brain sciences

MDPI

Article

Exploiting the Cone of Influence for Improving the Performance of Wavelet Transform-Based Models for ERP/EEG Classification

Xiaoqian Chen [1], Resh S. Gupta [2] and Lalit Gupta [1,*]

[1] School of Electrical, Computer, and Biomedical Engineering, Southern Illinois University, Carbondale, IL 62901, USA
[2] Center of Excellence for Stress and Mental Health, VA San Diego Healthcare System, San Diego, CA 92161, USA
* Correspondence: lgupta@siu.edu

Abstract: Features extracted from the wavelet transform coefficient matrix are widely used in the design of machine learning models to classify event-related potential (ERP) and electroencephalography (EEG) signals in a wide range of brain activity research and clinical studies. This novel study is aimed at dramatically improving the performance of such wavelet-based classifiers by exploiting information offered by the cone of influence (COI) of the continuous wavelet transform (CWT). The COI is a boundary that is superimposed on the wavelet scalogram to delineate the coefficients that are accurate from those that are inaccurate due to edge effects. The features derived from the inaccurate coefficients are, therefore, unreliable. In this study, it is hypothesized that the classifier performance would improve if unreliable features, which are outside the COI, are zeroed out, and the performance would improve even further if those features are cropped out completely. The entire, zeroed out, and cropped scalograms are referred to as the "same" (S)-scalogram, "zeroed out" (Z)-scalogram, and the "valid" (V)-scalogram, respectively. The strategy to validate the hypotheses is to formulate three classification approaches in which the feature vectors are extracted from the (a) S-scalogram in the standard manner, (b) Z-scalogram, and (c) V-scalogram. A subsampling strategy is developed to generate small-sample ERP ensembles to enable customized classifier design for single subjects, and a strategy is developed to select a subset of channels from multiple ERP channels. The three scalogram approaches are implemented using support vector machines, random forests, k-nearest neighbor, multilayer perceptron neural networks, and deep learning convolution neural networks. In order to validate the performance hypotheses, experiments are designed to classify the multi-channel ERPs of five subjects engaged in distinguishing between synonymous and non-synonymous word pairs. The results confirm that the classifiers using the Z-scalogram features outperform those using the S-scalogram features, and the classifiers using the V-scalogram features outperform those using the Z-scalogram features. Most importantly, the relative improvement of the V-scalogram classifiers over the standard S-scalogram classifiers is dramatic. Additionally, enabling the design of customized classifiers for individual subjects is an important contribution to ERP/EEG-based studies and diagnoses of patient-specific disorders.

Keywords: event-related potentials; electroencephalography; continuous wavelet transform; scalogram; cone of influence; subsample averaging; customized classifier design; channel ranking

check for updates

Citation: Chen, X.; Gupta, R.S.; Gupta, L. Exploiting the Cone of Influence for Improving the Performance of Wavelet Transform-Based Models for ERP/EEG Classification. *Brain Sci.* **2023**, *13*, 21. https://doi.org/10.3390/brainsci13010021

Academic Editor: Giovanni Pellegrino

Received: 22 November 2022
Revised: 10 December 2022
Accepted: 19 December 2022
Published: 22 December 2022

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

1. Introduction

The continuous wavelet transform (CWT) and the discrete wavelet transform (DWT) are indispensable signal processing tools that are extensively used to analyze and develop classifiers for event-related potential (ERP) and electroencephalography (EEG) signals. Like most practical signals, ERPs and EEG epochs have finite lengths. The CWT and DWT of finite-length signals suffer from border distortions which are called cone of influence (COI) effects and border effects, respectively. This study focuses primarily on

emphasizing the importance of taking the COI into account in the design of ERP/EEG machine learning classification models which use features derived from the CWT. The COI is a boundary that is superimposed on the CWT scalogram to delineate the accurately computed wavelet coefficients from those that are inaccurate (artifact coefficients) due to the wavelet extending beyond the observation interval of the signal [1–4]. These edge-effect artifacts are due to the mechanics of convolving the signal and the wavelet near the beginning and near the end of the signal. For a given wavelet, the number of artifact coefficients computed depends on the wavelet scale. Larger scales result in a larger number of artifact coefficients; consequently, the boundary takes on a cone shape in the scalogram. Although it is common practice to exclude unreliable artifact coefficients outside the COI in scalogram analyses, surprisingly, the same cannot be said about classification. This is especially true when wavelet scalograms are used in conjunction with deep learning convolution networks (CNNs). Many recent studies have shown that the conversion of one-dimensional time-domain signals such as EEGs and ERPs into a time-scale matrix representation through the CWT not only offers simultaneous time-scale information but also offers an ideal matrix input format for CNNs [5–11]. However, due to the mechanics of matrix convolutions in the CNNs, the convolutions are carried out over the entire scalogram in the first stage as well as the subsequent convolution stages. Features extracted from the scalogram are also combined into feature vectors for input to many other classifiers such as support vector machines (SVMs) [7,12–15], random forests (RFs) [16,17], k-nearest neighbor (k-NN) [15,18,19], and multilayer perceptron (MLP) neural networks [20].

Considering the widespread use of the CWT for classifying ERPs and EEGs as well as numerous other types of non-stationary signals and the fact that COI edge-effect artifacts are well known, it is remarkable how little attention has been paid to addressing the following question: how can knowledge of the COI contribute to the design of wavelet-based classifiers? Our specific goal, therefore, is to demonstrate the importance of exploiting the information provided by the COI by developing three classification approaches that employ features from the (a) entire scalogram, (b) entire scalogram but with zeroed-out coefficients outside the COI, and (c) cropped scalogram consisting of only coefficients inside the COI. It will be shown that the entire scalogram is obtained via "same" convolution and the cropped scalogram is obtained via "valid" convolution; therefore, the entire and cropped scalograms will be referred to as the S-scalogram and V-scalogram, respectively. The scalogram with the zeroed-out coefficients will be referred to as the Z-scalogram. The S-scalogram approach is the "standard" method employed in most wavelet classifiers and can, therefore, serve as a baseline for performance comparisons with the two other approaches. We hypothesize that the Z-scalogram-based classifiers will outperform the S-scalogram-based classifiers, and the V-scalogram-based classifiers will outperform the Z-scalogram-based classifiers. Most importantly, we hypothesize that the improvement of V-scalogram classifiers over the standard baseline S-scalogram classifiers will be dramatic.

The formulations of the scalogram-based classifiers are quite general and are not tied to any particular application. This study focuses on ERP classification; however, the methods developed are equally applicable to classifying EEGs and other non-stationary signals. ERPs are voltage fluctuations in the EEG that occur as a result of an external or internal event and are frequently used to assess sensory, cognitive, motor, and emotion-related processes [21]. EEG is a non-invasive method used to assess neurophysiological function. EEG measures the electrical activity of large, synchronously firing populations of neurons in the brain with electrodes placed on the scalp [20]. EEG signals are often contaminated by various sources of noise, which can be physiological (e.g., eye blinks, eye movements, muscle artifacts) and non-physiological (e.g., electrode movement relative to the scalp, electrode wire movement, electromagnetic noise) in nature. Thus, preprocessing steps, including filtering, artifact correction, and artifact rejection procedures, are typically applied to the raw data [22]. Several other useful and novel methods have been proposed to reduce these sources of noise [23,24]. Altogether, EEG signals have been used in a wide variety of studies ranging from emotion classification [25] to qualifying seizure dynamics [26].

In order to validate the hypotheses and to also show that the performance trends are consistent across classifiers, the three approaches are implemented using a diverse set of classifiers which are trained and tested on the 64-channel ERPs of 5 subjects engaged in a binary semantic task which involved distinguishing between synonymous and non-synonymous word pairs. The problems with developing customized ERP classifiers for individual subjects are identified, and a subsampling strategy is developed to generate small-sample average ensembles to facilitate the design. A rank-of-rank sum channel selection strategy which uses an interclass separation measure is developed to select a subset of channels that carry the most useful discriminatory information for classifying the synonymous and non-synonymous ERPs. Classification experiments are designed using the (5 subjects) (8 channels) = 40 data sets, and it is shown that:

(a) the Z-scalogram classifiers outperform the standard S-scalogram classifiers;
(b) the V-scalogram classifiers significantly outperform the S-scalogram and Z-scalogram classifiers;
(c) the relative improvement of the V-scalogram classifiers over the standard S-scalogram classifiers is dramatic;
(d) the improvement trends across the three scalogram approaches are remarkably consistent across the classifiers, ERP channels, and subjects;
(e) the region outside the COI does not carry useful discriminatory information;
(f) the subsampling strategy to generate small-sample ERP ensembles enables customized classifier design for single subjects.

To the best of our knowledge, we are not aware of any other study that systematically demonstrates how the performance of standard scalogram-based classifiers can be improved through the V-scalogram approach, which excludes the artifact coefficients outside the COI. Furthermore, the subsampling strategy, which enables the design of customized classifiers for individual subjects, is unique.

2. Materials and Methods

This section presents (a) the definitions of the COI, S-scalogram, Z-scalogram, V-scalogram, and V-complement scalogram, (b) methods to reduce effects of edge artifacts, (c) measures to indicate the quality of scalograms, (d) the development of the S-scalogram, Z-scalogram, and V-scalogram approaches, (e) the subsampling strategy to generate small-sample ERPs to enable the development of ERP classifiers for single subjects, (f) the development of the algorithm to rank and select channels for classification problems, (g) the diverse set of classifiers selected to implement the 3 scalogram approaches for ERP classification, (h) the 64-channel dichotomous EEG data of 5 subjects that is used to implement the various classifiers for the three scalogram approaches, (i) the experiments designed to evaluate the performances of the various implementations of the three classification approaches, and (j) details of the wavelet choice, averaging parameter selection, feature vector generation, classifier architectures and hyperparameters, and cross-validation.

2.1. COI

The primary focus of this study is on the development of classifiers that modify the scalogram for feature extraction based on the COI. It is, therefore, important to understand how the COI is generated in the scalogram, which, in turn, requires an understanding of how the scalogram is generated from the CWT. Given a wavelet basis function $\Psi(t)$, the CWT of a signal $x(t)$ is given by

$$W(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \Psi^* \left(\frac{t-b}{a} \right) dt \quad (1)$$

where a and b are the scale and translation parameters, respectively. That is, the CWT is the convolution of $x(t)$ and scaled versions of $\Psi(t)$. Applying the CWT to $x(t)$ results in a set of wavelet coefficients which are a function of scale and position. A plot of

the coefficients provides a time-scale representation of $x(t)$. In order to implement on a computer, the CWT is discretized; consequently, $W(a,b)$ is discrete both in scale and in time. The discretized CWT is often referred to as the DCWT to distinguish it from the discrete wavelet transform (DWT). Using s and n to denote the discrete-scale and discrete-translation variables, respectively, $W(s,n)$ plotted as a function of s and n is called the CWT coefficient matrix. The scalogram is the amplitude $|W(s,n)|$ or power $|W(s,n)|^2$ which can be represented by a matrix $G(s,n)$, $s = 0,1,\ldots(J-1); n = 0,1,\ldots,(N-1)$, where J is the number of scale bands, and N is the signal duration. In this zero-indexed matrix representation, row 0 corresponds to scale S_0, row 1 corresponds to scale $S_1,\ldots$, row $(J-1)$ corresponds to scale S_{J-1}, with $S_j < S_{j+1}$.

The COI is best explained by first understanding the convolution operations that are used to generate the coefficient matrix and to identify the COI. Each row of the coefficient matrix $W(s,n)$ is obtained from the linear convolution of $x(n)$ with a wavelet of a particular scale. There are several different forms of linear convolution, which include "full", "same" and "valid", which are useful for identifying the COI in the coefficient matrix. The three types of convolutions, which are described in detail in Appendix A, differ in the manner in which the start and end points are computed. The standard scalogram, which we refer to as the S-scalogram, is generated by same-convolution. The S-scalogram contains artifact coefficients because it suffers from edge effects. The valid convolution operation yields coefficients that are accurate and is used to identify the COI in the coefficient matrix. The boundary that separates the artifact coefficients from the accurate coefficients defines the COI in the S-scalogram. We refer to the part of the S-scalogram that is inside the COI as the V-scalogram because it corresponds to the coefficients associated with valid convolutions. Furthermore, the entire S-scalogram with zeroed-out coefficients outside the COI is referred to as the Z-scalogram. Figure 1 shows an ERP and the 3 different types of scalograms generated from the Morlet wavelet transform of the ERP.

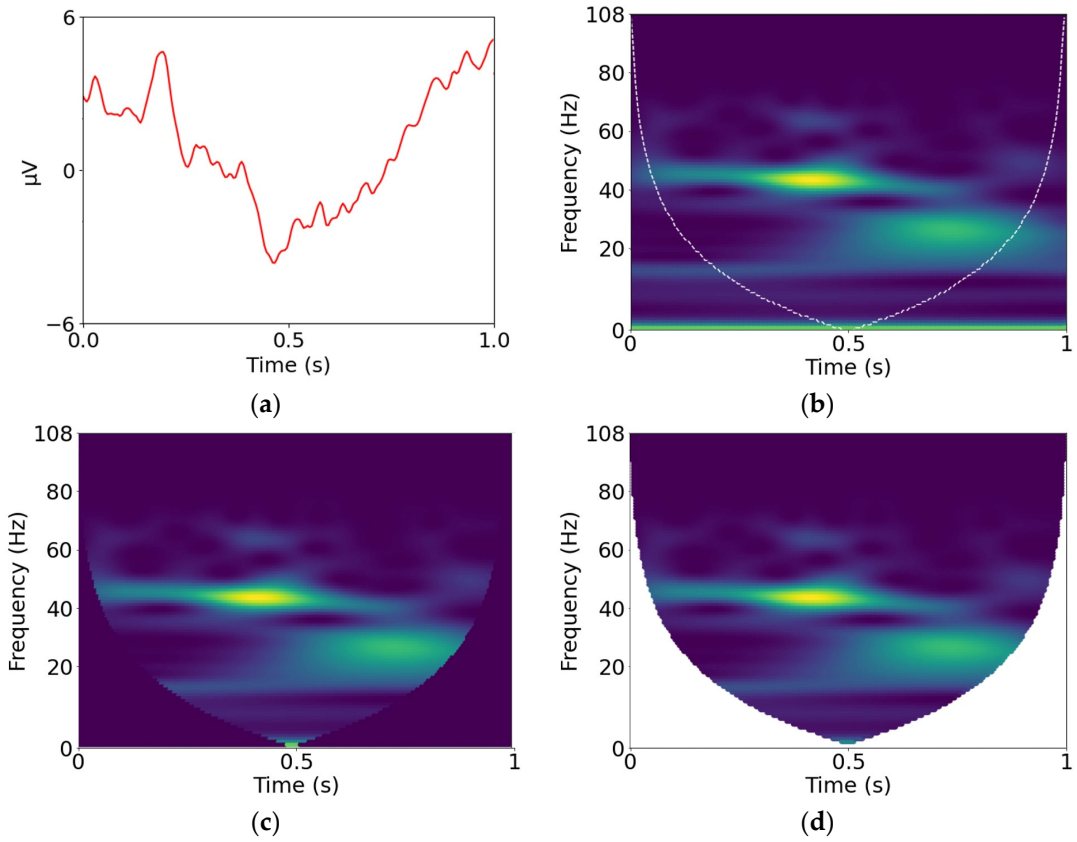


Figure 1. (**a**) ERP, (**b**) S-scalogram with superimposed COI, (**c**) Z-scalogram, (**d**) V-scalogram.

2.1.1. Edge Artifacts

It is inevitable that the CWT applied to a finite duration signal will lead to edge artifacts in the standard S-scalogram due to the need for extending the signal through padding. The inaccuracies of the artifact coefficients are the highest at the edges and decrease towards the COI. Various notable methods have been made to reduce, but not remove, the effects of the artifacts by smoothing the extensions [27–29], forecasting the extensions [30] and modifying the wavelet transform instead of extending the signal [31,32]. The effects of several extension methods have also been compared in [33,34]. For least-squares wavelet analysis (LSWA), which is used to analyze non-stationary signals that are not sampled evenly, the stochastic confidence level surface acts like the COI but is based on degrees of freedom [35].

Methods have also been proposed to approximate the COI, which include (a) using the time constant $1/e$ to delineate the borders of the cone of influence at each scale [2,36,37], (b) defining the extent of the COI at each scale as the point where the wavelet transform magnitude decays to 2% of its peak value [38,39], (c) delineating the COI by adding and subtracting (1/2) the wavelet footprint at the beginning and end of the observation interval at each scale [37,40,41], and (d) using the wavelet time interval that encompasses 95% of the wavelet's energy for Morse wavelets [42–44].

2.1.2. Scalogram Quality

In order to analyze the oscillatory behavior in most physiological signals, the scalogram is represented in terms of the discrete frequency variable f, instead of the scale variable. The discrete-time and discrete-frequency S-scalogram can be represented by the matrix $G_S(f,n)$, $f = 0, 1, \ldots, (J-1)$, where J in this case is the number of discrete-frequency bands. The COI depends not only on the wavelet choice and frequencies but also on the signal duration. For a given scalogram, the ratio ρ of the number of V-scalogram coefficients to the total number of S-scalogram coefficients can be used as a measure of the quality of the scalogram. For a $(J \times N)$ scalogram, the ratio is given by

$$\rho = \left(\sum_{f=0}^{J-1} N_f\right)/[(J)(N)], \tag{2}$$

where N_f is the number of coefficients in row f that are inside the COI. For a given scalogram, a quality ratio β can also be defined in terms of the power enclosed by the COI to the total power in the following manner:

$$\beta = \frac{\sum_f \sum_n |W(f,n)|^2}{\sum_{f=0}^{J-1} \sum_{n=0}^{N-1} |W(f,n)|^2}, \tag{3}$$

where the summations in the numerator are taken for values of (f,n) that fall inside the COI. Higher values of ρ and β indicate better scalogram quality.

2.2. The Three Scalogram Approaches

Our goal is to compare the performances of classifiers that employ S-scalogram, Z-scalogram, and V-scalogram features; therefore, the most straightforward way to demonstrate this goal is to design classifiers for each ERP channel. Clearly, the same goal can be demonstrated by designing classifiers that combine information from multiple ERP channels [45,46]. However, the performance will then also depend on the combination methods which could mislead the performance comparisons. Therefore, the assumption hereafter is that the classifiers are single-channel classifiers.

The strategy to demonstrate the influence of the COI on classification performance is to develop three classification approaches in which the feature vectors are derived from (a) the S-scalogram $G_S(f,n)$, (b) the Z-scalogram $G_Z(f,n)$, and (c) the V-scalogram $G_V(f,n)$. The scalogram $G_S(f,n)$ consists of the signal power at each time-frequency location. For classification problems, transforming a signal $x(n)$ into $G(f,n)$ is regarded as a feature

extraction process. Clearly, additional operations can be performed on the scalogram feature matrix to generate other features such as the mean band power, ratios of power in select bands, and those which are typically used in image classification, such as texture, entropies, higher-order statistical features, and PCA-derived features. However, the classification results would then depend on the choice of the features making it difficult to draw direct conclusions about the influence of the COI on classifier performance. Therefore, in order to avoid this problem, no further modifications of the scalogram features are employed in the development of the classifiers. That is, the features are simply the scalogram coefficients.

The most direct way to demonstrate the performance trends across the three classification approaches is to concatenate the rows of the scalogram into a feature vector. In the formulation of the three classification approaches, the region outside the COI is represented by $G_O(f,n)$ such that

$$G_S(f,n) = G_V(f,n) \cup G_O(f,n) \tag{4}$$

2.2.1. S-Scalogram Approach

The feature vector in this approach is formed by concatenating the rows of the $(J \times N)$ S-scalogram. If Δ and T represent the row concatenation and transpose operations, the $[(J)(N) \times 1]$ feature vector G_S is given by

$$G_S = [\Delta_{f=0}^{J-1} G_S(f,:)]^T \tag{5}$$

where, $G_S(f,:)$ is row f in $G_S(f,n)$. The feature vector G_S includes a combination of accurately computed and artifact features. It can, therefore, be expected that the performance will be impacted negatively by the inclusion of the artifact features.

2.2.2. Z-Scalogram Approach

In this method, the S-scalogram is modified by multiplying it with a mask $M(f,n)$ such that the resulting scalogram has elements set to zero outside the COI and are unchanged inside the COI. That is, the zeroed-out S-scalogram is given by

$$G_Z(f,n) = G_S(f,n) \times M(f,n) \tag{6}$$

where

$$M(f,n) = \begin{cases} 1, & all\ (f,n)\ \in G_V(f,n) \\ 0, & all\ (f,n)\ \in\ G_O(f,n) \end{cases} \tag{7}$$

The feature vector G_Z, which has the same dimension $[(J)(N) \times 1]$ as G_S, is given by

$$G_Z = [\Delta_{f=0}^{J-1} G_Z(f,:)]^T \tag{8}$$

Although the dimension of the feature vector G_Z is the same as G_S, it can be hypothesized that the performance would improve by zeroing out the unreliable features that are outside the COI. However, including features with zero values across all classes is completely uninformative, providing no discriminatory value. Furthermore, the unnecessarily large feature vector can lead to overfitting, which can negatively impact the performance.

2.2.3. V-Scalogram Approach

In this method, the S-scalogram is cropped with a COI-shaped mask $V(f,n)$ which selects the features inside the COI and discards the features outside the COI. The resulting COI-shaped 2-dimensional V-scalogram $G_V(s,n)$ is given by

$$G_V(f,n) = G_S(f,n) \blacksquare V(f,n) \tag{9}$$

where ■ is used to denote the cropping operation, and the mask selects or discards features according to

$$V(f,n) = \begin{cases} select, & all\ (f,n)\ inside\ COI \\ discard, & all\ (f,n)\ outside\ COI \end{cases} \quad (10)$$

The $[(J)(N_V) \times 1]$ feature vector G_V is given by

$$G_V = [\Delta_{f=0}^{J-1} G_V(f,:)]^T \quad (11)$$

where, $N_V = \sum_{f=0}^{J-1} N_f$, and N_f is the duration of the frequency band f inside the COI. Unlike the two previous cases, the durations of the frequency bands are not equal; consequently, the feature vector G_V does not have the same dimension as G_S. By excluding the features outside the COI, the dimension $(J)(N_V)$ of G_V will be smaller than the dimensions of G_S and G_Z. Through the simultaneous exclusion of unreliable features included in the S-scalogram approach and the cropping out of irrelevant features included in the Z-scalogram approach, it can be hypothesized that classifiers using G_V for the feature vector should outperform the classifiers using the feature vectors of the S-scalogram and Z-scalogram approaches.

2.2.4. V-Complement Scalogram Classification

In order to erase any doubts about the importance of excluding the features outside the COI, classifiers were also developed using only the features outside the COI, that is, the features contained in $G_O(f,n)$. This approach, represented by the $\overline{V}$-scalogram, is referred to as the V-complement scalogram approach because the feature set is the complement of the V-scalogram feature set. If these features are truly unreliable, as claimed throughout this study, the classification accuracies of the $\overline{V}$-scalogram classifiers should be worse than those of the baseline S-scalogram classifiers. The V-complement approach is not considered to be 1 of the 3 scalogram classification approaches but is simply developed to demonstrate that the features outside the COI do not carry any useful discriminatory information.

2.3. Subsample ERPs for Classifier Design

In statistical analyses, an ERP is formed by averaging all the time-locked single trials collected from a channel. The averaging operation is performed independently on the single trials of each channel. That is, 1 ERP is obtained for each channel. A different approach is required for ERP classification because classifiers are designed using the data split into two mutually exclusive subsets: the training set needed to train the model and the test set needed to evaluate the model. The term "design" includes classifier training and testing. Designing customized classifiers for individual subjects, whether it is through direct ERP-based classification or feature-based ERP classification, is impossible if each ERP category is represented by a single ERP. Group-based classifiers would require a very large number of subjects to generate enough ERPs for training and testing, which, in many practical cases, may be impossible. Also, it is important to note that the performance of a classifier is highly dependent on the number of ERPs and the quality of the training set. A small training set would not be representative of the variations in the ERPs, which results in poor generalization. Furthermore, large training sets are required to overcome the "curse of dimensionality" often encountered in the design of parametric classifiers [47]. Poor-quality (poor SNR) training sets lead to poor learning, which, in turn, results in poor performance.

The most obvious approach to increase the size of the training and test sets is to simply use single trials without averaging. However, high classification accuracies cannot be expected with single trial signals due to the poor SNR. Because the SNR improves through averaging [48–50], it can be expected that the classification accuracy will improve by averaging over a small number of single trials. The classification accuracy can be expected to increase even further by increasing the number of trials averaged. For convenience, the signal formed by averaging m single trials will be referred to as an m-ERP, and m will

be referred to as the "averaging parameter". Furthermore, the full-sample ERP formed by averaging all Ω single trials in an ensemble will be referred to as the Ω-ERP, and, for convenience, a single trial will be referred to as a 1-ERP. In order to form an *m*-ERP, the 1-ERP ensemble can be partitioned sequentially into blocks of size *m* and then averaging the 1-ERPs in each block. Alternatively, subsamples of size *m* single trials can be drawn randomly without replacement and averaged to form distinct *m*-ERPs. However, in both cases, the sizes of the *m*-ERP training and test sets, which are equal to the number of 1-ERPs in the respective sets divided by *m*, "shrink" as *m* increases, which leads to overfitting. The obvious solution to this problem is to increase the size of the single trial ensembles. However, collecting large ensembles of single trials to form a large ensemble of *m*-ERPs for training and testing is difficult in practice because subjects tend to lose focus on the task due to fatigue and lapses in concentration over long data collection sessions. Another alternative is to average *m* single trials drawn with replacement to generate large ensembles of *m*-ERPs. However, there is the risk of generating identical *m*-ERPs, which can result in singular covariance matrices. This could be problematic for parametric classifiers such as those with Gaussian discriminant functions, which require the computation of the inverse of the covariance matrix [47].

The real challenge, therefore, in designing ERP classifiers for many applications is to avoid having to repeat the stimulus presentation numerous times to be able to form high SNR *m*-ERPs, especially during real-time testing. To be practical, most applications require accurate classification of ERPs averaged over a small number of single trials. The goal, therefore, is to enable the design of practical ERP classifiers that can accurately classify ERPs averaged over a small number of single trials without having to collect a prohibitively large ensemble of 1-ERPs.

2.3.1. Generation of *m*-ERPs through "*m*-Subsample Averaging"

In order to overcome the shrinking problem that accompanies sequential averaging and averaging random subsamples without replacement, we employ a subsampling and averaging strategy introduced and implemented in [45,51,52]. We will refer to this method of generating subsample ERPs as "*m*-Subsample Averaging (*m*-SA)". For a given 1-ERP ensemble of size Ω_0, the *m*-SA strategy can be summarized as follows:

(a) Draw a random subsample of 1-ERPs of size m, $m < \Omega_0$. The m 1-ERPs in the subsample are drawn without replacement;
(b) Average the m 1-ERPs in the subsample to obtain an *m*-ERP;
(c) Replace the m 1-ERPs of the subsample into the single trial ensemble;
(d) Repeat steps (a)–(c) q times to generate an ensemble of Ω_q *m*-ERPs. Each ensemble generation is referred to as a "run";
(e) Repeat steps (a)–(d) Q times to yield Q runs, with each run containing an *m*-ERP ensemble of size Ω_q.

In order to maintain mutual exclusivity between the training and test sets, the 1-ERP ensemble is first partitioned into training and test sets, and subsequently, m-ERPs are generated from each of these sets. Each run, therefore, begins with randomly partitioned 1-ERP training and test sets. The specified number of *m*-ERPs for the training and test sets set is generated from the 1-ERPs of the training and test sets, respectively. Consequently, none of the 1-ERPs used to form the *m*-ERPs of the training set are included in forming the *m*-ERPs of the test set.

The key point to note is that *m*-SA offers the flexibility of generating large ensembles of *m*-ERPs for classifier design. Most notably, the *m*-ERP ensembles can be generated from the 1-ERP ensembles of individual subjects, thus enabling the design of customized classifiers for individual subjects. Clearly, *m*-SA is equally applicable for generating large ensembles for multi-subject group-based classifier design. Increasing the amount of data for training in this fashion is a form of data augmentation that can be especially beneficial for improving the performance of data-driven classifiers such as deep learning CNNs. Also noteworthy is that the training set can be made up of *m*-ERPs with different values of *m* to increase the

amount of data and the variability in the training set. An additional point to note is that the averaging parameter does not have to be the same for the training and test sets. That is, the classifier could be trained with m-ERPs and tested on n-ERPs. In practice, it is desirable to keep both m and n small in order to avoid having to collect a large number of single trials for training and testing, respectively. It is especially important to obtain high accuracies for small values of n so that trials do not have to be repeated many times during testing.

2.4. Rank-of-Rank-Sum (RRS) Channel Ranking

The primary reasons for channel selection are to extract a subset of channels that carry useful discriminatory information and to reduce the computational burden. For ERP analysis, temporospatial PCA can be used to identify channels and time windows that best capture effects elicited by stimuli [53,54]. Therefore, the classification performance can be expected to improve by using only the identified channels and time windows. However, the drawback of using such prior knowledge for channel selection with respect to multichannel ERP classifier design is that the information that a particular classifier uses to discriminate between the ERP categories may differ from the information identified by PCA which is aimed at seeking projections that best represent the data in a least-square sense. A different approach which is aimed at selecting channels that best discriminate between the ERP classes is required for classification problems. A common approach is to use a suitable criterion with backward elimination to develop greedy algorithms which start with the full set of channels and decrease the number of channels, one channel at a time, to determine a subset of channels that yield the best criterion score. References [55–57] review several notable channel selection methods that have been developed for EEG channel selection.

The rank-of-rank-sum (RRS) algorithm developed in this study is feature-blind and quite general because it is formulated to rank channels for classification problems involving multiple classes (polychotomous) and multiple subjects. Binary classification and single-subject cases are treated as special cases of the RRS algorithm. The channels are ranked according to the interclass separations of the ERPs generated at the channels. The interclass separation measure selected is a scatter-normalized measure of separation between two cluster means. In general, higher classification accuracies can be expected if clusters have higher interclass separations. However, the interclass separation, which is a pairwise measure, is not directly applicable for multi-channel and polychotomous ranking because it provides a separation measure only for a pair of channels and a pair of ERP classes. Moreover, it is not directly applicable to multiple subjects. To solve the first problem, channel ranking is obtained via the ranking of the sum of the pairwise rankings of the $U(U-1)/2$ pairs of ERP classes where U is the number of ERP classes. One approach to solving the second problem is to combine the B subjects into a single group. This approach, however, is sensitive to the inter-subject variabilities of the ERPs. Instead, the RRS algorithm solves the second problem by determining the rankings of the sum of the channel rankings across the B subjects. The RRS algorithm is described in detail in Appendix A.

2.5. Choice of Classifiers

In order to test the performance-related hypotheses and to determine whether the performance trends are consistent across classifiers, channels, and subjects, a range of diverse classifiers is selected to implement the three scalogram approaches for ERP classification. The following well-known classifiers are selected: SVMs, RFs, k-NN, MLPs, and deep learning CNNs. Each of these classifiers has its advantages and limitations. Details of these well-known classifiers can be found in excellent books on machine learning [58–62].

Two variants of CNN classifiers are designed: vector input CNN-1 and matrix input CNN-2. The CNN-1 classifier accepts vector inputs and can, therefore, be implemented for all 3 approaches through the row concatenation operations described in Section 2.2. As mentioned in the introduction, the matrix scalogram is an ideal format for CNN inputs. Implementing matrix input CNN classifiers for the first 2 approaches is straightforward because the S-scalogram and the Z-scalogram have matrix formats. However, implementing

CNN classifiers capable of accepting the unequal row V-scalogram matrix is problematic because the CNN matrix convolutions require equal row matrices. Such an implementation is beyond the scope of this study.

2.6. Data Sets

The EEG data used to demonstrate the application and evaluate the performance of the 3 classification approaches was downloaded from:

https://eeglab.org/tutorials/10_Group_analysis/study_creation.html#description-of-the-5-subject-experiment-tutorial-data (accessed on 1 October 2022).

This data set was selected because it was freely available and ideal for demonstrating the goals of this study. However, the methods formulated in this study are quite general and are not tied to this particular data set or any other data set. According to the description of the study on the website, the EEG data were collected from 5 subjects performing an auditory binary semantic task in which they were asked to distinguish between synonymous and non-synonymous word pairs. The second word was presented 1 s after the presentation of the first word. Data epochs were extracted 2 s before the second-word onset to 2 s after the second-word onset. EEGs were collected from 64 channels using a sampling rate of 200 Hz. After decomposing each subject's data by ICA, 2 EEG datasets were extracted: the synonymous data set comprising trials in which the second word was synonymous with the first one and the non-synonymous data set in which the second word was not a synonym of the first. Thus, the study included 5 binary data sets or (5 subjects) (2 conditions) = 10 datasets for each of the 5 subjects. Both datasets of each subject were recorded in a single session. The number of epochs for each condition varied from 196 to 235 across the 10 data sets.

For the purpose of this study, the single trials (1-ERPs) for the classification experiments consisted of the 1 s segment extracted after the second-word onset in each epoch. Consequently, the total number of samples in each 1-ERP was 200. In order to facilitate 5-fold cross-validation and to have an equal number of epochs across the 10 data sets, the first 195 epochs were selected from each data set. The top 8 ranked channels for each subject determined by the RRS algorithm were selected for the experiments. The rankings are listed in Table 1. The rankings of the channels differ across the 5 subjects, which is not unexpected. Whether these channels differ from those extracted from temporospatial PCA is not of concern because the focus is on channel selection for classification and not for analyses. Furthermore, whether the top-ranked channels are correlated proximal channels is not taken into account because the goal is to simply select 1-ERPs from a subset of channels, instead of all channels, that can be used as data sets for the experiments. If needed, correlation analyses could be conducted on the ranked channels to systematically include low-correlated channels starting with the top-ranked channel.

Table 1. The top 8 ranked channels for each subject.

Subject	**Top 8 Ranked Channels**							
b_1	P3	P1	P5	Pz	P7	POz	PO2	PO1
b_2	PO4	F3	CP5	PO2	Oz	PO6	TP7	O1
b_3	O2	Oz	O1	PO6	PO4	PO2	TP7	POz
b_4	O2	Oz	TP7	O1	F3	F1	F5	CP5
b_5	C2	P5	Cz	C4	P3	P7	C1	FCz

Each classifier was designed on (5 subjects) (8 channels) = 40 binary data sets. The 40 binary data sets are large enough to adequately observe the performance trends across the three approaches. Adding more channels will simply increase the size of the already large tables that are included in the results without contributing to the trend analyses. Figure 2 shows the full-sample Ω-ERPs of subject b_1 formed by averaging the 1-ERPs in the binary data sets. Observe that the Ω-ERPs of the two classes appear to be fairly well separated in all 8 channels.

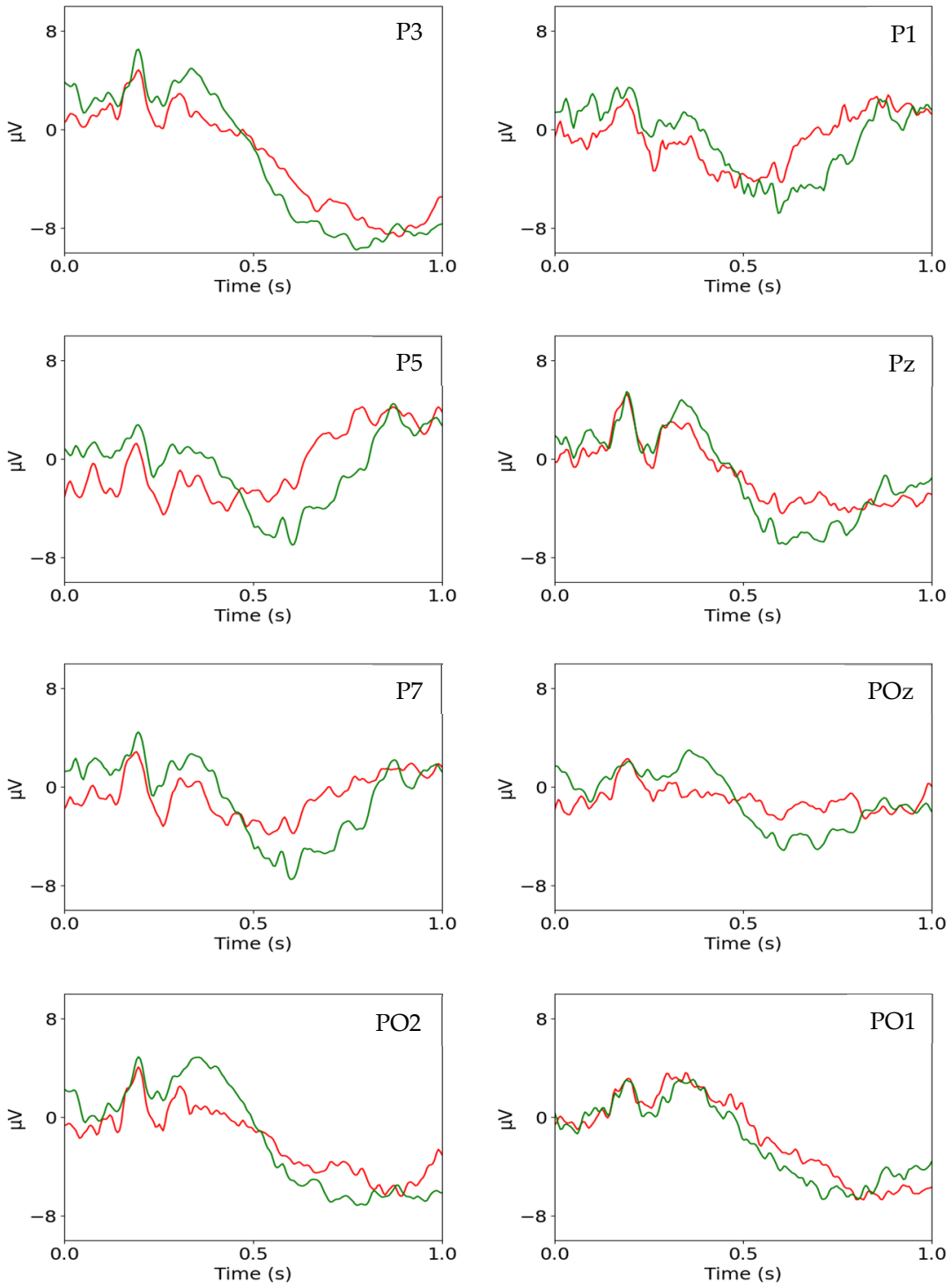


Figure 2. The full-sample synonymous (green) and non-synonymous (red) Ω-ERPs of the 8 selected channels of subject b_1.

2.7. Classification Experiments

This section describes various aspects of the design of experiments to test the implementations of the three classification approaches using the 5 different types of classifiers and the 5-subject 8-channel ERP data sets.

2.7.1. Wavelet Choice

Over the years, many different CWTs have been proposed, which include, among many others, Morlet [63,64], Morse [44], Gaussian [2], and Mexican Hat [65,66]. The choice of the wavelet is application dependent, and in this study, the analytic Morlet wavelet is selected because it is frequently used to analyze the oscillatory behavior of ERPs and EEGs. The analytic Morlet mother wavelet, a product of a complex exponential signal of frequency f_0 and a zero-mean Gaussian window with variance σ^2, is given by

$$\Psi_{f_0}(t) = (A)(e^{-\frac{t^2}{2\sigma^2}})\left(e^{j(2\pi f_0)t}\right) \tag{12}$$

where the constant A ensures that the wavelet energy is equal to one. The analytic wavelet coefficients are complex, and the power spectrum is zero for negative frequencies. Note that the wavelet is defined in terms of a frequency parameter which is preferred in applications involving oscillatory analyses.

2.7.2. Selection of the Averaging Parameter

The aim of this study is not focused on obtaining the highest classification accuracies but on analyzing the performance trends across the 3 classification approaches using the data sets. In general, classifiers trained and tested on high SNR data sets yield good accuracies across the classifiers, making it difficult to compare their performances. Furthermore, it may not be practical to repeat trials a large number of times to generate high-SNR m-ERPs during real-time testing. It is, therefore, important to be able to obtain high accuracies for small values of m. For these reasons, the smallest value, $m = 4$, which enabled the analyses of the performance trends across the 3 approaches, was selected for generating the m-ERP ensembles. In the experiments to follow, a classifier was developed to test the 4-ERPs of each channel collected from each subject independently.

2.7.3. Feature Vector Generation

The feature vectors for each approach were generated from the Morlet wavelet power scalogram through row concatenation, as described in Section 2.2. The question as to whether the scalogram is computed from 1-ERPs or m-ERPs has to be considered. In general, the scalogram of the average of the single trials is used for evoked power analysis, whereas the average of the scalograms of each single trial, which gives the total power, is used for induced power analysis. The total power is the sum of the evoked and induced power. In this study, for the purpose of feature extraction, a single scalogram is computed for each m-ERP where m is equal to 4. The duration of each 4-ERP (same as the single trial duration) was N = 200 samples, and the number of frequency bands J was 108. The dimensions of the scalograms were, therefore, 108 $\times$ 200. Hence, the dimension of the feature vectors G_S and G_Z were (21,600 $\times$ 1). Using the process outlined in Section 2.2 for selecting the features inside the COI, the dimension of the feature vector G_V was 17,056 $\times$ 1. The ratio ρ of the number of V-scalogram coefficients to the total number of S-scalogram coefficients for this case is (17,056/21,600) (100) = 78.96%. That is, approximately 21% of the scalogram coefficients were artifact coefficients.

2.7.4. Classifier Architectures and Parameters

This subsection describes the architectures and hyperparameters of the 6 different classifiers that were implemented. No serious attempt was made to optimize the architectures and hyperparameters because the goal is to compare the relative performances of a given

classifier using the S-scalogram, Z-scalogram, and V-scalogram feature vectors. Details of the classifiers are as follows:

SVM Classifiers: The SVM classifiers were implemented using the Gaussian radial basis function kernel. The best combination of the regularization parameter C and influence parameter γ was selected using an exhaustive grid search.

RF Classifiers: The following hyperparameters were selected: number of trees = 10, maximum depth of a tree = 475, minimum number of samples required to split an internal node = 3, minimum number of samples required to be leaf node = 1, splitting quality criterion = entropy.

k-NN Classifiers: The Euclidean distance metric was used to measure proximity. Through trial and error, it was found that k = 20 yielded good performance across all classifiers.

MLP Classifiers: The MLP classifiers consisted of four layers of neurons: the first layer had 1024 neurons, the second layer had 512 neurons, the third layer had 256 neurons, and the output layer had 2 neurons. The sigmoidal activation function was used in all 4 layers. The following training options were used: initialization = uniform random, optimizer = Adam, learning rate = 0.001, number of epochs = 50, drop-out probabilities = 0.15.

CNN-1 Classifier: The CNN-1 classifiers consisted of three convolution layers followed by a max pooling layer and terminated with a fully connected network (FCN) consisting of four layers of neurons. The convolution layers used the ReLU activation functions, the first three layers in the FCN used sigmoidal activations, and the last layer used softmax activations. The "same" operation was used in the convolution layers. The number of filters in the convolution layers was 32. The filter dimensions in the convolution layers were (9×1). A (2×1) max pooling filter with a stride of 2 was used in the pooling layer. The FCN had 1024, 512, 256, and 2 neurons in the four layers. The following training options were used: initialization = uniform random, optimizer = Adam, learning rate = 0.001, number of epochs = 50, drop-out probabilities = 0.15.

CNN-2 Classifiers: The CNN-2 classifiers consisted of a convolution layer, max pooling layer, convolution layer, max pooling layer, convolution layer, max pooling layer, and terminated with a fully connected network (FCN) consisting of four layers of neurons. The convolution filters had dimensions (3×3), and the max pooling filter had dimensions (2×2) with a stride equal to 2. The "same" operation was used in both convolution layers. The number of filters in both convolution layers was 32. The convolution layers used the ReLU activation functions, the first three layers in the FCN used sigmoidal activations, and the last layer used softmax activations. The FCN had 1024, 512, 256, and 2 neurons in the four layers. The following training options were used: initialization = uniform random, optimizer = Adam, learning rate = 0.001, number of epochs = 50, drop-out probabilities = 0.15.

In summary, the SVM, RF, k-NN, MLP, and CNN-1 classifiers were developed for all three scalogram approaches, and the CNN-2 classifiers were developed for the first 2 approaches to give a total of 17 classifiers. Each classifier was trained and tested independently on the 40 binary data sets. The classifiers were implemented using the PyTorch library.

2.7.5. Cross-Validation

The steps outlined in Section 2.3.1 were used to generate runs consisting of the m-ERPs of the training and test sets. For each run that started with randomly ordered 1-ERPs, the 195 1-ERPs of each class were partitioned into 5 folds, each containing 39 1-ERPs. The training set consisted of 4 folds (156 1-ERPs), and the test was the remaining fold. From the 1-ERP training set, an equal number (156) of 4-ERPs were generated for the training set. Similarly, 39 4-ERPs were generated for the test set. This method of generating 4-ERP training and test sets was conducted for each channel. The channel classifiers were trained and tested using 5-fold cross-validation for each run. For each run, the total number 4-ERPs for training and testing each channel classifier consisted of (195 4-ERPs/class) (2 classes) = 390 4-ERPs. The classification accuracy for each run, expressed as a percentage, was estimated as the ratio of the correctly classified 4-ERPs to the total number (390) of 4-ERPs of the 2 classes. The procedure was repeated over 50 runs, and

the final classification accuracy of each classifier was estimated by averaging the accuracies of the 50 runs. That is, the final classification accuracy of each classifier was averaged across (390) (50) = 19,500 4-ERPs belonging to the 2 classes. Importantly, all the classifiers were implemented using identical training and test sets.

3. Results

This section presents the results of the various experiments designed to observe the performance trends of the 3 classification approaches across classifiers, channels, and individual subjects.

3.1. m-ERP Classification Results

In order to observe the trends across the 3 approaches closely, the final classification accuracies obtained from the classifiers across the three approaches are presented separately in five "full" tables, which contain the results for each subject. Due to the large size, the five full tables are presented in Appendix B (Tables A1–A5). Each entry in the tables is the average classification accuracy obtained from testing (390 4-ERPs × 50 runs) = 19,500 4-ERPs. The best result for each channel is highlighted in bold font. The entries are empty for the CNN-2 classifiers for the V-scalogram approach because they were not implemented. Any inconsistent result (expected improvement is not observed) is highlighted in red. Out of the total of 680 results presented in the five full tables, only 10 (<1.5%) were minorly inconsistent in the sense that the expected improvements of the Z-scalogram classifiers over the S-scalogram classifiers were not observed. Table 2 summarizes the overall average classification across the five subjects and eight channels for each approach. Except in the row labeled Global Average, each entry in Table 2 is the average classification accuracy obtained from testing (five subjects × eight channels × 19,500 4-ERPs) = 780,000 4-ERPs. The row labeled Global Average is the average of the classifiers across each approach (column averages). An asterisk on CNN-2 indicates that the CNN-2 results were not included in the computation of the Global Average. The CNN-2 averages are included below the Global Averages to emphasize this point.

Table 2. Average classification accuracies of the six different types of classifiers across the entire test set consisting of 780,000 4-ERPs. The global average is the average of the column entries in the first five rows of the table.

Classifier	Approach		
	S-Scalogram	Z-Scalogram	V-Scalogram
SVM	58.33	62.49	70.95
RF	75.30	81.74	89.07
k-NN	58.85	62.92	70.86
MLP	75.84	81.27	89.34
CNN-1	77.68	83.91	92.96
Global Average	69.20	74.46	82.64
CNN-2 *	81.18	92.73	——

* indicates that the CNN-2 results were not included in the computation of the Global Average.

Figure 3 shows the 95% confidence interval error bars for the Global Averages in Table 2. The V-scalogram error bar does not overlap with the S-scalogram and Z-scalogram error bars. Therefore, the improvements of the V-scalogram approach over the S-scalogram and Z-scalogram approaches are statistically significant, with p-values less than 0.05.

3.2. Single Trial Results

Up to this point, we have assumed that single trial classification will yield poor classification accuracies. In order to support this assumption, the classifiers were trained and tested on the 1-ERPs of Subject b_3. The results are shown in Table A6 in Appendix B. For brevity, the results of the 4 other subjects are not included because the trends are the

same as those observed for Subject b_3. As expected, the results for the 1-ERP classification are inferior to the 4-ERP classification. Interestingly, the CNN-1 results using V-scalogram features are quite good for single trial classification. The significant jumps in the CNN results from the S-scalogram accuracies to the V-Scalogram accuracies are noteworthy.

3.3. V-Complement Scalogram Results

Classifiers were developed using only the $(21600 - 17056) = 4544$ features outside the COI. Table 3 summarizes the classification accuracies of the classifiers trained using this $\overline{V}$-scalogram approach. The S-scalogram results are also included in the table to facilitate comparison. The $\overline{V}$-scalogram average classification accuracy of 51.95% is close to the 50% accuracy expected for random binary classification. Clearly, the artifact coefficients do not carry useful discriminatory information. Therefore, the performance should improve by zeroing the artifact coefficients and improve even further by removing them completely, as confirmed by the experimental results in Appendix B.

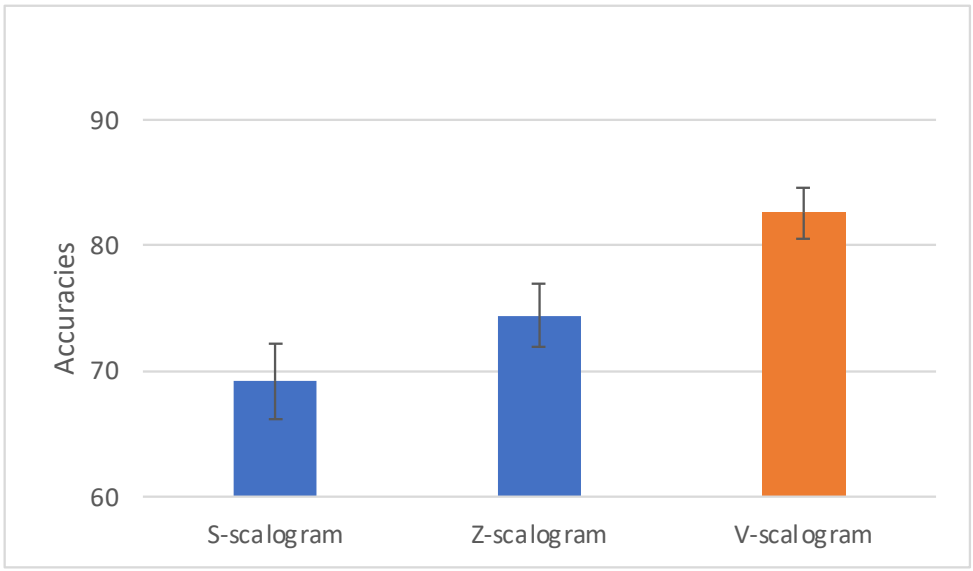


Figure 3. The 95% CIs for the global mean accuracies of the 3 approaches.

Table 3. A comparison of the average classification accuracies of the S-scalogram and the $\overline{V}$-scalogram approaches.

Classifier	Approach	
	S-Scalogram	**$\overline{V}$-Scalogram**
SVM	58.33	50.77
RF	75.30	51.92
k-NN	58.85	47.08
MLP	75.84	53.85
CNN-1	77.68	56.15
Global Average	69.20	51.95

4. Discussion

This section discusses the findings presented in the Section 3.

4.1. Performance Trends

The following general trends in the 6 full tables in Appendix B and the summary table are important to note: for each classifier, there is an improvement in the performance of the Z-scalogram classifiers over the S-scalogram classifiers (including CNN-2) and a consistent improvement of the V-scalogram classifiers over the Z-scalogram classifiers.

This is also reflected across the overall averages for the three approaches. Importantly, the CNN-1 results in Tables A1–A5 also show that the two-class semantic ERPs can be classified with accuracies exceeding 94% for all 5 subjects using a value as small as $m = 4$ for subsample averaging with the V-scalogram. As noted in the subsampling section, even higher accuracies can be obtained by increasing the averaging parameter m.

For the CNN-2 classifier, the improvement of the Z-scalogram approach over the S-scalogram approach is also notable. Quite interestingly, even the Z-scalogram approach can yield accuracies greater than 93% for all five subjects. This impressive performance can be attributed to the fact that the convolution filters are trained to extract discriminatory features from input during the training phase. That is, the zeroed-out region outside the COI is ignored to some degree by the filters. The filters are influenced by the zeroes to a greater extent when they overlap with scalogram regions that are both inside and outside the COI boundary. Based on the observed trends of the other classifiers across the three approaches and the trends between the two CNN-2 implementations across the first two approaches, we can expect a dramatic improvement in the CNN-2 implementation of the V-scalogram approach over the standard S-scalogram approach if the convolution mechanisms in the CNNs can be modified to accommodate unequal row matrices. Furthermore, through an understanding of how the filter weights are trained by convolving over entire matrices for S and Z-scalograms containing artifact and zeroed-out coefficients, respectively, it can be expected that the convolution filters trained only over the correct V-scalogram information are bound to lead to better performance. It is important to note that modifying wavelet-based CNN classifiers to restrict the matrix convolutions within the COI would be a major breakthrough, given the huge attention focused on developing and applying wavelet-based CNN classifiers to numerous problems in various fields. Future efforts will focus on developing direct/indirect methods to restrict matrix convolutions within the COI.

4.2. Relative Improvements

The consistent improvements of the Z-scalogram and V-scalogram approaches over the baseline S-scalogram approach are quite remarkable because they are observed across the m-ERPs of all 5 subjects across all 8 channels (all 40 data sets). The relative improvement in the performance of an approach B over an approach A, denoted by $\Gamma(B, A)$, can be measured using

$$\Gamma(B, A) = \left[\frac{\alpha(B) - \alpha(A)}{\alpha(A)}\right] \times 100 \tag{13}$$

where $\alpha(A)$ and $\alpha(B)$ are the classification accuracies of approaches A and B, respectively. Table 4 summarizes the relative improvements for the average accuracies listed in Table 2. These results clearly confirm that the Z-scalogram approach outperforms the S-scalogram approach and the V-scalogram approach outperforms the Z-scalogram approach. Most importantly, the improvements of the V-scalogram approach over the standard S-scalogram approach are dramatic.

Table 4. Relative improvements in the accuracies listed in Table 2.

Classifier	$\Gamma(Z,S)$	$\Gamma(V,Z)$	$\Gamma(V,S)$
SVM	7.13	13.54	21.64
RF	8.55	8.97	18.28
k-NN	6.93	12.62	20.42
MLP	7.15	9.94	17.80
CNN-1	8.02	10.80	19.68
Global Average	7.61	10.97	19.42
CNN-2	14.24	—	—

4.3. Comparison of Classifiers

Although the primary goal of this study is to show that for any given classifier, the performance of the classifier using the standard S-scalogram features can be improved

by using Z-scalogram features, and further improvements can be obtained by using V-scalogram features, the results in the tables can also be used to rank the 6 different classifiers to pick the best performing classifier type. From the averages in Table 2, it can be concluded that the best results for the V-scalogram approach across classifiers, channels, and individual subjects were obtained with the CNN-1 classifiers. The superiority of the CNN classifier models can be attributed to the fact that the filters in the convolution layer have the ability to extract complex features that are relevant to classification. Most importantly, the scalograms can be input directly to the CNNs without having to extract a set of hand-engineered features. Consequently, the performance is not dependent on the hand-engineered feature set, which is often determined through trial and error. The size and the shape of the convolution filters can couple information in different ways to extract different types of features from the same input presented to the CNN. For example, row filters couple information from individual frequency bands in the scalogram. A matrix filter can be used to couple information across frequency bands. If needed, the complexity of the features can be increased by increasing the number of convolution layers. In general, CNNs are highly scalable for larger classification problems by increasing the network depth and increasing training. Therefore, CNNs should be a good choice for multi-subject group-based classification. Furthermore, there are several elegant ways of fusing "information" from multiple channels with CNNs to improve performance. The information can be data/features of the channels or decisions of the channel classifiers. The study in [67] involving the classification of multi-axial (multi-channel) signals using CNNs shows that, in general, multi-axial fusion classification models outperform the best uniaxial (single-channel) classification model.

5. Conclusions

The goal of this study was to demonstrate the importance of exploiting information offered by the COI in designing CWT-based classifiers. It was hypothesized that the performance would improve by zeroing out the unreliable scalogram features outside the COI and that the performance would improve even further by completely excluding the unreliable features outside the COI. A subsampling strategy was developed to generate small-average ensembles to enable customized design for individual subjects/patients, and a rank-of-rank sum channel selection strategy was developed to select a subset of channels from multiple channels. The S-, Z-, and V-scalogram classification approaches developed to validate the performance hypotheses were implemented with a set of diverse classifiers and tested on semantic task ERPs of 5 subjects. The results from 40 data sets confirmed the expected relative improvements, and it was noted that the improvements were remarkably consistent across channels, classifiers, and subjects. Most importantly, the relative improvements of the V-scalogram classifiers over the standard S-scalogram classifiers were dramatic. Among the 5 classifier types implemented, the CNN classifiers yielded the best results consistently.

In conclusion, this study demonstrated unequivocally that excluding the region outside the COI for feature extraction results in a significant improvement in classifier performance. The V-scalogram approach is not restricted to ERP/EEG classification and is applicable to improving the performance of wavelet classifiers designed to classify other physiological signals as well as signals in many other fields, including engineering, science, and economics. Undeniably, this is a significant contribution, considering the vast number of wavelet-based classifiers that have been reported but have not exploited the COI for feature extraction in the manner described in this study. Furthermore, the ability to design customized classifiers for individual subjects which yield high classification accuracies using subsample averaging is an important contribution for studying and diagnosing patient-specific disorders using ERPs and EEGs.

Author Contributions: Conceptualization, L.G., X.C. and R.S.G.; methodology, X.C., L.G. and R.S.G.; software, X.C.; validation, L.G., X.C. and R.S.G.; formal analysis, X.C. and L.G; investigation, X.C. and L.G., writing—original draft preparation, X.C. and L.G.; writing—review and editing, L.G., X.C. and R.S.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the anonymous reviewers for their helpful suggestions to improve the paper.

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

Appendix A

The convolution operations that lead to the definitions of the S-scalogram, Z-scalogram, V-scalogram, and COI are described in detail in this Appendix. The formulation of the generalized RRS channel selection algorithm applicable to single/multiple subjects and dichotomous/polychotomous classification is also presented in this Appendix.

Appendix A.1. Types of Convolution

The question that is addressed in this section is: what type of convolution causes the edge artifacts to occur in the scalogram coefficient matrix? In order to answer this question, three convolution mechanisms are described by assuming the durations of the signal $x(n)$ and the zero-centered mother wavelet filter $h(n)$ are N and $(2M+1)$, respectively. That is, the compact support of $h(n)$ is $[-M, M]$. It is also assumed that $M < N$.

Appendix A.1.1. Full-Convolution

In general, the discrete convolution sum is given by

$$y(n) = \sum_{\lambda=-M}^{M} x(\lambda)h(n-\lambda) \tag{A1}$$

where λ is a dummy variable. That is, the convolution operation is performed by shifting a folded version of $h(\lambda)$ along $x(\lambda)$ and finding the dot product at each shift n. It is assumed that the shift occurs in unit steps. In order to simplify the discussion, it will be assumed that $h(n)$ is symmetric. In full convolution, the rightmost sample of $h(\lambda)$ is aligned with the leftmost sample of $x(\lambda)$ at the start of convolution, and the leftmost sample of $h(\lambda)$ is aligned with the rightmost sample of $x(\lambda)$ at the end of convolution. By convention, the filtered output is at the filter center; therefore, the duration of the filtered output is $\widetilde{N} = [N + (2M+1) - 1 = N + 2M]$. In order to perform the dot product at the start and end, $x(\lambda)$ has to be padded by $2M$ samples on the left and on the right. A common practice is to pad the ends with zeros; however, other methods, such as padding with the mean value of the signal, linear padding, symmetric padding, periodic extension padding, and fitting polynomial functions, can also be employed [27,68,69]. For the given signal and filter durations, the full-convolution sum can be expressed as

$$y(n) = \sum_{\lambda=-M}^{M} \hat{x}(\lambda)h(n-\lambda), \quad n = 0,\, 1, \ldots, \left(\widetilde{N}-1\right), \tag{A2}$$

where, $\hat{x}(\lambda)$ is $x(\lambda)$ padded with $2M$ samples on both ends. It is important to observe that the first $2M$ and the last $2M$ samples of the output are computed from partial overlaps between $x(\lambda)$ and $h(\lambda)$, which are the artifact coefficients in this case. Irrespective of the padding method employed, the outputs computed from the partial overlaps are the inaccurate artifact coefficients.

In practice, full convolution is performed very efficiently in the frequency domain by multiplying the DFTs of $x(n)$ and $h(n)$. The filtered output is given by the inverse DFT of the multiplication. The DFTs are computed by augmenting both sequences with zeros to have durations equal to or greater than $\widetilde{N}$ for the resulting circular convolution to be identical to the desired linear convolution. The DFTs are computed efficiently using fast Fourier transform (FFT) algorithms [70].

Appendix A.1.2. Same-Convolution

This form of convolution is the one used to generate the standard CWT coefficient matrix to ensure that the rows have the same duration as the input. Each row of the coefficient matrix is obtained by performing the same-convolution operation between $x(n)$ and $h(n)$. The convolution operation starts by aligning the filter center with the rightmost sample of $x(\lambda)$ and ends with the alignment of the filter center with the rightmost sample of $x(\lambda)$. The signal $x(\lambda)$ is padded with M samples at both ends; therefore, the first M and the last M samples of the output are computed from partial overlaps, which result in artifact coefficients. For the given signal and filter durations, the same-convolution sum can be written as

$$y(n) = \sum_{\lambda=-M}^{M} \hat{x}(\lambda)h(n-\lambda), \quad n = 0,\ 1, \ldots, (N-1), \tag{A3}$$

where, $\hat{x}(\lambda)$ is $x(\lambda)$ now padded with M samples on both ends. We refer to the resulting scalogram, which has rows with the same duration as the input signal, as the same (S)-scalogram. The shorter duration same-convolution result can be obtained from full-convolution by cropping out the first M and last M samples. In general, row j of the standard scalogram is generated by same-convolving $x(n)$ with a wavelet having an effective support $[-(S_j)(M), (S_j)(M)]$; therefore, the first and last $(S_j)(M)$ coefficients in the row are the artifact coefficients, and the coefficients in between are the accurate coefficients.

Appendix A.1.3. Valid Convolution

The valid convolution operation yields coefficients that are accurate and is used to identify the COI in the coefficient matrix. In this form of convolution, the leftmost sample of the centered filter is aligned with the leftmost sample of $x(\lambda)$ at the start, and the rightmost sample of the centered filter is aligned with the rightmost sample of $x(\lambda)$ at the end of the convolution. Consequently, the duration of the output is $[N - 2M]$. Valid convolution requires no padding at the ends of $x(\lambda)$; thus, none of the output samples are computed from partial overlaps resulting in no artifact coefficients. That is, all valid coefficients are computed accurately. Valid convolution can be obtained from same-convolution by cropping the first and last M samples and by cropping the first and last $2M$ samples from full-convolution. For the given signal and filter durations, the valid convolution sum can be written as

$$y(n) = \sum_{\lambda=-M}^{M} x(\lambda)h(n-\lambda), \quad n = 0,\ 1, \ldots, (N-2M-1). \tag{A4}$$

As noted in Section 2.1, the COI is the boundary that separates the artifact coefficients from the accurate coefficients in the S-scalogram, the part of the S-scalogram that is inside the COI is the V-scalogram, and the entire S-scalogram with zeroed-out coefficients outside the COI is referred to as the Z-scalogram.

Appendix A.2. RRS Algorithm

The channel rankings can be computed from the 1-ERPs or the m-ERPs. The RRS algorithm is described in terms of m-ERPs and is applicable to 1-ERPs by setting m = 1. Given C channels, the ranking $R_m(c)$, $c = 1, 2, \ldots, C$, is such that the channel with the highest multi-class separation is assigned the highest $rank = 1$ and the channel with the

smallest is assigned the lowest $rank = C$. Based on this ranking, channels can be selected by evaluating the performance, starting with the best channel, and adding the next best channels until the performance starts to drop. Alternatively, a subset of channels with the highest ranks can be selected to decrease the computational requirements, especially for dense-electrode arrays. If L_{mxcb_i} and L_{mycb_i} are the m-ERPs of subject b_i, $i = 1, 2, \ldots B$, elicited by stimuli I_x and I_y at channel c, respectively, and $||L_1 - L_2||$ is the Euclidean distance between L_1 and L_2, the interclass separation between the m-ERPs belonging to a pair of classes, ω_x and ω_y, at channel c is given by

$$\rho_{(x,y)mb_i}(c) = \frac{\left|\left|\overline{L_{mxcb_i}} - \overline{L_{mycb_i}}\right|\right|^2}{\sum_{q=1}^{N_x}\left|\left|L_{mxcb_i}^{(q)} - \overline{L_{mxcb_i}}\right|\right|^2 + \sum_{q=1}^{N_y}\left|\left|L_{mycb_i}^{(q)} - \overline{L_{mycb_i}}\right|\right|^2}, \; c = 1, 2, \ldots, C \quad \text{(A5)}$$

where $L_{mxcb_i}^{(q)}$ and $L_{mycb_i}^{(q)}$ are the q^{th} m-ERPs; $\overline{L_{mxcb_i}}$ and $\overline{L_{mycb_i}}$ are the means, and N_x and N_y are the number of m-ERPs in the ω_x and ω_y m-ERP ensembles, respectively. Note that the pair (x, y) takes on the following values:

$$(x, y) = (\omega_1, \omega_2), (\omega_1, \omega_3), \ldots, (\omega_1, \omega_J), (\omega_2, \omega_3), (\omega_2, \omega_4), \ldots, (\omega_2, \omega_J), \ldots, \left[\omega_{(U-1)}, \omega_U\right]. \quad \text{(A6)}$$

The rank $R_{(x,y)mb_i}(c)$ of $\rho_{(x,y)mb_i}(c)$ for each subject b_i is determined for all possible pairs of x and y. Then, the rank-sum (RS) of channel c for subject b_i across all possible $(U)(U-1)/2$ pairs of ERP classes is given by

$$(RS)_{mb_i}(c) = \sum_{x=1}^{U-1}\sum_{y=x+1}^{U} R_{(x,y)mb_i}(c) \quad \text{(A7)}$$

The ranking of the channels for subject b_i is obtained from the rank of the rank-sum given by

$$R_{mb_i}(c) = Rank\left\{(RS)_{mb_i}(c)\right\}, \; c = 1, 2, \ldots, C \quad \text{(A8)}$$

The ranking of the channels across all B subjects is determined by first summing the ranks to obtain the rank-sum

$$(RS)_m(c) = \sum_{i=1}^{S} R_{mb_i}(c) \quad \text{(A9)}$$

and the final ranking of the channels across the U ERP classes and B subjects is determined from the rank of the rank-sum given by

$$R_m(c) = Rank\{(RS)_m(c)\}. \quad \text{(A10)}$$

The entire sequence of steps can be summarized as follows:

$$\left\{Rank\left\{\begin{matrix}Sum\ across\\ subjects\end{matrix}\left\{Rank\left\{\begin{matrix}Sum\ across\\ class\ pairs\end{matrix}\left\{Rank\left\{\rho_{(x,y)mb_i}(c)\right\}\right\}\right\}\right\}\right\}\right\} \quad \text{(A11)}$$

For the special dichotomous (2-class) case, the ranking of the channels for each subject is determined directly from $R_{mb_i}(c)$ which is the ranking of $\rho_{mb_i}(c)$, the interclass separation of the two classes x and y. The ranking across all subjects is obtained from the rank-sum $(RS)_m(c)$ of channel c, which is given by

$$(RS)_m(c) = \sum_{i=1}^{S} R_{mb_i}(c) \quad \text{(A12)}$$

and the final ranking of channel c is given by the rank-of-the rank-sum

$$R_m(c) = Rank\{(RS)_m(c)\}. \quad \text{(A13)}$$

A final comment regarding the application of the RRS algorithm is that the channel rankings are determined by applying the RRS algorithm to the ERPs of only the training set.

Appendix B

Table A1. Classification accuracies of the ERPs of Subject b_1. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	RF	KNN	MLP	CNN-1	CNN-2
P3	1	59.09	77.36	53.77	73.58	76.42	83.74
	2	63.64	83.02	60.38	82.26	83.02	**94.32**
	3	72.73	89.77	69.81	87.74	90.57	—
P1	1	60.91	73.58	56.60	78.49	77.36	81.13
	2	66.36	81.51	50.94	80.57	80.19	92.05
	3	71.82	86.36	63.21	90.57	**92.62**	—
P5	1	58.18	76.42	58.49	71.51	76.04	79.14
	2	59.43	84.91	61.32	72.45	84.91	90.94
	3	69.81	87.74	68.11	86.79	**91.38**	—
Pz	1	57.27	74.91	59.43	73.40	81.89	80.38
	2	56.60	80.38	62.26	76.23	86.79	91.51
	3	66.04	88.64	71.70	84.91	**92.46**	—
P7	1	60.00	76.42	57.55	67.92	75.09	82.45
	2	65.09	85.08	65.66	72.83	82.26	**93.40**
	3	70.81	90.91	73.77	87.36	90.09	—
POz	1	60.91	79.25	54.72	71.89	83.10	84.72
	2	66.04	84.15	59.06	74.34	87.36	92.22
	3	76.23	90.57	66.04	85.85	**94.34**	—
PO2	1	56.36	78.87	53.02	76.42	80.57	81.70
	2	57.55	79.62	60.38	80.06	84.90	94.34
	3	64.16	88.68	68.49	90.38	**95.00**	—
PO1	1	54.55	74.34	50.94	77.36	79.43	79.81
	2	61.32	80.38	56.60	80.38	85.09	**92.45**
	3	71.70	85.85	67.92	88.68	91.89	—

Table A2. Classification accuracies of the ERPs of Subject b_2. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	RF	KNN	MLP	CNN-1	CNN-2
PO4	1	55.46	72.73	64.14	72.76	73.47	80.53
	2	60.00	81.82	65.86	83.62	83.67	91.68
	3	72.73	88.18	78.45	90.52	**91.86**	—
F3	1	60.91	75.00	68.79	77.86	76.53	84.21
	2	64.55	82.55	75.35	76.07	81.63	**93.69**
	3	76.36	90.91	79.10	88.95	91.84	—
CP5	1	54.55	81.83	65.52	75.52	75.00	79.31
	2	66.36	85.45	71.55	81.03	79.59	91.58
	3	70.00	**93.64**	82.83	88.08	90.89	—
PO2	1	63.64	72.76	60.52	77.79	79.59	76.21
	2	68.18	79.31	66.38	83.91	83.33	90.42
	3	78.18	86.36	79.83	90.91	**91.67**	—
Oz	1	59.09	73.64	61.21	74.83	76.95	81.52
	2	61.82	85.45	65.52	85.69	88.56	**93.89**
	3	71.82	91.82	77.59	90.51	93.63	—
PO6	1	50.91	80.52	66.38	84.48	77.10	81.13
	2	59.09	85.35	66.67	89.66	83.73	92.03
	3	67.27	91.38	75.35	95.52	**92.75**	—
TP7	1	60.91	75.86	64.66	79.31	73.33	78.64
	2	65.55	82.21	70.69	80.07	79.24	90.86
	3	72.73	90.24	79.14	88.79	**90.84**	—
O1	1	58.72	78.79	63.79	77.41	77.79	83.10
	2	62.93	81.72	70.69	76.72	86.36	92.76
	3	73.07	90.52	74.75	87.27	**94.67**	—

Table A3. Classification accuracies of the ERPs of Subject b_3. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	**RF**	**KNN**	**MLP**	**CNN-1**	**CNN-2**
O2	1	61.21	72.41	68.10	79.32	78.62	82.66
	2	68.10	79.31	68.97	82.76	81.72	92.69
	3	75.00	92.24	75.52	91.38	**93.10**	—
Oz	1	68.97	72.06	62.93	78.97	71.72	80.21
	2	67.59	86.55	64.66	87.07	72.41	90.83
	3	78.10	**92.68**	72.41	92.83	88.28	—
O1	1	63.28	82.77	62.07	69.83	75.00	81.79
	2	74.14	89.10	65.52	86.21	84.83	94.41
	3	76.72	93.14	75.86	91.38	**95.69**	—
PO6	1	67.24	75.00	64.14	75.86	75.86	82.76
	2	74.14	84.83	62.07	84.48	82.76	**95.69**
	3	78.45	90.41	68.10	92.24	94.83	—
PO4	1	68.10	71.72	61.21	82.76	80.69	80.17
	2	68.97	83.97	65.52	89.66	89.66	90.86
	3	79.48	89.41	71.55	95.52	**96.55**	—
PO2	1	62.07	73.47	61.72	75.17	81.90	78.45
	2	73.28	89.66	65.17	81.72	86.21	89.66
	3	77.55	**94.83**	74.14	90.52	92.24	—
TP7	1	64.66	75.86	67.24	78.62	79.31	79.32
	2	67.24	82.76	69.83	84.83	84.14	93.97
	3	72.41	90.10	77.59	93.97	**97.41**	—
POz	1	68.97	76.55	60.34	80.69	72.07	74.14
	2	62.07	80.34	65.52	83.79	81.04	**92.24**
	3	75.86	89.21	74.14	91.14	90.52	—

Table A4. Classification accuracies of the ERPs of Subject b_4. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	**RF**	**KNN**	**MLP**	**CNN-1**	**CNN-2**
O2	1	57.55	73.58	52.83	74.15	76.42	81.89
	2	60.75	80.38	58.49	80.19	83.02	**93.77**
	3	69.81	87.74	64.15	88.68	92.51	—
Oz	1	54.72	76.23	52.45	77.36	79.81	80.75
	2	58.46	84.91	59.43	82.08	84.91	92.83
	3	66.98	89.62	63.77	90.56	**93.38**	—
TP7	1	51.89	70.57	56.60	69.81	76.98	80.19
	2	57.36	77.36	61.32	76.42	82.83	**94.34**
	3	63.21	85.85	68.30	83.96	92.89	—
O1	1	50.94	75.66	56.23	77.92	78.30	80.00
	2	53.20	79.06	55.85	80.94	86.79	92.66
	3	63.21	86.79	67.17	89.62	**93.58**	—
F3	1	56.60	79.25	60.38	75.47	81.13	83.96
	2	61.13	83.02	62.26	82.08	87.74	90.23
	3	70.38	89.43	66.04	90.94	**93.77**	—
F1	1	54.72	76.79	51.51	78.11	80.57	82.45
	2	62.26	83.02	54.53	83.92	85.85	95.09
	3	70.19	90.57	65.09	90.19	**96.70**	—
F5	1	58.49	74.34	59.06	70.76	74.53	80.94
	2	60.38	80.19	62.26	79.62	80.02	**93.58**
	3	71.70	88.68	73.58	86.79	92.75	—
CP5	1	53.77	76.42	58.49	77.17	79.25	81.02
	2	53.21	80.94	62.08	83.02	83.40	90.36
	3	64.15	87.74	69.81	89.06	**92.45**	—

Table A5. Classification accuracies of the ERPs of Subject b_5. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	**RF**	**KNN**	**MLP**	**CNN-1**	**CNN-2**
C2	1	51.89	75.51	51.02	76.73	77.55	81.25
	2	57.55	79.17	59.18	84.69	84.38	**92.71**
	3	64.15	87.50	60.20	89.80	92.63	—
P5	1	52.83	77.55	57.14	79.59	74.49	80.61
	2	58.33	81.63	63.27	80.41	83.47	91.67
	3	65.31	88.54	68.37	87.76	**92.76**	—
Cz	1	58.16	69.80	53.06	74.49	79.59	81.63
	2	61.46	71.63	61.22	78.57	85.71	**92.86**
	3	69.39	86.46	67.35	86.94	91.84	—
C4	1	56.25	74.29	55.31	75.51	79.69	83.68
	2	60.42	79.59	63.64	81.86	84.69	92.92
	3	69.17	85.42	66.12	88.98	**93.88**	—
P3	1	54.79	71.43	58.17	72.45	81.63	83.33
	2	56.88	78.16	64.49	79.39	86.74	**93.88**
	3	70.83	84.69	68.37	85.71	92.45	—
P7	1	56.67	73.47	62.22	76.53	81.08	82.45
	2	58.96	79.39	62.04	80.21	88.78	93.90
	3	68.55	87.76	67.35	89.80	**94.83**	—
C1	1	53.06	74.08	55.10	72.45	75.00	81.63
	2	59.38	77.55	57.14	81.22	79.80	92.86
	3	66.67	85.71	68.98	87.76	**93.20**	—
FCz	1	54.82	71.02	46.94	73.47	76.33	84.08
	2	59.80	74.29	53.06	79.59	84.69	92.22
	3	65.31	86.74	64.29	85.31	**93.82**	—

Table A6. Single trial classification accuracies of the 1-ERPs of Subject b_3. Inconsistent results are highlighted in red. The best result for each channel is highlighted in bold font.

Channel	Approach	Classifier					
		SVM	**RF**	**KNN**	**MLP**	**CNN-1**	**CNN-2**
O2	1	44.64	52.73	40.77	54.46	55.10	63.47
	2	50.21	58.16	47.64	57.14	59.23	76.73
	3	55.10	63.54	50.86	65.63	**82.53**	—
Oz	1	45.92	51.38	40.19	53.64	58.16	63.85
	2	47.00	53.10	44.83	57.27	65.31	78.54
	3	54.90	61.40	50.64	66.36	**83.62**	—
O1	1	44.83	52.76	40.38	54.14	52.79	67.08
	2	49.79	58.62	42.94	59.09	61.22	75.71
	3	55.17	63.79	50.43	68.42	**81.63**	—
PO6	1	42.42	49.06	43.40	54.72	51.07	61.88
	2	48.93	56.60	45.28	59.48	57.14	79.30
	3	53.27	60.38	50.09	67.24	**80.41**	—
PO4	1	42.31	47.17	40.56	55.66	54.72	61.23
	2	41.53	54.63	40.40	59.43	63.83	**80.61**
	3	53.96	61.32	51.89	66.57	77.55	—
PO2	1	42.59	44.45	40.74	52.24	50.86	69.79
	2	43.08	50.96	43.70	55.66	64.29	**80.18**
	3	54.72	57.55	51.13	64.15	79.80	—
TP7	1	45.28	51.85	40.94	51.85	55.32	61.43
	2	48.15	51.67	48.15	56.44	65.96	78.82
	3	56.60	62.26	52.83	63.77	**80.83**	—
POz	1	43.43	50.00	40.03	50.56	51.50	66.15
	2	49.23	57.41	47.96	59.26	63.86	72.73
	3	54.34	60.18	54.19	62.45	**79.59**	—

References

1. Stephen, M. Singularity detection and processing with wavelets. *IEEE Trans Inf. Theory* **1992**, *38*, 617–643.
2. Torrence, C.; Compo, G.P. A practical guide to wavelet analysis. *Bull. Am. Meteorol. Soc.* **1998**, *79*, 61–78. [CrossRef]
3. Nobach, H.; Tropea, C.; Cordier, L.; Bonnet, J.P.; Delville, J.; Lewalle, J.; Farge, M.; Schneider, K.; Adrian, R. Review of some fundamentals of data processing. In *Springer Handbooks*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 1337–1398.
4. Lilly, J.M. Element analysis: A wavelet-based method for analysing time-localized events in noisy time series. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **2017**, *473*, 20160776. [CrossRef]
5. Wei, X.; Zhou, L.; Chen, Z.; Zhang, L.; Zhou, Y. Automatic seizure detection using three-dimensional CNN based on multi-channel EEG. *BMC Med. Inform. Decis. Mak.* **2018**, *18*, 71–80. [CrossRef]
6. Türk, Ö.; Özerdem, M.S. Epilepsy detection by using scalogram based convolutional neural network from EEG signals. *Brain Sci.* **2019**, *9*, 115. [CrossRef]
7. Lee, H.K.; Choi, Y.S. Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface. *Entropy* **2019**, *21*, 1199. [CrossRef]
8. Mao, W.; Fathurrahman, H.; Lee, Y.; Chang, T. EEG dataset classification using CNN method. In *J. phys. Conf. Ser.*; 2020; Volume 1456, p. 012017.
9. Mammone, N.; Ieracitano, C.; Morabito, F.C. A deep CNN approach to decode motor preparation of upper limbs from time–frequency maps of EEG signals at source level. *Neural Netw.* **2020**, *124*, 357–372. [CrossRef]
10. Aslan, Z.; Akin, M. A deep learning approach in automated detection of schizophrenia using scalogram images of EEG signals. *Phys. Eng. Sci. Med.* **2022**, *45*, 83–96. [CrossRef]
11. Kaur, A.; Shashvat, K. Implementation of convolution neural network using scalogram for identification of epileptic activity. *Chaos Solitons Fractals* **2022**, *162*, 112528. [CrossRef]
12. Kumar, J.L.M.; Rashid, M.; Musa, R.M.; Razman, M.A.M.; Sulaiman, N.; Jailani, R.; Majeed, A.P.A. The classification of EEG-based wink signals: A CWT-transfer learning pipeline. *ICT Express* **2021**, *7*, 421–425. [CrossRef]
13. Buriro, A.B.; Ahmed, B.; Baloch, G.; Ahmed, J.; Shoorangiz, R.; Weddell, S.J.; Jones, R.D. Classification of alcoholic EEG signals using wavelet scattering transform-based features. *Comput. Biol. Med.* **2021**, *139*, 104969. [CrossRef]
14. Kant, P.; Hazarika, J.; Laskar, S. Wavelet transform based approach for EEG feature selection of motor imagery data for braincomputer interfaces. In Proceedings of the 2019 Third International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 10–11 January 2019; pp. 101–105.
15. Yasoda, K.; Ponmagal, R.; Bhuvaneshwari, K.; Venkatachalam, K. Automatic detection and classification of EEG artifacts using fuzzy kernel SVM and wavelet ICA (WICA). *Soft Comput.* **2020**, *24*, 16011–16019. [CrossRef]
16. Kumar, J.L.M.; Rashid, M.; Musa, R.M.; Razman, M.A.M.; Sulaiman, N.; Jailani, R.; Majeed, A.P.A. The classification of EEG-based winking signals: A transfer learning and random forest pipeline. *PeerJ* **2021**, *9*, e11182. [CrossRef] [PubMed]
17. Fraiwan, L.; Lweesy, K.; Khasawneh, N.; Wenz, H.; Dickhaus, H. Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier. *Comput. Methods Programs Biomed.* **2012**, *108*, 10–19. [CrossRef] [PubMed]
18. Altameem, A.; Sachdev, J.S.; Singh, V.; Poonia, R.C.; Kumar, S.; Saudagar, A.K.J. Performance Analysis of Machine Learning Algorithms for Classifying Hand Motion-Based EEG Brain Signals. *Comput. Syst. Sci. Eng.* **2022**, *42*, 1095–1107. [CrossRef]
19. Ieracitano, C.; Mammone, N.; Hussain, A.; Morabito, F.C. A novel multi-modal machine learning based approach for automatic classification of EEG recordings in dementia. *Neural Netw.* **2020**, *123*, 176–190. [CrossRef] [PubMed]
20. Light, G.A.; Williams, L.E.; Minow, F.; Sprock, J.; Rissling, A.; Sharp, R.; Swerdlow, N.R.; Braff, D.L. Electroencephalograph (EEG) and event-related potentials (ERPs) with human participants. *Curr. Protoc. Neurosci.* **2010**, *52*, 6–25. [CrossRef]
21. Kappenman, E.S.; Luck, S.J. Best practices for event-related potential research in clinical populations. *Biol. Psychiatry Cogn. Neurosci. Neuroimaging* **2016**, *1*, 110–115. [CrossRef]
22. Luck, S.J. *An Introduction to the Event-Related Potential Technique*; MIT Press: Cambridge, MA, USA, 2014.
23. Phadikar, S.; Sinha, N.; Ghosh, R.; Ghaderpour, E. Automatic Muscle Artifacts Identification and Removal from Single-Channe EEG Using Wavelet Transform with Meta-Heuristically Optimized Non-Local Means Filter. *Sensors* **2022**, *22*, 2948. [CrossRef]
24. Celka, P.; Le, K.N.; Cutmore, T.R. Noise reduction in rhythmic and multitrial biosignals with applications to event-related potentials. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 1809–1821. [CrossRef]
25. Ahmed, M.Z.I.; Sinha, N.; Phadikar, S.; Ghaderpour, E. Automated Feature Extraction on AsMap for Emotion Classification Using EEG. *Sensors* **2022**, *22*, 2346. [CrossRef] [PubMed]
26. Follis, J.L.; Lai, D. Modeling Volatility Characteristics of Epileptic EEGs using GARCH Models. *Signals* **2020**, *1*, 26–46. [CrossRef]
27. Montanari, L.; Basu, B.; Spagnoli, A.; Broderick, B.M. A padding method to reduce edge effects for enhanced damage identification using wavelet analysis. *Mech. Syst. Signal Process.* **2015**, *52*, 264–277. [CrossRef]
28. Su, H.; Liu, Q.; Li, J. Boundary effects reduction in wavelet transform for time-frequency analysis. *Wseas Trans. Signal Process.* **2012**, *8*, 169–179.
29. Kharitonenko, I.; Zhang, X.; Twelves, S. A wavelet transform with point-symmetric extension at tile boundaries. *IEEE Trans. Image Process.* **2002**, *11*, 1357–1364. [CrossRef]

30. Zhu, Y.y.; Man, Z.l.; Pei, W.; Wang, J. Research of a boundary prolongation method in runoff forecast based on wavelet transform. In Proceedings of the 2009 IEEE International Conference on Automation and Logistics, Shenyang, China, 5–7 August 2009; pp. 1254–1258.
31. Silva, V.; De Sa, L. General method for perfect reconstruction subband processing of finite length signals using linear extensions. *IEEE Trans. Signal Process.* **1999**, *47*, 2572–2575. [CrossRef]
32. Pacola, E.; Quandt, V.; Schneider, F.; Sovierzoski, M. The wavelet transform border effect in EEG spike signals. In Proceedings of the World Congress on Medical Physics and Biomedical Engineering, Beijing, China, 26–31 May 2012; Springer: Berlin/Heidelberg, Germany, 2013; pp. 593–596.
33. Unser, M. A practical guide to the implementation of the wavelet transform. In *Wavelets in Medicine and Biology*; Routledge: London, UK, 2017; pp. 37–74.
34. Asman, S.H.; Abidin, A.F. Comparative Study of Extension Mode Method in Reducing Border Distortion Effect for Transient Voltage Disturbance. Indones. *J. Electr. Eng. Comput. Sci* **2017**, *6*, 628.
35. Ghaderpour, E.; Pagiatakis, S.D. Least-squares wavelet analysis of unequally spaced and non-stationary time series and its applications. *Math. Geosci.* **2017**, *49*, 819–844. [CrossRef]
36. Ideião, S.M.A.; Santos, C.A.G. Analysis of precipitation time series using the wavelet transform. *Soc. Nat.* **2005**, *1*, 736–745. [CrossRef]
37. Lee, W.S.; Kassim, A.A. Signal and image approximation using interval wavelet transform. *IEEE Trans. Image Process.* **2006**, *16*, 46–56. [CrossRef]
38. De Moortel, I.; Munday, S.; Hood, A.W. Wavelet analysis: The effect of varying basic wavelet parameters. *Sol. Phys.* **2004**, *222*, 203–228. [CrossRef]
39. Li, Y.F. Image denoising based on undecimated discrete wavelet transform. In Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition, Beijing, China, 2–4 November 2007; Volome 2; pp. 527–531.
40. Dragotti, P.L.; Vetterli, M. Wavelet footprints: Theory, algorithms, and applications. *IEEE Trans. Signal Process.* **2003**, *51*, 1306–1323. [CrossRef]
41. Mota, H.d.O.; Vasconcelos, F.H.; de Castro, C.L. A comparison of cycle spinning versus stationary wavelet transform for the extraction of features of partial discharge signals. *IEEE Trans. Dielectr. Electr. Insul.* **2016**, *23*, 1106–1118. [CrossRef]
42. Liu, Y.; San Liang, X.; Weisberg, R.H. Rectification of the bias in the wavelet power spectrum. *J. Atmos. Ocean. Technol.* **2007**, *24*, 2093–2102. [CrossRef]
43. Cazelles, B.; Chavez, M.; Berteaux, D.; Ménard, F.; Vik, J.O.; Jenouvrier, S.; Stenseth, N.C. Wavelet analysis of ecological time series. *Oecologia* **2008**, *156*, 287–304. [CrossRef]
44. Lilly, J.M.; Olhede, S.C. Generalized Morse wavelets as a superfamily of analytic wavelets. *IEEE Trans. Signal Process.* **2012**, *60*, 6036–6041. [CrossRef]
45. Gupta, L.; Chung, B.; Srinath, M.D.; Molfese, D.L.; Kook, H. Multichannel fusion models for the parametric classification of differential brain activity. *IEEE Trans. Biomed. Eng.* **2005**, *52*, 1869–1881. [CrossRef] [PubMed]
46. Kota, S.; Gupta, L.; Molfese, D.; Vaidyanathan, R. Diversity-Based Selection of Polychotomous Components for Multi-Sensor Fusion Classifiers. *J. Eng. Med.* **2013**, *227*, 655–662.
47. Hart, P.E.; Stork, D.G.; Duda, R.O. *Pattern classification*; John Wiley Sons: Hoboken, NJ, USA, 2006.
48. Woody, C.D. Characterization of an adaptive filter for the analysis of variable latency neuroelectric signals. *Med. Biol. Eng.* **1967**, *5*, 539–554. [CrossRef]
49. Aunon, J.I.; McGillem, C.D.; Childers, D.G. Signal processing in evoked potential research: Averaging and modeling. *Crit. Rev. Bioeng.* **1981**, *5*, 323–367.
50. Gupta, L.; Molfese, D.L.; Tammana, R.; Simos, P.G. Nonlinear alignment and averaging for estimating the evoked potential. *IEEE Trans. Biomed. Eng.* **1996**, *43*, 348–356. [CrossRef] [PubMed]
51. Gupta, L.; Phegley, J.; Molfese, D.L. Parametric classification of multichannel averaged event-related potentials. *IEEE Trans. Biomed. Eng.* **2002**, *49*, 905–911. [CrossRef]
52. Kota, S.; Gupta, L.; Molfese, D.L.; Vaidyanathan, R. A dynamic channel selection strategy for dense-array ERP classification. *IEEE Trans. Biomed. Eng.* **2008**, *56*, 1040–1051. [CrossRef] [PubMed]
53. Dien, J.; Spencer, K.M.; Donchin, E. Parsing the late positive complex: Mental chronometry and the ERP components that inhabit the neighborhood of the P300. *Psychophysiology* **2004**, *41*, 665–678. [CrossRef] [PubMed]
54. Gupta, R.S.; Kujawa, A.; Vago, D.R. A preliminary investigation of ERP components of attentional bias in anxious adults using temporospatial principal component analysis. *J. Psychophysiol.* **2021**, *35*, 223–236. [CrossRef]
55. Alotaiby, T.; Abd El-Samie, F.E.; Alshebeili, S.A.; Ahmad, I. A review of channel selection algorithms for EEG signal processing. *EURASIP J. Adv. Signal Process.* **2015**, *2015*, 66. [CrossRef]
56. Baig, M.Z.; Aslam, N.; Shum, H.P. Filtering techniques for channel selection in motor imagery EEG applications: A survey. *Artif. Intell. Rev.* **2020**, *53*, 1207–1232. [CrossRef]
57. Guttmann-Flury, E.; Sheng, X.; Zhu, X. Channel selection from source localization: A review of four EEG-based brain–computer interfaces paradigms. *Behav. Res. Methods* **2022**, *2022*, 1–24.
58. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 4.

59. Gareth, J.; Daniela, W.; Trevor, H.; Robert, T. *An Introduction to Statistical Learning: With Applications in R*; Spinger: Berlin/Heidelberg, Germany, 2013.
60. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef]
61. Chollet, F. *Deep Learning with Python*; Simon and Schuster: New York, NY, USA, 2021.
62. Murphy, K. *Probabilistic Machine Learning: Advanced Topics*; MIT Press: Cambridge, MA, USA, 2022.
63. Auger, F.; Flandrin, P.; Gonçalvès, P.; Lemoine, O. *Time-Frequency Toolbox*; CNRS: Paris, France; Rice University: Houston, TX, USA, 1996; p. 46.
64. Cohen, M.X. A better way to define and describe Morlet wavelets for time-frequency analysis. *NeuroImage* **2019**, *199*, 81–86. [CrossRef]
65. Unser, M.; Aldroubi, A. A review of wavelets in biomedical applications. *Proc. IEEE* **1996**, *84*, 626–638. [CrossRef]
66. Daubechies, I. *Ten Lectures on Wavelets*; SIAM: New Delhi, India, 1992.
67. Amerineni, R.; Gupta, L.; Steadman, N.; Annauth, K.; Burr, C.; Wilson, S.; Barnaghi, P.; Vaidyanathan, R. Fusion Models for Generalized Classification of Multi-Axial Human Movement: Validation in Sport Performance. *Sensors* **2021**, *21*, 8409. [CrossRef] [PubMed]
68. Mallat, S.G. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **1989**, *11*, 674–693. [CrossRef]
69. Taswell, C.; McGill, K.C. Algorithm 735: Wavelet transform algorithms for finite-duration discrete-time signals. *ACM Trans. Math. Softw.* **1994**, *20*, 398–412. [CrossRef]
70. Oppenheim, A.V. *Discrete-Time Signal Processing*; Pearson Education India: Bengaluru, India, 1999.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.