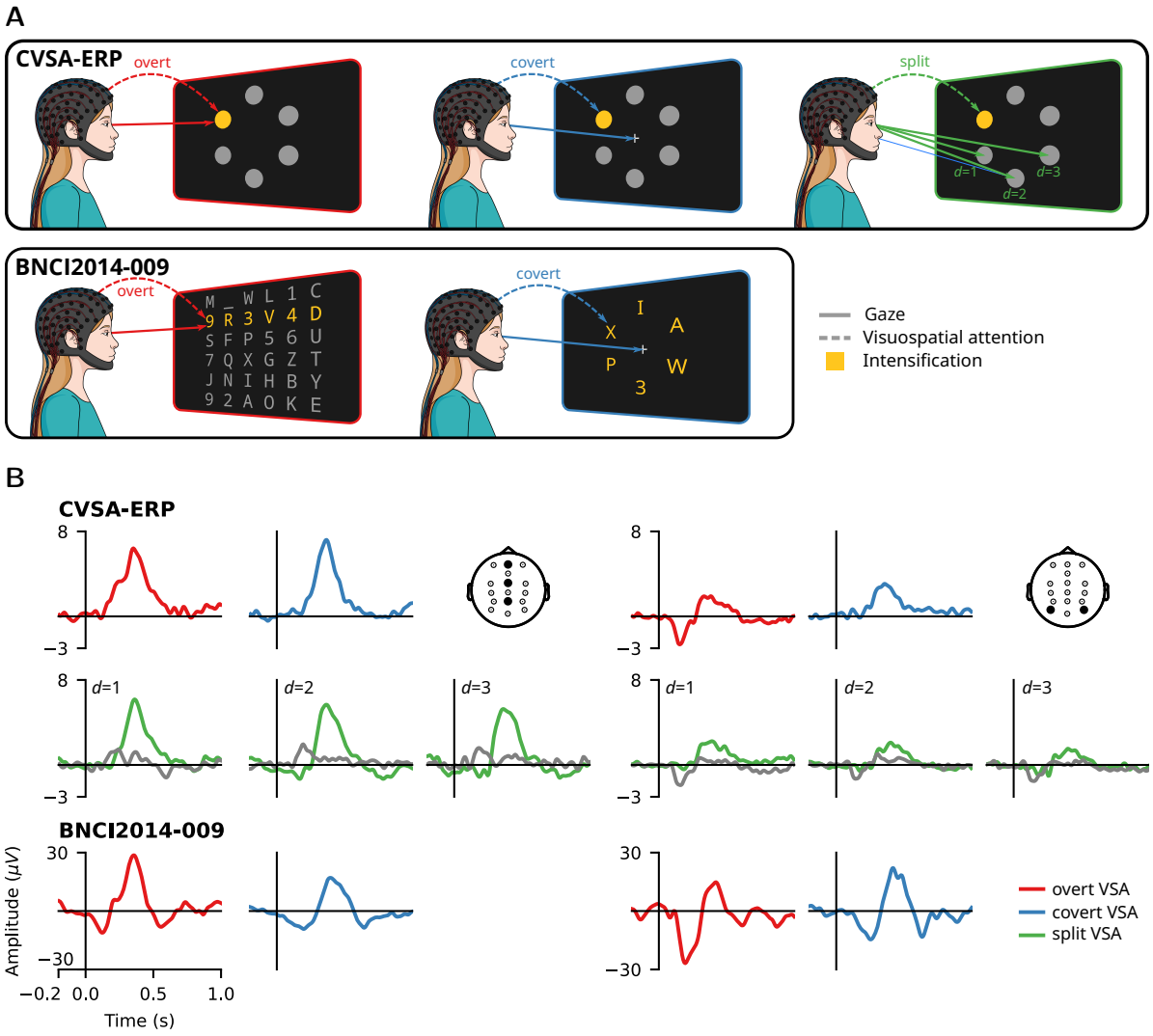

Figure 3: **A** Interfaces and visuospatial attention (VSA) conditions in the CVSA-ERP and BNCI2014-009 datasets. In the CVSA-ERP oddball BCI interface, screen targets are intensified one after the other in pseudorandom order while the participant can either pay overt, covert, or split VSA to the cued target. In the BNCI2014-009 overt VSA interface, entire rows and columns are intensified at once. In its covert counterpart, groups of 6 letters are intensified one after the other, partly relying on feature attention. **B** Contrast between target (color) and non-target, and distractor (gray) and non-target grand average event-related potentials per VSA condition and dataset. Overt VSA yields a strong modulation of the N1 component in both datasets; the P3 amplitude decreases with the degree of split VSA. In split VSA, N1 and P2 are more prominently evoked by the distractor, while the P3 is evoked by the target.

letter) are exploited. For a detailed description of the paradigm and dataset, we refer to Aloise et al. (2012a).

3.2.3. Data preprocessing and analysis Analysis was performed using Python and the MNE software package (version 1.3.1) (Gramfort, 2013). All datasets were band-pass filtered between 0.1Hz and 20Hz with a 4th-order Butterworth filter. Bad channels in the data were automatically detected using the RANSAC method (Fischler and Bolles, 1981) and rejected. The recorded EEG was re-referenced offline to the average of the mastoid electrodes TP9 and TP10. Next, the EEG signals were corrected for eye movement artifacts using an Independent Component Analysis (ICA). Since we have access to EOG data for the CVSA-ERP dataset components correlating significantly with the EOG were rejected. For the BNCI2014-009 dataset, ICA components were manually rejected. Finally, the EEG signal is divided into epochs ranging from 100ms before stimulus onset to 700ms after stimulus onset and down-sampled to 128Hz. In both datasets, only 16 channels were kept for analysis (Fz, FCz, Cz, CPz, Pz, Oz, F3, F4, C3, C4, CP3, CP4, P3, P4, PO7 and PO8).

To evaluate performance, 6-fold cross-validation without shuffling was performed for both datasets. At each fold, classifiers were trained on five target selection blocks (300 epochs) and tested on one block (60 epochs) without overlap for CVSA-ERP. For each subject and run in the BNCI2014-009 dataset, classifiers were trained on five symbol selections (480 epochs) and tested on one symbol selection (96 epochs) without overlap at each fold. A window ranging from 0ms to 600ms after stimulus onset was used for CBLE and WCBLE. With epochs ranging from -100ms to 700ms relative to stimulus onset, this allows for extracting latencies ranging from -100ms to +100ms.

4. Results

4.1. BCI decoding performance

We evaluated the BCI decoding performance in a single-trial classification experiment, as well as in a target selection experiment reflecting BCI operation.

Figure **4A** shows a comparison of areas under the Receiver Operating Characteristic curve (ROC-AUC) for all pairs of tLDA, CBLE and WCBLE for single-trial classification to investigate the contributions of CBLE and WCBLE relative to their first-stage classifier tLDA. For this evaluation, epochs were rejected when the peak-to-peak amplitude exceeded $800\mu V$ and, for the CVSA-ERP dataset, if the user's gaze differed more than 10 degrees of visual angle from the fixation crosshair. Wilcoxon signed-rank tests controlled for multiple comparisons by Benjamini and Hochberg's False Discovery Rate procedure (FDR) revealed that for the BNCI2014-009 dataset, WCBLE significantly outperformed tLDA ($\Delta\text{ROC} - \text{AUC} = 0.019$, $p = 0.004$) and CBLE ($\Delta\text{ROC} - \text{AUC} = 0.016$, $p = 0.036$) for covert VSA but was significantly outperformed by tLDA in overt VSA decoding ($\Delta\text{ROC} - \text{AUC} = -0.004$, $p = 0.040$). For the

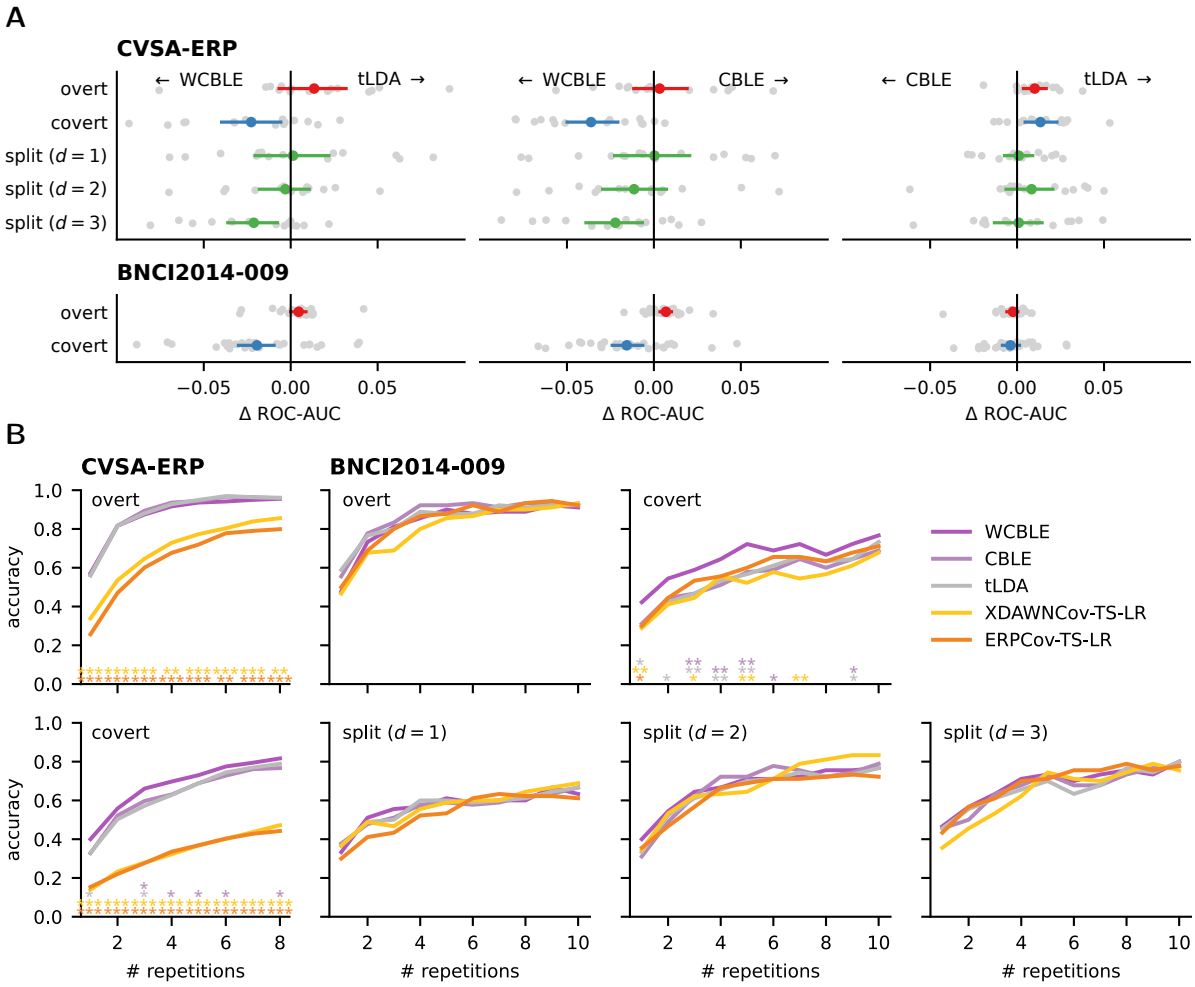

Figure **4**: **A** Difference in cross-validated single-trial classification receiver operating characteristic curve (ΔROC-AUC) between Classifier-based Latency Estimation (CBLE), Woody CBLE (WCBLE), and their first-stage classifier (tLDA). 95% confidence intervals were determined using $k = 1000$ bootstrapping. Our proposed WCBLE decoder outperforms tLDA and CBLE for covert and split ($d = 3$) visuospatial attention (VSA). CBLE scores on par with tLDA. **B** Cross-validated target selection accuracy for all decoders plotted n function of the number of test repetitions in different VSA conditions. Significance was determined using one-sided (WCBLE $>$ other) Wilcoxon rank-sum tests using False Discovery Rate correction ($* = p < 0.05$, $** = p < 0.01$, $*** = p < 0.001$). WCBLE generally achieves highest covert VSA target selection accuracy.

CVSA-ERP dataset, WCBLE also achieved significantly better covert VSA performance than tLDA ($\Delta\text{ROC} - \text{AUC} = 0.023$, $p = 0.041$) and CBLE ($\Delta\text{ROC} - \text{AUC} = 0.036$, $p = 0.024$) . We found no significant difference in WCBLE performance over tLDA in the split VSA conditions in the CVSA-ERP dataset, but results show a clear trend of increase in WCBLE performance over tLDA and CBLE as d increases CBLE failed to significantly outperform its first-stage classifier tLDA in all evaluated VSA conditions. Table B1 reports all single-trial classification scores for all considered models, datasets and conditions.

Figure **4B** shows the cross-validated BCI selection accuracy on the BNCI2014-009

and the CVSA-ERP dataset for all investigated decoders. Accuracy was determined by, for each block, selecting the character with the highest (stage-two if applicable) classifier score and comparing it to the cued target. Significance was calculated using one-sided Wilcoxon rank-sum tests ($p = 0.05$) corrected for FDR over decoders. For this evaluation, no epochs were rejected to keep the trial-based structure of BCI operation intact. For all datasets and VSA conditions, CBLE scores approximately on par with tLDA. Yet, WCBLE yields an improved decoding accuracy for covert VSA in both datasets, which is greatest for smaller numbers of repetitions and decreases as the number of repetitions increases. This covert VSA accuracy increase over tLDA is significant in the BNCI2014-009 dataset for 1 and 3 repetitions and in CVSA-ERP for 1-5 and 10 repetitions. Furthermore, while we reported a relative decrease in single-trial ROC-AUC for WCBLE in overt VSA, this does not seem to result in a consistent decrease in target selection accuracy. No significant increase of WCBLE over other methods was found in split VSA. While Riemannian methods are significantly outperformed by tLDA, CBLE and WCBLE in the BNCI2014-009 dataset, they perform approximately on par with tLDA and WCBLE in CVSA-ERP.

Overall, we observed a 5.10%pt. accuracy increase with WCBLE over tLDA for covert VSA in the BNCI2014-009 dataset and 5.55%pt. in the CVSA-ERP dataset. These results compare to the performance gain in Zisk et al. (2022). They observed a 5.63%pt. accuracy increase with 1-10 selection repetitions over step-wise Linear Discriminant Analysis (SWLDA) for 6 ALS patients, whose SWLDA performance also suffered from jitter. Note that interpretation of this comparison may be challenging due to differences in interface design (number of targets, inter-stimulus interval), subject population, EEG recording procedure and available training data.

4.2. Gaze-independence through cross-condition transfer

To further back our claim of gaze-independence in the case where eye motor control cannot be assumed, we evaluate our proposed decoder in a transfer learning setting between VSA conditions. While performing covert attention requires gaze redirection for each target selection, performing covert or split VSA continuously still requires sustained gaze fixation, which might not be possible for some patients that could benefit from such an application. Studying the transfer between conditions simulates what happens when the user performs different VSA conditions throughout the experimental session. Furthermore, if our decoder performs well in transfer-learning settings, it must capture some information about the ERP responses that is independent of the VSA condition, and hence does not depend on gaze redirection to perform these conditions. We introduce an additional setting of interest here, namely on a combination of VSA conditions, which represents those cases where patients cannot redirect their gaze and hence can be in any one of the VSA conditions depending on the target they attend. For BNCI2014-009, this is implemented as an equal mix of overt and covert VSA, for CVSA-ERP the combined condition represents an equal mix of overt, covert and split

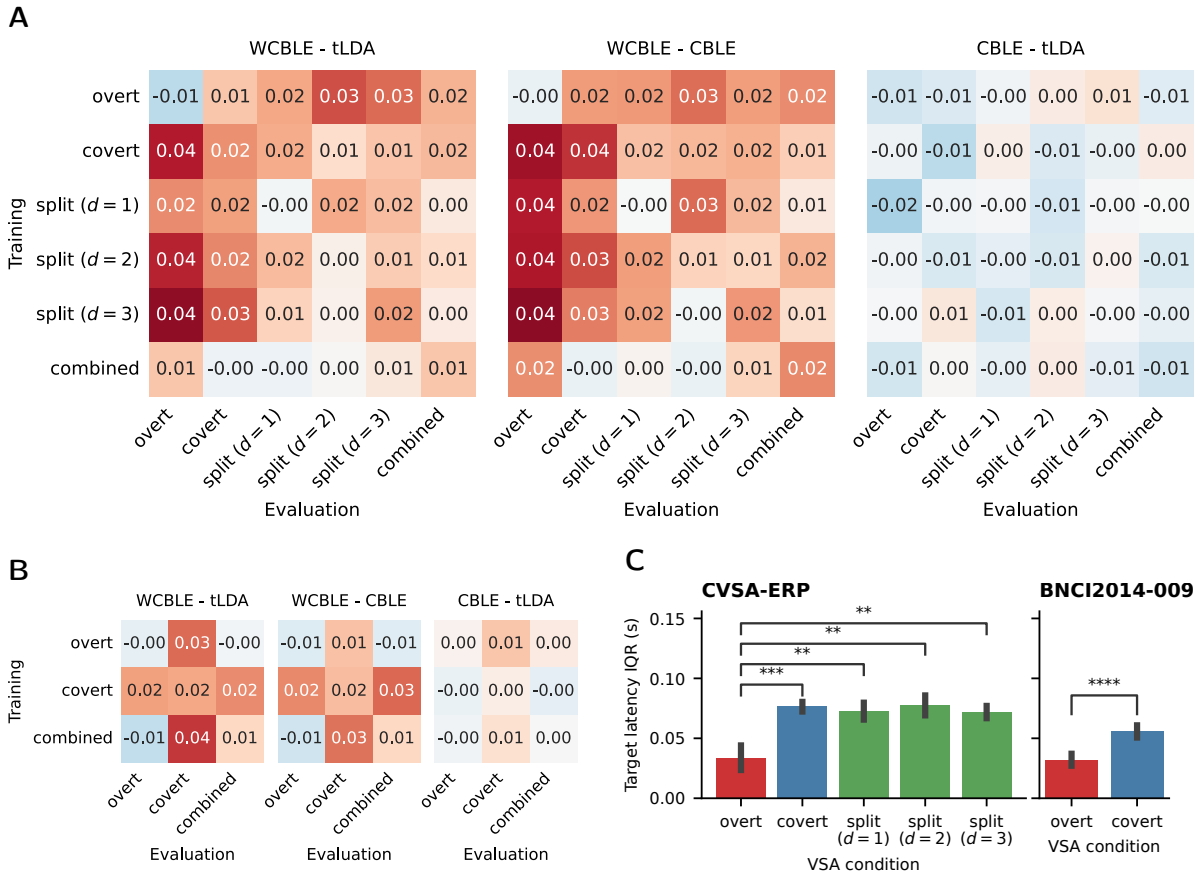

Figure **5**: **A**,**B** Difference in cross-validated area under the receiver operating characteristic curves between Classifier-based Latency Estimation (CBLE) and Woody CBLE (WCBLE) across conditions for the CVSA-ERP and BNCI2014-009 datasets respectively. A decoder is each time trained on a visuospatial attention (VSA) condition and tested on all VSA conditions. WCBLE yields an improvement in most non-overt VSA settings, indicating it is more invariant to eye gaze than CBLE and tLDA. **C** Jitter characterized as the inter-quartile range (IQR) of target epochs for different VSA conditions. Overt VSA exhibits lower jitter than other conditions. Significance of differences was determined with two-sided Wilcoxon rank-sum tests with False Discovery Rate correction on per-subject jitter ($* = p < 0.05$, $** = p < 0.01$, $*** = p < 0.001$, $**** = p < 0.0001$).

VSA, disregarding parameter d.

Figure **5A** and figure **5B** show the pair-wise differences in area under the ROC curve (ΔROC-AUC) between the investigated decoders. In this evaluation, bad epochs were rejected as in section 4.1. When comparing CBLE and tLDA, we do not observe large differences in any of the evaluated settings, similar to the within-subject conditions. On the contrary, when considering the comparisons between WCBLE and tLDA, we see that performance is on par or greater using WCBLE for most conditions, except for within-overt VSA decoding.

5. Discussion

Figure **4A** shows that WCBLE significantly improves covert VSA decoding. This is advantageous for the development of a class of ERP-BCI interfaces for patients who prefer to rest their gaze on a fixation cross on the screen, avoiding the effort of redirecting their eye gaze for every selection. Furthermore, the performance gain over the first-stage classifier in the split VSA ($d = 3$) and between-VSA condition transfer settings are promising for patients with even less eye motor control that can experience involuntary eye movements or fixation fatigue hence cannot keep their gaze fixed throughout an entire BCI operation session. WCBLE would allow them to operate a BCI comfortably while directing their gaze to whichever portion of the screen they prefer, even when there is another target present or this location varies during the course of operation. Although WCBLE did not significantly improve overt VSA single-trial decoding, figure **4B** shows that this does not negatively impact target selection accuracy. While target selection accuracy also did not improve for split VSA, the increase in single-trial performance in split ($d = 3$) shows that an iterative alignment procedure has the potential to improve over CBLE and its first-stage classifier for this case as well.

We believe the relative increase in performance of our proposed decoder in covert and split VSA, and the lack thereof in overt VSA, could stem from the following: (1) Covert and split VSA exhibit higher P3 jitter than overt VSA. In covert and split VSA, participants have to execute a dual task by dissociating their visuospatial attention and gaze fixation. Evidence shows that ERP latency variability is higher when attention is divided (Polich, 2007; Aricò et al., 2014). Aricò et al. (2014) also partly attribute higher latency jitter to the covert VSA task performed in the BNCI2014-009 dataset since the GeoSpell interface requires both spatial and feature attention. (2) In overt VSA, the first-stage classifier can rely mostly on the modulation of early visually evoked potentials (VEPs) like N1 than on the P3 (Treder and Blankertz, 2010). These VEPs are closely related to visual processing hence exhibit lower jitter, contrary to P3 which is more prone to the effects of attention and workload (Hu et al., 2010), reducing the contribution of alignment. (3) This property can also result in the estimation of VEP latencies instead of the P3 latency and WCBLE would in this case fail to increase the P3 SNR, which still could be somewhat jittered in overt VSA. (4) Aligning to the P3 will lower the SNR of early VEPs while aligning to VEPs will lower P3 SNR, since they are not time-locked to each other. (5) Covert and split VSA ERPs may exhibit lower SNR than overt VSA due to lower P3 amplitudes or even due to the presence of higher P3 jitter itself. Higher SNR in overt VSA results in higher decoding performance of state-of-the-art classifiers, leaving less room for relative improvement in this case.

Although it is not immediately clear if WCBLE actively corrects for higher P3 jitter present in covert and split VSA compared to overt VSA, we justify our approach similar to Hardiansyah et al. (2020) by observing that the increased discrimination performance of a machine-learning model accounting for jitter forgoes the need for characterizing the underlying physiological processes while still objectively quantifying the presence of

jitter between data classes. Furthermore, some evidence points towards higher P3 jitter as the main contributing factor. While figure **3B** shows no visible smearing effect in the shape of the ERPs, a more quantitative analysis on latencies in figure **5C** indicates the opposite. To obtain comparable results across VSA conditions, WCBLE was evaluated per session and trained on all combined VSA conditions as in section 4.2. Since all conditions have the P3 in common, the estimated latencies can be interpreted as P3 latencies. Two-sided Wilcoxon rank-sum tests on jitter, expressed as the inter-quartile range (IQR) of the estimated latencies of target trials, revealed that overt VSA exhibited significantly lower jitter than all other conditions for both datasets. No other significant differences were found. p-values were corrected for the family-wise error rate using Bonferroni correction. Additionally Aricò et al. (2014) prove that P3 jitter does play a non-negligible role in covert VSA by comparing performance between overt and covert VSA, while including or excluding early VEPs. This analysis showed that the absence of the N1 and other early VEPs is not the only factor hampering covert VSA decoding performance. However, the large increase in performance for covert VSA in Aricò et al. (2014) could also be explained by overfitting on artifacts amplified by aligning, since their method is not evaluated on unseen data, as opposed to ours.

Finally, we argue that the increased performance of WCBLE is partly due to the fact that tLDA is a suitable first-stage classifier. Firstly, imposing a Toeplitz-covariance structure strongly regularizes the problem (Sosulski and Tangermann, 2022; Van Den Kerchove et al., 2022), at the benefit of decoding performance. Secondly, this method has a synergy with CBLE since both make the same assumption about the short-time stationary of the EEG background noise within an epoch. CBLE does not retrain the first-stage classifier for each time shift, but rather trains it once within the given window. After training, the classifier parameters represent some information about the expected ERP waveform and background noise. By applying the trained classifier to different time shifts, it assumes this ERP waveform can be shifted in time, but since the classifier's information about the background noise was only obtained from the initial window, CBLE assumes its properties do not vary throughout the epoch. The block-Toeplitz covariance structure of tLDA also assumes that the background noise represented by this covariance after subtracting the class averages is stationary within the epoch (Sosulski and Tangermann, 2022).

We found that CBLE did not improve gaze-independent decoding performance significantly and also did not increase performance over its first-stage classifier in overt VSA contrary to what was reported by Mowla et al. (2017). While they report that CBLE is relatively independent to the first-stage classifiers evaluated in their work, it is evident here that applying any given classifier in the CBLE scheme does not necessarily increase its performance. In our case, this could be due to characteristics of the tested dataset, e.g. the presence of jitter, or the generally higher performance of tLDA as compared to the first-stage classifiers tested by Mowla et al. (2017), leaving less performance to be gained.

Thompson et al. (2012) already attempted applying CBLE in an iterative scheme

but did not report any results due to convergence issues. We mitigated this by combining the robust latency estimation presented in section 3.1.2 with the alignment of both target and non-target epochs. Aligning only the target epochs containing the jittered P3 component is prone to overfitting by aligning non-discriminate noise that is present in both classes, such as environment noise, oscillatory background rhythms or non-modulated VEPs. If SNR is low, residual noise varying slightly between classes could dominate the expected response of the first-stage classifier and subsequently dominate WCBLE from the start, preventing convergence to a meaningful solution. Our procedure circumvents this problem by aligning both classes to the time points where the expected separation between classes is greatest. This way, noise of which the latencies are estimated in a given iteration will be perfectly time-locked in all classes in the next iteration. The first-stage classifier can then more easily suppress this noise since it is now clear it is present in both classes and non-discriminative. This aids the method in converging to a more robust classifier by iteratively increasing SNR for both classes and class separation over time.

Zisk et al. (2022) addressed P3 jitter in ALS patients by augmenting the training data once with time-shifted copies based on CBLE-estimated jitter. While we aim to train the first-stage classifier without the effects of jitter in its parameters, they do the opposite by intentionally jittering the training data. As their focus was on ALS patients and inter-session stability, they did not assess how their method interacts with visuospatial attention. We achieved a similar performance gain with our jitter compensation method, but argue that our method can cope with more granular latency differences, as Zisk et al. (2022) augment the data with just one positive and negative time shift.

While Hardiansyah et al. (2020) decoded covert VSA more effectively by contributing single-trial latency and amplitude features to decoder. Contrary to our approach, they did not then correct these amplitude features for the jitter in their latencies by e.g. aligning trials to achieve better separability. Hence, their approach would not in principle render the classifier more robust to jitter. Furthermore we incorporated estimated latency features in both CBLE and WCBLE, yet only WCBLE improved covert VSA performance. This shows that the incorporation of latency features is not the only driver of covert VSA decoding performance increase.

Despite encouraging results, our study faces some limitations that we plan to tackle in the future. Firstly, multiple ERP components can be time-locked to different neural processes, each with their own jitter, hampering the performance of single-trial latency estimation and their interpretability. Adaptations could be made to incorporate prior time windows or probability distributions on the latency of specific ERP components or to simultaneously estimate set of multiple component (clusters) latencies per ERP such as in Residue Iteration Decomposition (Ouyang et al., 2017). Future efforts should investigate how strong spatiotemporal filtering can be combined with methods that allow for a more flexible non-stationarity of the ERP like Dynamic Time Warping (DTW) or other techniques lent from time series classification, or methods that explicitly

model multiple time displacements present in one ERP. Secondly, performance might be improved by venturing beyond the classical target/non-target binary classification problem. Due to e.g. the perifoveal stimulus cruciform model (Vanegas et al., 2013), covert and split VSA responses might differ based on the relative position in the field of view of their related stimulus which could be exploited in a multi-class classification problem. Similarly, explicitly taking into account the characteristic of the distractor ERP response might have a beneficial effect. Thirdly, results were obtained in an off-line and within-session evaluation, which does not reflect true BCI operation. Using multiple sessions with on-line feedback, the user could optimize their performance over time by controlling attention or gaze. Finally, since this work was conducted with patient applications in mind, we should highlight that gaze of participants in the conducted experiments was cued and fixed, which is per definition impossible for the patients we consider. We are currently conducting a patient study to further investigate whether the studied VSA conditions are appropriate and to what extent they occur in natural patient BCI operation.

6. Conclusion

Our aim was to improve gaze-independent BCI performance for spatially organized visual event-related potential (ERP) paradigms by using a suited decoder. Earlier results on BCI performance in covert visuospatial attention (VSA) performance prediction have shown that accounting for single-trial latency jitter could improve gaze-independent decoding performance. We applied Classifier-based Latency Estimation (CBLE) as a decoder robust to latency jitter but found no increase in gaze-independent decoding performance. To remedy this, we improved CBLE by adapting it into CBLE with Woody iterations (WCBLE), an iterative scheme using probabilistic latency estimation. Results for WCBLE within and across VSA condition decoding show that gaze-independent BCI performance can be improved at the decoding stage. Overt decoding performance was not improved, but our proposed method can provide added value for patients who are unable to operate a visual BCI in overt attention mode. Later studies should confirm whether our findings hold in patient populations suffering from a variety of eye-motor impairments, and develop a solution that is capable of properly handling multiple non-time locked ERP components.

Acknowledgments

AVDK is supported by the special research fund of the KU Leuven (GPUDL/20/031) MMVH is supported by research grants received from the European Union's Horizon Europe Marie Sklodowska-Curie Action program (grant agreement No. 101118964), the European Union's Horizon 2020 research and innovation program (grant agreement No. 857375), the special research fund of the KU Leuven (C24/18/098), the Belgian Fund for Scientific Research – Flanders (G0A4118N, G0A4321N, G0C1522N), and the Hercules

Foundation (AKUL 043).

The authors wish to acknowledge Arno Libert, Tjaša Mlinarič and Yide Li for their help in data collection.

Appendix A. Stimulation and Recording

EEG for the CVSA-ERP dataset was recorded using a SynAmps RT amplifier (Compumedics Neuroscan, Australia) at 2048Hz and 62 Ag/AgCl active electrodes arranged in the international 10-10 layout fitted to a standard electrode cap (EASYCAP GmbH, Germany), with electrodes located at AFz and FCz as ground and reference respectively. Using electrolyte gel, electrode impedances were brought below 5kΩ. Electrodes TP9 and TP10, used for off-line re-referencing, were directly attached to the skin using stickers for better contact. The power line frequency in Belgium is 50 Hz. Participant's eye gaze was registered using an EyeLink 1000 Plus eye tracker (SR Research, Canada) in non-fixation mode.

Participants signed the informed consent form and were seated at a distance of 60 cm before a CRT-emulating monitor (VPixx Technologies, Canada) operating at a refresh rate of 120Hz, displaying 6 circular white targets with a diameter of 4.15° visual angle and laid out on a hexagon with a radius of 12.28° of visual angle centered on the midpoint of the screen, conforming to the interface proposed by Treder and Blankertz (2010) (figure A1**A**). A hexagonal layout interface with an empty center and a low number of targets counteracts target crowding and, as long as the subject's gaze is within the hexagon of targets, no other target can be between the subject's gaze and a covertly attended target. Targets are full-contrast white and were intensified by scaling them to a larger size (5.60° of visual angle, figure A1**B**) instead of changing the contrast to avoid Troxler-fading§ (Treder and Blankertz, 2010) in the peripheral visual field. Stimuli were presented using PsychopPy (version 2023.1.3) (Peirce et al., 2019).

The participant was instructed to press the space bar when ready for a block of stimulations. Then, one target was indicated as the cue and the participant was instructed to count the number of intensifications of the cued target during the following block of stimulations. After pressing the space bar again, a blue crosshair appeared, and the subject was instructed to fixate their gaze on the blue crosshair for the duration of the stimulation block (figure A1**C** and figure A1**D**). The position of this crosshair determined the VSA condition for this trial: overt VSA when the crosshair was at the same location as the cued target, covert VSA when the crosshair appeared in the center of the screen, and split VSA when the crosshair appeared on a different target than the cued one. After pressing the space bar again and a delay of 5 seconds, the stimulation block starts. All targets were intensified for a duration of 100 ms, in pseudorandom order. The inter-stimulus-interval (ISI), the time between the onsets of subsequent intensifications, was variable and consisted of a fixed 300ms interval (of which 100ms

§ The optical illusion of disappearing unchanging stimuli experienced when visually fixating (Troxler, 1804).

with an intensified target onscreen) with 200ms uniform jitter added, resulting in an ISI between 200 and 400 ms. Inter-stimulus intervals were jittered to counteract steady-state effects and residue in averaging. A longer inter-stimulus interval will increase component amplitude and aid in counteracting temporal autocorrelation for a higher statistical test precision. In a block of stimulations, each target was intensified a pseudorandom number of times between 10 and 15. This led to stimulation blocks with an average duration of 26.25 seconds. After a block of stimulations, an input prompt appeared to enter the mentally counted number of intensifications. After inputting this number, the subject was allowed to pause until pressing the space bar again. In total, six blocks were presented for overt VSA, six blocks for covert VSA, 12 blocks for split ($d = 1$) VSA, 12 blocks for split ($d = 2$) VSA and 6 blocks for split ($d = 3$) VSA, covering all possible combinations of VSA conditions, cued targets and crosshair locations. The experiment started with five non-recorded practice stimulation blocks, one for each of the 5 VSA conditions. During these practice blocks, the participant received feedback about their gaze position and counting accuracy. Counting the instructions and the participant's response to the input prompts, a block lasted about 30 seconds. In sum, the experiment featured approximately 45 minutes of stimulation time. After blocks 14 and 28, the participant was allowed to take a longer break. Including these longer breaks, the experiment lasted approximately one hour.

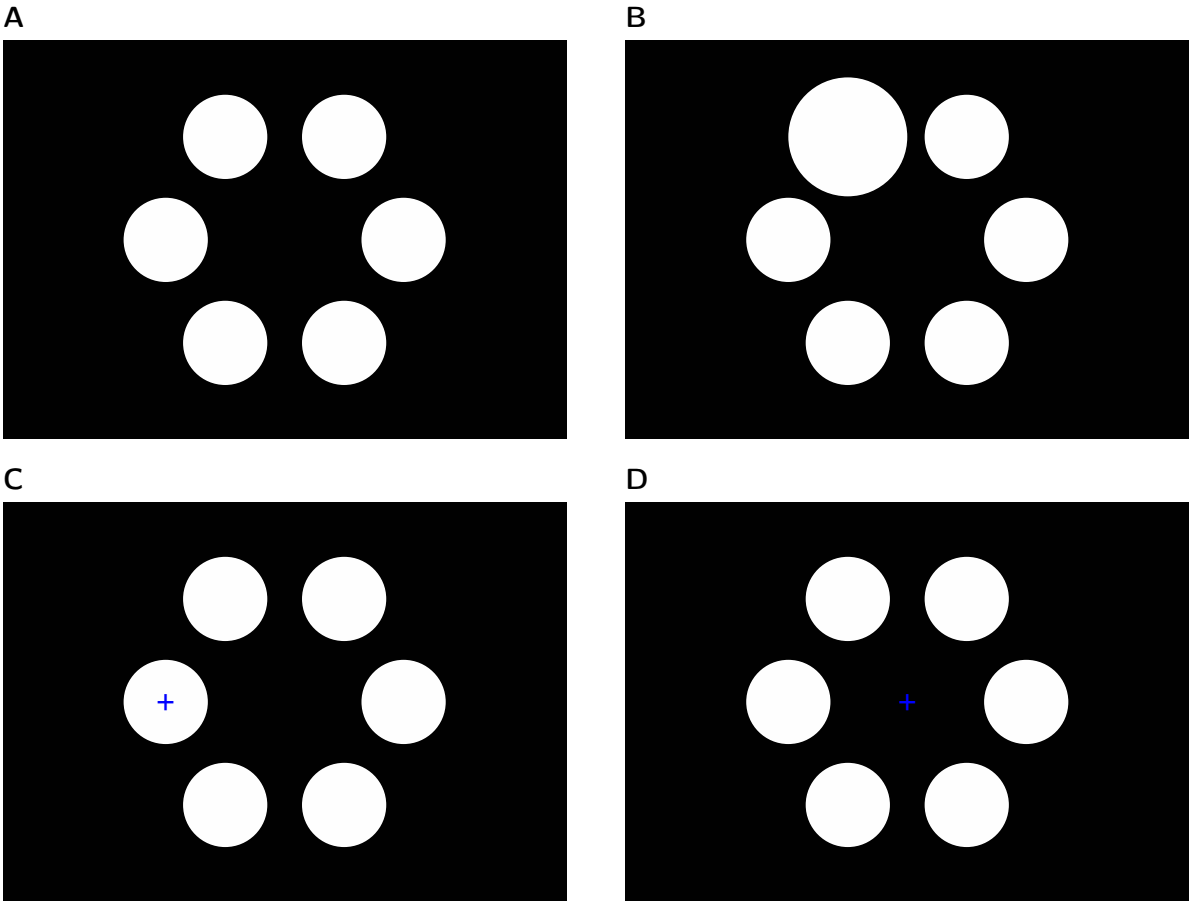

Figure A1: **A** Visual features of the stimulation interface used in the CVSA-ERP dataset. **B** Targets were intensified by shortly expanding in size. **C** The cue for gaze fixation in the overt and split conditions. **D** The cue for gaze fixation in covert conditions.

Appendix B. Single-trial classification performance

dataset	BNCI2014-009		CVSA-ERP				
VSA condition	overt	covert	overt	covert	split ($d = 1$)	split ($d = 2$)	split ($d = 3$)
ERPCov-TS-LR	0.9000	0.7450	0.7956	0.6880	0.6690	0.7252	0.7214
XDAWNCov-TS-LR	0.9044	0.7464	0.8113	0.6780	0.6683	0.7259	0.7136
tLDA	0.9474	0.7890	0.**8660**	0.7111	**0.7097**	0.7550	0.7474
CBLE	**0.9497**	0.7928	0.8559	0.6977	0.7087	0.7467	0.7465
WCBLE	0.9428	**0.8084**	0.8525	**0.7338**	0.7084	**0.7581**	**0.7687**

Table B1: Cross-validated single-trial classification area under the receiver operating characteristic curve for all evaluated models, visuospatial attention conditions and datasets

Appendix C. Algorithms

Algorithm 1 Classifier-Based Latency Estimation

TRAIN

Input: $\{\mathbf{X}_n^{\text{train}}\}_{n=1}^{N}, \mathbf{l}^{\text{train}}, \mathcal{C}(\cdot, f, \Pr), s_1, s_2$

$\theta \leftarrow \text{train}_{\mathcal{C}}(\{\mathbf{X}_n^{\text{train}}[:, s_1 : s_2]\}_{n=1}^{N}, \mathbf{l})$ ▷ Train stage 1

for $n = 1 \ldots N$ **do** ▷ Feature extraction for stage 2

  for $s = 1 \ldots R$ **do**

    $y_{n,s}^{\text{train}} \leftarrow f(\mathbf{X}_n^{\text{train}}[:, s : s + (s_2 - s_1)], \theta)$

  end for

  $s_{n,\text{target}}^{\text{train}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_n^{\text{train}}, \theta, \text{target})\right]$

  $s_{n,\text{non-target}}^{\text{train}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_n^{\text{train}}, \theta, \text{non-target})\right]$

end for

Output: $\theta, \mathbf{Y}^{\text{train}}, \mathbf{s}_{\text{target}}^{\text{train}}, \mathbf{s}_{\text{non-target}}^{\text{train}}$

EVALUATE

Input: $\{\mathbf{X}_m^{\text{test}}\}_{m=1}^{N}, \mathcal{C}(\theta, f, \Pr), s_1, s_2$

for $m = 1 \ldots M$ **do** ▷ Feature extraction for stage 2

  for $s = 1 \ldots R$ **do**

    $y_{m,s}^{\text{test}} \leftarrow f(\mathbf{X}_n^{\text{test}}[:, s : s + (s_2 - s_1)], \theta)$

  end for

  $s_{m,\text{target}}^{\text{test}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_m^{\text{test}}, \theta, \text{target})\right]$

  $s_{m,\text{non-target}}^{\text{test}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_m^{\text{test}}, \theta, \text{non-target})\right]$

end for

Output: $\mathbf{Y}^{\text{test}}, \mathbf{s}_{\text{target}}^{\text{test}}, \mathbf{s}_{\text{non-target}}^{\text{test}}$

Algorithm 2 Classifier-Based Latency Estimation with Woody Iterations

TRAIN

Input: $\{\mathbf{X}_n^{\text{train}}\}_{n=1}^{N}, \mathbf{l}, \mathcal{C}(\cdot, f, \Pr), s_1, s_2$

$\mathbf{X}'_n \leftarrow \mathbf{X}_n^{\text{train}} \quad \forall \quad n = 1 \dots N$ ▷ Train stage 1
repeat
  $\theta^* \leftarrow \text{train}_{\mathcal{C}}(\{\mathbf{X}'_n[:, s_1 : s_2]\}_0^{N-1}, \mathbf{l}$
  for $n = 1 \dots N$ **do**
    $s_n \leftarrow \text{median}\,[Pr(s|\mathbf{X}'_n, \theta^*, l_n)]$
    $\mathbf{X}'_n \leftarrow \text{align}(\mathbf{X}_n^{\text{train}}, s_n^*)$
  end for
until convergence or maximum iterations reached
for $n = 1 \dots N$ **do** ▷ Feature extraction for stage 2
  for $s = 1 \dots R$ **do**
    $y_{n,s}^{\text{train}} \leftarrow f(\mathbf{X}_n^{\text{train}}[:, s : s + (s_2 - s_1)], \theta^*)$
  end for
  $s_{n,\text{target}}^{\text{train}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_n^{\text{train}}, \theta^*, \text{target})\right]$
  $s_{n,\text{non-target}}^{\text{train}} \leftarrow \text{median}\left[Pr(s|\mathbf{X}_n^{\text{train}}, \theta^*, \text{non-target})\right]$
end for

Output: $\theta^*, \mathbf{Y}^{\text{train}}, \mathbf{s}_{\text{target}}^{\text{train}}, \mathbf{s}_{\text{non-target}}^{\text{train}}$

EVALUATE

Input: $\{\mathbf{X}_m^{\text{test}}\}_{m=1}^{N}, \mathcal{C}(\theta^*, f, \Pr), s_1, s_2$

for $m = 1 \dots M$ **do** ▷ Feature extraction for stage 2
  for $s = 1 \dots R$ **do**
    $y_{m,s}^{\text{test}} \leftarrow f(\mathbf{X}_n^{\text{test}}[:, s : s + (s_2 - s_1)], \theta^*)$
  end for
  $s_{m,\text{target}}^{\text{test}} \leftarrow \text{median}\,[Pr(s|\mathbf{X}_m^{\text{test}}, \theta^*, \text{target})]$
  $s_{m,\text{non-target}}^{\text{test}} \leftarrow \text{median}\,[Pr(s|\mathbf{X}_m^{\text{test}}, \theta^*, \text{non-target})]$
end for

Output: $\mathbf{Y}^{\text{test}}, \mathbf{s}_{\text{target}}^{\text{test}}, \mathbf{s}_{\text{non-target}}^{\text{test}}$

References

Abanda, A., Mori, U. and Lozano, J. A. (2019). A review on distance based time series classification, *Data Mining and Knowledge Discovery* **33**(2): 378–412.
URL: *https://doi.org/10.1007/s10618-018-0596-4*

Aloise, F., Aricò, P., Schettini, F., Riccio, A., Salinari, S., Mattia, D., Babiloni, F. and Cincotti, F. (2012a). A covert attention P300-based brain–computer interface: Geospell, *Ergonomics* **55**(5): 538–551.
URL: *https://doi.org/10.1080/00140139.2012.661084*

Aloise, F., Schettini, F., Aricò, P., Salinari, S., Babiloni, F. and Cincotti, F. (2012b). A comparison of classification techniques for a gaze-independent P300-based brain–computer interface, *Journal of Neural Engineering* **9**(4): 045012.
URL: *https://dx.doi.org/10.1088/1741-2560/9/4/045012*

Aricò, P., Aloise, F., Schettini, F., Salinari, S., Mattia, D. and Cincotti, F. (2014). Influence of P300 latency jitter on event related potential-based brain–computer interface performance, *Journal of Neural Engineering* **11**(3): 035008.
URL: *https://dx.doi.org/10.1088/1741-2560/11/3/035008*

Aydarkhanov, R., Uščumlić, M., Chavarriaga, R., Gheorghe, L. and del R Millán, J. (2020). Spatial covariance improves BCI performance for late ERPs components with high temporal variability, *Journal of Neural Engineering* **17**(3): 036030.

Brunner, P., Joshi, S., Briskin, S., Wolpaw, J. R., Bischof, H. and Schalk, G. (2010). Does the 'P300' speller depend on eye gaze?, *Journal of Neural Engineering* **7**(5): 056013.
URL: *https://dx.doi.org/10.1088/1741-2560/7/5/056013*

Cecotti, H., Barachant, A., King, J.-R., Bornot, J. S. and Prasad, G. (2017). Single-trial detection of event-related fields in MEG from the presentation of happy faces: Results of the biomag 2016 data challenge, *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, IEEE, pp. 4467–4470.

Chaudhary, U., Birbaumer, N. and Ramos-Murguialday, A. (2016). Brain–computer interfaces for communication and rehabilitation, *Nature Reviews Neurology* **12**(9): 513–525.
URL: *https://www.nature.com/articles/nrneurol.2016.113*

de Neeling, M. and Van Hulle, M. M. (2019). Single-paradigm and hybrid brain computing interfaces and their use by disabled patients, *Journal of Neural Engineering* **16**(6): 061001.
URL: *https://doi.org/10.1088/1741-2552/ab2706*

Dimitriadis, S. I., Brindley, L., Evans, L. H., Linden, D. E. and Singh, K. D. (2018). A Novel, Fast, Reliable, and Data-Driven Method for Simultaneous Single-Trial Mining and Amplitude—Latency Estimation Based on Proximity Graphs and Network Analysis, *Frontiers in Neuroinformatics* **12**.
URL: *https://www.frontiersin.org/articles/10.3389/fninf.2018.00059*

D'Avanzo, C., Schiff, S., Amodio, P. and Sparacino, G. (2011). A Bayesian method to estimate single-trial event-related potentials with application to the study of the P300 variability, *Journal of Neuroscience Methods* **198**(1): 114–124.
URL: *https://www.sciencedirect.com/science/article/pii/S0165027011001579*

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM* **24**(6): 381–395.
URL: *https://dl.acm.org/doi/10.1145/358669.358692*

Frenzel, S., Neubert, E. and Bandt, C. (2011). Two communication lines in a 3 × 3 matrix speller, *Journal of Neural Engineering* **8**(3): 036021.
URL: *https://doi.org/10.1088/1741-2560/8/3/036021*

Gramfort, A. (2013). MEG and EEG data analysis with MNE-Python, *Frontiers in Neuroscience* **7**.
URL: *http://journal.frontiersin.org/article/10.3389/fnins.2013.00267/abstract*

Hardiansyah, I., Pergher, V. and Van Hulle, M. M. (2020). Single-trial EEG responses classified using latency features, *International Journal of Neural Systems* **30**(06): 2050033.

Hu, L., Mouraux, A., Hu, Y. and Iannetti, G. D. (2010). A novel approach for enhancing the signal-to-noise ratio and detecting automatically event-related potentials (ERPs) in single trials, *NeuroImage* **50**(1): 99–111.
URL: *https://www.sciencedirect.com/science/article/pii/S105381190901297X*

Hwang, H.-J., Ferreria, V. Y., Ulrich, D., Kilic, T., Chatziliadis, X., Blankertz, B. and Treder, M. (2015). A Gaze Independent Brain-Computer Interface Based on Visual Stimulation through Closed Eyelids, *Scientific Reports* **5**(1): 15890.
URL: *https://www.nature.com/articles/srep15890*

Iturrate, I., Chavarriaga, R., Montesano, L., Minguez, J. and Millán, J. (2014). Latency correction of event-related potentials between different experimental protocols, *Journal of Neural Engineering* **11**(3): 036005.
URL: *https://doi.org/10.1088/1741-2560/11/3/036005*

Kim, M., Kim, J. and Kim, S.-P. (2020). Enhancing ERP component detection by estimating ERP latency variability using hidden process model, *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1262–1268. ISSN: 2577-1655.

Krell, M. M., Seeland, A. and Kim, S. K. (2018). Data Augmentation for Brain-Computer Interfaces: Analysis on Event-Related Potentials Data, *Technical report.* arXiv:1801.02730 [cs, q-bio] type: article.
URL: *http://arxiv.org/abs/1801.02730*

Lees, S., Dayan, N., Cecotti, H., McCullagh, P., Maguire, L., Lotte, F. and Coyle, D. (2018). A review of rapid serial visual presentation-based brain–computer interfaces, *Journal of Neural Engineering* **15**(2): 021001.
URL: *https://dx.doi.org/10.1088/1741-2552/aa9817*

Lotte, F., Bougrain, L., Cichocki, A., Clerc, M., Congedo, M., Rakotomamonjy, A. and Yger, F. (2018). A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update, *Journal of neural engineering* **15**(3): 031005.

Luck, S. J. (2014). *An introduction to the event-related potential technique*, MIT press.

Mowla, M. R., Gonzalez-Morales, J. D., Rico-Martinez, J., Ulichnie, D. A. and Thompson, D. E. (2020). A Comparison of Classification Techniques to Predict Brain-Computer Interfaces Accuracy Using Classifier-Based Latency Estimation, *Brain Sciences* **10**(10): 734.
URL: *https://www.mdpi.com/2076-3425/10/10/734*

Mowla, M. R., Huggins, J. E. and Thompson, D. E. (2017). Enhancing P300-BCI performance using latency estimation, *Brain-Computer Interfaces* **4**(3): 137–145.
URL: *https://doi.org/10.1080/2326263X.2017.1338010*

Naci, L., Monti, M. M., Cruse, D., Kübler, A., Sorger, B., Goebel, R., Kotchoubey, B. and Owen, A. M. (2012). Brain–computer interfaces for communication with nonresponsive patients, *Annals of Neurology* **72**(3): 312–323.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/ana.23656*

Ouyang, G., Hildebrandt, A., Sommer, W. and Zhou, C. (2017). Exploiting the intra-subject latency variability from single-trial event-related potentials in the P3 time range: A review and comparative evaluation of methods, *Neuroscience & Biobehavioral Reviews* **75**: 1–21.
URL: *https://www.sciencedirect.com/science/article/pii/S0149763416305590*

Peirce, J., Gray, J. R., Simpson, S., MacAskill, M., Höchenberger, R., Sogo, H., Kastman, E. and Lindeløv, J. K. (2019). PsychoPy2: Experiments in behavior made easy, *Behavior Research Methods* **51**(1): 195–203.
URL: *https://doi.org/10.3758/s13428-018-01193-y*

Pelo, P. D., Tommaso, M. D., Monaco, A., Stramaglia, S., Bellotti, R. and Tangaro, S. (2018). Trial latencies estimation of event-related potentials in EEG by means of genetic algorithms, *Journal of Neural Engineering* **15**(2): 026016.

URL: *https://doi.org/10.1088/1741-2552/aa9b97*

Pires, G., Nunes, U. and Castelo-Branco, M. (2011). GIBS block speller: Toward a gaze-independent P300-based BCI, *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 6360–6364. ISSN: 1558-4615.

Polich, J. (2007). Updating P300: An integrative theory of P3a and P3b, *Clinical Neurophysiology* **118**(10): 2128–2148.
URL: *https://www.sciencedirect.com/science/article/pii/S1388245707001897*

Reichert, C., Dürschmid, S., Bartsch, M. V., Hopf, J.-M., Heinze, H.-J. and Hinrichs, H. (2020b). Decoding the covert shift of spatial attention from electroencephalographic signals permits reliable control of a brain-computer interface, *Journal of Neural Engineering* **17**(5): 056012.
URL: *https://doi.org/10.1088/1741-2552/abb692*

Riccio, A., Mattia, D., Simione, L., Olivetti, M. and Cincotti, F. (2012). Eye-gaze independent EEG-based brain–computer interfaces for communication, *Journal of Neural Engineering* **9**(4): 045001.
URL: *https://doi.org/10.1088/1741-2560/9/4/045001*

Ron-Angevin, R., Garcia, L., Fernández-Rodríguez, Á., Saracco, J., André, J.-M. and Lespinet-Najib, V. (2019). Impact of speller size on a visual P300 brain-computer interface (BCI) system under two conditions of constraint for eye movement, *Computational Intelligence and Neuroscience* **2019**.

Schaeff, S., Treder, M. S., Venthur, B. and Blankertz, B. (2012). Exploring motion VEPs for gaze-independent communication, *Journal of Neural Engineering* **9**(4): 045006.
URL: *https://dx.doi.org/10.1088/1741-2560/9/4/045006*

Sosulski, J. and Tangermann, M. (2022). Introducing Block-Toeplitz Covariance Matrices to Remaster Linear Discriminant Analysis for Event-related Potential Brain-computer Interfaces, *arXiv:2202.02001 [cs, q-bio]* . arXiv: 2202.02001.
URL: *http://arxiv.org/abs/2202.02001*

Souloumiac, A. and Rivet, B. (2013). Improved estimation of EEG evoked potentials by jitter compensation and enhancing spatial filters, *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, IEEE, pp. 1222–1226.

Thompson, D. E., Warschausky, S. and Huggins, J. E. (2012). Classifier-based latency estimation: a novel way to estimate and predict BCI accuracy, *Journal of neural engineering* **10**(1): 016006.

Treder, M. S. and Blankertz, B. (2010). (C)overt attention and visual speller design in an ERP-based brain-computer interface, *Behavioral and brain functions* **6**: 1–13.

Treder, M. S., Porbadnigk, A. K., Shahbazi Avarvand, F., Müller, K.-R. and Blankertz, B. (2016). The LDA beamformer: Optimal estimation of ERP source time series using linear discriminant analysis, *NeuroImage* **129**: 279–291.
URL: *https://www.sciencedirect.com/science/article/pii/S1053811916000252*

Treder, M. S., Schmidt, N. M. and Blankertz, B. (2011). Gaze-independent brain–computer interfaces based on covert attention and feature attention, *Journal of Neural Engineering* **8**(6): 066003.
URL: *https://doi.org/10.1088/1741-2560/8/6/066003*

Troxler, D. (1804). Ueber das Verschwinden gegebener Gegenstande innerhalb unseres Gesichtskreises, *Ophthalmologische Bibliothek* **2**: 1–119.
URL: *https://ci.nii.ac.jp/naid/10020081652/*

Van Den Kerchove, A., Libert, A., Wittevrongel, B. and Van Hulle, M. M. (2022). Classification of Event-Related Potentials with Regularized Spatiotemporal LCMV Beamforming, *Applied Sciences* **12**(6): 2918.
URL: *https://www.mdpi.com/2076-3417/12/6/2918*

Vanegas, M. I., Blangero, A. and Kelly, S. P. (2013). Exploiting individual primary visual cortex geometry to boost steady state visual evoked potentials, *Journal of neural engineering* **10**(3): 036003.

Woody, C. D. (1967). Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals, *Medical and biological engineering* **5**(6): 539–554.

Xu, W., Gao, P., He, F. and Qi, H. (2022). Improving the performance of a gaze independent P300-BCI by using the expectancy wave, *Journal of Neural Engineering* **19**(2): 026036.
URL: *https://dx.doi.org/10.1088/1741-2552/ac60c8*

Zhang, D., Maye, A., Gao, X., Hong, B., Engel, A. K. and Gao, S. (2010). An independent brain–computer interface using covert non-spatial visual selective attention, *Journal of Neural Engineering* **7**(1): 016010.
URL: *https://doi.org/10.1088/1741-2560/7/1/016010*

Zisk, A. H., Borgheai, S. B., McLinden, J., Deligani, R. J. and Shahriari, Y. (2022). Improving longitudinal P300-BCI performance for people with ALS using a data augmentation and jitter correction approach, *Brain-Computer Interfaces* **9**(1): 49–66.
URL: *https://doi.org/10.1080/2326263X.2021.2014678*