Classifier Transferability in the Detection of Error Related Potentials from Observation to Interaction

Su Kyoung Kim*† and Elsa Andrea Kirchner *†
* Research Group Robotics,
University of Bremen, Bremen, Germany
† Robotics Inovation Center (RIC),
German Research Center for Artificial Intelligence (DFKI) GmbH,
Robert-Hooke-Strasse 5, 28359, Bremen, Germany
Email: {sukim, ekir}@informatik.uni-bremen.de

***Abstract*—A challenge in brain computer interface (BCI) applications is the reduction of time required for the acquisition of training data, which is needed for a user specific calibration of a BCI. This paper proposes an application oriented approach to minimize the calibration time by transferring a classifier between different types of error related potentials (ErrPs). A classifier trained to detect a certain brain pattern is used to later detect a brain pattern which is expected to be similar to the first one. In the here presented approach, two different tasks (interaction task/observation task) are performed within one scenario which is developed to generate two types of ErrPs: interaction ErrPs and observation ErrPs. Since almost twice as much training data can be generated while performing the observation task compared to the interaction task within the same calibration time, we use the classifier trained on the data containing observation ErrPs to evaluate it's performance on the data containing the interaction ErrPs. Presented results support our approach. We show that a single trial detection of interaction ErrPs using the classifier trained on observation ErrPs is possible and results on average in a high detection performance of 0.77 balanced accuracy [(TPR+TNR)/2], i.e., an average of recognition rate of correct and erroneous trials of 77%. Without such classifier transfer, the classification performance of observation and interaction ErrPs is on average slightly higher (0.80 and 0.81 balanced accuracy, respectively). Our results suggest that classifier transfer is feasible and reduces calibration time. This is a relevant result from the perspective of applying an ErrP-based brain-computer interface in a realistic scenario in robotics.**

***Index Terms*—Error related potentials (ErrPs), brain computer interface, classifier transfer, single trial detection.**

I. Introduction

Brain computer interfaces (BCIs) link the user's intent and an external device (computer, system, robot, etc.) by interpreting the user's brain signals measured by, e.g., the electroencephalogram (EEG). Using such interface the user's intent can be translated into a control signal for an external device [1], [2] with different aims of applications (see [3] for a review). Although the P300 has been used for most BCI applications [4]–[6], the investigation of error related potentials (ErrPs) has been the focus of research in recent years. It has not only been applied as a verification tool, e.g., during single trial P300 detection, but also as an independent tool for simple error monitoring or adaptation of erroneous behavior [7]–[13]. Hence, the ErrP can be used to improve the performance of an external device by correcting the errors.

Such improvement of performance of a human or artificial agent can be realized by single trial detection of ErrPs, which are differently elicited depending on the context of application. For example, ErrPs can be elicited in case that a) Own errors are recognized after a false response (*response error* [14]), b) Own errors or errors made by another agent (human or artificial agent) are recognized after receiving the feedback indicating the incorrectness of action (*feedback error* [15]), c) Errors of an agent or other human are recognized when observing their behavior (*observation error* [16], [17]), or d) Errors of the interface linking human and an external system can be recognized (*interaction error* [9]). The different types of errors elicit different brain signatures, i.e., event related potentials: a) response ErrP, b) feedback ErrP, c) observation ErrP, and d) interaction ErrP.

In BCIs, the interaction ErrP has often been applied to correct the interface. For example, the interaction ErrP can be used to improve the performance of an external device, in case that the external device (robot or other system) executes an action that violates the user's intent [9]. One reason for such a wrong action of the external device is a failure of the interface, i.e., the interface fails to interpret the user's intent and delivers a wrong command to the external device. When applying error monitoring, e.g., during the behavior of a robot, the robot's behavior can be corrected or suitably adapted with respect to the context of the situation/application by detecting ErrPs in the observers EEG (i.e., observation ErrP) [17].

The focus of this study was to investigate the applicability of interaction ErrPs and observation ErrPs in a realistic task environment (e.g., targets with a simple semantics have to be checked among the obstacles) for robotic applications and brain-computer interfaces. Furthermore, we investigate the transferability of a classifier between two types of ErrPs. We propose that a classifier transfer is useful to reduce calibration time in case that more training data can be collected for one type of ErrP compared to another type of ErrP within the same time of data collection. We found a work where a classifier transfer between two different tasks containing the same type of ErrP (observation ErrPs) was performed [13]. However, to our knowledge there is no study on the transfer of a classifier trained on one type of ErrP (e.g., observation ErrP) and tested on another type of ErrP (e.g., interaction ErrP).

To investigate the feasibility of classifier transfer we used the same scenario for the detection of interaction ErrPs and observation ErrPs. Hence, we developed one scenario, in which two different tasks could be performed to generate interaction ErrPs and observation ErrPs separately. Before investigating the classifier transferability, the performance of single trial detection of interaction ErrPs and that of observation ErrPs were separately evaluated. Afterwards interaction ErrPs were detected by using the classifier trained on the observation ErrPs. As mentioned above, this direction of classifier transfer (transfer from observation ErrP to interaction ErrP) was chosen because of the higher amount of training instances that could be recorded in the observation task within the same time of data collection.

In this paper, we present experimental results for four subjects during the monitoring of errors made by the interface (interaction ErrP) and when observing the actions of an artificial agent (observation ErrP). Based on the earlier mentioned research goal the main findings are structured in two parts: a) single performance: Performance of single trial detection of interaction ErrPs and observation ErrPs trained and tested without transfer and b) classifier transfer: Performance in single trial detection of interaction ErrPs with a classifier trained on examples of observation ErrPs.

II. Methods

A. Data Acquisition

EEGs were acquired from four subjects (one female, age: 27 ± 3.16, normal or corrected-to normal vision) during the monitoring of the erroneous behaviors in two different experimental environments. 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 $1\,\mathrm{kHz}$.

B. Experimental Setup

We developed a scenario which allows to detect two types of ErrPs (observation ErrP/interaction ErrP) separately, depending on the task that was performed (interaction task/observation task). The developed scenario was based on the same principle of the scenario from [9], in which interaction ErrPs were elicited during the monitoring of simulated errors of a classifier.

Unlike the scenario described in [9], our scenario was closer to a realistic, more application oriented one, since the task had to be performed with respect to simple rules. These rules were defined by target semantics, i.e., each target contained a simple semantics (labeled number) based on which the order of targets was defined. Further, there were restrictions in how to reach the targets. These restrictions were realized by obstacles. The task was to find an optimal way to reach the targets in an

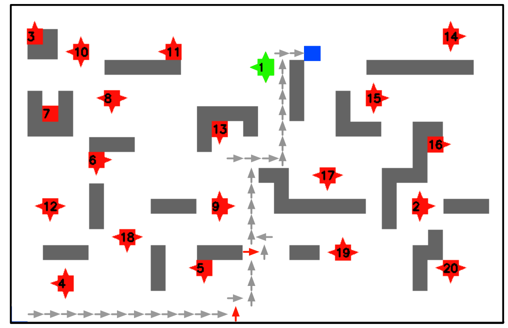


Fig. 1. Interaction task: 20 Targets placed among the obstacles (gray objects) and spikes of targets had to be reached in a numeric order. To move the cursor, subjects had to press four keys of a computer keyboard to move to the left, right, up or down (details, see text).

numeric order while avoiding obstacles. Due to this realistic character of the scenario we could not exclude errors made by the subject (i.e., response errors). To differentiate between both types of evoked ErrPs (evoked by response errors or interface errors), we labeled them differently as response error and as interface error. In this paper, we focused on the interaction ErrP, but not on the response ErrP.

Using this scenario two tasks (interaction task/observation task) were performed with a *counter-balanced* measure design. That means, subjects were divided into two groups. One group began with the observation task followed by the interaction task and vice versa. Since we collected 7 data sets for each task, the tasks were also *counter-balanced* within a subject. This means, each subject began with the one task (e.g. interaction task) followed by the other (e.g. observation task), and vice versa.

1) Interaction ErrP task: The task was to move the cursor (blue) to reach 20 targets (red) within a numerical order (see Fig. 1). After moving the cursor, its direction was depicted as gray arrow in the previous position of the cursor. All stimuli (cursor and targets) were displayed on a monitor placed approx. $30\,\mathrm{cm}$ in front of the subject. The subjects were instructed to bring the cursor to one of the 20 targets using four keys of a computer keyboard to move to the left, right, up or down. In case that a target was checked in the correct order, the color changed from red to green. Otherwise the target color remained red. To approach a realistic scenario, there were some obstacles placed between targets (gray objects). Further, the targets could only be reached from one side of a target. The other three sides were blocked by spikes (red). In case of touching these spikes, the cursor went back to the start position as a penalty. The task was finished after checking all 20 targets in the correct order. Interface errors that occurred in case that the actual cursor movement did not correspond to the chosen key that was pressed by the subjects were simulated with the probability of 9%. The possible directions of wrong movements were uniformly distributed. Wrong movements left traces depicted as red arrows (direction of errors). The task was repeated seven times and thus 7 data sets for each subject were recorded. Each set contained about 48 erroneous trials and 480

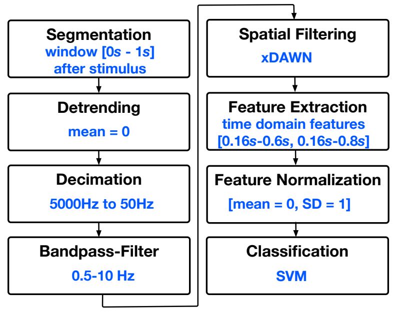


Fig. 2. Data flow: The continuous EEGs were segmented, normalized, decimated, band pass filtered, and the signal to noise ratio was enhanced by applying a spatial filter (xDAWN). The features that were extracted from the spatial filter were normalized and finally used to train the classifier. A support vector machine (SVM) was trained on the two classes: correct and erroneous events.

correct trials. To avoid the same task pattern, the target order was randomized for each run. All subjects needed about 2 minutes to reach all 20 targets (i.e., each run took 2 min). Three different labels for the classification were generated during the task: 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*): the errors made by the subject (e.g., touching the spikes of a target or violating the target order). In this study, we focused only on two labels: *Corr* and *InterErr*.

2) Observation ErrP task: The task was to observe the performance of an artificial agent. Unlike in the interaction task, not the subjects but an artificial agent performed the task. The subjects were only observing the behavior of the agent. As in the interaction task, all stimuli (cursor and targets) were displayed on a monitor placed approx. 30 cm in front of the subject. Cursor movements left traces depicted as gray arrows pointing towards the chosen direction. The errors of the agent could be recognized by the movements deviated from the correct path to reach the targets. Such wrong movements of the cursor left traces depicted as red arrows (directions of errors). The subjects could recognize the wrong movements of the cursor without developing and executing a strategy to find the correct path. The path to reach the targets and its deviation (errors) from the correct path were hard coded, in which 99 erroneous events were generated. To obtain the fixed determined amount of erroneous trials, the chosen path to reach the targets were not optimal compared to the path chosen by the subjects in the interaction task. The empirical ratio of error and correct trials after the task were 1:10 as for the interaction task. The speed of key pressings were also hard coded. Since subjects paused quite often to find the correct path, their average movement speed was significantly slower compared to the agent. As for the interaction task, 7 runs were collected (each run took 2 min) and the target order was randomized per run. For the observation task two labels were generated: a) correct trial (*Corr*): the movements that did not deviate from the path to reach the targets (i.e. correct movements) and b) erroneous trial (*ObsErr*): the movements that deviated from the path to reach the targets (i.e. wrong movements).

C. Data Set

We recorded seven data sets for each subject per task (interaction task/observation task).

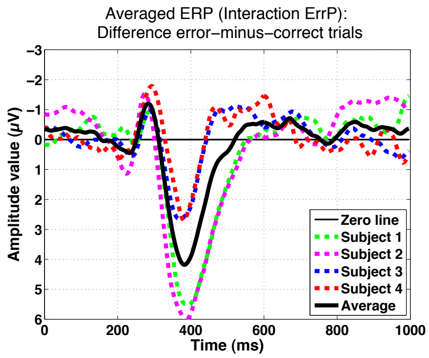
1) Single trial detection of interaction ErrP and observation ErrP: To enable a fair comparisons in detection performance between the interaction ErrP and observation ErrP, four data sets recorded during the interaction task were merged into one data set (total approx. 192 erroneous trials and 1920 correct trials, calibration time of 8 min) for each subject (inter-set design) and two data sets recorded during the observation task were merged into one data set (total 198 erroneous trials and 1980 correct trials, calibration time of 4 min) for each subject (inter-set design). In this way, an approximately equal number of erroneous trials were used for the evaluation for each task. Note that the ratio of correct and erroneous trials (1:10) were the same for the both tasks.

2) Single trial detection of interaction ErrP using the classifier trained on the observation ErrP: From the perspective of the application it is practical to use the data collected by the observation task to train the classifier, since the calibration time for detecting interaction ErrPs can be reduced compared to using the data collected by the interaction task to train the classifier. To test such possible advantage, we used one data set collected from the observation task (99 erroneous trials, calibration time of 2 min) for training the classifier and one data set collected from the interaction task (48 erroneous trials) for evaluation.

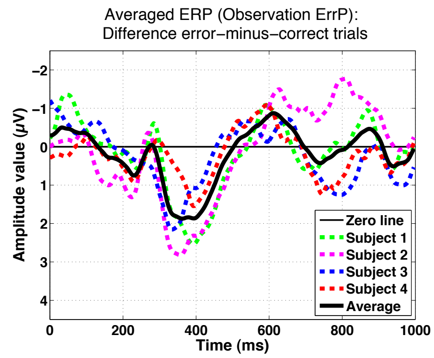
D. Preprocessing and Classification

Fig. 2 illustrates the data flow for preprocessing and classification. Based on the type of events (correct/erroneous trial) the continuous EEG signal was segmented into epochs from 0 ms to 1000 ms after each event type. 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 [18] was used as a spatial filter to enhance signal-to-noise ratio. After applying the xDAWN the number of 64 physical channels was reduced to 8 pseudo channels.

To determine whether a large time window is necessary for a successful detection of interaction ErrP and observation ErrP, respectively, we investigated the classification performance in case of using the features extracted from a large time window compared to the features from a small time window. As shown in Fig. 3, the ERP curve for each subject showed a more clear separation of second and third negativity peak for observation ErrPs compared to interaction ErrPs. For the observation ErrP, the first negativity and second positivity were reduced for Subject 1 and Subject 2, whereas the third negativity was increased for those subjects. Based on the pattern of the second and third negativity peak of interaction ErrP and observation ErrP for each subject, two time windows were used for



(a) Interaction ErrP



(b) Observation ErrP

Fig. 3. Averaged Event Related Potential (ERP) for the difference error-minus-correct trials at channel FCz for each subject and the grand average over all subjects. Only artifact-free EEG trials were used. a) Interaction ErrP: A first negative peak was observed around 270 ms after the erroneous events, followed by a positive peak around 380 ms. After that, a negative peak occurred around 600 ms after the erroneous events. b) Observation ErrP: Similar to the interaction ErrP, a first negative peak occurred around 250 ms after the erroneous events, followed by a positive peak around 350 ms. Also two negative peaks were observed around 600 ms and 850 ms after the erroneous events. For Subject 1 and Subject 2, the first negativity and second positivity were reduced for the observation ErrP compared to the interaction ErrP, whereas the third negativity was increased for the observation ErrP.

TABLE I

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 THAT TWO TIME WINDOWS WERE USED FOR FEATURE EXTRACTION: 0.16 S–0.6 S AND 0.16 S–0.8 S

	Interaction ErrP					Observation ErrP				
	Training instances (erroneous/correct): approx. 192/1920 calibration time of 8 min					Training instances (erroneous/correct): 198/1980 calibration time of 4 min				
0.16–0.6 s	Subject 1	Subject 2	Subject 3	Subject 4	Average	Subject 1	Subject 2	Subject 3	Subject 4	Average
bACC	0.79±0.06	0.81±0.06	0.82±0.04	0.80±0.06	0.81±0.01	0.79±0.04	0.83±0.05	0.78±0.05	0.77±0.06	0.79±0.03
TPR	0.70±0.12	0.70±0.11	0.72±0.10	0.72±0.11	0.71±0.01	0.76±0.10	0.75±0.12	0.72±0.10	0.66±0.11	0.72±0.05
TNR	0.88±0.04	0.92±0.02	0.93±0.03	0.89±0.03	0.91±0.02	0.83±0.04	0.92±0.03	0.84±0.03	0.88±0.03	0.87±0.04
0.16–0.8 s	Subject 1	Subject 2	Subject 3	Subject 4	Average	Subject 1	Subject 2	Subject 3	Subject 4	Average
bACC	0.78±0.06	0.81±0.05	0.82±0.04	0.80±0.06	0.81±0.02	0.81±0.05	0.85±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	0.70±0.01	0.77±0.10	0.75±0.10	0.71±0.10	0.69±0.11	0.73±0.04
TNR	0.87±0.04	0.93±0.02	0.92±0.03	0.89±0.03	0.90±0.03	0.85±0.04	0.94±0.02	0.84±0.03	0.91±0.03	0.89±0.05

feature generation: a) [0.16 s–0.6 s] and b) [0.16 s–0.8 s]. Thus, features were extracted from 8 channels after spatial filtering, between 0.16 s and N s where N = [0.6, 0.8], for a total of 240 features (8 channels × 30 data points= 240) for first window [0.16 s–0.6 s] and 320 features (8 channels × 40 data points = 320) for second window [0.16 s–0.8 s].

The extracted features were used to train the classifier. We used a linear support vector machine (SVM) [19] to classify the correct and erroneous trials. For each training, SVM parameters were optimized with an internal 5-fold cross validation using a grid search among the determined complexity values of the SVM $[10^0, 10^{-1}, \ldots, 10^{-6}]$. Due to the unbalanced ratio of the *erroneous* and *correct* trials (1:10), we determined the class weight of 5.

E. Evaluation

As a metric for classification performance we used the arithmetic mean of true positive rate (TPR) and true negative rate (TNR), so-called balanced accuracy (*bACC*), where the erroneous trials belonged to the positive class.

First, we evaluate the performance of single trial detection of interaction ErrP and observation ErrP separately. For evaluation 10×10-fold cross validation was performed on the merged data set for each task (interaction task/observation task). The merged data set was split into 9 training sets and 1 validation set. For training, total approx. 192 erroneous trials and 1920 correct trials (calibration time of 8 min) were used for the interaction task and total 198 erroneous trials and 1980 correct trials (calibration time of 4 min) for the observation task. To find whether a small 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 two different types of ErrPs for each time window. For that the classification performance was analyzed using repeated measures ANOVA with *time window* [0.16 s–0.6 s, 0.16 s–0.8 s] and *ErrP type* (observation, interaction), and *subject* (subject 1–subject 4) as within-subjects factors.

Second, we evaluate the classifier transferability between two different types of ErrPs. In our case, the classifier transfer

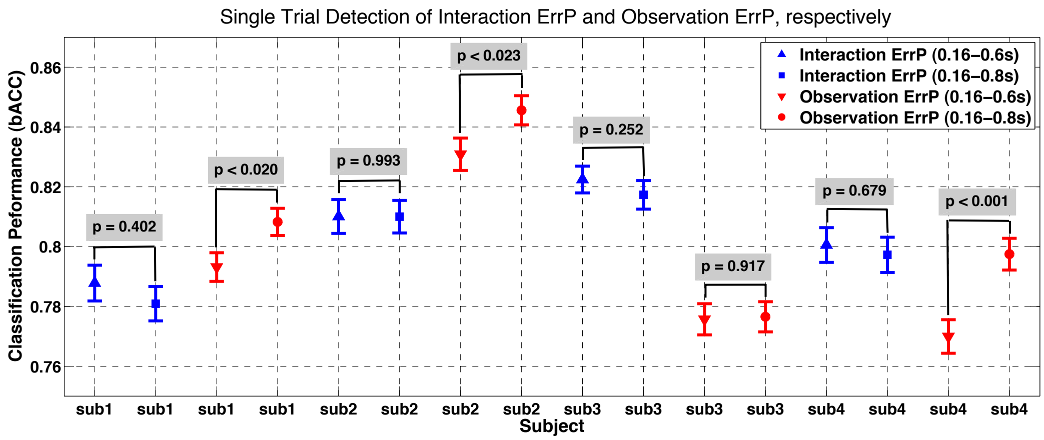


Fig. 4. Comparison in classification performance (bACC) between the two time windows [0.16-0.6 s, 0.16-0.8 s] for each type of ErrP [interaction ErrP/observation ErrP] and for each subject, respectively (interaction of classification performance with *time window* and *subject*). For pairwise comparisons, Bonferroni correction was applied. Note: mean balanced accuracy (bACC = [(TPR+TNR/2)]) with standard error were presented for each time window and each type of ErrP (blue triangle: first time window for interaction ErrP, blue square: second time window for interaction ErrP, red reversed triangle: first time window for observation ErrP, red circle: second time window for observation ErrP).

from the observation ErrP to the interaction ErrP was of interest, since we could collect more data from the observation task compared to the interaction task for the same duration of data collection per run. Thus, the classifier was trained on the data containing the observation ErrP, collected by one set of the observation task (99 erroneous trials, calibration time of 2 min). After that, the trained classifier was used to evaluate the data containing the interaction ErrP collected by one set of interaction task (48 erroneous trials). To compare the single trial detection of interaction ErrP using a classifier trained on the data containing the interaction ErrP (8 min calibration time) compared to the using a classifier trained on only one data set containing the observation ErrP (2 min calibration time), the data was analyzed using repeated measures ANOVA with *transfer* (transfer, no transfer) and *subject* (subject 1–subject 4) as within-subjects factors.

III. Results

A. *Classification performance for each type of ErrP*

Table I and Fig. 4 show the classification performance on correct and erroneous single trials for two different types of ErrPs (interaction ErrP/observation ErrP) in the same scenario.

We obtained a classification performance for interaction ErrPs and observation ErrPs with an average balanced accuracy of 0.81 and 0.80, respectively. Across subjects and time windows, there was no significant difference in classification performance between interaction ErrP and observation ErrP [main effect of *ErrP type*: $F(1, 99) = 1.337, p = 0.25$].

For the observation task, a higher classification performance for the second time window [0.16 s–0.8 s] was achieved compared to the first time window [0.16 s–0.6 s] except for one subject [interaction of *time window* with *ErrP type* and *subject*: $F(3, 297) = 1.17$, $p = 0.32$, first window vs. second window: the statistical values, see Fig. 4]. The difference in classification performance between first and second window was not observed for the interaction task (an average balanced accuracy of 0.81 for both time window, details, see Fig. 4).

Based on this result, only the first window was selected to later detect interaction ErrPs using a classifier trained on the data containing observation ErrPs. By selecting the small time window [0.16 s–0.6 s] we could also reduce the dimensionality of feature space which could be relevant for an application.

TABLE II
CLASSIFICATION PERFORMANCE (MEAN±SD) OF INTERACTION ERRP USING THE CLASSIFIER TRAINED ON THE OBSERVATION ERRP. NOTE THAT THE TIME WINDOW FOR FEATURE EXTRACTION WAS 0.16–0.6 S.

Classifier Transfer: Observation ErrP $\rightarrow$ Interaction ErrP					
Training instances (erroneous/correct): 99/990 calibration time of 2 min					
	Subject 1	Subject 2	Subject 3	Subject 4	Average
bACC	0.82±0.01	0.70±0.04	0.76±0.01	0.81±0.01	0.77±0.06
TPR	0.73±0.01	0.55±0.13	0.58±0.03	0.80±0.02	0.67±0.12
TNR	0.91±0.01	0.84±0.06	0.94±0.01	0.82±0.02	0.88±0.06

B. *Classification performance for classifier transfer*

Table II show the classification performance on correct and erroneous single trials collected during the interaction task, where a classifier trained on data collected during the observation task was used.

We obtained an average balanced accuracy of 0.77 across all subjects. The success of classifier transfer from the observation ErrP to the interaction ErrP was subject-specific: For two 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*: $F(3, 27) = 11.81$, $p < 0.001$, classifier transfer vs. no classifier transfer: the statistical values, see Fig. 5]. In other words, for two subjects we obtained a high

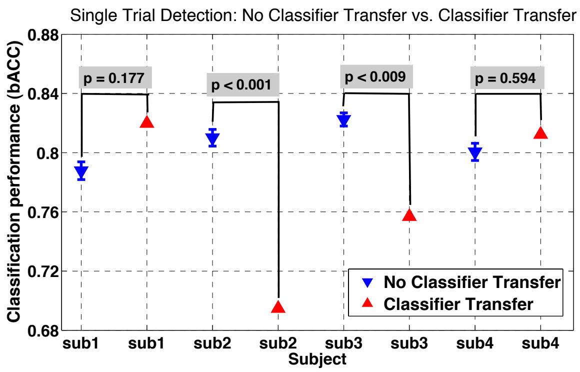


Fig. 5. Comparison in classification performance (mean and standard error) between two cases: no classifier transfer and classifier transfer. The standard error was very low for the case of classifier transfer. For pairwise comparisons, Bonferroni correction was applied.

classification performance using the classifier transfer with the calibration time of 2 min which was as good as in the case of no classifier transfer with the calibration time of 8 min.

IV. Conclusion

In this study, we have achieved a high performance in single trial detection of interaction ErrP and observation ErrP, respectively (an average balanced accuracy of 0.80 and 0.81, respectively) in a more realistic, application oriented scenario if compared to the study described in [9]. The result from two different lengths of time windows proved that the small 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 the larger time window [0.16 s–0.8 s].

Furthermore, our approach shows the feasibility of classifier transfer, i.e., the applicability of a classifier that is trained on observation ErrPs to classify interaction ErrPs. Since we could collect twice as much training data containing the observation ErrP compared to the interaction ErrP during the same time of data collection in our scenario, it was reasonable to test whether the classifier trained on observation ErrPs could be used for single trial detection of interaction ErrPs. Our results indicate a successful application of a classifier trained on the data containing the observation ErrP to the data containing the interaction ErrP.

Although the performance of detecting interaction ErrPs in case of using the classifier trained on observation ErrPs compared to using the classifier trained on interaction ErrPs is more subject-specific, from the perspective of application such classifier transfer is very useful to reduce the calibration time, i.e., two minutes of EEG recording is sufficient to calibrate the system for some subjects who show no significant difference between transfer case and no transfer case (*no classifier transfer* with calibration time of 8 min vs. *classifier transfer* with calibration time of 2 min: 0.79 vs. 0.82 for subject 1, 0.80 vs. 0.81 for subject 4, see details Results and Table I Interaction ErrP vs. Table II).

Acknowledgment

This work was funded by the German Ministry of Economics and Technology (grant no. 50 RA 1011 and grant no. 50 RA 1012). We thank Anett Seeland, Marc Tabie, and Yohannes Kassahun for constructive comments.

References

[1] 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.

[2] K.-R. Müller, M. Krauledat, G. Dornhege, G. Curio, and B. Blankertz, “Machine learning techniques for brain-computer interfaces,” *Biomedical Engineering*, vol. 49, pp. 11–21, 2004.

[3] 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 Neuroprosthetics*, vol. 4, no. 161, pp. 1–15, 2010.

[4] D. Krusienski, E. Sellers, F. Cabestaing, S. Bayoudh, D. McFarland, T. Vaughan, and J. Wolpaw, “A comparison of classification techniques for the P300 speller,” *Neural Engineering*, vol. 3, pp. 229–305, 2006.

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

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.

[13] 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, pp. 1–12, 2013.

[14] 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.

[15] C. Holroyd and M. Coles, “The neural basis of human error processing: Reinforcement learning, dopamine and the error-related negativity,” *Psychological Review*, vol. 109, no. 4, pp. 679–709, 2002.

[16] H. T. van Schie, R. B. Mars, M. G. H. Coles, and H. Bekkering, “Modulation of activity in medial frontal and motor cortices during error observation,” *Nature Neuroscience*, vol. 7, no. 5, pp. 549–554, 2004.

[17] I. Iturrate, L. Montesano, and J. Minguez, “Single trial recognition of error-related potentials during observation of robot operation,” in *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, 2010, pp. 4181–4184.

[18] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert, “xDAWN algorithm to enhance evoked potentials: Application to brain-computer interface,” *IEEE Transaction Biomedical Engineering*, vol. 56, no. 8, pp. 2035–2043, 2009.

[19] 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. 27:1–27:27, May 2011.