Handling few training data: classifier transfer between different types of error-related potentials

Su Kyoung Kim, *Member, IEEE* and Elsa Andrea Kirchner

***Abstract*—This paper proposes an application oriented approach that enables to transfer a classifier trained within an experimental scenario into a more complex application scenario or a specific rehabilitation situation which do not allow to collect sufficient training data within a reasonable amount of time. The proposed transfer approach is not limited to be applied to the same type of event-related potential. We show that a classifier trained to detect a certain brain pattern can be used successfully to detect another brain pattern, which is expected to be similar to the first one. In particular a classifier is transferred between two *different* types of error-related potentials (ErrPs) within the same subject. The classifier trained on observation ErrPs is used to detect interaction ErrPs, since twice as much training data is collected for observation ErrPs compared to interaction ErrPs during the *same* calibration time. Our results show that the proposed transfer approach is feasible and outperforms another approach, in which a classifier is transferred between different subjects but the *same* type of ErrP is used to train and test the classifier. The proposed approach is a promising way to handle few training data and to reduce calibration time in ErrP-based brain-computer interfaces.**

***Index Terms*—human-machine interaction, brain-computer interfaces, electroencephalogram, interaction errors, observation errors, error-related potentials, single-trial detection, classifier transfer.**

I. Introduction

BRAIN-computer interfaces (BCIs) link [1], a user and an external system by detecting a specific brain activity correlated with the user's intent (e.g., movement intention/planning [2]–[7]), mental states or cognitive processes [8]–[12]), which is measured by, e.g., electroencephalography (EEG).

EEG-based BCIs have been developed in different application contexts (for reviews, see [13], [14]), e.g., for enabling users to communicate with the external world (P300-based BCIs for review, see [15] or visual-evoked potential (VEP)-based BCIs [16], [17]), for assisting users's motor functions in a daily life environment and in rehabilitation respectively [18]–[23], or for other applications, like simulated driver support [24] or in robotics [25]–[28].

A core ability of adaptive systems (e.g., autonomous robots) is self-monitoring of their own performance to automatically self-correct erroneous behavior. Learning models used for such self-adaptation of system's behavior can be improved by using external evaluations, e.g., using the so called error-related potentials (ErrPs) measured on a human evaluator. In recent years, ErrPs have been used to improve the system's performance by correcting the errors within the system itself or in the interfaces that link human and machines [26], [29]–[43].

Such improvement of system performance can be realized by single-trial detection of ErrPs. Two types of ErrPs, among others, have widely been used to adapt systems: a) interaction ErrPs and b) observation ErrPs (for review of ErrP-based BCIs, see [44]). Interaction ErrPs have been used in cases that the interface fails to interpret the user's intent and delivers a wrong command to an external device. Such a failure of an interface (i.e., interaction errors) elicits a specific brain activity called interaction ErrPs [31], [33]. Thus, the detection of interaction ErrPs has been used as a verification tool for other BCI-systems such as P300-based BCIs [37]–[40] or in VEP-based BCIs [41]. In robotic applications, the behavior of a robot has been adapted with respect to the context of situations by single-trial detection of observation ErrPs, which are elicited in an observer's EEG who monitors the robot's erroneous behavior [26].

However, real-world applications using ErrPs are challenging for different reasons. First, in general erroneous behavior does not often occur in real-world applications. This leads to a long recording time to obtain enough training data. Second, the daily use of BCIs in real-world environments (e.g., in specific rehabilitation situations) is often limited with respect to recording time. For example, it is not always possible for patients to train an interface for a long time, since their health condition may change.

One possibility to handle few training data in a real-world scenario is to develop an experimental scenario which enables to collect enough training data and then transfer the classifier trained within the developed scenario to a real-word application scenario.

In general, such classifier transfer between scenarios can reduce the recording time during data collection (i.e., calibration time). However, the scenario used to train a classifier does not always allow to collect a sufficient amount of training instances within a reasonable time. We propose that in such cases, a classifier transfer between ERP types which are similar to each other can reduce the calibration time, when we can collect considerably more training instances for one

S.K. Kim (corresponding author) is with the Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI) GmbH, Bremen, Germany. e-mail: su-kyoung.kim@dfki.de

E.A. Kirchner is with the Faculty of Mathematics and Computer Science, University of Bremen & Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI) GmbH, Bremen, Germany. e-mail: elsa.kirchner@dfki.de

This work is supported by the German Ministry of Economics and Technology (grant no. 50 RA 1011 and grant no. 50 RA 1012).

Manuscript received July 30, 2015; revised February 28, 2015, July 3, 2015; accepted November 4, 2015.

type of ERP compared to the other type of ERP during the same recording time.

In this study, we propose an application oriented approach to handle few training data and to reduce the calibration time by transferring a classifier between different types of ErrPs. To test the applicability of such a transfer approach, we detect two different types of ErrPs. The idea of the proposed approach is to use a classifier trained on one type of ErrP and to test on another type of ErrP in case that considerably more training data is obtained for the former type of ErrP compared to the latter during the *same* time of data collection. Such classifier transfer allows us to reduce the calibration time, which is needed for a user-specific calibration of a BCI. In recent studies [42], [43], the feasibility of classifier transfer between different tasks has been investigated within the same type of ErrP. In both studies, the same mental task was performed to elicit observation ErrPs (i.e., subjects observe the behavior of a cursor or a robotic arm). However, the type of stimulus (e.g, different types of cursor presentations or robotic arm instead of cursor) was differently presented with the goal of varying the cognitive workload. Such differences resulted in differences in temporal features of the observation ErrPs (such as differences in latency) between different tasks. To our knowledge, there is no study on the transfer of a classifier trained on one type of ErrP and tested on another type of ErrP in order to reduce the calibration time.

In the presented approach, an error monitoring task is performed in two different ways. Errors are monitored during the interaction with an external device (called *interaction* errors) or while observing the operation of an artificial agent (called *observation* errors). The corresponding ErrPs elicited in the two different application contexts (*interaction* ErrPs/*observation* ErrPs) are detected by using signal processing and machine-learning techniques. In our application, we obtain more training instances in the observation task compared to the interaction task during the same recording time. For this reason, the classifier developed for detecting the observation ErrPs is used to detect the interaction ErrPs. The concept of the proposed approach was tested with few subjects in a previous study [45]. In this paper, we test our approach on more subjects. In addition, the proposed approach is compared to another approach used to reduce calibration time, in which a classifier is transferred between different subjects but the *same* type of ErrPs is used to train and test the classifier. We evaluate the applicability of two transfer approaches by comparing them to the baseline (i.e., no transfer case).

This paper presents experimental results from eight subjects during the monitoring of two different types of errors (interaction and observation errors). The main findings are structured in five parts: 1) single-trial detection of ErrPs of each type of ErrP (no transfer case), 2) investigation of a time window of interest to extract features for the proposed transfer approach, 3) single-trial detection of ErrPs in case of classifier transfer between different types of ErrPs, 4) single-trial detection of ErrPs in case of classifier transfer between different subjects within the same type of ErrPs, 5) comparison between the three cases: no transfer case, transfer case at ErrP level, and transfer case at subject level.

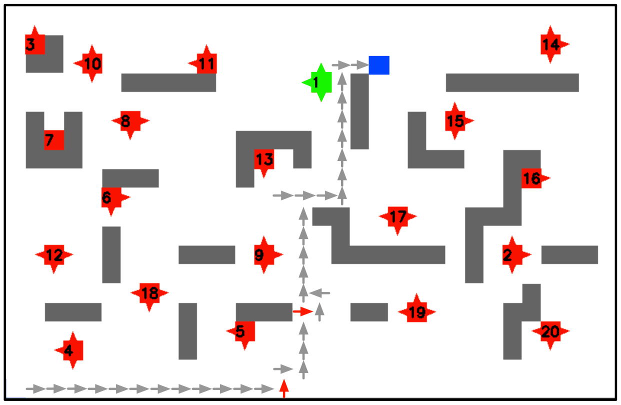


Fig. 1. Experimental paradigm used for different application contexts: 1) interaction task: subjects were instructed to bring the cursor (blue) to one of the 20 targets (red) placed among the obstacles (gray objects) in numeric order; 2) observation task: an artificial agent controls the movements of the cursor to check all targets. Task rules were the same as for the interaction task. Subjects observed the behaviors of the artificial agent. The track of cursor movements is depicted by gray arrows towards the chosen direction and the track of wrong movements of cursor is depicted by red arrows (direction of errors). The depicted paradigm is an example of the interaction task.

II. Approach

A. Experimental Scenario

We developed a scenario based on a similar principle of the scenario developed in [33], in which interaction ErrPs were detected during the monitoring of interaction errors that were simulated with a certain probability. Compared to their approach, our scenario was designed to allow us to detect different types of ErrPs (interaction ErrPs/observation ErrPs) depending on which task (interaction task/observation task) was performed (see Fig. 1).

1) Interaction task to detect interaction ErrPs: The task was to reach all 20 targets (red) by moving the cursor (blue) (see Fig. 1). All stimuli (cursor and targets) were displayed on a monitor placed in front of the subject. To move the cursor the subjects used four arrow keys of a computer keyboard (left, right, up or down). Here, the subjects performed self paced key pressings to move the cursor to obtain a realistic character of the scenario (i.e. natural interactions between subjects and device). Thus, the interval between events was not predetermined. Each cursor movement corresponded to one type of events (correct/erroneous). Thus, the number of cursor movements was equal to the number of events. The track of cursor movements is depicted by gray arrows towards the chosen direction.

The subjects had to perform the task according to specific rules. First, the targets have to be reached in numeric order. Each target has a semantics (labeled number) based on how the order of targets is defined. Second, the obstacles placed on the way to the targets and spikes of the targets have to be avoided when finding a way to reach the targets. In case of checking a target point in the correct order the color of the targets is changed from red to green. Due to the spikes all targets can only be reached from one side of a target. In case of touching a spike the cursor go back to the start position as

TABLE I
OVERVIEW OF DATA SET AND EVALUATION PROCEDURE. HERE, WE DEFINED CALIBRATION TIME AS THE TIME WHICH WAS NEEDED TO RECORD TRAINING AND TEST DATA DURING PERFORMING THE TASK. EACH TASK TOOK ABOUT TWO MINUTES. IN OUR CASE, CALIBRATION TIME WAS THE SAME AS THE TIME WHICH WAS NEEDED TO FINISH THE TASK.

ErrP type used to train	ErrP type used to test	data set used to train and test	amount of instances (erroneous/correct)	calibration time	evaluation
interaction	interaction	four sets are merged	192/1920	8 min	10 fold-cross validation each subject separately
observation	observation	two sets are merged	198/1980	4 min	10 fold-cross validation each subject separately
observation	interaction	one set for each type of ErrP	train: 99/990 test: 48/480	2 min	train: observation / test: interaction each subject separately
interaction	interaction	one set from each subject	336/3360	0 min	leave-one-subject-out cross validation

a penalty. The task was finished after reaching all 20 targets in the correct order.

In this task, we expected two kinds of erroneous behaviors. First, erroneous behavior made by the subjects themselves (i.e., *response errors* [46]) could occur due to the realistic character of this task (e.g., semantics of targets, obstacles, and spikes of targets). Second, erroneous actions of the system (i.e., wrong movements of the cursor) could occur since we simulated wrong movements of the cursor with a probability of 9% to generate *interaction errors*. Such simulated wrong movements did not correspond to the chosen key that was pressed by the subjects. The possible directions of wrong movements were uniformly distributed. Wrong movements left traces depicted as red arrows (direction of errors).

Hence, three different labels for the classification were generated: a) correct trial (*Corr*): cursor movements that corresponded to the pressed key (i.e., correct movements), b) erroneous trial type I (*InterErr*): cursor movements that did not correspond to the key pressing of subjects (i.e., simulated interface errors), and c) erroneous trial type II (*RespErr*): errors made by the subject (e.g., touching spikes of a target or violating the target order). In this study, we focused only on two labels: *Corr* and *InterErr*.

The task was repeated seven times and thus seven data sets were recorded for each subject. Each set contained about 48 erroneous trials and 480 correct trials. To avoid the same task pattern, the order of targets was randomized for each set. All subjects needed about 2 minutes to finish one set.

2) Observation task to detect observation ErrPs: Unlike in the interaction task, not the subject but an artificial agent performed the task. The task rules, the way of stimulus presentation, and the probability of simulated errors were the same as for the interaction task. Subjects were instructed to observe the behavior of the agent.

When observing the actions of the agent, it is not clear whether the actual cursor movement is erroneous or not, since there are more than one shortest way to the targets. For this reason, we constructed clear cases for erroneous events by hard coding the path to the targets and its deviations (errors). In this way, the subjects clearly recognize the wrong movements of the cursor without developing an own strategy to find the correct path. However, our hard coding of the path to targets led to suboptimal ways to the targets. We obtained 99 erroneous events for each set. The empirical ratio of erroneous and correct trials was 1:10 as for the interaction task. The speed of key pressings was also hard coded. Since subjects paused quite often to find the correct path, their average movement speed was slower compared to the agents speed. Thus, we obtained more erroneous trials compared to the interaction task within the same time of data collection.

Here, we expected one type of error, i.e., the erroneous behavior committed by the agent. Such errors could be recognized by the movements which deviated from the correct path to reach the targets. The track of wrong movements of the cursor is depicted by red arrows (directions of errors). Accordingly, the subjects could recognize wrong movements of the cursor without developing and executing a strategy to find the correct path.

Hence, two different labels for classification were generated: a) correct trial (*Corr*): movements did not deviate from the path to reach targets (i.e., correct movements) and b) erroneous trial (*ObsErr*): movements deviated from the path to reach targets (i.e., wrong movements).

Again seven data sets (each took 2 minutes) were collected as for the interaction task. Each set contained about 99 erroneous trials and 990 correct trials. To avoid the same task pattern, the target order was randomized for each set.

B. Transfer Approaches

In the following we explain the proposed and a comparative transfer approaches which both allow to reduce calibration time.

1) Classifier transfer between different types of ErrPs: Using the developed scenario we could collect twice as much training instances in the observation task compared to the interaction task during the same recording time per set. Hence, from the perspective of the application it is practical to use the data collected from the observation task to train the classifier for detecting the interaction ErrPs in the application case. Such reuse of a classifier allows us to reduce the calibration time. Also in general an investigation of such classifier transfer is of interest since the application contexts can be changed due to different application environments in a daily application of an interface. Here, we used data collected during the observation task (i.e., data containing observation ErrPs) to train the classifier. The trained classifier was then used to test data collected during the interaction task (i.e., data containing interaction ErrPs).

2) Classifier transfer between different subjects within the same type of ErrPs: The calibration time can also be reduced by reusing of the classifier trained on data that was already obtained from another subject for the same type of ErrP. In this case, no explicit recording of training data from the current subject is needed. However, the scenario used to train a classifier is also used to test a classifier. Hence this approach is beneficial, only in cases that the scenario enables to record a sufficient amount of data within a reasonable time. Otherwise, a reasonable number of subjects is needed to train a classifier on historic session.

III. Methods

A. Subjects

Eight subjects (two females, six males, age: 26.5 ± 3.25, right-handed, normal or corrected-to normal vision) participated in this study. Each subject provided written consent to participate in the study approved by the ethics committee of the University of Bremen. The study was conducted according to the Declaration of Helsinki.

B. Data Acquisition

EEGs were acquired for each participant during two experiments: a) interaction task and 2) observation task. Experiments were performed with a *counter-balanced* measures design. Subjects were divided into two groups: one group began with the observation task followed by the interaction task and vice versa. Since each task had seven sets, both tasks were performed alternately within a subject. EEGs were recorded using the actiCap system (Brain Products GmbH, Munich, Germany), in which 64 active electrodes were arranged in accordance to an extended 10-20 system with reference at FCz. Impedance was kept below $5\,\mathrm{k}\Omega$. EEG signals were sampled at $5\,\mathrm{kHz}$, amplified by two 32 channel Brain Amp DC Amplifiers (Brain Products GmbH, Munich, Germany), and filtered with a low cut-off of $0.1\,\mathrm{Hz}$ and high cut-off of $1\,\mathrm{kHz}$.

C. Analysis of Event Related Potential (ERP)

We computed averaged ERPs for each event (correct/erroneous) per subject. Here we used the same data sets which were used for singe-trial classification. For ERP peak analysis we used only eleven fronto-central channels: FC1, FC2, FC3, FC4, C1, C2, C3, C4, Fz, FCz, Cz. We measured maximum ERP peaks in the following time windows: a) early time window: first negativity in the time window of $0.2\,\mathrm{s}$–$0.3\,\mathrm{s}$ and positivity in the time window of $0.3\,\mathrm{s}$–$0.4\,\mathrm{s}$ and b) late time window: negativity in the time window of $0.4\,\mathrm{s}$–$0.8\,\mathrm{s}$.

In a descriptive manner (see Figure 3), we observed a similar ERP shape in the early time window $[0.2\,\mathrm{s}$–$0.4\,\mathrm{s}]$ for both ErrP types (i.e., similar shapes in first negativity around $0.27\,\mathrm{s}$ followed by positivity around $0.38\,\mathrm{s}$), whereas differences in ERP shapes between both ErrP types were observed in the late time window $[0.4\,\mathrm{s}$–$0.8\,\mathrm{s}]$. Thus, we divided the late window into two late sub-windows: $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$ and $0.6\,\mathrm{s}$–$0.8\,\mathrm{s}$. In summary, four time windows were used to measure ERP peaks: a) negativity in the time window of $0.2\,\mathrm{s}$–$0.3\,\mathrm{s}$, b) positivity in the time window of $0.3\,\mathrm{s}$–$0.4\,\mathrm{s}$, c) negativity in the time window of $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$, and d) negativity in the time window of $0.6\,\mathrm{s}$–$0.8\,\mathrm{s}$.

Peak amplitudes in the predefined time windows were analyzed using repeated measures ANOVA with two within-subjects factors: *ErrP type* and *time window*. We compared peak amplitudes between both ErrP types in the early time window: a) first negativity in interaction ErrPs vs. first negativity in observation ErrPs and b) positivity in interaction ErrPs vs. positivity in observation ErrPs. Furthermore, both late sub-windows were compared within each ErrP type, i.e., a maximum peak in the time window of $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$ was compared with a maximum peak in the time window of $0.6\,\mathrm{s}$–$0.8\,\mathrm{s}$ for each ErrP type. Where necessary, the Greenhouse-Geisser correction was applied and the corrected p-value is reported.

D. Data Set

Seven data sets were collected for each task (interaction ErrP/observation ErrP). As mentioned earlier a different amount of training data was collected for each task during the same recording time (i.e., a different number of training instances was recorded per set). Thus the recorded data sets were merged differently depending on the ErrP task to enable a fair comparison of detection performance of each type of ErrP. In this way, an approximately equal number of erroneous trials was used for the evaluation for each ErrP task. Note that the ratio of correct and erroneous trials (1:10) was the same for both tasks and we used the first four data sets for the interaction task and the first two data sets for the observation task. An overview of data preparation for the four different investigations can be seen in Table I.

1) Single-trial detection of interaction ErrP: Four data sets were merged into one data set. Approximately 192 erroneous trials and 1920 correct trials (calibration time of 8 min) were used to train and test a classifier for each subject (inter-set design).

2) Single-trial detection of observation ErrP: Two data sets were merged into one data set. Approximately 198 erroneous trials and 1980 correct trials (calibration time of 4 min) were used to train and test a classifier for each subject (inter-set design).

3) Classifier transfer between different types of ErrPs: To detect interaction ErrPs using the classifier trained on data containing observation ErrPs, one data set collected from the observation task (99 erroneous trials) was used to train the classifier and one data set collected from the interaction task (48 erroneous trials) was used for evaluation.

4) Classifier transfer between different subjects within the same ErrP type: To detect interaction ErrPs using the classifier trained on the data collected from another subject within the same ErrP task, we used the inter-subject design. Here, we used one data set which was collected from one subject during

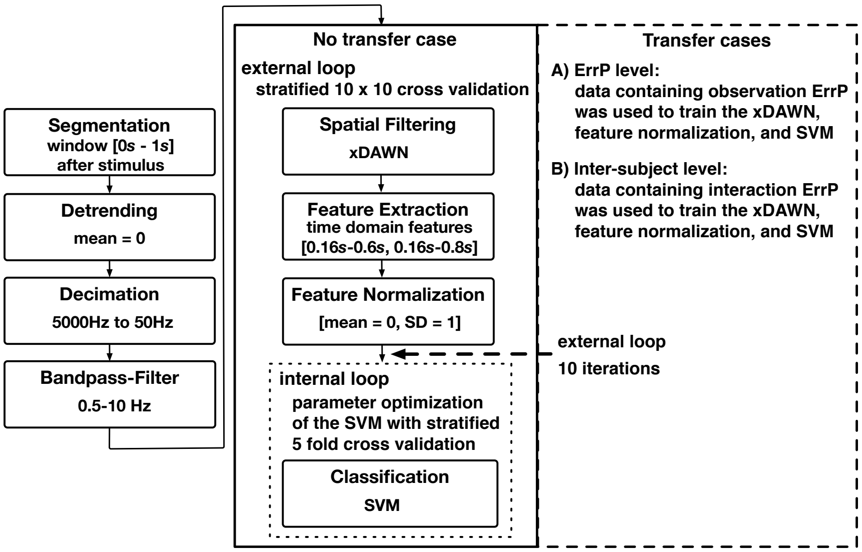


Fig. 2. Data flow [47]: The continuous EEGs were segmented, normalized, decimated, band pass filtered, and the signal to noise ratio was enhanced by applying a spatial filter called xDAWN [48]. The features that were extracted from the spatial filter were normalized over all trials and finally used to train the classifier called SVM [49]. For single trial detection of each type of ErrPs, we trained the spatial filter (xDAWN), feature normalization over all trials and classification (SVM) with stratified 10×10 fold cross validation (external loop, solid-line box). For the parameter optimization of the SVM, we additionally used an internal loop (stratified 5 fold cross validation, dotted-line box). For classifier transfer, the same data flow was used from segmentation to feature normalization (external loop, dashed-line box). Here, the cost parameter of the SVM was also optimized with an internal stratified 5 fold cross validation (internal loop, dotted-line box). This was repeated 10 times (dashed-line box). Details, see text.

the interaction task (48 erroneous trials) for evaluation and one data set which was collected from the other subjects (i.e., one of seven sets) during the interaction task (7 × 48 = 336 erroneous trials) to train the classifier.

E. Preprocessing and Classification

Figure 2 illustrates the data flow for preprocessing and classification. The continuous EEG signal was segmented into epochs from 0 s to 1 s after each event type (correct/erroneous trial). Since we tried to develop a realistic scenario (details, see Section II-A), some segmented correct trials overlapped with the following erroneous events. The correct trials were excluded when an erroneous trial occurred 1 s before or after the correct event during the segmentation of the continuous EEGs. Thus, only the correct trials without any error-related activity were labeled as correct.

All epochs were normalized to zero mean for each channel, decimated to 50 Hz, and band pass filtered (0.5 to 10 Hz). The xDAWN [48] was used as a spatial filter to enhance the signal-to-noise ratio. By applying the xDAWN the number of 64 physical channels was reduced to 8 pseudo channels.

For a successful classifier transfer between different types of ErrPs we investigated two different windows of training data which can be used to extract features for a classifier. First, we investigated a window of interest which leads to the highest detection performance of each type of ErrP. To this end, we compared two time windows with different window lengths.

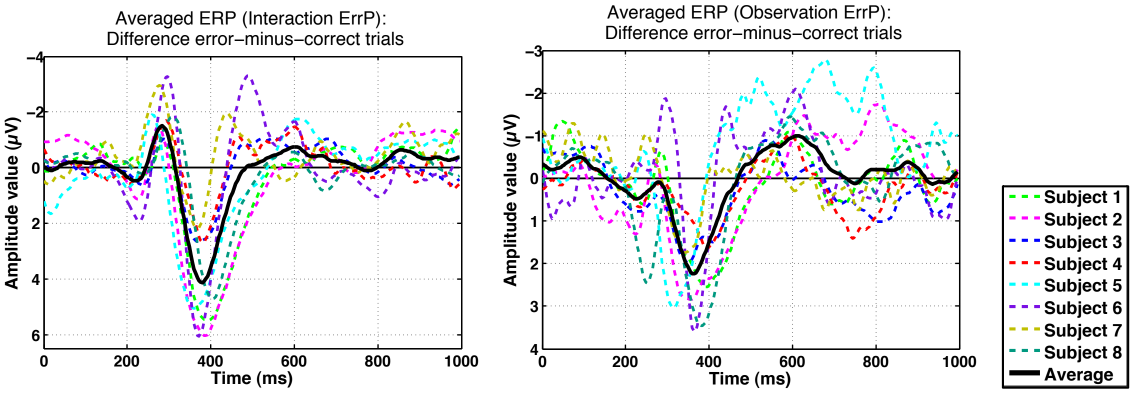
As shown for Fig. 3 both types of ErrPs showed a similar shape of averaged ERP curve in the early time window [0.16 s–0.4 s]: a first negative peak around 0.27 s after the erroneous events was followed by a positive peak around 0.38 s. However, differences in ERP shapes between both ErrP types were observed in the late time window [0.4 s–0.8 s]. A broader negativity including two negative peaks [0.4 s–0.6 s and 0.6 s–0.8 s] was observed for the averaged observation ErrP compared to the interaction ErrP. In contrast, a narrow negativity peak in the later time window [0.4 s–0.6 s] was observed for the averaged interaction ErrP (details see Section III-C and IV-A).

Based on the difference in the shape between the interaction ErrPs and observation ErrPs in the time window of 0.6 s–0.8 s, two time windows were investigated: a short [0.16 s–0.6 s] and a long [0.16 s–0.8 s] window. We compared the classification performance of both time windows for each type of ErrP. Such comparison allows us to determine whether a longer time window is necessary for a successful detection of interaction ErrPs and observation ErrPs.

Thus, the following two time windows were used for feature generation: a) [0.16 s–0.6 s] and b) [0.16 s–0.8 s]. Features were extracted from 8 channels after spatial filtering, between 0.16 s and N s where $N \in \{0.6, 0.8\}$, for a total of 176 features (8 channels × 22 data points = 176) for the shorter time window [0.16 s–0.6 s] and 256 features (8 channels × 32 data points = 256) for the longer time window [0.16 s–0.8 s].

The extracted features were normalized over all trials and used to train a classifier. We used a linear support vector machine (SVM) [49] to classify correct and erroneous trials.

For single trial detection of both types of ErrPs, we trained and validated the xDAWN, feature normalization over all trials, and the SVM by using stratified 10×10 fold cross validation. In each of the 10 iterations of cross validation (CV), the data was randomly split into 10 folds (splits) and 10 train-test pairs were constructed where each fold was used exactly once as test data. The remaining 9 folds were used



(a) Interaction ErrP

(b) Observation ErrP

Fig. 3. Averaged event-related potentials (ERPs) for the difference error-minus-correct trials at channel FCz for each subject. Here, we used a common average reference (CAR) and re-calculated the data on the channel FCz. Only artifact-free EEG trials were used. a) Interaction ErrP: for most subjects, a first negative peak was observed around 270 ms after the erroneous events, followed by a positive peak around 380 ms and a narrow negative peak between 400 ms and 600 ms. b) Observation ErrP: similar to the interaction ErrP, for most subjects, a first negative peak occurred around 270 ms after the erroneous events, followed by a positive peak around 380 ms. However, the first negativity and positivity were reduced (except for subject 6) for the observation ErrP compared to the interaction ErrP. Especially the second negativity in the window between 600 ms–800 ms was increased for the observation ErrP compared to the interaction ErrP. Finally, we observed a broad negative peak between 400 ms and 800 ms only for the observation ErrP.

for training. Here, the ratio of the two classes was considered during the construction of the train-test pairs so that each fold contained approximately the same proportions of both types of class labels (*stratified* 10×10 fold CV).

For the parameter optimization of the SVM, we additionally used an internal loop. For each training, the cost parameter of the SVM (i.e., regularization constant [50]) was optimized with an internal stratified 5 fold cross validation using a grid search among the predetermined values [10^0, 10^{-1}, ... , 10^{-6}]. Here, 9 splits, which were used for training, were divided into 5 train-test pairs for the parameter optimization of the SVM. The best parameter evaluated by this 5 fold cross validation (internal loop, see Figure 2 *dotted-line box*) was used for the external loop (see Figure 2 *solid-line box*).

Due to the unbalanced ratio of *erroneous* and *correct* trials (1:10), different penalty constants were used for the two different classes [51]. We determined a class weight of 5 for the under-represented class as penalty so that making errors on under-represented instances was costlier than making errors on over-represented instances.

In both cases of classifier transfer, the same data flow was used from *segmentation* to *feature normalization* (see Figure 2 *dashed-line box*). The xDAWN, feature normalization over all trials, and the SVM were trained in the external loop (see Figure 2 *dashed-line box*). Again, the cost parameter of the SVM (i.e., regularization constant [50]) was optimized with an internal stratified 5 fold cross validation (see Figure 2 *dotted-line box*). This was repeated ten times in the external loop (see Figure 2 *dashed-line box*), which resulted in different splits for each repetition.

F. Evaluation

As a metric for classification performance we used the arithmetic mean of true positive rate (TPR) and true negative rate (TNR), the so-called *balanced accuracy (bACC)*, where the erroneous trials were the positive instances. This metric is less sensitive to imbalanced data (i.e., unbalanced ratio of the two classes) compared to other metrics, e.g., accuracy (details, see [52]). Thus, the unbalanced ratio of the two classes was considered both during training of the classifier (class weight of 5 was used for the under-represented instances) and in the evaluation metric. The erroneous trials belonged to the positive class.

1) Single-trial detection of interaction ErrPs and observation ErrPs: Classification performances for the detection of interaction ErrPs and observation ErrPs were evaluated separately. For evaluation 10 × 10-fold cross validation was performed on the merged data set (192 erroneous trials and 1920 correct trials for the interaction task; 198 erroneous trials and 1980 correct trials for the observation task).

2) Classifier transfer between different types of ErrPs: We evaluated the classifier transferability between two different types of ErrPs. In our case, classifier transfer from the observation task to the interaction task was of interest, since we could collect more data in the observation task compared to the interaction task for the same duration of data collection per set. Thus, the classifier was trained on one data set containing observation ErrPs, (99 erroneous trials, calibration time of 2 min). After that, the trained classifier was used to evaluate one data set containing interaction ErrPs (48 erroneous trials).

3) Classifier transfer between different subjects within the same type of ErrPs: We evaluated the classifier transferability between different subjects. Unlike in the case of classifier transfer between different types of ErrPs, the same type of ErrP was used to train and to test the classifier. However, the classifier was trained on the data from different subjects to detect the interaction ErrP.

For evaluation the *leave-one-subject-out* cross validation was used, in which the data from one subject (i.e., the current subject) was selected for testing and the data from the other subjects was used to train the classifier.

G. Statistical Analysis

1) Single-trial detection of interaction ErrPs and observation ErrPs: To find out, whether a short time window could be sufficient to detect two different types of ErrPs and whether there could be a difference in classification performance depending on the length of time window, we compared the classification performances between the predefined two time windows within the same type of ErrPs. Also the classification performance of interaction ErrPs and observation ErrPs were compared for each time window.

To this end, classification performances (100 classification performances: stratified 10×10 fold cross validation (CV): 10 iterations, 10 splits CV, see Figure 2 *solid-line box*) were analyzed using repeated measures ANOVA with *time window* [$0.16\,\mathrm{s}$–$0.6\,\mathrm{s}$, $0.16\,\mathrm{s}$–$0.8\,\mathrm{s}$], *ErrP type* (observation, interaction), and *subject* (subject 1–subject 8) as within-subjects factors. Where necessary, the Greenhouse-Geisser correction was applied and the corrected p-value is reported. For multiple comparisons, the Bonferroni correction was applied.

2) Classifier transfer cases: To find out a possible effect of both types of classifier transfer on the classification performance, we compared classification performances obtained by two different kinds of classifier transfer with the no transfer case: 1) classifier transfer between different types of ErrPs within the same subject (i.e., transfer at the ErrP level), 2) classifier transfer between different subjects within the same type of ErrP (i.e., transfer at the inter-subject level), and 3) single-trial detection of interaction ErrPs without classifier transfer but with a long calibration time (i.e., no transfer case).

To this end, the classification performances (10 classification performances: 10 iterations, see Figure 2 *dashed-line box*) were analyzed by repeated measures ANOVA with *transfer type* (no transfer, transfer at ErrP level, transfer at inter-subject level) and *subject* (subject 1–subject 8) as within-subjects factors.

Here, we divided the 100 values obtained by the no transfer setup in 10 groups and averaged all values in each group. In this way, we obtained ten classification performances (i.e., sample size of 10) in no transfer case and they were compared to the ten classification performances in each transfer cases. Where necessary, the Greenhouse-Geisser correction was applied and the corrected p-value is reported. For multiple comparisons, the Bonferroni correction was applied.

IV. Results

A. ERP Results

We found no significant peak differences between both ErrP types in first negativity in the time window of $0.2\,\mathrm{s}$–$0.3\,\mathrm{s}$ [$F(1,7) = 4.48, p = 0.072$], whereas a significant reduced peak amplitude in positivity in the time window of $0.3\,\mathrm{s}$–$0.4\,\mathrm{s}$ was observed for observation ErrPs compared to interaction ErrPs [$F(1,7) = 29.21, p < 0.001$].

Furthermore, we found peak difference between both late sub-time windows for interaction ErrPs, but not for observation ErrPs. That means, interaction ErrPs had a negative peak in the time window of $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$, but not in the time window of $0.6\,\mathrm{s}$–$0.8\,\mathrm{s}$. In contrast, we found two negative peaks for both late sub-time windows [interaction between *ErrP type* and *time window*: $F(1,7) = 15.16, p < 0.006$, $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$, vs. $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$: $p < 0.005$ for interaction ErrPs and $p = 0.44$ for observation ErrPs]. In summary, both ErrP types contained a negative peak in the time window of $0.4\,\mathrm{s}$–$0.6\,\mathrm{s}$. Therefore, we used the time window of $0.16\,\mathrm{s}$–$0.6\,\mathrm{s}$ for classifier transfer between both ErrP types.

B. Classification performance of interaction ErrPs and observation ErrPs

Table II and Fig. 4 show the classification performance on correct and erroneous single trials for each ErrP task. We obtained a classification performance with an averaged bACC of 0.82 and 0.81 for interaction ErrPs and 0.79 and 0.81 for observation ErrPs for each time window.

For the observation task, a higher classification performance for the longer time windows [$0.16\,\mathrm{s}$–$0.8\,\mathrm{s}$] was achieved compared to the shorter time window [$0.16\,\mathrm{s}$–$0.6\,\mathrm{s}$] across subjects [short window: bACC of 0.79, long window: bACC of 0.81, interaction between *ErrP type* and *time window*: $F(1,99) = 16.88$, $p < 0.001$, short window vs. long window: $p < 0.001$]. The higher classification performance on the longer time window was observed for five subjects [interaction of *time window* with *ErrP type* and *subject*: $F(3,693) = 2.01$, $p = 0.052$, short window vs. long window: the statistical values, see Fig. 4].

However, the difference in classification performance between short and long windows was not observed for the interaction task. All subjects showed no difference between both time windows (short window vs. long window: $p = n.s.$ statistical values, see Fig. 4). Based on this result, only the short time window was selected to detect interaction ErrPs using a classifier trained on data containing observation ErrPs. By selecting the short time window [$0.16\,\mathrm{s}$–$0.6\,\mathrm{s}$] we could also reduce the dimensionality of the feature space which could be relevant for an application.

C. Classification performance in case of classifier transfer between different ErrP types (transfer at the ErrP level)

Table III–A(I) shows the classification performance of interaction ErrPs using the classifier trained on the data collected during the observation task.

We obtained an averaged bACC of 0.79 across all subjects. The success of classifier transfer from observation ErrPs to interaction ErrPs was subject-specific. For four subjects the classification performance in case of using a classifier trained on observation ErrPs was reduced compared to the case of using a classifier trained on interaction ErrPs [interaction of *transfer* with *subject*: $F(14,126) = 82.21$, $p < 0.001$, no classifier transfer vs. classifier transfer at the ErrP level: the statistical values, see Fig. 5 (upper)].

On the other hand, for the other four subjects we did not find a reduction in classification performance when applying classifier transfer using a calibration time of 2 minutes. The classification performance after transfer was as good as (subject 7) or even higher compared to no classifier transfer with a calibration time of 8 min (subject 1, subject 4, subject 8).

TABLE II

CLASSIFICATION PERFORMANCE (MEAN±STANDARD DEVIATION) OF EACH SUBJECT ON CORRECT AND ERRONEOUS SINGLE TRIALS AND THE AVERAGE OF THEM FOR TWO DIFFERENT TYPES OF ERRP: INTERACTION ERRP AND OBSERVATION ERRP (INTER-SET DESIGN). NOTE: TWO TIME WINDOWS WERE USED FOR FEATURE EXTRACTION: SHORT TIME WINDOW (0.16 S–0.8 S) AND LONG TIME WINDOW (0.16 S–0.8 S)

Single-trial detection of interaction ErrPs									
	Training instances (erroneous/correct): approx. 192/1920, calibration time of 8 min (4 sets were merged)								
0.16-0.6 s	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Average
bACC	0.79±0.06	0.81±0.06	0.82±0.04	0.80±0.06	0.89±0.04	0.83±0.05	0.78±0.04	0.81±0.06	0.82±0.03
TPR	0.70±0.12	0.70±0.11	0.72±0.10	0.72±0.11	0.82±0.07	0.72±0.11	0.69±0.09	0.73±0.12	0.73±0.04
TNR	0.88±0.04	0.92±0.02	0.93±0.03	0.89±0.03	0.96±0.01	0.93±0.03	0.87±0.04	0.89±0.05	0.91±0.03
0.16-0.8 s	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Average
bACC	0.78±0.06	0.81±0.05	0.82±0.04	0.80±0.06	0.88±0.05	0.82±0.05	0.78±0.05	0.80±0.05	0.81±0.03
TPR	0.69±0.12	0.69±0.11	0.71±0.10	0.71±0.11	0.83±0.10	0.73±0.12	0.67±0.10	0.72±0.11	0.72±0.05
TNR	0.87±0.04	0.93±0.02	0.92±0.03	0.89±0.03	0.90±0.04	0.91±0.04	0.89±0.04	0.87±0.04	0.90±0.02

Single-trial detection of observation ErrPs									
	Training instances (erroneous/correct): approx. 198/1980, calibration time of 4 min (2 sets were merged)								
0.16-0.6 s	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Average
bACC	0.79±0.05	0.83±0.05	0.78±0.05	0.77±0.06	0.85±0.04	0.75±0.05	0.79±0.05	0.76±0.06	0.79±0.03
TPR	0.76±0.10	0.75±0.12	0.72±0.10	0.66±0.11	0.82±0.07	0.65±0.11	0.71±0.09	0.66±0.12	0.72±0.05
TNR	0.83±0.04	0.92±0.03	0.84±0.03	0.88±0.03	0.87±0.04	0.85±0.03	0.86±0.03	0.86±0.03	0.87±0.02
0.16–0.8 s	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	Subject 6	Subject 7	Subject 8	Average
bACC	0.81±0.05	0.85±0.05	0.78±0.05	0.80±0.05	0.87±0.04	0.76±0.05	0.79±0.05	0.78±0.06	0.81±0.04
TPR	0.77±0.10	0.75±0.10	0.71±0.10	0.69±0.11	0.83±0.10	0.67±0.10	0.72±0.10	0.70±0.11	0.73±0.05
TNR	0.85±0.04	0.94±0.02	0.84±0.03	0.91±0.03	0.90±0.04	0.85±0.02	0.86±0.03	0.86±0.03	0.88±0.04

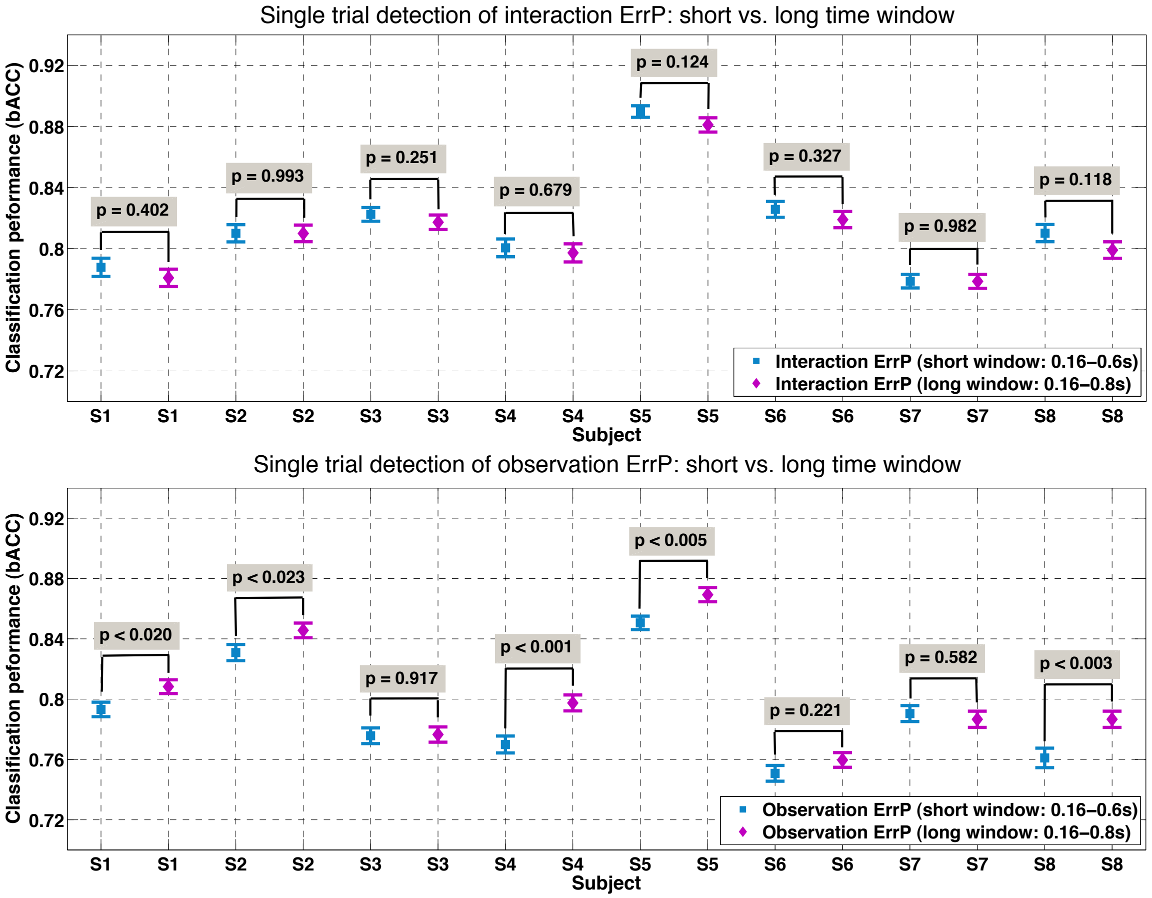


Fig. 4. Comparison of classification performance between the two time windows (0.16 s-0.6 s, 0.16 s-0.8 s) for each type of ErrP and each subject: no interaction effect of *time window* with *ErrP type* and *subject*: $F(3, 693) = 2.01$, $p = 0.052$. For multiple comparisons, Bonferroni correction was applied. Note: mean bACC (bACC = (TPR+TNR)/2) with standard error were presented for each time window and each subject.

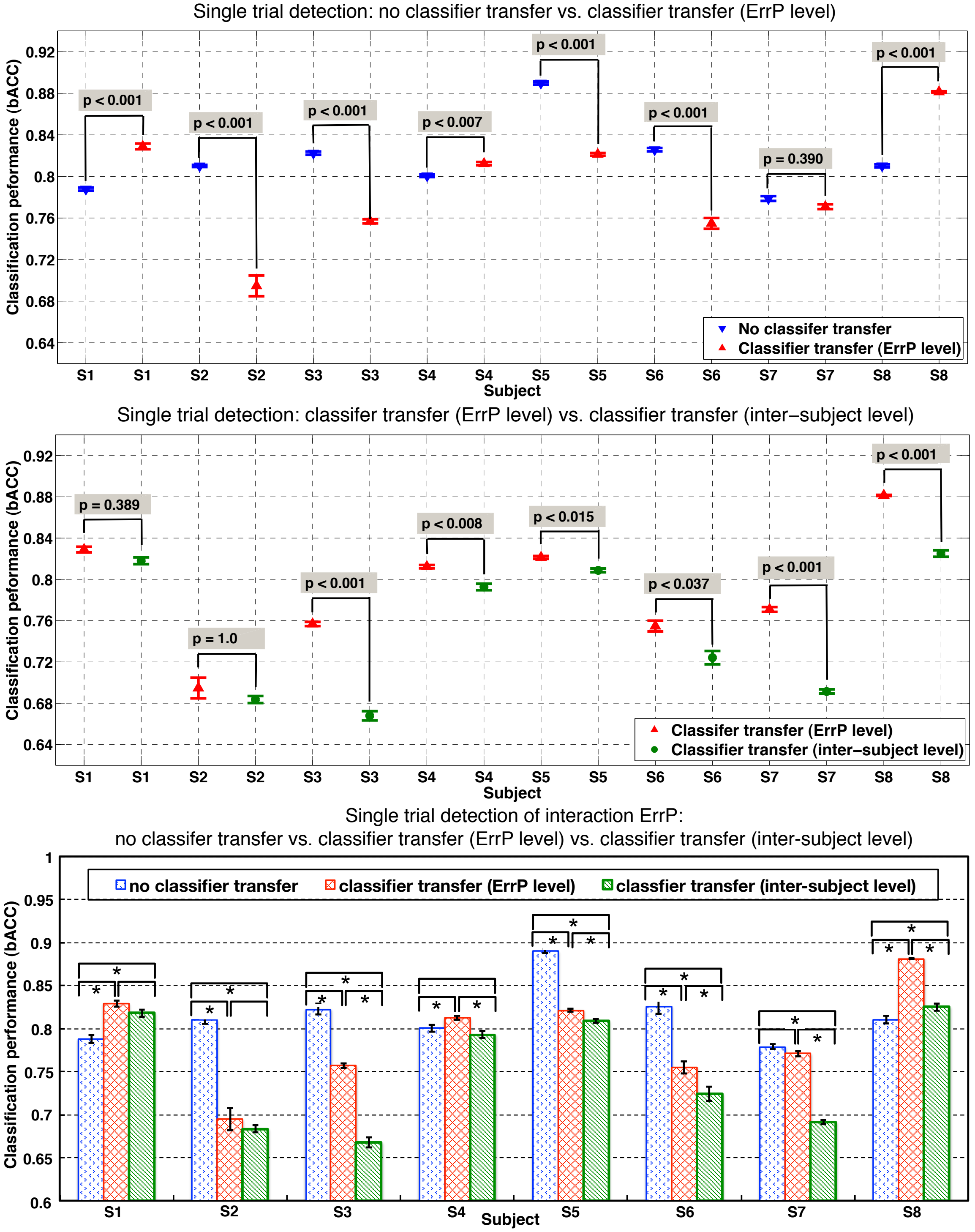


Fig. 5. Comparison between two different types of classifier transfer and the no transfer case: no classifier transfer vs. classifier transfer at the ErrP level (upper), classifier transfer at the ErrP level vs. classifier transfer at the inter-subject level (middle), no classifier transfer vs. classifier transfer at ErrP level vs. classifier transfer at inter-subject level (below). Note that the p value that is reported in each plot was corrected based on the comparison under three conditions (no classifier transfer, classifier transfer at the ErrP level, and classifier transfer at the inter-subject level). For multiple comparison, Bonferroni correction was applied. * denotes a significant difference

TABLE III
Classification performance (mean±standard deviation) of interaction ErrPs using two kinds of classifier transfer (A-II and A–II) and observation ErrPs using classifier transfer at the ErrP level (B). Note: Features used by the classifier were extracted from the time window of 0.16 s–0.6 s.

<table>
<tr><td colspan="10">A) Single trial detection of interaction ErrPs using different types of classifier transfer: I and II</td></tr>
<tr><td colspan="10">I. Classifier Transfer between different types of ErrPs
Observation ErrP (Training) → Interaction ErrP (Test): transfer at the ErrP level</td></tr>
<tr><td colspan="10">Training instances (erroneous/correct): approx. 99/990, calibration time of 2 min</td></tr>
<tr><td></td><td>Subject 1</td><td>Subject 2</td><td>Subject 3</td><td>Subject 4</td><td>Subject 5</td><td>Subject 6</td><td>Subject 7</td><td>Subject 8</td><td>Average</td></tr>
<tr><td>bACC</td><td>0.83±0.01</td><td>0.70±0.04</td><td>0.76±0.01</td><td>0.81±0.01</td><td>0.82±0.01</td><td>0.76±0.02</td><td>0.77±0.01</td><td>0.88±0.01</td><td>0.79±0.06</td></tr>
<tr><td>TPR</td><td>0.74±0.01</td><td>0.55±0.13</td><td>0.58±0.03</td><td>0.80±0.02</td><td>0.75±0.05</td><td>0.77±0.04</td><td>0.66±0.05</td><td>0.81±0.01</td><td>0.71±0.05</td></tr>
<tr><td>TNR</td><td>0.91±0.01</td><td>0.84±0.06</td><td>0.94±0.01</td><td>0.82±0.02</td><td>0.88±0.03</td><td>0.74±0.02</td><td>0.89±0.02</td><td>0.94±0.01</td><td>0.87±0.07</td></tr>
<tr><td colspan="10">II. Classifier Transfer between different subjects within the same type of ErrPs
7 subjects (Training) → 1 subject (Test): transfer at the inter-subject level</td></tr>
<tr><td colspan="10">Training instances (erroneous/correct): approx. 336/3360, calibration time of 0 min</td></tr>
<tr><td></td><td>Subject 1</td><td>Subject 2</td><td>Subject 3</td><td>Subject 4</td><td>Subject 5</td><td>Subject 6</td><td>Subject 7</td><td>Subject 8</td><td>Average</td></tr>
<tr><td>bACC</td><td>0.82±0.01</td><td>0.68±0.04</td><td>0.67±0.01</td><td>0.79±0.01</td><td>0.81±0.01</td><td>0.72±0.02</td><td>0.69±0.01</td><td>0.83±0.01</td><td>0.75±0.07</td></tr>
<tr><td>TPR</td><td>0.83±0.01</td><td>0.47±0.13</td><td>0.35±0.03</td><td>0.71±0.02</td><td>0.74±0.01</td><td>0.59±0.04</td><td>0.49±0.02</td><td>0.79±0.03</td><td>0.62±0.17</td></tr>
<tr><td>TNR</td><td>0.80±0.02</td><td>0.89±0.01</td><td>0.99±0.01</td><td>0.86±0.02</td><td>0.87±0.01</td><td>0.85±0.01</td><td>0.90±0.02</td><td>0.86±0.01</td><td>0.87±0.02</td></tr>
<tr><td colspan="10">B) Single trial detection of observation ErrPs using classifier transfer between different types of ErrPs
Interaction ErrP (Training) → Observation ErrP (Test): transfer at the ErrP level</td></tr>
<tr><td colspan="10">Training instances (erroneous/correct): approx. 96/960 (2 sets were merged), calibration time of 4 min</td></tr>
<tr><td></td><td>Subject 1</td><td>Subject 2</td><td>Subject 3</td><td>Subject 4</td><td>Subject 5</td><td>Subject 6</td><td>Subject 7</td><td>Subject 8</td><td>Average</td></tr>
<tr><td>bACC</td><td>0.72±0.02</td><td>0.63±0.04</td><td>0.70±0.03</td><td>0.73±0.02</td><td>0.72±0.02</td><td>0.72±0.02</td><td>0.73±0.02</td><td>0.84±0.01</td><td>0.72±0.05</td></tr>
<tr><td>TPR</td><td>0.55±0.02</td><td>0.42±0.03</td><td>0.49±0.03</td><td>0.62±0.02</td><td>0.56±0.02</td><td>0.57±0.02</td><td>0.71±0.02</td><td>0.75±0.01</td><td>0.58±0.11</td></tr>
<tr><td>TNR</td><td>0.88±0.02</td><td>0.84±0.02</td><td>0.91±0.02</td><td>0.85±0.02</td><td>0.88±0.02</td><td>0.86±0.02</td><td>0.75±0.02</td><td>0.93±0.01</td><td>0.86±0.05</td></tr>
</table>

D. Classification performance for classifier transfer between different subjects within the same ErrP type (transfer at the inter-subject level)

Table III–A(II) shows the classification performance of interaction ErrPs when the classifier was transferred between different subjects but the same type of ErrPs was used to train and test the classifier.

We obtained an averaged bACC of 0.75 across all subjects. All subjects showed a reduced classification performance compared to the case of classifier transfer at the ErrP level except for two subjects (subject 1, subject 2) [interaction of *transfer* with *subject*: $F(14, 126) = 82.21$, $p < 0.001$, classifier transfer at ErrP level vs. classifier transfer at inter-subject level: the statistical values, details, see Fig. 5 (middle)].

Compared to the single-trial detection of interaction ErrPs without any classifier transfer the classification performance was significantly reduced for all subjects except for one subject (subject 4) [no classifier transfer vs. classifier transfer at the inter-subject level, statistical values, details, see Fig. 5 (below)].

E. Summary

We achieved a high performance in single-trial detection of interaction ErrPs and observation ErrPs (an averaged bACC of $0.82/0.81$ and $0.79/0.81$).

The classification performance was slightly reduced when transferring the classifier between different types of ErrPs: from bACC of 0.82 to bACC of 0.79. This reduction of performance is subject-specific. For half of the subjects the transfer between different types of ErrP was successful. The detection performance in this case was as good as in the case of no transfer or even higher.

In contrast, the classification performance was significantly reduced when transferring the classifier between different subjects within the same type of ErrP: from bACC of 0.82 to bACC of 0.75. Such reduction was not subject specific. All subjects showed a reduced classificaton performance except for one subject.

Furthermore, the results from two different lengths of time windows proved that the shorter window [0.16 s–0.6 s] is sufficient to detect the interaction ErrP, whereas the observation ErrP can be detected with a higher classification performance in case of using a longer time window [0.16 s–0.8 s]. Based on this investigation we could select the shorter window to extract features for classifier training.

V. Discussion and Conclusion

A. Feasibility of classifier transfer

In this study, we have achieved a high performance in single-trial detection of interaction ErrPs and observation ErrPs in a more realistic, application oriented scenario, when compared, e.g., to the study described in [33]. The evaluation of the proposed approach showed that the classifier transfer between different types of ErrPs is feasible. Although a success of the proposed classifier transfer was subject-specific, from the perspective of application, such classifier transfer was very useful to reduce calibration time, i.e., two minutes of EEG recording was sufficient to calibrate the system for half of the subjects. Furthermore, the classification performance obtained by the proposed transfer approach outperformed the approach of transfer between different subjects within the same type of ErrP for most subjects.

B. Directions of classifier transfer between different ErrP types

In this study, we performed the transfer of observation ErrP → interaction ErrP, since this transfer direction matches the goal of the present study (i.e., reduction of calibration time) and the concept of the proposed approach (i.e., use of data with a shorter calibration time to test on data with a longer calibration time). However, the reversed transfer direction (i.e., transfer of interaction ErrP → observation ErrP) could be interesting for other applications. Thus, we additionally performed the transfer with the reversed transfer direction. To this end, we used the same data sets as for the transfer of observation ErrP → interaction ErrP. To obtain the same amount of training instances, we merged two data sets from the interaction task. Here, the same time window [0.16 s–0.6 s] was used for feature extraction. As shown for Table III–B, the transfer of interaction ErrP → observation ErrP was successful, but performed worse compared to the transfer of observation ErrP → interaction ErrP (see Table III–A(I) vs. Table III–B).

A possible explanation is that properties of the data used for training and testing a classifier could affect classification performance in case of classifier transfer. We observed more variability between single trials for the observation task compared to the interaction task. Figure 6 shows an example of variability between single trials for each ErrP type in a descriptive manner. Most subjects showed more variability between single trials for observation ErrPs than interaction ErrPs (e.g., subject 6). However, a few subjects (e.g., subject 8) showed less differences in variability between single trials depending on the ErrP type. In a descriptive manner, the data obtained from the observation task was more heterogenous than the data from the interaction task.

Based on this hypothesis, one can assume that the model learned on data with more variability between single trials may cover the whole range of data with less variability between single trials. In contrast, the model learned on data with less variability between single trials may not cover completely the whole range of data with more variability between single trials. However, we can not ensure that variability between single trials could have a direct impact on classification performance. Hence, we need further systematic investigations with more clear concepts and hypotheses to test this assumption. In future work, variability between single trials will be investigated to evaluate its impact on transferability of classifiers between scenarios (e.g., different ErrP types, different tasks).

C. Classifier transfer based on similarity in ERP shapes

In this study, we showed that it is possible to use a classifier trained on one type of ErrP to detect another type of ErrP. In our case, the context of application (e.g., the scenario, the task has been changed and this change elicited a different type of ErrP (e.g., interaction ErrPs) compared to the learned ErrP type (e.g., observation ErrPs). However, both ErrP types share a similar pattern (e.g., similar ERP shapes), since they are elicited by recognizing errors. Accordingly, we could show that a learned model can be used to detect a brain pattern which is partly similar to the learned brain pattern.

Thus, it was relevant to find features which reflect this similarity. One possibility is to investigate ERP shapes of both ErrP types. In this study, we found a difference in ERP peaks between both ErrP types in the late time window of 0.4 s and 0.8 s: interaction ErrP contained a negative peak in the time window of 0.4 s–0.6 s, but not in the time window of 0.6 s–0.8 s, whereas observation ErrP contained a negative peak in both late time windows [0.4 s–0.6 s and 0.6 s–0.8 s]. Accordingly, we selected the time window in which both ErrP types shared a similar ERP shape, i.e., the time window of 0.16 s–0.6 s. We assume that a good choice of time window might contribute to a successful transfer between both ErrP types, especially when data points in the time domain are used for feature extraction. In future work, a systematic investigation could be performed to find an optimal time window for such transfer.

Transfer based on similarity in ERP shapes has been also found in recent studies that reveal a successful transfer between different tasks using the same ErrP type [42], [43], [53]. In our previous study, we also used this transfer concept and transferred a classifier in the context of P300 detection in specific applications [27]. Here, we detected P300 elicited by recognizing target stimulus among frequent standard stimuli (oddball paradigm). We observed that subjects sometimes missed targets when they performed multiple tasks. In case of no response to a target stimulus we repeated this *missed* target stimulus. Further, we observed that subjects recognized target stimuli but sometimes postponed to respond to target stimuli when they had a critical situation while performing the other task. In this case, we did not want to repeat a target stimulus due to a delayed response to a target stimulus. Hence, in applications, the distinction between *missed* target stimuli (i.e., without response) and *recognized* target stimuli (i.e., with response) could help to optimize an interface between human and machine. To distinguish *missed* target stimuli from *recognized* target stimuli, we build a classifier trained on data containing *standard* stimuli and *recognized* target stimuli and used the trained classifier to evaluate data containing *missed* and *recognized* target stimuli. This classifier transfer was also based on our observation in which ERP shape of *missed target* stimuli were similar to the ERP shape of *standard* stimuli. In this special case the P300 was under both conditions (missed target stimuli and standard stimuli) not or only weekly evoked. Here, we did not transfer different ERP types, but the used concept of transfer is the same, i.e., transfer based on similarity in signal characteristics.

Therefore, we carefully assume that classifier transfer between different ERP types is feasible when different ERP types share a similar pattern (e.g., similar ERP shapes).

D. Benefit of the proposed transfer approach and its application possibilities

This study shows that the higher classification performance obtained by the proposed approach is not the only advantage

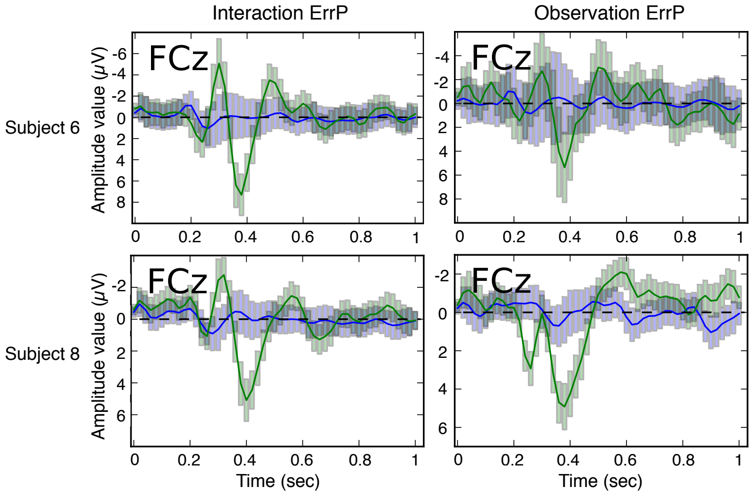


Fig. 6. An example of variability between single trials for each ErrP type (solid line: average of all single trials, shaded area: standard error). Violet lines and their shaded areas correspond to correct trials. Green lines and their shaded areas correspond to erroneous trials.

compared to the transfer approach between different subjects within the same type of ErrPs. The proposed transfer approach is a more beneficial way to handle few training instances and to reduce calibration time, since the transfer approach between different subjects requires data from a reasonable number of subjects when events correlated with specific EEG pattern, that are detected by the BCI, occur very seldom. However, there is also a possible limitation of our approach, which is based on similarity in ERP pattern. In a recent study, different kinds of ErrPs (execution ErrPs/outcome ErrPs) with different patterns (e.g., different ERP shapes) were found [54]. Reasons for such differences and whether classifier transfer is feasible in such cases must be investigated in future work.

Furthermore, in future work, the proposed transfer approach and the possible advantages obtained by using the proposed transfer approach will be tested in real applications. For example, in specific rehabilitation situations, the proposed approach (i.e., reduction of calibration time by classifier transfer) and both underlying concept of transfer (1. transfer by using data with a shorter calibration time to test on data with a longer calibration time and 2. transfer based on similarity in signal characteristics) could be beneficial, since some specific situations (patients health state, other unfavorable environments) do not always allow to collect a sufficient amount of training data. In this case, calibration time which is needed in applications can be reduced by using the proposed transfer approach. In most application cases, ErrPs have been used to correct an external device control, for example P300 speller (e.g., [38], [40]) or robot arm movements control (e.g., [55], [56]). When a successful transfer of classifier trained on data from other scenarios is feasible, it could not be necessary to collect training data containing ErrPs, for example, to correct wrong movement prediction/detection in the actual application scenario.

Furthermore, the proposed approach could be applied in real robotic applications. Possibilities for improving the learning of a model by using ErrPs have been already investigated in [35], [36]. Such an investigation with a humanoid robot could be interesting, especially by using the proposed transfer approach e.g., by transferring a classifier trained on data from a simplified scenario into real applications of the humanoid robot.

Acknowledgment

This work is supported by the German Ministry of Economics and Technology (grant no. FKZ 50 RA 1011, grant no. FKZ 50 RA 1012 and grant FKZ 50 RA 1301). The authors would like to thank anonymous reviewers for valuable comments. We also thank Foad Ghaderi, Anett Seeland and Marc Tabie for constructive remarks. Finally we thank all participants of the experiments and Elke Neubauer for her help in recording EEG data.

References

[1] S. K. Kim, E. A. Kirchner, A. Stefes, and F. Kirchner, "Intrinsic interactive reinforcement learning using error-related potentials for real world human-robot interaction," vol. 7: 17562, December 2017.

[2] M. Fatourechi, R. Ward, and G. Birch, "Evaluating the performance of a self-paced BCI with a new movement and using a more engaging environment," in *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2008, pp. 650–653.

[3] O. Bai, V. Rathi, P. Lin, D. Huang, H. Battapady, D.-Y. Fei, L. Schneider, E. Houdayer, X. Chen, and M. Hallett, "Prediction of human voluntary movement before it occurs," *Clinical Neurophysiology*, vol. 122, no. 2, pp. 364–372, Feb. 2011.

[4] M. Folgheraiter, E. A. Kirchner, A. Seeland, S. K. Kim, M. Jordan, H. Wöhrle, B. Bongardt, S. Schmidt, J. Albiez, and F. Kirchner, "A multimodal brain-arm interface for operation of complex robotic systems and upper limb motor recovery," in *Proceedings of the 4th International Conference on Biomedical Electronics and Devices (BIODEVICES)*, P. Vieira, A. Fred, J. Filipe, and H. Gamboa, Eds. Rome: SciTePress, Jan 2011, pp. 150–162.

[5] D. Novak, X. Omlin, R. Leins-Hess, and R. Riener, "Predicting targets of human reaching motions using different sensing technologies." *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 9, pp. 2645–2654, 2013.

[6] P. Ahmadian, S. Cagnoni, and L. Ascari, "How capable is non-invasive EEG data of predicting the next movement? A mini review," *Frontiers in Human Neuroscience*, vol. 7, p. 124, 2013.

[7] E. A. Kirchner, M. Tabie, and A. Seeland, "Multimodal movement prediction - towards an individual assistance of patients," *PLoS ONE*, vol. 9, no. 1, p. e85060, Jan 2014.

[8] J. Kohlmorgen, G. Dornhege, M. Braun, B. Blankertz, K.-R. Müller, G. Curio, K. Hagemann, A. Bruns, M. Schrauf, and W. Kincses, "Improving human performance in a real operating environment through real-time mental workload detection," *Toward Brain-Computer Interfacing*, pp. 409–422, 2007.

[9] J. H. Metzen, S. K. Kim, T. Duchrow, E. A. Kirchner, and F. Kirchner, "On transferring spatial filters in a brain reading scenario," in *IEEE Statistical Signal Processing Workshop (SSP)*, June 2011, pp. 797–800.

[10] J. H. Metzen, S. K. Kim, and E. A. Kirchner, "Minimizing calibration time for brain reading," in *Pattern Recognition*, ser. Lecture Notes Computer Science. Springer, 2011, vol. 6835, pp. 366–75.

[11] E. A. Kirchner, H. Wöhrle, C. Bergatt, S. K. Kim, J. H. Metzen, D. Feess, and F. Kirchner, "Towards operator monitoring via brain reading – an EEG-based approach for space applications," in *Proceeding of 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (ISAIRAS)*, Sapporo, 2010, pp. 448–455.

[12] C. Mühl, C. Jeunet, and F. Lotte, "EEG-based workload estimation across affective contexts," *Frontiers in Neuroscience*, vol. 8, p. 114, 2014.

[13] J. R. Wolpaw, N. Birbauer, D. J. McFarland, G. Pfurtscheller, and T. Vaughan, "Brain-computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, pp. 767–791, 2002.

[14] J. d. R. Millán, R. Rupp, G. Müller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. Kübler, R. Leeb, C. Neuper, K.-R. Müller, and D. Mattia, "Combining brain-computer interfaces and assistive technologies: State-of-the-art and challenges," *Frontiers in Neuroscience*, vol. 4, no. 161, 2010.

[15] J. Mak, Y. Arbel, J. Minett, L. McCane, B. Yuksel, D. Ryan, D. Thompson, L. Bianchi, and D. Erdogmus, "Optimizing the P300-based brain–computer interface: current status, limitations and future directions," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025003, 2011.

[16] G. Bin, X. Gao, Y. Wang, B. Hong, and S. Gao, "VEP-based brain–computer interfaces: time, frequency, and code modulations," *IEEE Computational Intelligence Magazine*, vol. 4, no. 4, pp. 22–26, 2009.

[17] G. Bin, X. Gao, Y. Wang, Y. Li, B. Hong, and S. Gao, "A high-speed BCI based on code modulation VEP," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025015, 2011.

[18] J. R. Wolpaw and D. J. McFarland, "Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proceedings of the National Academy of Sciences*, vol. 101, no. 51, pp. 17 849–17 854, Dec 2004.

[19] F. Cincotti, D. Mattia, F. Aloise, S. Bufalari, G. Schalk, G. Oriolo, A. Cherubini, M. G. Marciani, and F. Babiloni, "Non-invasive brain–computer interface system: towards its application as assistive technology," *Brain research bulletin*, vol. 75, no. 6, pp. 796–803, 2008.

[20] G. Müller-Putz, R. Scherer, G. Pfurtscheller, C. Neuper, and R. Rupp, "Non-invasive control of neuroprostheses for the upper extremity: temporal coding of brain patterns," in *Proceedings of the 31th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009, pp. 3353–3356.

[21] I. K. Niazi, N. Jiang, O. Tiberghien, J. F. Nielsen, K. Dremstrup, and D. Farina, "Detection of movement intention from single-trial movement-related cortical potentials," *Journal of Neural Engineering*, vol. 8, no. 6, p. 066009, Dec. 2011.

[22] J. Ibáñez, J. I. Serrano, M. D. Castillo, L. Barrios, J. A. Gallego, and E. Rocon, "An EEG-Based Design for the Online Detection of Movement Intention," in *Advances in Computational Intelligence*, ser. Lecture Notes in Computer Science, J. Cabestany, I. Rojas, and G. Joya, Eds., vol. 6691. Springer Berlin Heidelberg, 2011, pp. 370–377.

[23] E. Lew, R. Chavarriaga, S. Silvoni, and J. d. R. Millán, "Detection of self-paced reaching movement intention from EEG signals," *Frontiers in Neuroengineering*, vol. 5, pp. 13–13, 2011.

[24] H. Zhang, R. Chavarriaga, L. Gheorghe, and J. d. R. Millán, "Inferring driver's turning direction through detection of error related brain activity," in *Proceedings of the 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2013, pp. 2196–2199.

[25] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. Rao, "Control of a humanoid robot by a noninvasive brain–computer interface in humans," *Journal of Neural Engineering*, vol. 5, no. 2, p. 214, 2008.

[26] I. Iturrate, L. Montesano, and J. Minguez, "Single trial recognition of error-related potentials during observation of robot operation," in *Proceedings of the 32th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2010, pp. 4181–4184.

[27] E. A. Kirchner, S. K. Kim, S. Straube, A. Seeland, H. Wöhrle, M. M. Krell, M. Tabie, and M. Fahle, "On the applicability of brain reading for predictive human-machine interfaces in robotics," *PLoS ONE*, vol. 8, no. 12, p. e81732, Dec 2013.

[28] A. Seeland, H. Wöhrle, S. Straube, and E. A. Kirchner, "Online movement prediction in a robotic application scenario," in *Proceeding of 6th International IEEE EMBS Conference on Neural Engineering (NER)*, Nov 2013, pp. 41–44.

[29] L. Parra, C. Spence, A. Gerson, and P. Sajda, "Response error correction -a demonstration of improved human-machine performance using real-time EEG monitoring," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 173–177, June 2003.

[30] B. Blankertz, G. Dornhege, C. Schafer, R. Krepki, J. Kohlmorgen, K. Muller, V. Kunzmann, F. Losch, and G. Curio, "Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial eeg analysis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 127–131, 2003.

[31] P. W. Ferrez and J. d. R. Millán, "You are wrong! - automatic detection of interaction errors from brain waves," in *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2005, pp. 1413–1418.

[32] A. Buttfield, P. W. Ferrez, and J. d. R. Millán, "Towards a robust BCI: Error recognition and online learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 43, pp. 128–131, 2006.

[33] P. W. Ferrez and J. d. R. Millán, "Error-related EEG potentials generated during simulated brain-computer interaction," *IEEE Transaction on Biomedical Engineering*, vol. 55, no. 3, pp. 923–929, March 2008.

[34] M. Lehne, K. Ihme, A.-M. Brouwer, J. B. van ErP, and T. Zander, "Error-related EEG patterns during tactile human-machine interaction," in *International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2009, pp. 1–9.

[35] R. Chavarriaga and J. d. R. Millán, "Learning from EEG error-related potentials in noninvasive brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 4, pp. 381–388, 2010.

[36] I. Iturrate, L. Montesano, and J. Minguez, "Robot reinforcement learning using EEG-based reward signals," in *IEEE International Conference of on robotics and automation (ICRA)*, 2010, pp. 4181–4184.

[37] B. D. Seno, M. Matteucci, and L. Mainardi, "Online detection of P300 and error potentials in a BCI speller," *Computational Intelligence and Neuroscience*, p. 307254, 2010.

[38] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI," *Clinical Neurophysiology*, vol. 123 (7), pp. 1328–37, 2012.

[39] N. M. Schmidt, B. Blankertz, and M. Treder, "Online detection of error-related potentials boots the performance of mental typewriters," *BMC Neuroscience*, vol. 13, p. 19, 2011.

[40] A. Combaz, N. Chumerin, N. V. Manayakov, A. Robben, J. A. K. Suykens, and M. M. Van Hulle, "Towards the detection of error-related potentials and its integration in the context of a P300 speller brain-computer interface," *Neurocomputing*, vol. 80, pp. 73–82, 2012.

[41] M. Spüler, W. Rosenstiel, and M. Bogdan, "Online adaptation of a c-VEP brain-computer interface (BCI) based on error-related potentials and unsupervised learning," *PLoS ONE*, vol. 7, no. 12, p. e51077, 2012.

[42] I. Iturrate, R. Chavarriaga, L. Montesano, and J. d. R. Minguez, Millán, "Latency correction of error potentials between different experiments reduces calibration time for single-trial classification," in *Proceedings of 34th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 3288–3291.

[43] I. Iturrate, L. Montesano, and J. Minguez, "Task-dependent signal variations in EEG error-related potentials for brain-computer interfaces," *Journal of Neural Engineering*, vol. 10, no. 2, p. 026024, 2013.

[44] R. Chavarriaga, A. Sobolewski, and J. d. R. Millán, "Errare machinale est: the use of error-related potentials in brain-machine interfaces," *Front. Neurosci.*, vol. 8, 2014.

[45] S. K. Kim and E. A. Kirchner, "Classifier transferability in the detection of error related potentials from observation to interaction," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, (SMC)*, 2013, pp. 3360–3365.

[46] M. Falkenstein, J. Hoormann, S. Christ, and J. Hohnsbein, "ERP components on reaction errors and their functional significance: A tutorial," *Biological Psychology*, vol. 51, pp. 87–107, 2000.

[47] M. M. Krell, S. Straube, A. Seeland, H. Wöhrle, J. Teiwes, J. H. Metzen, E. A. Kirchner, and F. Kirchner, "pySPACE - a signal processing and classification environment in Python," *Frontiers in Neuroinformatics*, vol. 7, no. 40, 2013.

[48] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert, "xDAWN algorithm to enhance evoked potentials: Application to brain-computer interface,"

IEEE Transaction on Biomedical Engineering, vol. 56, no. 8, pp. 2035–2043, 2009.

[49] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, pp. 27:1–27, May 2011.

[50] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural computation*, vol. 12, no. 5, pp. 1207–1245, 2000.

[51] K. Veropoulos, C. Campbell, N. Cristianini *et al.*, "Controlling the sensitivity of support vector machines," in *Proceedings of the international joint conference on artificial intelligence*, 1999, pp. 55–60.

[52] S. Straube and M. M. Krell, "How to evaluate an agent's behavior to infrequent events? – reliable performance estimation insensitive to class distribution," *Frontiers in Computational Neuroscience*, vol. 8, no. 43, 2014.

[53] I. Iturrate, R. Chavarriaga, L. Montesano, J. Minguez, and J. Millán, "Latency correction of event-related potentials between different experimental protocols," *Journal of neural engineering*, vol. 11, no. 3, p. 036005, 2014.

[54] M. Spüler and C. Niethammer, "Error-related potentials during continuous feedback: using EEG to detect errors of different type and severity," *Frontiers in Human Neuroscience*, vol. 9:155, 2015.

[55] A. Kreilinger, C. Neuper, and G. R. Müller-Putz, "Error potential detection during continuous movement of an artificial arm controlled by brain–computer interface," *Medical & Biological Engineering & Computing*, vol. 50, no. 3, pp. 223–230, 2012.

[56] S. Bhattacharyya, A. Konar, and D. Tibarewala, "Motor imagery, p300 and error-related eeg-based robot arm movement control for rehabilitation purpose," *Medical & Biological Engineering & Computing*, vol. 52, no. 12, pp. 1007–1017, 2014.