Online single trial ERN detection as an interaction aid in HCI applications

Chi Thanh Vi
Interaction & Graphics
University of Bristol
chi.vi@bristol.ac.uk

Sriram Subramanian
Interaction & Graphics
University of Bristol
sriram@compsci.bristol.ac.uk

Abstract
Error-related negativity (ERN) is a form of an Event Related Potential signal which can be triggered in the brain when a user either makes a mistake or the application behaves differently from the expectation of the user. The first step to harness the benefits of ERN in HCI applications is to detect the patterns in real-time on a single trial basis. In this paper we present our initial results in detecting ERN. Using a logistic regression technique, we have achieved a 70% recognition rate of erroneous and correct single trials. We then explored several designs, e.g. using ERN to help a user in the moment of bad decision in a map navigation task. Through multiple designs and careful user testing, we aim to identify guidelines and design principles that can help HCI researchers to include ERN as an interaction aid in their applications

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Keywords
Brain Computer Interface, Error Related Negativity, Single Trial Detection

ACM Classification Keywords
H5.2. Information interfaces and presentation (e.g., HCI): User Interfaces (D.2.2, H.1.2, I.3.6).

General Terms
Human Factors, Design

Introduction
Error Related Negativity (ERN) is a form of Event Related Potentials (ERPs - brain responses to an internal or external stimulus). An ERN is produced when a user is aware of the obvious error(s) that he or she has made; either through the system feedback or individual realization [4]. It also appears but with lower amplitude when a user is confused about the decision he or she has made [3]. Since the late 80s, researchers have developed different analysis methods to detect ERN to confirm their existence and nature [3].

However, most of the methods focus on the offline results. These methods are highly efficient in detecting a clear pattern of ERN. Nevertheless, from a HCI's point of view, ERN is useful when it can be detected and applied right after its occurrence. There is also little knowledge about designing guidelines for ERN in HCI. This paper will investigate these problems further.

The use of Error Related Negativity

Nowadays, both disabled and able-users can benefit from BCI systems as an alternative communication method with the outside world. However, even a well designed BCI system cannot guarantee 100% of success in interpreting the users' intended goals. This is because of the misunderstandings between users and machines that may arise on either side [1]. These misunderstandings may reduce the user's performance and irritate them while interacting.

This problem can be overcome by using ERN to predict and then correct those mismatches. From examples of using ERN in the P300 speller [7] and imaginary movements [4], it can be concluded that ERN may be used as an integrated error detection and correction module. This module runs in parallel and in an open feedback loop with the user interface. Consequently, HCI systems have the ability to cross validate the result or predict intuitively the user's intended goals.

c

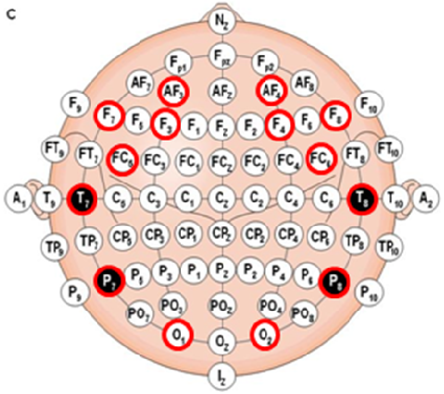

Figure 1: Emotiv neuroheadset electrode positions

Single trial ERN detection – Why and How?

Like other ERPs, ERN was discovered using the conventional averaging method [3]. This method provides an easier way to achieve a clear pattern as noise is reduced by the procedure.

However, in HCI, ERN will be the most useful when it is used immediately after it is detected. Hence, there must be a technique to recognize this pattern when capturing EEG which is known as the online detection mechanism. One solution that was tried was caching and averaging few patterns in order to retrieve a reasonably clear ERN pattern. Nevertheless, as ERN is a manifestation of error awareness, it may take a long time to have collected enough signals. The caching procedure may delay the correction progress as well. Therefore, it is crucial to detect this pattern using only single trial in real-time as closely as possible to the user's moment of decision.

This can be done by applying a moving time window through continuous EEG signals then pass them to a feature extractor and classifier. The output will indicate whether there is an ERN pattern or not.

Method

Linear Discrimination

Assuming x(t) is one window of values from the channel M. We calculated the coefficient matrix **v** so that $\mathbf{y(t)} = \mathbf{v}^T * \mathbf{x(t)}$ is maximally discriminated between two types of responses: correct (non-error) trial and incorrect (error) trial. The coefficient matrix **v** is calculated by applying a logistic regression technique [2] on parts of the data. Other parts were used for testing on each trial to calculate the accuracy of the online prediction algorithm.

Procedure

We collected data from two participants on a standard Flanker Task with 4 blocks and 40 trials per block. The procedure of each trial is as follow, a fixation cross was displayed in the center of the screen for 500 ms. It was then replaced with one of four stimuli (<<<<<, >>>>>, <<><<, >><>>). The stimulus was presented for 100 ms after which the screen was cleared. The screen remains clear until 500 ms after the participant’s response. At a viewing distance of

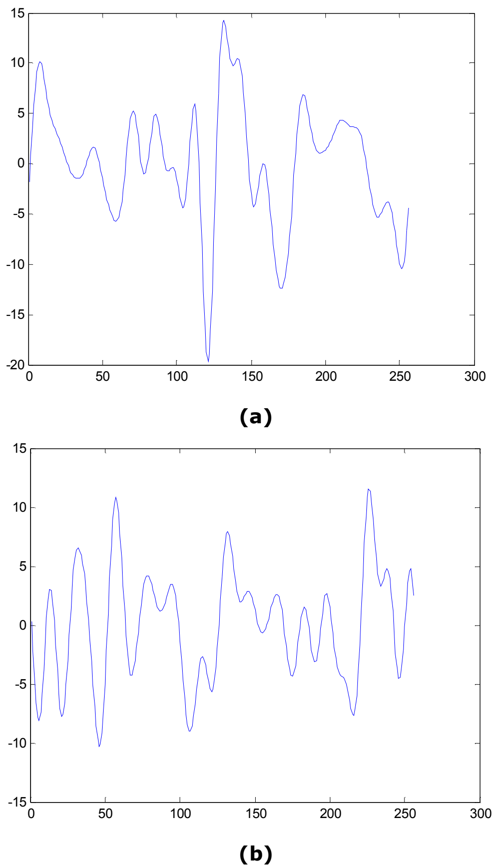

Figure 2: Examples of single trial ERN for two cases: incorrect (a) and correct (b). The x axis is number of samples (1 sec = 128 samples). The y axis is amplitude (micro volt)

around 100 cm, the visual angle of the arrow stimuli was 0.4° vertically, 0.6° horizontally, and space between them was 0.3°. Participants had 2 minutes to rest after each block. They also were instructed to sit comfortably minimize eye movement and blink as infrequently as possible while performing the task.

Signal Processing

The EEG signals, captured from an Emotiv headset, were divided into 2s epochs (1000 ms before and 1000 ms after the key press moment). The first 200ms were used to remove DC offset following which all epochs were filtered in 1-10 Hz to remove components that are not in the ERN frequency bands. Figure 2 shows examples of single trials ERN in the two cases when the trial was incorrect (Fig 2a) and correct (Fig 2b).

For each channel, half of the trials were used for training via a logistic regression technique and other half were for testing. Out of the 160 test trials we found that using data from F4 channel, we can discriminate two types of responses: roughly 70% of erroneous trials were classified as incorrect and about 70% of correct trials were classified as correct. This initial result suggests that we can get ERN detection rates that can be acceptable for interactive applications where the detection rates can be further improved through careful feedback to the user.

Next Steps

To date most experiments involving ERN detection are done on Flanker tasks. Some studies showed that non-Flanker tasks such as speaking one's second language under time pressure [5] or playing musical instruments [6] can also elicit ERN. However, these studies focused on detecting ERN in a different environment instead of using ERN as a source of input for a HCI system.

To examine the applicability of our algorithm in a non-Flanker task we plan to run a study involving map navigation. In our pilot study we will investigate ERN detection in a first person maze walking game with the ultimate aim of exploring use of ERN in mobile map navigation tasks. At the beginning of the game, participants will see a top-down view of the maze showing all the paths and the escape route which will involve 40 turning points to complete the maze in the shortest time possible. After this, the view changes to a first person's perspective. The users are expected to navigate through the maze based on the escape route showed earlier.

Our algorithm will detect the elicited ERN signals while users are navigating through the maze. A gauge will appear when ERN signal is detected. This will serve as a form of indication to inform the users that they have potentially encountered a confused (or incorrect) decision during the game. The users need to press the space bar if they disagree with the gauge's message.

Discussion

Challenges of using ERN in HCI

The first challenge of applying ERN to HCI is adapting to user's environment. Muscle movements may produce noise in the EEG data and compromise the classifying algorithm. One solution is to measure and subtract those noises from captured signals (i.e. place electrodes to measure EOG then subtract these from EEG signals to eliminate noises caused by blinking).

Another related challenge is the variability of ERN shapes and sizes in EEG signals. Although a solution could be to provide more features of the ERN pattern to the classifier this could compromise the speed and real-time requirements imposed by HCI applications. Our approach is going to explore application specific closed-loop visual feedback to elicit stronger ERN signals in moments of confusion.

Applications designs using ERN

We present two possible types of using ERN in HCI. The first is predicting the misunderstandings between user and the system before the action is processed. The second one is detecting mismatches after a committed action so that it can be reverted.

1. SELECTING OBJECTS TASK:

Selecting an object is a common task in HCI. When an object is selected, the system usually provides a graphical/ audio feedback to confirm the selection. Based on the feedback, the user will decide whether there is a mismatch between their intentions and the system. In case of a confirmed mismatch, the user typically needs to undo the action and then repeat the selection. A system with an integrated ERN module can detect those moments of misunderstanding, give suggestions and re-do the selection process. This method may be very useful when the selecting function is difficult to graphically represent (i.e. save) or the selecting operation is complex.

2. MAP NAVIGATIONS

Sometimes, a user still needs help in using and navigating using electronic maps on the fly (i.e. Google maps). One example is the users may not know the location they are in if the GPS device either is not integrated or has weak signals. Another example is a pedestrian who does not know which road to take at a junction if there are no road signs. Here, the application can produce a pseudo action/ initial feedback showing one direction and the locations it leads to. Captured EEG following the pseudo action is inspected to determine if the ERN appears. Based on the result, the pseudo action can be executed or ceased.

References

[1] Bellotti, V., M. Back, W.K. Edwards, R.E. Grinter, A. Henderson, and C. Lopes, Making sense of sensing systems: five questions for designers and researchers, in *CHI*. (2002) p. 415-422

[2] Christoforos, A., T. Dimitris, A. Niall, and H. David, Temporally adaptive estimation of logistic classifiers on data streams. *Proc. of ADAC*. 3(3): (2009) p. 243-261

[3] Coles., M.K.S.a.M.G.H., Performance Monitoring in a Confusing World: Error-Related Brain Activity, Judgments of Response Accuracy, and Types of Errors. *Proc. of JEPHPP*. 26(1): (2000) p. *141-151*

[4] Ferrez, P.W. and J.D.R. Millán, You are wrong!: automatic detection of interaction errors from brain waves, in *Proc. of IJCAI*. (2005) p. 1413-1418

[5] Ganushchak, L.Y. and N.O. Schiller, Effects of time pressure on verbal self-monitoring: An ERP study. *Brain Research*. 1125(1): (2006) p. 104-115

[6] Katahira, K., D. Abla, S. Masuda, and K. Okanoya, Feedback-based error monitoring processes during musical performance: An ERP study. *Neuroscience Research*. 61(1): (2008) p. 120-128

[7] Seno, B.D., M. Matteucci, and L. Mainardi, Online detection of p300 and error potentials in a BCI speller. *Intell. Neuroscience*: (2010) p. 1-1