

# On the use of brain decoded signals for online user adaptive gesture recognition systems

Kilian Förster<sup>1</sup>, Andrea Biasiucci<sup>2,3</sup>, Ricardo Chavarriaga<sup>2</sup>, José del R. Millán<sup>2</sup>, Daniel Roggen<sup>1</sup>, and Gerhard Tröster<sup>1</sup>

<sup>1</sup> ETH Zurich, IFE, Wearable Computing Lab  
CH-8092 Zurich, Switzerland

<sup>2</sup> EPFL, CNBI, Center for Neuroprosthetics,  
CH-1015 Lausanne, Switzerland

<sup>3</sup> University of Genova

Department of Informatics, Systems and Telematics (DIST),  
16126 Genova, Italy

{foerster,roggen,troester}@ife.ee.ethz.ch  
{andrea.biasiucci,ricardo.chavarriaga,jose.millan}@epfl.ch

**Abstract.** Activity and context recognition in pervasive and wearable computing ought to continuously adapt to changes typical of open-ended scenarios, such as changing users, sensor characteristics, user expectations, or user motor patterns due to learning or aging. System performance inherently relates to the user’s perception of the system behavior. Thus, the user should be guiding the adaptation process. This should be automatic, transparent, and unconscious.

We capitalize on advances in electroencephalography (EEG) signal processing that allow for error related potentials (ErrP) recognition. ErrP are emitted when a human observes an unexpected behavior in a system. We propose and evaluate a hand gesture recognition system from wearable motion sensors that adapts online by taking advantage of ErrP. Thus the gesture recognition system becomes self-aware of its performance, and can self-improve through re-occurring detection of ErrP signals.

Results show that our adaptation technique can improve the accuracy of a user independent gesture recognition system by 9.58% when ErrP recognition is perfect. When ErrP recognition errors are factored in, recognition accuracy increases by 3.29%. We characterize the boundary conditions of ErrP recognition guaranteeing beneficial adaptation. The adaptive algorithms are applicable to other forms of activity recognition, and can also use explicit user feedback rather than ErrP.

## 1 Introduction

Human activity and gesture recognition from body worn motion sensors using machine learning techniques [1] enables activity based computing [2].

**Motivation** Activity recognition systems are trained in a user-independent manner for ‘out of the box’ operation. Training data is collected from multiple

subjects to build generic statistical activity models. Exhaustive data collection is time consuming and may not be practical. The recognition of simple activities may already be difficult in a user independent manner [3]. As activities get more complex, this becomes a major challenge. Some highly complex gesture recognition systems weren't even trained for user independence [4].

User specific models usually perform better than user independent models but are less able to generalize to new subjects [5, 6, 3]. They are trained on the target user to reflect individual characteristics. This individual training phase may not be practical when deploying a system.

User independent and user specific systems are trained once at design time, respectively during first use, and remain static throughout operation. Thus they are not able to adapt to so far unseen situations typical of open-ended scenarios. These systems also have no knowledge about their instantaneous performance, as it was characterized at training time on a specific dataset. Therefore no action can be taken if runtime performance drops. This can be important for example in critical applications, where stopping the context-aware system may be better than letting it operate with reduced performance.

**Contribution** Activity and context recognition in pervasive and wearable computing ought to continuously adapt to changes typical of real-world applications, such as a new user of the system, changing sensor characteristics, changing user expectations, or changing user motor patterns due to learning or aging. System performance inherently relates to the user's perception of the system behavior. Thus, the user should be guiding the adaptation process. This should be automatic, transparent, and unconscious.

In order to guide adaptation according to the user's run-time expectation, a feedback signal is required. We capitalize on advances in electroencephalography (EEG) signal processing that allow for error related potentials (ErrP) recognition. ErrP occur when a human observes an unexpected behavior in a system [7–9]. We propose and evaluate a hand gesture recognition system from wearable motion sensors that adapts online by taking advantage of ErrP. Essentially the activity recognition system turns into an autonomous system with performance self-awareness and self-improvement capabilities.

In this work, we focus on user specific adaptation from a user-independent model through ErrP signal occurrences. Specific contributions include:

- An experimental setup (gesture based HCI scenario) that allows the joint investigation of activity recognition, ErrP detection, and the combination of both into an autonomously adaptive activity recognition system.
- A dataset of EEG signals, electro-oculography (EOG), hand acceleration and electro-myography (EMG), with 18'000+ gesture instances on 7 subjects.
- The baseline ErrP detection and non-adaptive gesture recognition accuracy.
- A method to estimate instantaneous recognition performance from ErrP.
- A comparative analysis of three strategies to adapt the gesture recognition system to a specific user from a user independent model based on ErrP.

**Paper content** In section 2 we review adaptive approaches applied to activity recognition and explain the nature of ErrP. In section 3 we describe the experimental setup we use to investigate adaptive activity recognition driven by ErrP. In section 4 we present the ErrP detection results. In section 5 we show how instantaneous system performance can be derived from ErrP signals. We comparatively evaluate user adaptation strategies driven by ErrP signals in section 6. We discuss results in section 7 and conclude in section 8.

## 2 State of the art

**Adaptation strategies and limitations** Adaptive techniques can improve the performance for individual users without affecting generalization capability. A user-independent model can adapt to a specific user during a short calibration phase. This has been investigated in handwriting [10] and speech [11] recognition, where it remains a major research topic [12]. Similar approaches were proposed in gesture recognition [13]. Calibration-based approaches are time consuming with large number of activities and activity models remain static after calibration. In dynamic model selection a pre-existing model that best corresponds to the current user or his environment is selected at run-time. This has been applied in speech processing [14]. In activity recognition it has been used to adapt to the user’s on-body sensor placement preferences, by selecting models corresponding to the automatically detected sensor location [15]. Such approaches require extensive training data to build multiple models. Other adaptation techniques rely on the unsupervised tracking of clusters of activities in the feature space [16]. While devised for sensor placement adaptation, similar principles may apply to user adaptation. Such approaches can adapt at run-time because they rely on underlying data structure properties. However they do not guarantee to adapt in a way that reflects the user’s perception of system performance.

Current adaptation strategies do not take into account the user’s perception of the system’s behavior. Guiding adaptation according to the user’s run-time expectation requires a feedback signal. Explicit interaction may provide this feedback, such as a button that is pressed when the behavior of the context-aware system is not satisfactory. In the vision of wearable and pervasive computing, however, feedback should be transparent: automatic and unconscious.

**Brain signals related to unexpected action perception** Several studies have suggested the existence of a neural system responsible for error processing [17]. Specifically, stereotypical electrophysiological signals have been consistently reported to appear as a response to erroneous actions [18] or unexpected action outcomes [9]. These signals, —termed *Error-related negativity* (ERN) and *Feedback-related negativity* (FRN)— are characterized by a negative deflection of the EEG signals in fronto-central areas of the scalp, followed by a centro-parietal positive peak. Typical signal latencies are 50 to 100ms in the case ERNs and around 250ms for FRNs. Neurophysiological studies have provided evidence of error-based learning. Specifically, it has been suggested that these signals reflect conscious error processing; post-error adjustment of response strategies [18], and reward-based adaptive behavior [9].

Moreover, research on Brain-Computer Interfaces (BCI) has shown that it is possible to recognize EEG error-related signals (ErrP) on single trials above random levels [19–21]. Based on this fact, these signals have been proposed to be used to correct erroneous motor action in speed-response human-computer interaction [21], as well as to increase the information transfer rate of EEG-based BCI systems [19]. Experimental measures taken over different time periods (up to two years) show that these potentials are stable over time, despite the delay between recordings. Current protocols for EEG signal analysis require motionless subjects to avoid that EMG signals (1-30mV) from muscle activity contaminate the subtler EEG signals (10-100 $\mu$ V) [22]. In order to use EEG system in naturalistic settings, however, researchers now start to investigate limited subject mobility.

### 3 ErrP-based adaptive gesture recognition scenario

We investigate the use of ErrP to guide the adaptation of a gesture recognition system in an HCI scenario. This scenario is based on a game to maintain the user’s involvement during experimental sessions [23,19]. It is designed so that a large number of gesture instances can be acquired in a comparatively short amount of time. It allows movements of the user’s arm with limited amplitude, to investigate EEG signal analysis in more realistic situations than state of the art EEG protocols. EEG signal and hand acceleration from a wearable sensor is recorded during the scenario to assess adaptation strategies in offline simulations.

**Gesture-controlled computer game** The subjects played a computerized version of a “memory game” consisting of 8 image pairs (fig 1). The 16 images are randomly distributed in a four by four matrix and hidden behind question marks. The subjects have to find identical pairs of images, which are then removed from the screen. If two images are flipped they are hidden again before a new image can be selected. The game is finished when all image pairs were correctly found.

The game input interface is based on five hand gestures. Left, right, up and down hand movements shift the image selection cursor in the corresponding direction. Each directional gestures starts and ends at a central home position. Flipping an image is controlled by closing and opening the hand.

**Measurement setup** The online recognition of the gestures is based on light barriers and a reed switch. This ensures accurate gesture recognition for the collection of a reference dataset. Three horizontal and three vertical infrared light barriers detect the hand position (see fig. 1). The closing gesture is detected from a reed switch on the subjects hand activated by a magnet on the subjects fingers.

A tri-axial acceleration sensor at the subjects fingertips records the motion of the hand for offline acceleration-based gesture recognition. The acceleration sensor is sampled at 64 Hz and connected via USB to the experiment computer. This computer also ran the memory game. Another computer recorded EEG, as well as arm EMG and eye movements using EOG with the Biosemi ActiveTwo

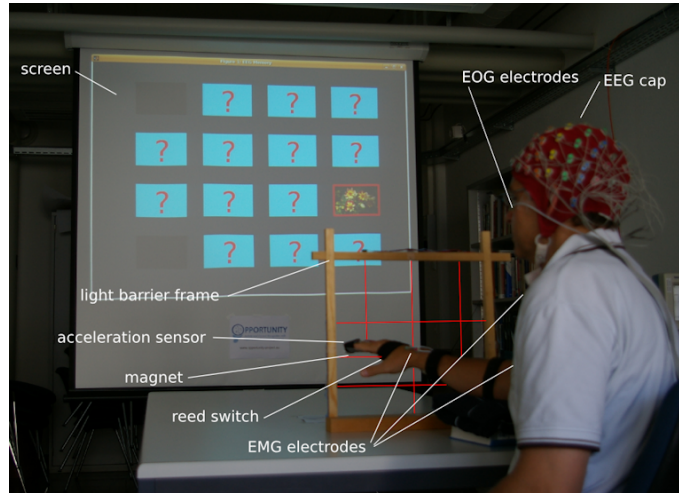


Fig. 1: The computer game is presented on the screen; the light-barrier frame, magnet and reed switches capture game control gestures; the acceleration sensor, EMG electrodes (right wrist, biceps and shoulder), EOG electrodes, and EEG electrode cap stream data to a PC for recording and offline analysis.

system with active electrodes. EMG and EOG allow for motion artifacts cancellation by adaptive filtering. Both computers were interconnected to ensure a synchronized data recording using a shared data line.

**Experimental protocol** Seven healthy male subjects aged 25 to 47 participated. For each subject we recorded 14 sessions with a duration of three to five minutes. One session corresponds to one memory game. Between recording sessions the subjects could rest for one to two minutes. We recorded more than 2700 hand gestures per subject. The experiment lasted about two hours per subject including setup.

In each session we randomly artificially induced between 5% and 33% of gesture recognition errors to provoke ErrP events. In an error case the game selects a random command instead of the user command. For example if the subject closes his hand to turn a card, the card is not turned but instead the cursor is moved in a random direction.

## 4 EEG-ErrP Single-Trial Recognition

Following previous studies [23, 19], we perform classification using the time signal of electrodes FCz and Cz as input features for a Bayesian filter [24], since EEG ErrP are characterized by a fronto-central distribution along the midline. EEG potentials were spatially filtered by subtracting from each electrode the average potential (i.e. the common average reference) at each time step to suppress average brain activity and keep the information from the local sources below each

electrode. A 1–10-Hz bandpass filter was applied as ErrPs are relatively slow cortical potentials [25]. EEG signals were subsampled to 64Hz before classification, which is based on temporal features. The input vector for the classifier (described below) is composed by the time samples on electrodes FCz and Cz within a fixed time window after the feedback onset.

At each sampling time step, the Bayesian filter estimates the state probabilities according to the observations and the previous state estimations. In this case, we have discrete observations of a continuous EEG signal and we want to find the state for the action shown on the screen, i.e. an erroneous or correct movement.

To build the Bayesian filter, we define two possible states at each time  $t$ :  $S_t = 1$  for erroneous recognition, and  $S_t = 0$  for correct recognition. At each sampling time step  $t$  observations  $O_t$  are given by a vector with components  $FCz$  and  $Cz$  corresponding to the electrodes of the same name:  $O_t = [FCz_t, Cz_t]$ . Observations and states from time zero to  $T$  are respectively noted  $O_{0:T}$  and  $S_{0:T}$ .

A transition model is defined by a first order Markov hypothesis for states over time:  $P(S_t|S_{0:t-1}) = P(S_t|S_{t-1})$  for  $t = 0 \dots T$ . Since the state during a single trial doesn't change, the transition model corresponds to the identity matrix:  $P(S_t|S_{t-1}) = 1$  if  $S_t = S_{t-1}$  and zero otherwise.

The sensor model is given by the probability distribution  $P(O_t|S_t)$  which predicts observations given the state. Then the decomposition of the joint probability is given by:

$$P(S_{0:T}O_{0:T}) = P(S_0)P(O_0|S_0) \prod_{t=1}^T (P(S_t|S_{t-1})P(O_t|S_t)) \quad (1)$$

The classification consists in estimating  $P(S_t|O_{0:t})$ , i.e. the probability of the state (error or correct) knowing the observations (EEG activity). It can be obtained in a recurrent manner; first, a *prediction* (2) of the state is done based on the transition model and then, second, the state *estimation* (3) is computed based on the sensor model.

$$P(S_t|O_{0:t-1}) = \sum_{S_{t-1}} (P(S_t|S_{t-1})P(S_{t-1}|O_{0:t-1})) \quad (2)$$

$$P(S_t|O_{0:t}) \propto P(O_t|S_t)P(S_t|O_{0:t-1}) \quad (3)$$

Given the identity transition matrix, the *prediction–estimation* recurrent calculus is simplified:

$$P(S_t = 1 | O_{1:t}) \propto P(O_t|S_t)P(S_{t-1} = 1 | O_{1:t-1}) \quad (4)$$

And correspondingly for  $P(S_t = 0 | O_{1:t})$ . Being  $Q_t$  be the quotient of the probabilities for both states, an erroneous trial is detected when  $\ln(Q_t)$  is positive, where  $\ln(Q_t) = \ln(Q_{t-1}) + \ln(P(O_t|S_t = 1)) - \ln(P(O_t|S_t = 0))$ .

Estimations from both channels are combined using a naive fusion,

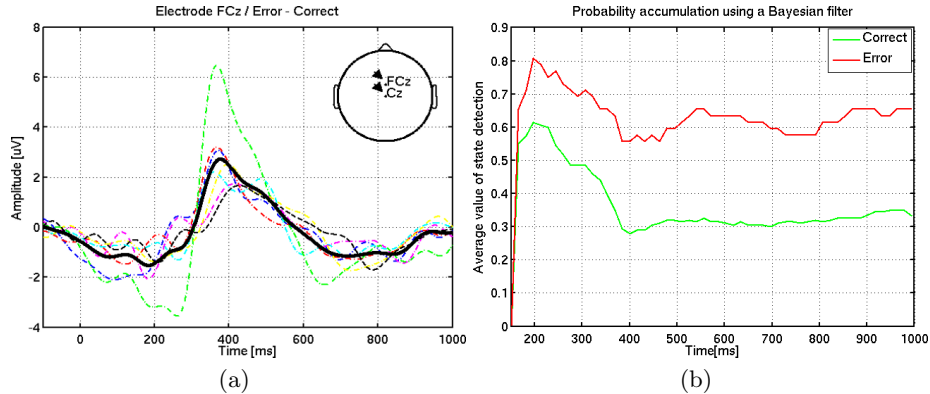


Fig. 2: (a) Grand average error-related potential on the FCz electrode, Error minus Correct condition, (*thick line*); individual subject averages are shown with *thin dashed lines*. Time ( $t=0$ ) is measured from the feedback onset. Electrodes positions are shown in the scalp plot. (b) Average value of the state detection for correct and erroneous trials of session 1 on subject V.

$$P(O_t|S_t) = P(FCz_t|S_t)P(Cz_t|S_t) \quad (5)$$

The sensor model  $P(O_t|S_t)$  is defined by a Gaussian distribution with a mean  $\mu_t$  and a variance  $\sigma_t^2$ , these parameter were estimated using the training dataset. Having two input channels and two possible states, we have four Gaussian distributions at each time  $t$ , and eight parameters to identify. This approach updates the estimated state probability as new samples are available, an example of the average estimated probability at different time points. Fig. 2 shows the average EEG activity in channel FCz (error-minus correct condition), as well as the average state estimation for both classes at different time points.

We divided the recorded dataset into seven folds of EEG activity for each subject; every single fold corresponds to two consecutive memory games. The classifier was trained using a Leave-One-Fold-Out Cross Validation, i.e. training with 6 folds, and testing on the 7th, then averaging results for all folds. We consider the activity of electrodes in the [150, 1000] ms time windows after the feedback presentation, estimating the state probabilities according to the observations and taking a choice at the time instant that maximizes State Recognition. Table 1 shows classification results for the described technique; we should stress the fact that the ErrP recognition can be affected by EMG artifact contamination due to subject movements. This indeed a major challenge in the integration of EEG activity in pervasive applications. Different filtering techniques can be applied for reducing such contamination [26], and its use will be object of further study.

	Subjects							Average
	I	II	III	IV	V	VI	VII	
<i>Sensitivity</i>	0.74	0.56	0.60	0.63	0.57	0.65	0.58	0.62
<i>Specificity</i>	0.48	0.71	0.73	0.59	0.75	0.65	0.63	0.65

Table 1: Sensitivity and specificity for the seven subjects using Leave-One-Fold-Out Cross Validation.

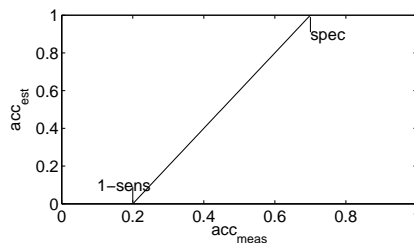


Fig. 3: The estimated accuracy vs. the measured accuracy for a specific specificity and sensitivity level.

## 5 Performance self-awareness through ErrP detection

A run-time system performance measure can be obtained from the detection of ErrP events. We define  $N_{err}$  the number of ErrP events and  $N_{gest}$  the number of executed gestures during a period of operation. The ErrP detection specificity and sensitivity is  $spec_{ErrP}$  and  $sens_{ErrP}$ , characterized during system training. We assume that sensitivity and specificity is stationary throughout operation (i.e. no fluctuations of  $spec_{ErrP}$  and  $sens_{ErrP}$ ). In this case we can estimate the true accuracy of the gesture recognition as follows:

$$acc_{est} = \frac{acc_{meas} - 1 + sens_{ErrP}}{spec_{ErrP} + sens_{ErrP} - 1} = \frac{1 - \frac{N_{err}}{N_{gest}} - 1 + sens_{ErrP}}{spec_{ErrP} + sens_{ErrP} - 1} \quad (6)$$

The specificity and sensitivity of the ErrP detection control slope and the offset of the dependency between  $acc_{meas}$ , the measured accuracy based on the ErrP signals, and  $acc_{est}$ , the estimated true accuracy of the system, as depicted in figure 3. As  $spec_{ErrP}$  and  $sens_{ErrP}$  get lower, slight errors in the measurement of  $acc_{meas}$  will have more effect on the estimated true accuracy (due to the steeper line slope in the figure).

## 6 ErrP-based gesture recognition adaptation

ErrP signals indicate when the action taken by a system is erroneous (see table 1), and thus whether a gesture was wrongly recognized in our game scenario. The open questions are what are the adaptive strategies suitable to incorporate the information provided by ErrP into a gesture recognition system.

**User-specific adaptation scenario** We consider an adaptation scenario where a user independent gesture recognition system is adapted to a specific user through ErrP occurrences. The gesture recognition system is trained in a user independent manner in the HCI scenario (i.e. it uses a user independent classifier  $C_{init}$ ). The system is then given to a so far unseen user. Each gesture performed by this user is classified by the gesture recognition system. EEG analysis indicates whether the action taken by the computer game, and thus the classification of the gesture, was correct or wrong. We adapt the gesture classification system through online learning (see below).

**Gesture classification** We distinguish the five game control gestures based on the hand acceleration. We segment the signal using the gesture-start and gesture-end signal provided by the light-barrier frame. During training of classifiers, the ground truth label of gesture instances was provided by the light-barrier frame. We did no dataset cleaning or outlier removal as this would not be possible in a real application of this kind.

For each gesture, we calculate the following acceleration features on three windows (full gesture, first half and second half of the gesture): mean, standard deviation, minimum, maximum and energy. We do this on the three axes of the acceleration signal as well as on its magnitude. In addition the correlation for each axes pair  $xy$ ,  $xz$  and  $yz$  is calculated. This yields 63 features. We perform a probabilistic feature selection [27] combined with a scatter search [28] to select a feature subset. This yields 6 features: the mean and the standard deviation on the  $x$  axis, the mean on the  $y$  axis, the minimum of the magnitude, the mean of the first half of the magnitude and the correlation between  $x$  and  $z$  axes.

We classify the gestures with the following classifiers: Naive Bayes [29], Bayes Networks [30] and  $k$  Nearest Neighbor (kNN) [31] implemented in the Weka Machine Learning Project [32]. For the kNN classifier we chose  $k = 13$  as it gave sufficiently good results for all subjects. A higher value for  $k$  would also increase the minimum number of training instances. We use a batch approach [33] for on-line learning.

**Strategies to exploit EEG** The absence of ErrP indicates that the classification result of the system is correct. We assume this result is the ground truth class label of the gesture. The presence of ErrP indicates a wrong classification, but does not provide indication of the class label. Therefore during operation we can collect labeled user specific samples (those where ErrP was not detected). We investigate three strategies to create user adapted classifiers. Essentially all strategies start from a user independent classifier  $C_{init}$  and operate by collecting a user specific training set  $S_x$ . They then train a user adapted classifier  $C_x$  on this set. The strategies differ in the way the training set is collected:

1. **AD 1 “Incremental knowledge integration”:** Starting from a user independent training set  $S_1 = S_{init}$ , the user adapted training set  $S_1$  grows by including a new (user specific) gesture instance whenever the gesture performed by the user is recognized and no ErrP is detected (see algorithm 1).

---

**Algorithm 1** Adaptation strategy 1

---

```
1: Initialize dataset  $S_1$  with  $n_{init}$  subject independent instances
2: Train classifier  $C_1$  on the subject independent initial training set  $S_1$ 
3: for each user specific instance  $i$  do
4:   classify instance  $i$  using  $C_1$  to class  $c$ 
5:   if no error detected then
6:     add instance together with label  $c$  to  $S_1$ 
7:     retrain  $C_1$  on the new  $S_1$ 
8:   end if
9: end for
```

---

After a new gesture is collected a classifier  $C_1$  is trained on  $S_1$  and replaces  $C_{init}$ .

2. **User specific batch training:** a batch of user specific training samples  $S_{2x}$  is collected whenever a gesture performed by the user is recognized and no ErrP is detected. There are two training sample collection variants:
  - (a) **AD 2a:** the user independent classifier  $C_{init}$  classifies the gestures (see algorithm 2)
  - (b) **AD 2b:** the classifier created with adaptation strategy 1 (AD1) classifies the gestures (see algorithm 3)

---

**Algorithm 2** Adaptation strategy 2a

---

```
1: Initialize dataset  $S_{init}$  with  $n_{init}$  subject independent instances
2: Initialize dataset  $S_{2a}$  to the empty set
3: Train classifier  $C_{init}$  on the subject independent initial training set  $S_{init}$ 
4: for each user specific instance  $i$  do
5:   classify instance  $i$  using  $C_{init}$  to class  $c$ 
6:   if no error detected then
7:     add instance  $i$  together with label  $c$  to  $S_{2a}$ 
8:     if  $S_{2a}$  contains sufficient instances then
9:       train classifier  $C_{2a}$  on  $S_{2a}$ 
10:    end if
11:  end if
12: end for
```

---

To train the user independent classifier  $C_{init}$  we combine the data of all subjects, leaving out the subject we want to adapt to. From this combined dataset we select randomly  $n_{init}$  instances which are used for the training. The data of the left out subject is split into an *adaptation set* and a *test set*. The *adaptation set* contains 2250 instances while the *test set* contains 500 instances. We purposely preserve the timely order of the data instances in the *adaptation set* to simulate the adaptation as close to reality as possible. During operation, the instances in the *adaptation set* are iteratively presented to the system for classification and adaptation.

---

**Algorithm 3** Adaptation strategy 2b

---

```
1: Initialize dataset  $S_1$  with  $n_{init}$  subject independent instances
2: Initialize dataset  $S_{2b}$  to the empty set
3: Train classifier  $C_1$  on the subject independent initial training set  $S_1$ 
4: for each user specific instance  $i$  do
5:   classify instance  $i$  using  $C_1$  to class  $c$ 
6:   if no error detected then
7:     add instance  $i$  together with label  $c$  to  $S_1$ 
8:     add instance  $i$  together with label  $c$  to  $S_{2b}$ 
9:     retrain  $C_1$  on the new  $S_1$ 
10:    if  $S_{2b}$  contains sufficient instances then
11:      train classifier  $C_{2b}$  on  $S_{2b}$ 
12:    end if
13:  end if
14: end for
```

---

### 6.1 Adaptation assuming perfect ErrP detection

For the following simulation we assume a perfect ErrP detection with a sensitivity and a specificity of 1. In figure 4 we show the evolution of the classification accuracies for the different adaptation strategies over time. Due to space constraints we limit ourselves to the Naive Bayes classifier, the other classification methods show the same trends. Each plot shows adaptation from a different number of initial subject independent training instances  $n_{init}$ . Every 45 iterations the accuracy of each classifier with respect to the subject dependent *test set* is given. Two baselines represent the performance of a non-adaptive subject independent and subject dependent classifier. The subject independent classifier is trained on the initial training set with  $n_{init}$  instances. The subject dependent classifier is trained on the *adaptation set*. It indicates the upper bound an adapted user specific classifier can achieve if all the user specific information would be available during the adaptation process.

The classifiers built based on adaptation strategy 1 show a benefit over the subject independent baseline. The larger  $n_{init}$  the smaller the gain in accuracy is, and adding more subject dependent instances (i.e. going towards higher iteration counts) shows less improvement than the other adaptation strategies.

The classifiers built based on adaptation strategies 2a and 2b outperform the subject independent baseline and also adaptation strategy 1. The adapted subject dependent classifier achieves an accuracy 4-8% higher than the subject independent classifier. As the number  $n_{init}$  of initial subject independent training instances increases, the 2a and 2b adapted classifiers reach a higher accuracy.

To show the effect of the teacher signal we also added the result for adaptation strategy 1 in a setting where every classification result is assumed to be correct and no teacher is present. Without the teacher the adaptation leads to a classifier performing worse than the user independent classifier.

To further the benefit of the additional ErrP information we also show a baseline for adaptation strategy 1 in a setting where every classification result is

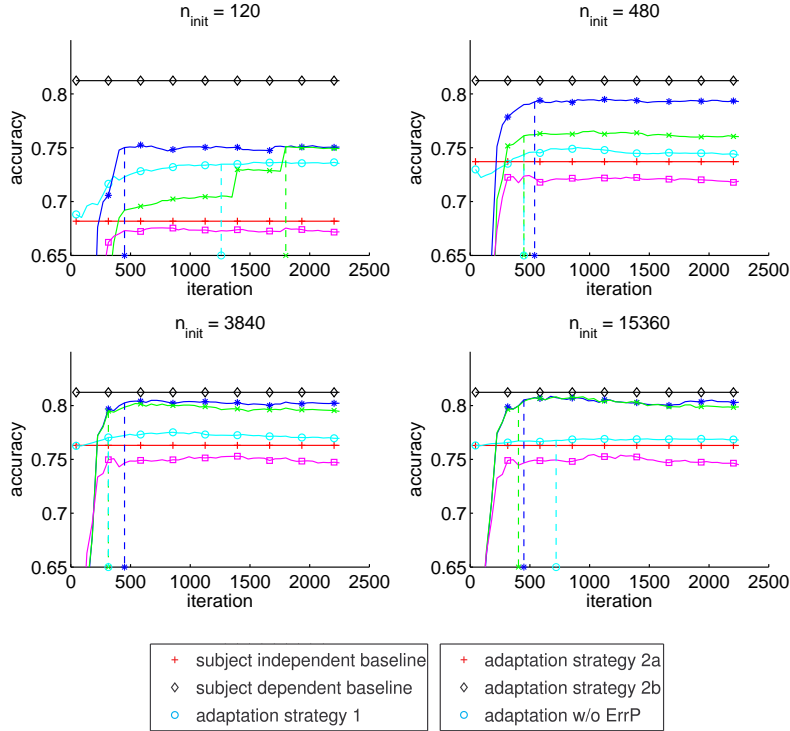


Fig. 4: Accuracies of the classifier built based on adaptation strategies 1, 2a and 2b over time for four different numbers of instances  $n_{init}$  in the initial training set. At each iteration the adapted classifiers are tested on the subject dependent *test set*. The dashed lines mark the point where 99.9% of the accuracy after all iterations is reached.

assumed to be correct - simulating the absence of ErrP. Without ErrP the adaptation leads to a classifier performing worse than the user independent classifier in this setting.

The marker in the plots show the points where the 1, 2a and 2b adapted classifiers reach 99.9% of the accuracy they achieve after all iterations. This indicates how many iterations are sufficient to build a good adapted classifier.

The results averaged over all subjects and over five simulation runs from different random seeds are listed in table 2. In all cases at least one of the user adaptation strategies leads to a classifier outperforming the subject independent classifier. The best adaptation is achieved with either strategy 2a or 2b. The average improvement achieved by the best adaptation strategy over the subject independent case is 5.1% for the Naive Bayes, 9.5% for the Bayes Network and 7.8% for the kNN.

The Symbols (+, \*) next to recognition accuracy indicate for how many of the subjects the increase in performance provided by the adaptation strategy is statistically significant compared to the subject independent accuracy (T-Test,  $p < 0.05$ , null hypothesis is that the performance after adaptation is identical to the subject independent accuracy).

	$n_{init}$	SID	AD 1		AD 2a		AD 2b	
Naive Bayes	120	68.2	73.5	(1260)	<b>75.1</b> <sup>+</sup>	(450)	74.9	(1800)
	480	73.7	74.3	(450)	<b>79.3</b> *	(540)	76.0	(450)
	SD = 81.2	3840	76.3	77.0	(315)	<b>80.2</b> *	(450)	79.5* (315)
	15360	76.3	76.8 <sup>+</sup>	(720)	<b>80.3</b> *	(450)	79.9* (405)	
Bayes Network	120	67.7	<b>82.2</b> <sup>+</sup>	(2025)	75.6*	(405)	<b>82.2</b> *	(1845)
	480	73.6	80.8*	(2205)	80.1*	(765)	<b>82.1</b> *	(2160)
	SD = 90.8	3840	75.8	80.2*	(2115)	81.8*	(540)	<b>83.4</b> *
	15360	77.1	80.0*	(2115)	83.9*	(585)	<b>84.6</b> *	(630)
kNN k = 13	120	69.0	75.0*	(2250)	77.4*	(1845)	<b>79.9</b> *	(2250)
	480	76.6	70.3*	(45)	82.9*	(2250)	<b>84.6</b> *	(2250)
	SD = 92.3	3840	80.1	85.9*	(2250)	85.4*	(2250)	<b>86.1</b> *
	15360	81.8	86.7*	(2205)	87.2*	(2160)	<b>87.9</b> *	(2250)

Table 2: Perfect ErrP detection: Accuracies achieved with the subject dependent (SD), subject independent (SID), 1 adapted (AD 1), 2a adapted (AD 2a) and 2b adapted (AD 2b) classifiers in %. The values in brackets give the number of iterations until 99.9% of the final accuracy is reached. (<sup>+</sup>: significant increase in accuracy for at least 3 out of 7 subjects, \*: significant increase in accuracy for at least 5 out of 7 subjects)

## 6.2 Adaptation using experimental ErrP detection performance

We present the results obtained from the experimentally measured ErrP detection accuracy. ErrP detection is challenging in this setup and the average sensitivity is 0.65, the average specificity 0.62, over all subjects and sessions, which is only slightly above chance. We repeated the simulations from section 6.1 taking into account the inaccuracies in the error detection. The simulation results are listed in table 3.

For all values of  $n_{init}$  at least one adapted classifier performs better than the subject independent classifier. The gain is however marginal in several cases and certain adapted classifier perform worse than the subject independent one.

The average improvement achieved by the best adapted classifier over the subject independent classifier is 0.4% for the Naive Bayes, 3.4% for the Bayes Network and 2.4% for the kNN.

## 6.3 Influence of the ErrP detection accuracy on the adaptation

The ErrP recognition performance is a key parameter for a successful adaptation. A perfect ErrP recognition leads to a performance of the user adapted classifiers higher than the user independent classifier. With the ErrP performance experimentally achieved in our setup the improvement is comparatively lower. Typical EEG ErrP recognition algorithms can be adjusted towards increased specificity or sensitivity following a ROC curve. By understanding the range of ErrP recognition sensitivity and specificity values where the adaptation of the gesture recognition shows benefit it becomes possible to adjust the ErrP recognition parameters along the ROC curve to ensure a benefit.

We consider the adaptation to be beneficial when the user adapted classifier performs significant better for at least 3 out of 7 subjects. We consider the 2a

	$n_{init}$	SID	AD 1		AD 2a		AD 2b	
Naive Bayes	120	68.2	67.0	(45)	<b>69.1</b>	(450)	67.0	(1800)
	480	73.7	68.4	(45)	<b>74.2</b>	(450)	69.3	(405)
	SD = 81.23	3840	76.3	74.2	(45)	<b>76.6</b> <sup>+</sup>	(585)	74.9
	15360	76.3	76.0	(45)	<b>76.3</b> <sup>+</sup>	(315)	75.3 <sup>+</sup>	(315)
Bayes Network	120	67.7	72.8	(1710)	70.0	(585)	<b>73.2</b> <sup>+</sup>	(2070)
	480	73.6	76.7 <sup>+</sup>	(1305)	75.3	(540)	<b>77.0</b> <sup>+</sup>	(900)
	SD = 90.8	3840	75.8	77.1	(1575)	77.4	(855)	<b>77.6</b> <sup>+</sup>
	15360	77.1	78.0	(720)	<b>80.0</b> *	(855)	79.7*	(720)
kNN k = 13	120	69.0	70.0	(1035)	<b>71.4</b>	(1260)	66.5	(315)
	480	76.6	76.7 <sup>+</sup>	(90)	<b>78.3</b> <sup>+</sup>	(990)	76.6	(495)
	SD = 92.3	3840	80.1	<b>82.9</b> <sup>+</sup>	(2025)	82.1 <sup>+</sup>	(2250)	82.5 <sup>+</sup>
	15360	81.8	<b>84.5</b> *	(2205)	83.5 <sup>+</sup>	(2205)	81.4*	(855)

Table 3: Experimental ErrP detection: Accuracies achieved with the subject dependent (SD), subject independent (SID), 1 adapted (AD 1), 2a adapted (AD 2a) and 2b adapted (AD 2b) classifiers in %. The values in brackets give the number of iterations until 99.9% of the final accuracy is reached. (+: significant increase in accuracy for at least 3 out of 7 subjects, \*: significant increase in accuracy for at least 5 out of 7 subjects)

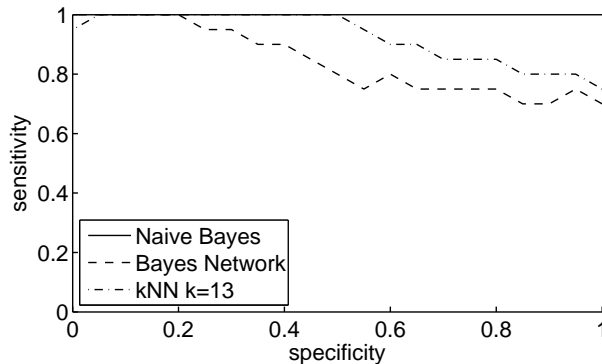


Fig. 5: The curves for each classification method give the ErrP detection performance for which the 2a adapted classifier improves significantly for at least three subjects over the subject independent classifier. For every point to the right or above these curves a benefit can be expected when adapting based on a ErrP detection system offering this performance.

adaptation strategy only with  $n_{init} = 480$  instances in the initial dataset to reduce the computation effort.

Figure 5 shows the decision line between sensitivity and specificity pairs beyond which the 2a adaptation strategy is beneficial. Any point right or above the line describes an ErrP detection sensitivity or specificity that allows for beneficial adaptation. The Bayes Network classifier is the most robust with respect to ErrP recognition performance. The Naive Bayes and the kNN classifier require better performance from the ErrP detection for the adaptation to be beneficial. All algorithms are more sensitive to a low sensitivity than to a low specificity.

## 7 Discussion

In this work we assess, for the first time, single-trial recognition of EEG error-related potentials in a complex, realistic task. This contrasts with previously reported experiments where these signals were studied using very simple stimuli, and subjects movements were restricted to minimize motion-related artifacts in the EEG signal [23, 19, 24]. The difference in the experimental protocol - i.e. subject moving during the recording, complex visual feedback, different cognitive demand of the experimental task -, together with the intrinsic variability, noise, and non-stationarities of brain signals, may explain the low classification accuracies obtained in the current study. Nevertheless, it should be considered that it is not possible to achieve perfect decoding of brain generated signals due to several reasons (i.e. low signal-to-noise ratio, EEG non-stationarity, muscular contamination). Indeed, classification performance for ErrP recognition in much simpler, controlled experiments is approximately 80% for both classes [23, 19]. Therefore our results are encouraging.

Despite the low ErrP recognition performance it was still possible to use this additional source of information to successfully adapt the gesture recognition system towards a specific user. The gain in accuracy achieved by the adaptation is depending on the ErrP recognition. The better the ErrP recognition performs the more improvement in the gesture classification can be expected. This is true for all classification methods we investigated. The Bayes Network seemed to be the most robust classification method regarding the adaptation from non-perfect ErrP recognition.

As we rely on a subject independent gesture recognition system as a basis for our adaptation it is indispensable that this initial system reaches a certain recognition performance. If for example the initial system could not at all recognize one specific class it would not be possible to build a new subject dependent classifier, as this class would be missing in the collected subject dependent data.

In our experimental setup for data collection we assume that the subject intention is correctly captured by the gesture recognition. There might still be cases where the subject performs a wrong gesture by mistake. This mistake might also be reflected in the brain signal as an error. As we do not capture the users intention directly we can not assess the influence of user mistakes.

The gesture recognition errors are added artificially and randomly so that the user can not adapt to it to improve the gesture recognition. Therefore the simulated improvements of the gesture recognition are independent of potential user adaptation.

One can arg that the simulations based on offline data are not meaningful for a live system as other effects, like user adaptation, may come into play. In a life system it is very difficult to investigate all parameters essential for such an adaptation scenario, though.

Even though the ErrP adapted classifier reaches a promising accuracy gain, the performance of a classifier trained in a pure subject dependent manner is still not reached. This can be explained by the fact that the adaptation only uses the instances which are correctly classified by the subject independent classifier. The instances which are too different and therefore not covered by the subject

independent model are excluded. These excluded instances potentially contain information important for building a good user dependent model.

In the adaptation schemes we propose we do not make use of confidence values which could be provided by the ErrP detection. These confidence values may be used to weight the instances for the adaptation process.

The online learning we made use of was based on batch learning. Batch learning in general puts higher memory requirements on a system, compared to incremental learning, as all training instances in the batch have to be stored. Especially for wearable systems with limited memory and processing capabilities incremental learning should be considered for online learning and adaptation.

## 8 Conclusion

We have investigated strategies for user adaptation within a gesture based HCI scenario making use of additional information provided by EEG ErrP analysis. To our knowledge, this is the first attempt to use brain signals related to the perception of errors for the improvement of activity recognition systems. Simulations of a perfect decoding of such signals show that theoretically the recognition accuracy can be increased by up to 9.58% over the user independent classifier. Using single-trial recognition of actual EEG data recorded during the gesture based HCI experiment, the accuracy increase for the adapted gesture recognition reached 3.29% in the best case. This shows that brain signals (i.e. EEG) generated during real human-computer interaction provide information that can be integrated into the activity recognition chain so as to improve its performance.

EEG-based user adaptation remains unlikely in real-world scenarios in the near future given the current state of the sensing technology, its sensitivity to motion artifacts, and the desire for invisible wearables. Miniaturized sensing platforms may become available [34], however there are also many professional occupations that require to wear a helmet or head protection gear (e.g. firefighters, soldiers, surgeons, pilots). In this case the integration of EEG within the head apparel can be envisioned. Since these are usually high stakes professions, a continuous self-monitoring of wearable system performance and its improvement over time may be strong factors supporting the inclusion of such technology. In general there are many potential applications, ranging from disabled people to entertainment [35], which could benefit from “human in the loop” strategies.

An immediate outcome of this work, however, is the comparative evaluation of user adaptation strategies, that are applicable to other forms of user feedback. For instance, a button integrated in a smart shirt or a user interface element could be used to signal a non-desired behavior triggering system adaptation.

Besides using the strategies presented here to adapt a generic classifier to a specific user, they may also be used to deal with changing user preferences or non stationarities, either using implicit EEG-based feedback, or explicit feedback.

In future work we plan to use the recorded EMG and EOG to filter out muscular artifacts that may contaminate the signals used for classification. This may lead to an increase in recognition performance and a higher robustness to contaminations.

To improve the user adaption we further plan to investigate how online learning methods can make additional use of user specific instances which were wrongly classified. Those instances may add valuable information to the adaptation process.

## References

1. Ward, J.A.: Activity Monitoring: Continuous Recognition and Performance Evaluation. PhD thesis, ETH Zurich (2006) Nr. 16520.
2. Davies, N., Siewiorek, D.P., Sukthankar, R.: Special issue: Activity-based computing. *IEEE Pervasive Computing* **7**(2) (2008) 20–21
3. Ravi, N., Dandekar, N., Mysore, P., Littman, M.L.: Activity recognition from accelerometer data. *American Association for Artificial Intelligence* (2005)
4. Stiefmeier, T., Roggen, D., Ogris, G., Lukowicz, P., Tröster, G.: Wearable activity tracking in car manufacturing. *IEEE Pervasive Computing Magazine* **7**(2) (2008) 42–50
5. Bao, L., Intille, S.S.: Activity recognition from user-annotated acceleration data. In: *Pervasive Computing: Proc. of the 2nd Int Conference*. (2004) 1–17
6. Lester, J., Choudhury, T., Borriello, G.: A practical approach to recognizing physical activities. In: *Lecture Notes in Computer Science : Pervasive Computing*. (2006) 1–16
7. Nieuwenhuis, S., Ridderinkhof, K.R., Blom, J., Band, G.P., Kok, A.: Error-related brain potentials are differentially related to awareness of response errors: Evidence from an antisaccade task. *Psychophysiology* **38**(5) (Sep 2001) 752–760
8. Yasuda, A., Sato, A., Miyawaki, K., Kumano, H., Kuboki, T.: Error-related negativity reflects detection of negative reward prediction error. *Neuroreport* **15**(16) (Nov 15 2004) 2561–2565
9. Frank, M.J., Woroch, B.S., Curran, T.: Error-related negativity predicts reinforcement learning and conflict biases. *Neuron* **47**(4) (Aug 2005) 495–501
10. Santosh, K.C., Nattee, C.: A comprehensive survey on on-line handwriting recognition technology and its real application to the nepalese natural handwriting. (2009)
11. Tang, Y., Rose, R.: Rapid speaker adaptation using clustered maximum-likelihood linear basis with sparse training data. *IEEE Transactions on Audio, Speech, and Language Processing* **16**(3) (2008) 607–616
12. Baker, J.M., Deng, L., Glass, J., Khudanpur, S., C.-H., L., Morgan, N., OShaughnessy, D.: Research developments and directions in speech recognition and understanding, part 1. *IEEE Signal Processing Magazine* **26**(3) (2009) 75–80
13. Ohmura, R., Hashida, N., Imai, M.: Preliminary evaluation of personal adaptation techniques in accelerometer-based activity recognition. In: *Proc. 13th IEEE Int. Symposium on Wearable Computers: Late Breaking Results*. (2009)
14. He, X., Zhao, Y.: Fast model selection based speaker adaptation for nonnative speech. *IEEE Trans. on Speech and Audio Processing* **11**(4) (2003) 298–307
15. Kunze, K., Lukowicz, P.: Using acceleration signatures from everyday activities for on-body device location. *Wearable Computers, 2007 11th IEEE International Symposium on* (Sep 2007) 115 – 116
16. Förster, K., Roggen, D., Tröster, G.: Unsupervised classifier self-calibration through repeated context occurrences: Is there robustness against sensor displacement to gain? In: *Proc. 13th IEEE Int. Symposium on Wearable Computers (ISWC)*. (2009) 77–84

17. Taylor, S.F., Stern, E.R., Gehring, W.J.: Neural systems for error monitoring: Recent findings and theoretical perspectives. *Neuroscientist* **13**(2) (2007) 160–172
18. Falkenstein, M., Hoormann, J., Christ, S., Hohnsbein, J.: ERP components on reaction errors and their functional significance: A tutorial. *Biol Psychol* **51**(2-3) (Jan 2000) 87–107
19. Ferrez, P.W., Millán, J.: Error-related EEG potentials generated during simulated brain-computer interaction. *IEEE Trans Biomed Eng* **55** (2008) 923–929
20. Schalk, G., Wolpaw, J.R., McFarland, D.J., Pfurtscheller, G.: EEG-based communication: Presence of an error potential. *Clin Neurophysiol* **111**(12) (Dec 2000) 2138–2144
21. Parra, L.C., Spence, C.D., Gerson, A.D., Sajda, P.: Response error correction—A demonstration of improved human-machine performance using real-time EEG monitoring. *IEEE Trans Neural Syst Rehabil Eng* **11**(2) (June 2003) 173–177
22. Fatourehchi, M., Bashashati, A., Ward, R.K., Birch, G.E.: EMG and EOG artifacts in brain computer interface systems: A survey. *Clin Neurophysiol* **118**(3) (Mar 2007) 480–494
23. Chavarriaga, R., Ferrez, P.W., Millán, J.: To err is human: Learning from error potentials in brain-computer interfaces. In: *International Conference on Cognitive Neurodynamics*. (2007)
24. Bollon, J.M., Chavarriaga, R., Millán, J., Bessière, P.: EEG error-related potentials detection with a Bayesian filter. In: *4th International IEEE EMBS Conference on Neural Engineering*, Antalya Turkey (2009)
25. Gehring, W.J., Goss, B., Coles, M.G.H., Meyer, D.E., Donchin, E.A.: Neural system for error-detection and compensation. *Psychol Sci* **4** (1993) 385–390
26. Schlögl, A., Keinrath, C., Zimmermann, D., Scherer, R., Leeb, R., Pfurtscheller, G.: A fully automated correction method of EOG artifacts in EEG recordings. *Clin Neurophysiol* **118**(1) (Jan 2007) 98–104
27. Liu, H., Setiono, R.: A probabilistic approach to feature selection - a filter solution, *Morgan Kaufmann* 319–327
28. García Lopez, F., García Torres, M., Melian Batista, B., Moreno Perez, J.A., Moreno-Vega, J.M.: Solving feature subset selection problem by a parallel scatter search. *European Journal of Operational Research* **169**(2) (March 2006) 477–489
29. John, G., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: *In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann (1995) 338–345
30. Castillo, E., Gutiérrez, J.M., Hadi, A.S.: *Expert Systems and Probabilistic Network Models*. Erste edn. Springer, New York, NY, USA (December 1996)
31. Aha, D.W., Kibler, D.: Instance-based learning algorithms. In: *Machine Learning*. (1991) 37–66
32. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations* (The Morgan Kaufmann Series in Data Management Systems). 1st edn. Morgan Kaufmann (October 1999)
33. Tsymbal, A.: The problem of concept drift: Definitions and related work. Technical report, Department of Computer Science, Trinity College (2004)
34. Casson, A., Smith, S., Duncan, J., Rodriguez-Villegas, E.: Wearable EEG: what is it, why is it needed and what does it entail? In: *Proc IEEE Eng Med Biol Soc*. (2008) 5867–5870
35. Garipelli, G., Galán, F., Chavarriaga, R., Ferrez, P.W., Lew, E., Millán, J.: The use of Brain-Computer Interfacing for Ambient Intelligence. In: *Intl Workshop on Human Aspects in Ambient Intelligence*. (2007)