Detecting Swipe Errors on Touchscreens using Grip Modulation

Mohammad Faizuddin Mohd Noor[*†1], **Simon Rogers**[†2] **and John Williamson**[†3]

[*]Malaysian Institute of Information Technology
Universiti Kuala Lumpur
Kuala Lumpur, Malaysia
mfaizuddin@unikl.edu.my

[†]School of Computing Science
University of Glasgow
Scotland, United Kingdom
[1]m.bin-md-noor.1@research.gla.ac.uk
[2]Simon.Rogers@glasgow.ac.uk
[3]JohnH.Williamson@glasgow.ac.uk

ABSTRACT

We show that when users make errors on mobile devices they make immediate and distinct physical responses that can be observed with standard sensors. We used three standard cognitive tasks (Flanker, Stroop and SART) to induce errors from 20 participants. Using simple low-resolution capacitive touch sensors placed around a standard mobile device and the built-in accelerometer, we demonstrate that errors can be predicted at low error rates from micro-adjustments to hand grip and movement in the period shortly after swiping the touchscreen. Specifically, when combining features derived from hand grip and movement we obtain a mean AUC of 0.96 (with false accept and reject rates both below 10%). Our results demonstrate that hand grip and movement provide strong and low latency evidence for mistakes. The ability to detect user errors in this way could be a valuable component in future interaction systems, allowing interfaces to make it easier for users to correct erroneous inputs.

Author Keywords

capacitive; touch; back-of-device; machine learning; accelerometer

ACM Classification Keywords

H.5.2. User Interfaces: Input devices and strategies

INTRODUCTION

Touch gestures are a common way of interacting with smartphones. Common examples are scrolling through a phonebook by flicking upwards, or accepting or rejecting a call by swiping left or right.

Despite the simplicity of touch gestures, users often make mistakes when interacting with mobile devices [19], particularly when in attention-demanding situations in mobile contexts. With simple swipes, these mistakes can be both errors in gesture performance and recognition and higher-level cognitive errors. Detection of these mistakes could be used to support recovery from error. For example, an error-detecting system could provide additional verification to check if the selection was as intended. This process requires timely detection of an *error signal* – a sensor measurement which correlates strongly with user error.

Example use case: Mobile application

Tinder[1] is a popular dating application that is based entirely on swipes; left to recommend/pass the person to someone else or right to like the person. In an event of a mistake, a user may swipe to the wrong direction, resulting in liking or disliking someone unintentionally – a potentially sensitive and embarrassing mistake. This occurs so frequently that Tinder has added a (premium) *Undo* feature to the application. Prediction of error could be used to identify if the swipe that the user just made was intentional or unintentional, allowing users to retract the swipe without explicit undo. A more sophisticated input could have the user simply reverse the direction of a swipe to cancel a detected possible error.

A similar approach could be applied to common scrolling-based applications such as contact lists. While flicking through a list of names, people often miss the name they were going for. If this "overshoot" error can be detected, the scrolling can be slowed and gently reversed to transparently recover from the slip. The detection of errors also has applications in instrumented usability and analytics, for logging likely mistakes or confused actions on the part of the user. Developers could use this to optimise error-prone elements of their applications.

To date, most techniques to identify interaction errors in HCI have been done through EEG-based brain-computer interface (BCI). This is achieved by observing the presence of error-related negativity (ErrN), a sub-component of error-related potential (ErrP) [10][8][5]. This is a distinct electrical variation in the EEG occurring 100-300ms after an error has been made. Over the years, BCI equipment has been simplified and made available commercially, allowing this technology to be more approachable by consumers for everyday use. For instance, Vi et. al. have used off-the-shelf EEG headset to detect error-related negativity (ErrN) when a user makes a mistake [28]. Using a similar headset, the ErrN signals present when observing someone else making a mistake are described in [29].

Whilst using brain waves to detect error is well established, it is not a practical approach for most interactions. There is less work on physiological signals and body channels which

[1] https://www.gotinder.com/

might produce similar signals when a mistake is made (other than measurements such as GSR or heart rate modulations, which tend to be too slow to identify individual errors in an interaction [26]).

In this paper we investigate the subtle hand movements that occur following mobile device gestures when an error is made. We examine back-of-device (BoD) capacitive and device motion to infer mistakes in interaction. The underlying hypothesis is that after making a mistake, a user will modulate their grip, affecting the contact with the phone and its orientation. We investigate whether these subtle hand-micro movements captured by BoD capacitive sensors and accelerometer can be used to reliably recognise user error.

There are two main factors that motivate using BoD and accelerometer. First, by leveraging built-in sensors, we are able to avoid using additional peripherals (i.e. headsets) to capture error-related signals, making it more practical for real devices. Second, since both BoD and accelerometer sensors can be built-in to a device at a low power cost, the sensors can give an always available channel for detection.

The contribution of this paper is a study investigating the potential of BoD and accelerometer sensors for detecting interaction mistake when using mobile application. We use a prototype device that can capture BoD hand grip strength and subtle hand movement information and we show the differences between mistake and non-mistake signals captured by these sensors before and after the mistake. We also demonstrate the optimal BoD sensor locations to capture salient information pertaining user's mistake from the sensors, and propose a minimum set of sensors for effective detection. We also provide insights into the performance gain from combining both front and BoD modalities. Finally, we discuss the various trade-offs in using BoD and accelerometer to assess user's intention.

These results form a concrete contribution to building a rich, multi-sensor mobile phone interaction error detection system. While hand grip and hand movement alone may be insufficient to perform precise detection, it can form part of an array of contributing virtual sensors in a hybrid in recognising a user's true intention.

RELATED WORK

The use of BCI in detecting human error is well-established [11]. When people make errors, there is a characteristic signal observable in EEG – the Error-related Negativity (ErrN). ErrN is a pattern that appears in the ongoing EEG signals when users have feedback about their response accuracy [15]. It also known to appear when users are confused or unsure about their actions [25] (i.e. without explicit correct/incorrect feedback).

In HCI, ErrN signals have been shown to be useful in detecting errors in spelling [8][21]. Besides BCI spelling applications, ErrN also has been used to detect errors in diverse interactive tasks [28][29]. These studies used a commercially available EEG headset to capture ErrN signals triggered from Flanker, button selection and pointing tasks. Using a logistic regression classifier, ErrN signals were classified with an accuracy of about 70%, 65%, 80% in Flanker, button selection and pointing tasks respectively. EEG-based BCI has also been used to detect ErrN during tactile human-machine interaction [17].

Various attempts have been made to bring BCI to mobile device. *NeuroPhone* for instance is a mobile phone that allows user to interact with mobile application using P300 neuro signals from the EEG headset [1], but this is impractical in almost any mobile context.

Outside of BCI, there is less research on physiological indicators of error in human computer interaction, particularly in mobile contexts. There are sophisticated auto-correct mechanism which detect and correct typing mistakes on mobile device through various techniques such as keypress timings [6], touch position distributions [13] and geometric pattern matching [16]. Weir et. al. on the other hand have used touch pressure as an uncertainty indicator before correcting the typing error [30]. Although these approaches can reduce typing time and error levels they can also be irritating [20].

Inertial sensors such as accelerometers and gyroscopes have been fundamental to mobile human activity recognition systems [23]. There have been attempts to study unusual or anomalous human dynamics from sensor data [2] to detect of unexpected events or accidents [18], although these focus on large scale events (falling over) rather than micro-movements related to touch screen gesturing.

GOAL OF THE STUDY

In this study, we investigate whether subtle fluctuations in hand grip and movement can be used to detect cognitive errors in gesture based interfaces on a mobile phone.

In particular, we are interested to see if hand grip and movement contain sufficient information to reliably detect error from swipe gestures. In order to do so, we collect swipe gestures from three swipe-based cognitive tasks using a prototype mobile phone and train classifiers to distinguish between error-related and correct swipes. We evaluate the performance of both input modalities, and the combination of the two over different period after the swipe is made.

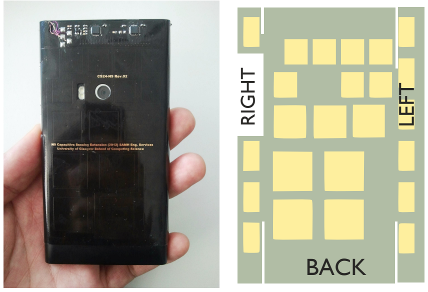
We hypothesize that when an error is made on a touchscreen interface, there are physical changes in the grip and pose of the device, such as gripping the phone more tightly or tilting it. Distinctive modulation of grip as a function of emotional state was detected by Coombes et al. [7], and Noteboom et al. [22] found that pinch grasps were modulated by the potential unpleasant stimuli. Our hypothesis is that when users realise they have made such errors they induce similar modulations of their grasp.

These post-error "flinches" should be detectable with contact and inertial sensors on the device. We set out to experimentally demonstrate the existence and character of these signals, by augmenting a standard device with wraparound capacitive sensing (and built-in inertial sensing), and designing a series of onscreen tasks to induce cognitive errors.

We use sensors that can be practically integrated into mobile devices: ultra-thin, lightweight, low-cost and low-power capacitive sensors alongside standard inertial sensors. Thus makes our approach suitable for devices such as tablets and smartphones. We hope that this study can serve as a foundation to sensor-based gestural error detection technique for mobile device.

EXPERIMENTAL SETUP

Hardware prototype



(a) (b)

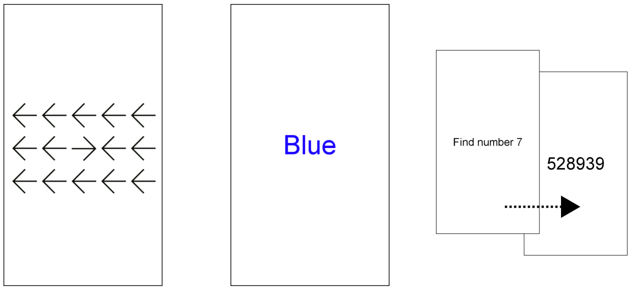
Figure 1. (a) Overview of the prototype device used in the experiment. The sensor pads are marked in yellow in (b).

Current smartphones do not have grip sensing around the device, so we used a custom BoD capacitive sensor. The prototype was based around a Nokia N9, modified to include around device sensing using a 0.1 mm thick flexible PCB, interfaced directly to the phones internal I2C bus with custom electronics. The prototype has 24 capacitive sensors distributed around the back and sides of the device to capture the users hand grip. The total size of this prototype is fractionally larger than the device itself, adding less than 1mm to all dimensions and less than 5g to the weight. In contrast to tactile sensing used in [14], we opted for capacitive sensing technology because it is a well proven touch sensing technology which is practically implementable on mobile devices. The prototype was configured to sample data from all 24 sensors at 150Hz, with 16 bit resolution. A Python client application was developed on a PC to coordinate the data acquisition from the prototype.

Cognitive Tests

To stimulate error-related responses, we designed three experiments that require significant cognitive effort, based on well-established protocols in the psychology literature. The first task was based on the Flanker task[9] where the goal is to specify the direction of a central arrow that is bordered by flanking arrows. There were two types of arrows, with each type had two stimuli: congruent stimuli (<<<<< and >>>>>) and the incongruent stimuli (<<><< and >><>>). We also used no-go stimuli (◇◇ <> ◇◇) where no specification of arrow direction is required.

Our second experiment is based on Stroop task[27]. The stimuli used in this task were texts of colours (i.e. *Blue, Yellow, Green*) where the objective is to determine whether the word matches the colour used by the text. Congruent stimuli are matching colour/word and incongruent stimuli otherwise. For example, the word "blue" written in blue is a congruent stimuli whilst the word "green" written in blue is incongruent. For no-go stimuli we used unrelated words within the context of colour (i.e. *Book, Grin, Back*).



(a) Flanker (b) Stroop (c) SART

Figure 2. Example of incongruent stimulus used in Flanker and congruent stimuli in Stroop and SART tasks. In (a) and (b) the correct swipe direction would be right, whereby in (c) it would be left.

The final task used Sustained Attention to Response Task (SART)[24]. There were two conditions used in this task: Non-Zero and Specific. In the former, a series of 6 digits is presented containing mostly zeros. The objective is to determine if the series contain a non-zero number. In the Specific condition, a series of 6 digits is presented containing different non-zero digits. Here the goal is to determine if a particular digit appears in the series. In contrast to Flanker and Stroop, there is no no-go stimulus used in SART.

We developed a Python application on the N9 to run the experimental software. Unlike classic computer-based cognitive tests which typically use a physical keyboard or mouse, we adapted the tests to use standard swipe-gestures on the N9 touchscreen. Participants responded to the stimulus by swiping the N9 touchscreen horizontally. We picked swiping over tapping since swiping does not require us to place occluding buttons on the stimulus screen and swiping is a very common task on modern touchscreen devices. We mapped the rightwards swipe for congruent stimuli and left swipe for incongruent stimuli for the SART and Stroop tests. For Flanker, however, the direction of swipe is based on the direction of the central arrow regardless of the stimulus type. Participants were required to withhold their responses (i.e. not touch the screen) when no-go stimuli are presented.

Swipe as a response

Since our experiments used swipes rather than key presses from dedicated buttons, it is important to clearly define valid swipes. For this purpose, we defined a swipe as a touch stroke with length of at least 30 pixels (3.04 mm). The direction of the swipe is determined by measuring the angle of the straight line connecting the start and end points of the swipe. Duration of swipes was required to be within 50 – 150ms. The lower limit is to ensure that the swipe was not accidental (click/tap) and the upper limit is to prevent participants from leaving their finger on the touchscreen.

Participants and Data Acquisition

N = 20 participants, 18 right-handed and 2 left-handed were recruited locally. This included 16 males and 4 females, aged between 25 – 35 (mean = 29.4, sd = 3.15) each with at least one year of experience in using a smartphone. They performed Flanker, Stroop and SART tests whilst seated in a lab, with the phone in a single handed grip, swiping with the thumb in a portrait orientation. All participants were briefed and given a practice session before the main experiment.

For each touch events, we recorded timestamps, accelerometer and capacitive readings from the back of the device for subsequent offline analysis. Each recording was performed in three sessions, separated by an approximately 5 minute break. This was done by asking participants to put down the phone on the table at the end of every session. This ensures that we are not observing temporary grip patterns, but a range of plausible grips for each user.

Each session consisted of 30 trials (swipe responses), resulting in 90 trials in total for each of the three tasks and a total of 270 trials per participant. Each trial begins with a start screen. Prior to showing the stimulus, a 2000ms waiting time is given to allow participants to prepare themselves. We used a red circle at the centre of the screen that would reduce its radius relative to the waiting time. The stimulus is shown immediately after the waiting time. Participants were required to respond to the stimulus within a designated time. We fixed the response time to 1000ms and reduced/increased the time based on the error rate after the 10th trial. We reduce the time by 100ms if the error rate stays below 25% or increase it by the same amount if the error rate is higher. The minimum and maximum response time were capped at 100ms and 1000ms respectively. This adaptive procedure induces errors at a consistent level across participants of varying skill and performance. A notification of correct or incorrect responses was given immediately at the end of each trial. We used plain black 'Incorrect' and 'Correct' texts on white background as feedback stimuli. This stimuli was shown for 2000ms before returning back to the start screen for next trial.

Overall experiment time ranged between 25 – 43 minutes per participant (mean = 33.74, sd = 5.89). A single Flanker session (30 trials) ranged between 1.9 – 2.4 minutes (mean = 2.14, sd = 0.12), Stroop between 1.88 – 2.35 minutes (mean = 2.11, sd = 0.14) and SART between 2.5 – 2.8 minutes (mean = 2.64, sd = 0.08).

Table 1 breaks down the complete set of trials over all participants. On average, performance across the three trials are broadly equivalent with many more correct swipes than incorrect. The number of mistakes per user varies quite dramatically. For the SART test, users have between 0 and 28 errors. Such an imbalance between the two classes poses a challenge to classification, which we discuss later.

ANALYSIS

Feature extraction

The prototype produces 24 capacitive and 3 acceleration time series for each swipe action (one time series per sensor). The BoD sensors in our prototype are occasionally affected by noise spikes that we remove by interpolating the gaps with a linear fit. This method is sufficient since all spikes in the signal were short (mean = 4 samples).

For each trial we extract a time segment consisting of 1000ms before and after the feedback (notification to the user of whether or not they have made a mistake). In order to create a fixed length representation from the time series (the sampling rate from the sensors exhibited small variability) sampling rate was down sampled linearly to 100Hz. Thus we are left with an equal length of 200 x 24 BoD and 200 x 3 accelerometer time series vectors (100 samples before and 100 samples after the feedback (notification)). Both BoD and accelerometer signals were smoothed using an $order-3$ Savitzky-Golay filter with frame size of 5. This applies very mild low-pass filtering.

The time series typically exhibited long term trends (e.g. gradually increasing grip strength across a session). An example is shown in Figure 3. To remove this, data was normalised for each trial by subtracting the mean of the values before feedback from the vector (i.e. for each sensor, the values were normalised to that the time series had zero mean for the first 100 values).

In all experiments we compared keeping all of the distinct sensor values as well as using the total magnitude (the square root of the sum of squared values). The total magnitude overcomes the problem of handedness (for both BoD and accelerometer) and gives, for BoD, a value that can be considered a proxy for the grip strength. Comparison of the individual sensors and the total magnitude can therefore help us to decide whether the classification signal is based on grip strength (total magnitude) or also requires grip shape (individual sensors).

Classification

The primary goal of this study is to measure the extent to which hand grip and hand movement can distinguish swipes that are related to mistakes. This is as a binary classification task where the problem is to determine whether the swipe belongs to either positive (mistake) or negative class (non-mistake/normal). As well as investigating if classification is possible, we are interested in investigating how classification performance varies with the time. To this end, in all experiments we investigate features derived from 6 different time segments. Each segment starts at -100ms (100ms before feedback) but has a different end point (0s, 200ms, 400ms, 600ms, 800ms and 1000ms). In all experiments, the data is split into independent training (70%) and testing (30%) sets. Due to very small incorrect response examples in every session, we randomly partitioned incorrect response data instead of partitioning them by session. To perform classification, we used Support Vector Machine (SVM) and Random Forests (RF) classifiers implemented in Matlab (for SVM we used MATLAB's libSVM toolbox [3]). The same train/test splits were used for the different classifiers to faciliate paired statistical testing.

	Flanker				Stroop				SART			
	Correct	**Incorrect**	**Invalid**	**Late**	**Correct**	**Incorrect**	**Invalid**	**Late**	**Correct**	**Incorrect**	**Invalid**	**Late**
$\bar{x}$	49.65	5.00	4.95	3.30	37.85	10.40	6.85	8.45	64.05	10.95	6.80	6.75
sd	4.60	4.63	2.33	2.68	4.51	7.88	3.76	3.32	8.91	7.55	3.04	3.08
min	40	0	0	2	29	1	4	2	42	0	3	2
max	57	15	9	10	46	26	15	14	77	28	11	11

Table 1. Sample statistics (mean, standard error, min (per user) and max (per user)) of correct, incorrect, invalid and late responses made by participants in the cognitive experiments.

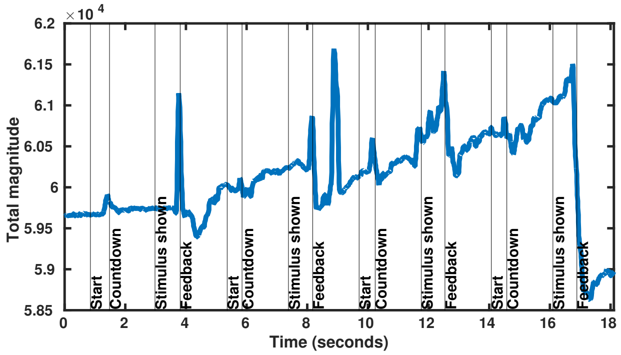


Figure 3. Example of the BoD total magnitude signal showing four consecutive trials for one participant in the Stroop task. The vertical lines denote the key moments in the trials.

For the RF we used an ensemble of 100 trees, and for the SVM we used Gaussian (RBF) kernels throughout, with the following form:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right), \ \gamma > 0,$$

where x_{nd} is the d-th feature of the n-th swipe object and γ is kernel parameter.

As well as being interested in the relative performance of BoD and accelerometer features, we also investigated combining the data sources (BoD and accelerometer) through kernel combination with the SVM and feature concatenation for the RF. In the kernel combination approach, a weighted sum of the two kernel matrices (BoD and accelerometer) is used, with an additional parameter a controlling the influence of each kernel:

$$K = aK_{BoD} + (1 - a)K_{accelerometer}$$

where $0 \leq a \leq 1$ with 0 representing only the accelerometer kernel and 1 representing only the BoD kernel. This is guaranteed to produce a correct (i.e. symmetric, positive, semi-definite) kernel, K. Optimising a allows us to measure the relative contribution of each data modality. For a review of multiple kernel learning, see e.g. [12].

Prior to training classifiers, features were normalised to have mean zero and standard deviation one. As already mentioned, the data are not balanced across the classes (we have many more correct swipes than error swipes). In the pooled setting (see below), data subsampling was applied only to give equal class distributions. This was done by randomly sampling data points from the negative class (correct swipes) to give the same number as in the positive class (errors). When we train classifiers on data from individual users, we are unable to do this due to the small sizes of the positive class. In these cases, we up-weight the C parameter for examples from the positive class by a factor equal to the ratio of points in the negative to positive classes (see e.g. [4]).

In our experiments, we quote various different performance measures: Accuracy, AUC, false accept rate (FAR) and false reject rate (FRR). Accuracy is the proportion of swipes classified correctly. AUC is calculated as the area under the ROC curve which describes the true and false positive rates of a classifier at different thresholds. For instance, for a classifier that outputs values between 0 and 1, we can obtain different true and false positive rates by varying the threshold above which we consider a data point to be in class 1. The ROC curve shows performance from all thresholds in one graph. The area under the ROC curve (AUC) quantifies performance from all thresholds into one single value. AUC can be interpreted as the probability that a randomly selected pair of test points (one from each point) will be given outputs in the correct order (i.e. the point in class 1 will have the higher output). Unlike accuracy, AUC is not distorted by imbalanced classes.

For FAR and FRR we consider how the system might be used in practice – users may provide correct or erroneous swipes and it is the job of the system to accept them (i.e. consider them as a correct input) or reject them (perhaps ask the user to enter the input again). The False Accept Rate is therefore the fraction of erroneous swipes (positive class) misclassified as correct swipes (positive class), whilst the False Reject Rate is the fraction correct swipes (negative class) misclassified as incorrect swipes (positive). In other words, a false accept is where an erroneous swipe is mistakenly taken by the system to be a correct input.

In all experiments, our baseline is the performance of a classifier with randomised test data labels (i.e. with no predictive power). This was created by multiple randomisations of the test labels. To produce a single figure for comparison, we quote classification performance for the dataset finishing 200ms after the feedback. To compute statistical significance of results, we use the non-parametric Wilcoxon Signed Rank Test with p-value threshold of 0.05. A non-parametric test was used as the upper bound of 1 on all performance measures would potentially make the examples highly non-Gaussian.

For the SVM, the C and γ parameters were selected via 3-fold cross-validation on the training set using a grid-search and the AUC as the performance metric. When investigating kernel combination, we fixed the γ parameters for each kernel to their optimal individual values and optimised C and a with an additional 3-fold cross-validation procedure. For the RF, we used 100 estimators to create the ensemble to classify data in both individual and pooled settings. To perform classification on composite data, we simply combine data from both modalities into a single input vector.

RESULTS

Preliminary Analysis

Before performing any classification, we visualised the mean trial time series across all participants. The time series can be seen in Figure 4 for BoD sensors (top three plots) and accelerometer (bottom three). In all plots, the blue lines show the mean (plus and minus standard error) for all correct swipes and the red for all incorrect swipes. Various interesting features are visible in these plots. Firstly, there appears to be a clear difference in the lines for the Stroop test, suggesting that there are signals that could be classified to detect errors. Secondly, the differences appear to start before feedback (dashed vertical line), suggesting that users know when they have made a mistake. Such changes are not so visible in the other tests (there is a small difference in the accelerometer plot for Flanker but nothing for SART) suggesting that the different tests reveal different physical responses from the participants. Note also that the BoD features for Flanker errors have quite large standard error. This is due to the relatively small number of errors in this test (see Table 1).

Experiment 1: Can we classify swipe errors automatically?

The grand average plots of Figure 4 suggest that there are systematic variations in the measured grip and pose when errors are made. To determine whether this could be reliably distinguished automatically, we trained classifiers to recognise error swipes.

Our first approach trained user-specific classifiers. The classifiers were trained using both the total magnitude and multi-sensor features. The results of this experiment are shown in Figure 5. This clearly shows that error swipes *can* be classified automatically from the sensors we have, with AUCs well above baseline performance. There is also significant user variation in classification performance, possibly due to the large variance in numbers of errors made by users.

In all cases, observing longer periods of sensor data did not improve classification, and performance 200ms after feedback was comparable with feedback including subsequent time points.

We found that the multi-sensor features (time series of 24 pad values) gave the better performance than total magnitude for the BoD sensors. The RF performed better than the SVM for Flanker and SART, giving AUC of 0.80 and 0.70 respectively after 200ms of data. In Stroop however, the performance of RF and SVM is similar, producing classification AUC of about 0.65 after 200ms observing BoD data.

For the accelerometer features, total magnitude performed much better than individual axis classification, with mean AUC of 0.80 at 200ms in all tests, compared to 0.60 at 200ms for individual axes. This may be due to the device being held at slightly different orientations, leading to big offsets in the individual axis data, but having no effect on the total magnitude of the accelerometer vector. The SVM performs better than RF for accelerometer classification, particularly in Stroop and SART tests.

The RF performance is very poor for Stroop and SART accelerometer classification, possibly due to the very small number of swipe error examples available for these tasks.

We required at least 5 training examples of both correct and incorrect responses to train the classifiers. Due to insufficient numbers of error swipe examples, we could not train classifiers for some participants. We used 8 users for Flanker, 15 for Stroop and 18 for SART.

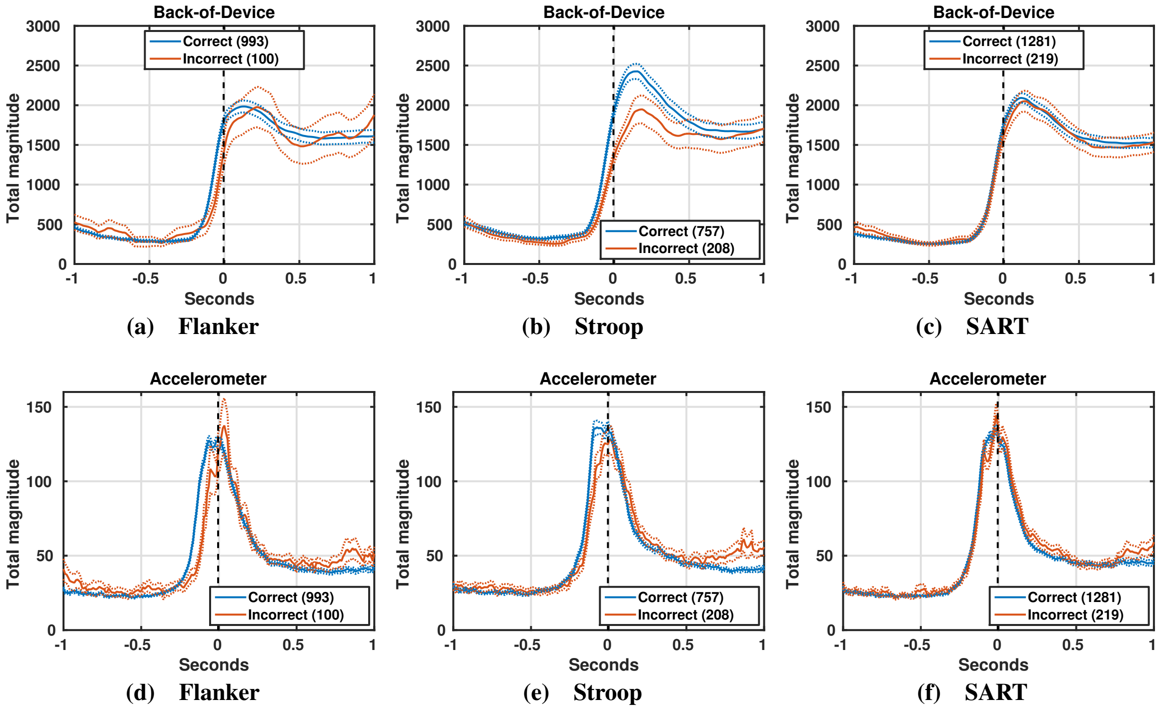
Experiment 2: Can we build classifiers that do not require individually trained models?

Having to train a specific classifier for each potential user is problematic in real-world use: it requires a potential user to explicitly enroll in the system, online training must be performed, and we must know which user is using the device at all times. A universal, user-insensitive model would be much more useful. We set out to determine if all participants share common error-related grip or hand motion patterns.

We constructed classifiers using swipe data sampled randomly from **all** participants, to determine if a general model can be trained to classify error swipes. Our results are summarised in Figure 6. Surprisingly, the general classification model performs much *better* than the individual model, with AUCs of > 0.9 for RF classification of BoD features 200ms after feedback. This improvement is likely to be due to the increase in training data available.

Again, we found that the best classification performance for BoD used the multi-sensor features while total magnitude worked best for the accelerometer. We also again found that additional time periods after 200ms did not significantly improve classification performance. For BoD the RF has an AUC > 0.90 at every point of time after the feedback in all tests. This suggests that there is highly discriminative hand grip pattern that happens early during error swipes. For accelerometer, the SVM matches the AUC of RF 100ms before the feedback but drops slightly when more data points is observed by the classifiers particularly at 200ms onwards in Stroop and SART tests. In contrast, the AUC using RF increases when more data point is used to build the classifiers.

Experiment 3: Can we improve performance by fusing the accelerometer and BoD sensors?



(a) Flanker (b) Stroop (c) SART

(d) Flanker (e) Stroop (f) SART

Figure 4. Mean total magnitude of signals from the 24 BoD capacitive sensors (a, b, c) and 3-axis built-in accelerometer sensor (d, e, f) from all participants. Blue and red solid lines correspond to correct and incorrect responses respectively. The x axis is the time before and after the notification moment. The y axis is total magnitude of the sensors. The blue and red dashed lines are $+/-$ standard error of the mean total magnitude. The black dashed line is the feedback moment.

Given that we have both the accelerometer and BoD sensors, we wanted to test whether they were providing the same information or if they had independent components; and if there was independent information, could this be used to improve classification performance. We built composite classifiers using the best performing feature of each input modality: multisensor for BoD and total magnitude for accelerometer.

For the SVM, we can use a weighted linear combination of kernels, parameterised by a the kernel weight a (0=accelerometer only to 1=BoD only). For the RF we simply concatenate the feature vectors. The relative influence of each input cannot be assessed with this approach.

The results are summarised in Figure 7. Composite classifiers give better AUCs than either accelerometer or BoD for both individual and pooled models. The best AUC for individual models uses the SVM classifiers, producing AUC > 0.80 in all tests. For pooled model, the RF classifiers outperform the SVM in all tests, producing AUCs > 0.95 for all time periods.

For the SVM model, we found that the best composite kernel weight a varied with the task: the Flanker test performed better with a high BoD weighting, while SART and Stroop performed better with a lower BoD weighting. In all cases, optimal a was neither 0 nor 1.0, indicating that the classifier found independent information from both sensors.

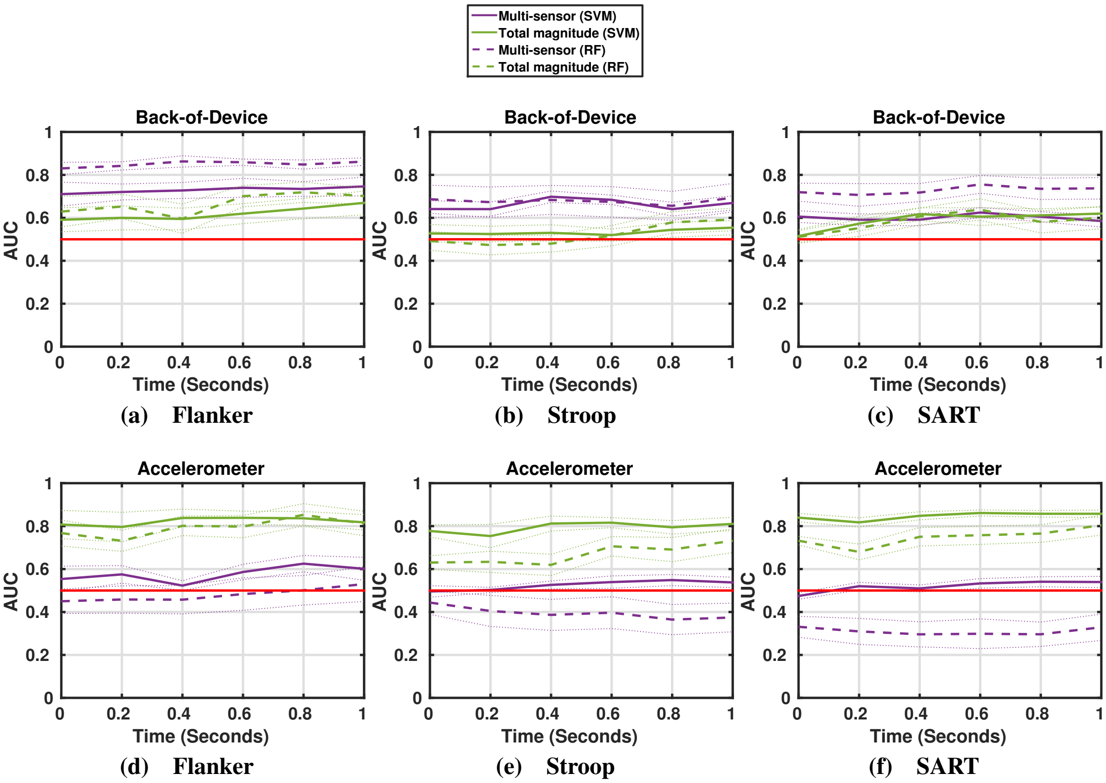
In order to investigate if the optimal a is statistically significant, we ran a Wilcoxon paired signed-rank test between the balanced composite features ($a = 0.5$) and the individual features for each BoD ($a = 1.0$) and accelerometer ($a = 0$) features at $t = 200$ms following feedback. From the test we found out that there were no significant improvements in all tests for composite features versus accelerometer. However for BoD, the improvement was significant in SART ($p < 0.01$) but not in Flanker and Stroop.

For the individual user models, we found statistically significant improvements for SART and Stroop tests for BoD versus composite ($p < 0.001$ and $p < 0.001$ respectively). Improvement over accelerometer alone was not statistically signficant for any task with the individual models. For the pooled model we found a statistically significant improvement over accelerometer alone for Flanker and Stroop ($p < 0.01$ and $p < 0.05$ respectively).

For the RF model, statistically significant improvements with the composite model were not observed for the individual or pooled models over either BoD or accelerometer.

We also compared performance of the SVM and RF classifiers on the composite features, again with a paired Wilcoxon signed-rank test. For the pooled models, the RF has significantly better AUC for Flanker, Stroop and SART ($p < 0.05$, $p < 0.01$, $p < 0.01$ respectively). For individual models, the SVM has significantly better AUC for SART ($p < 0.05$), but there is no significant difference for Flanker and Stroop.

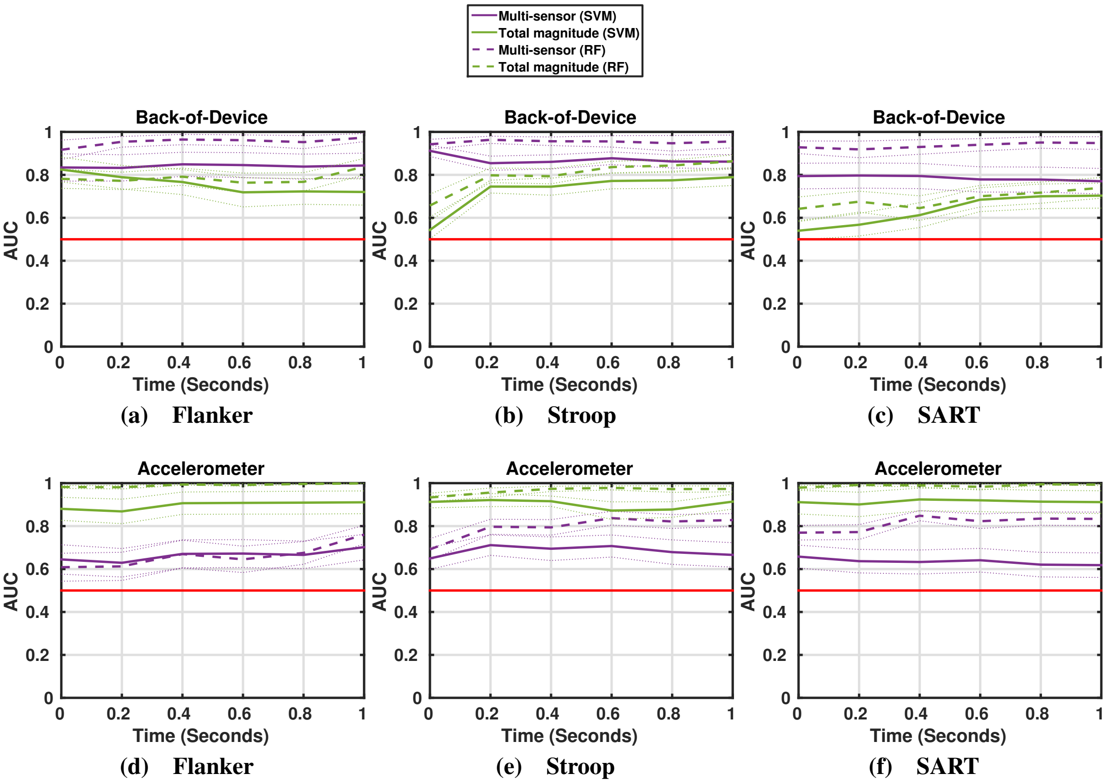
Table 2 shows the overall performance of the classifiers with the composite features. The random forest has better performance overall, and suffers much less from the class imbal-



(a) Flanker (b) Stroop (c) SART

(d) Flanker (e) Stroop (f) SART

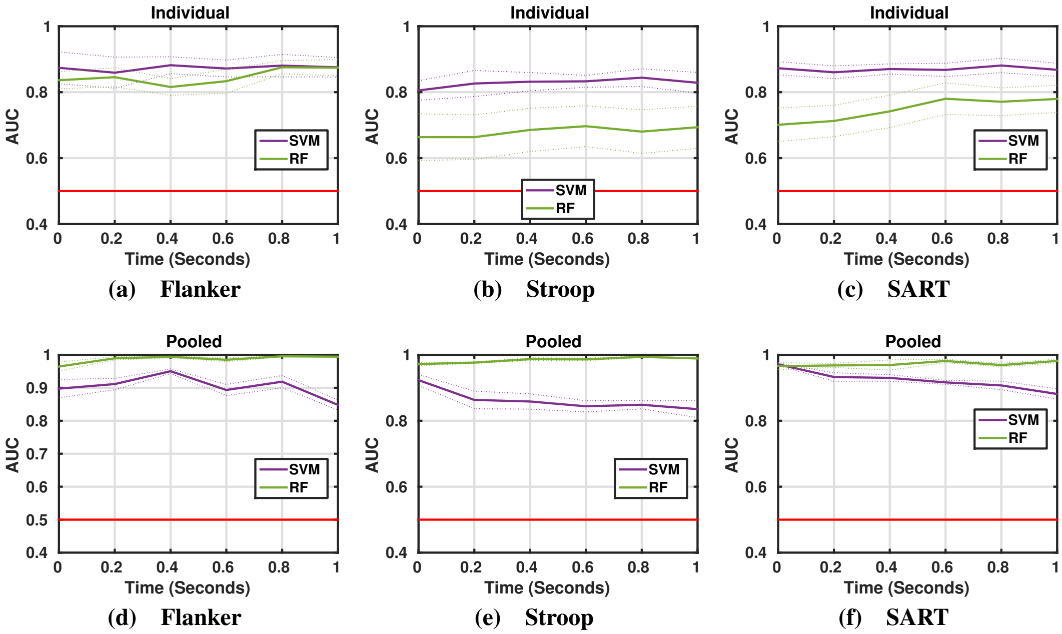
Figure 5. Classification results using individual model averaged across all participants for BoD (a, b, c) and accelerometer (d, e, f). Solid and dashed lines correspond to AUC (with standard errors) for SVM (green) and random forest (purple) classifiers respectively. Red solid line is the reference baseline.



(a) Flanker (b) Stroop (c) SART

(d) Flanker (e) Stroop (f) SART

Figure 6. Classification results using pooled model for BoD (a, b, c) and accelerometer (d, e, f). Solid and dashed lines correspond to AUC (with standard errors) for SVM and random forest classifiers respectively. Solid red line is the reference AUC baseline.



(a) Flanker (b) Stroop (c) SART

(d) Flanker (e) Stroop (f) SART

Figure 7. Classification results using combination of BoD and accelerometer kernels for individual (a, b, c) and pooled (d, e, f) models. Solid green and purple lines correspond to AUC for RF and SVM classifier respectively. Dashed lines are the standard errors. Reference AUC baseline is represented by the solid red lines.

	Task	Accuracy (%)	FAR (%)	FRR (%)
SVM	Flanker	78.9	0	42.1
	Stroop	80.7	15.2	23.2
	SART	89.3	0.4	20.9
RF	Flanker	95.8	5.3	3.2
	Stroop	91.8	8.6	7.7
	SART	89.8	10.7	9.8

Table 2. The overall performance of the classifiers for the pooled model in terms of accuracy, FAR and FRR at t=200ms with the composite features.

ance problems that cause the SVM to have extremely high FRR rates and very low FAR rates.

DISCUSSION

Swipe errors can be detected.

We have shown that – in our controlled lab setting and specific tasks – swipe errors have a distinctive signature that can be automatically classified. AUCs well above baseline were achieved for individual models (Figure 5) (e.g. AUC > 0.8 using the SVM on accelerometer data for all tasks).

Swipe errors patterns generalise over users

When we pooled data from multiple users and did not use user-specific models, we were still able to classify swipe errors reliably. In fact, AUC increased from around 0.62 – 0.83 to 0.91 – 0.99 with the pooled model. This suggests that classification performance may be limited by the available dataset size rather than inter-user variability.

Swipe errors are not task-specific

As shown in Table 3 training on specific *specific* models and testing on mismatched tasks (e.g. training on Flanker and testing on SART) has very good performance for the accelerometer (AUC 0.80 – 0.99) however weaker for the BOD, possibly due to the high dimensional features with relatively small training sets. Training with a multi-task model *significantly improves* test performance over training on just the matching task.

Specific sensors work best with different transforms.

BoD performance was best with the full 24 sensor time series compared to total magnitude (SVM AUC increased from 0.85 to 0.95 at 200ms). This suggest that there are relevant changes in the pose of the hand, and not just overall grip strength. The accelerometer, in contrast, had better performance with total magnitude than individual axis features. This may be because of variations in orientation which are removed by the magnitude transform.

Composite features are the best.

We found the best classification when combining the accelerometer and the BoD sensors, both with simple concatenation of the feature vectors (RF) and with composite kernels (SVM). This suggests that the accelerometer and BoD sensors contribute independent information about the error signal.

There is little change in classification performance over time.

In all of our results, classification performance reached a peak by 200ms after feedback and did not vary significantly after that. Even at feedback time ($t = 0$), before the user has had a chance to react to the feedback we are able to classify the error signal. This suggests the error-related movements happen early, and may anticipate the feedback (users are responding to internal knowledge of the error they have made).

Random forests generally perform better than the SVM.

		Test					
		Back-of-Device			*Accelerometer*		
		Flanker	Stroop	SART	Flanker	Stroop	SART
Train	Flanker	0.95	0.57	0.48	0.98	0.86	0.89
	Stroop	0.60	0.96	0.69	0.97	0.96	0.98
	SART	0.49	0.68	0.92	0.80	0.84	0.99
	Flanker + Stroop + SART	0.92	0.99	0.99	0.99	0.99	0.99

Table 3. Cross-task classification AUC using RF classifiers constructed using back-of-device (multi-sensor) and accelerometer (total magnitude) features at 200ms after the feedback.

The random forest classifier was more effective for all of the pooled models (although performed very poorly in the individual models where training data was sparse). As can be seen in Table 2 the SVM did not deal well with the imbalanced classes and led to very high FRR rates. The random forest had better overall performance for the pooled models, and a much more balanced FAR/FRR result.

Performance was similar for SART, Stroop and Flanker tasks. We were able to classify errors with a similar accuracy (Table 2) for errors made in SART, Stroop and Flanker tasks, suggesting that the results are indeed capturing a general error signal, and not simply a task-specific anomaly.

FUTURE WORK

Our study has identified that it is possible to identify errors in cognitive tests using data from either an accelerometer, back of device sensing, or both. This opens many interesting avenues for future investigation and development. Firstly, our data was collected in a very controlled setting using well-defined cognitive tasks. It is an important next step to demonstrate that detection is possible in a more realistic context. For example, it is unclear how user movement would effect predictions derived from the accelerometer, or how the signals would change when a user was less engaged with the task. In addition, we have only studied one type of gesture, and one type of error. It would be particularly interesting to look at whether similar performance is possible within a completely different domain. For example, when entering text, many different error types are possible: physical errors (hitting the wrong key), cognitive errors (spelling a word incorrectly) and system errors (incorrectly auto-correcting a word). It is possible that the signals from the user (if they exist) would be different in these three cases although detecting these different situations would allow us to make smart correction systems.

Our prototype phone has 24 back of device sensors. Our study has suggested that the spatial information encoded in these sensors is useful in detecting errors. It would be useful to be able to extract from such data where the sensors should be best positioned to make error predictions. The distinct differences between grip across users makes this impossible with the current data (we have too few errors per user). Indeed, a preliminary mutual information analysis with our data reveals no clear patterns in the information content of the different sensors. In our studies, we investigated the effect of time on the classification performance. On the whole, we found classification performance to remain fairly constant over time, suggesting that the error information is present very early in the time series. Visual inspection of Figure 4 suggests that the error signal appears slightly earlier in the accelerometer data than it does in the back of device data. This is worth investigating further, for which a larger dataset is required to ensure sufficient statistical power to identify these changes.

CONCLUSION

We addressed automatic detection of errors (swiping the wrong way) in swipe gesture based interfaces by detecting subtle fluctuations in hand grip and movement. We used BoD capacitive sensors and accelerometer to detect these changes. Three swipe-based cognitive tests based on the classic Flanker, Stroop and and SART tasks were used to induce errors. From the recorded sensor data, we trained classifiers Support Vector Machines and Random Forests which could reliably detect errors shortly after their performance.

Our results show that both BoD capacitive, and accelerometer sensing can be reliably used to detect swipe errors. Our approach can be implemented with standard, cheap hardware, and is viable for standard phones and tablets. As well as detecting errors with per-user trained models, we also show models trained on pooled data perform well, and therefore that user-specific training is not required. Best performance is obtained by combining the accelerometer and BoD data, suggesting each contributes some independent information about the error signal.

Our results suggest that hand grip and hand movement from BoD sensors and accelerometer can be used to reliably and rapidly detect cognitive errors in swipe-based interfaces. While back-of-device sensing is not yet mainstream, some commercial devices (e.g. the Doogee DG800 or the Oppo N1) already provide back-of-device touch sensors that may have sufficient resolution to capture grip modulations. Although our experimental models are specific to the cognitive tests that we used, many mobile tasks have very similar components: a mentally demanding task with simple swipe gestures as input. Our results are both practically relevant in developing forgiving interfaces that can recover from errors gracefully, and suggest that there is much more to be explored in error-related micro-movements in touch screen gesture interfaces.

ACKNOWLEDGEMENTS

Experimental data available at `http://dx.doi.org/10.5525/gla.researchdata.247`

REFERENCES

1. Andrew Campbell, Tanzeem Choudhury, Shaohan Hu, Hong Lu, Matthew K Mukerjee, Mashfiqui Rabbi, and Rajeev DS Raizada. 2010. NeuroPhone: brain-mobile phone interface using a wireless EEG headset. In *Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds*. ACM, 3–8.
2. Julin Candia, Marta C Gonzlez, Pu Wang, Timothy Schoenharl, Greg Madey, and Albert-Lszl Barabsi. 2008. Uncovering individual and collective human dynamics from mobile phone records. *Journal of Physics A: Mathematical and Theoretical* 41, 22 (2008), 224015.
3. Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27. Issue 3.
4. Chih-Jen Chang, Chih-Chung; Lin. 2011. LIBSVM: A Library for Support Vector Machines. *ACM Trans. Intell. Syst. Technol.* 2, 3, Article 27 (May 2011), 27:1–27:27 pages.
5. Ricardo Chavarriaga, Pierre W Ferrez, and José del R Millán. 2008. To err is human: Learning from error potentials in brain-computer interfaces. In *Advances in Cognitive Neurodynamics ICCN 2007*. Springer, 777–782.
6. James Clawson, Kent Lyons, Alex Rudnick, Robert A Iannucci Jr, and Thad Starner. 2008. Automatic whiteout++: correcting mini-QWERTY typing errors using keypress timing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 573–582.
7. Stephen A Coombes, Kelly M Gamble, James H Cauraugh, and Christopher M Janelle. 2008. Emotional states alter force control during a feedback occluded motor task. *Emotion* 8, 1 (2008), 104.
8. Bernardo Dal Seno, Matteo Matteucci, and Luca Mainardi. 2010. Online Detection of P300 and Error Potentials in a BCI Speller. *Intell. Neuroscience* 2010, Article 11 (Jan. 2010), 1 pages.
9. Barbara A Eriksen and Charles W Eriksen. 1974. Effects of noise letters upon the identification of a target letter in a nonsearch task. *Perception & psychophysics* 16, 1 (1974), 143–149.
10. Pierre W Ferrez and José del R Millán. 2005. You are wrong!-Automatic detection of interaction errors from brain waves. In *International Joint Conference on Artificial Intelligence*, Vol. 19. LAWRENCE ERLBAUM ASSOCIATES LTD, 1413.
11. WJ Gehring, MGH Coles, DE Meyer, and E Donchin. 1990. The error-related negativity: an event-related brain potential accompanying errors. *Psychophysiology* 27, 4 (1990), S34.
12. Mehmet Gönen and Ethem Alpaydın. 2011. Multiple kernel learning algorithms. *The Journal of Machine Learning Research* 12 (2011), 2211–2268.
13. Joshua Goodman, Gina Venolia, Keith Steury, and Chauncey Parker. 2002. Language modeling for soft keyboards. In *Proceedings of the 7th international conference on Intelligent user interfaces*. ACM, 194–195.
14. Yuta Higuchi and Takashi Okada. 2014. User Interface Using Natural Gripping FeaturesGrip UI. (2014).
15. Clay B Holroyd and Michael GH Coles. 2002. The neural basis of human error processing: reinforcement learning, dopamine, and the error-related negativity. *Psychological review* 109, 4 (2002), 679.
16. Per-Ola Kristensson and Shumin Zhai. 2005. Relaxing stylus typing precision by geometric pattern matching. In *Proceedings of the 10th international conference on Intelligent user interfaces*. ACM, 151–158.
17. M. Lehne, K. Ihme, A.-M. Brouwer, J. van Erp, and T.O. Zander. 2009. Error-related EEG patterns during tactile human-machine interaction. In *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*. 1–9.
18. Qiang Li, John Stankovic, Mark Hanson, Adam T Barth, John Lach, Gang Zhou, and others. 2009. Accurate, fast fall detection using gyroscopes and accelerometer-derived posture information. In *Wearable and Implantable Body Sensor Networks, 2009. BSN 2009. Sixth International Workshop on*. IEEE, 138–143.
19. I Scott MacKenzie. 2012. *Human-computer interaction: An empirical research perspective*. Newnes.
20. Jillian Madison. 2012. *Damn you, autocorrect!* Random House.
21. Perrin Margaux, Maby Emmanuel, Daligault Sébastien, Bertrand Olivier, and Mattout Jérémie. 2012. Objective and subjective evaluation of online error correction during P300-based spelling. *Advances in Human-Computer Interaction* 2012 (2012), 4.
22. J Timothy Noteboom, Kerry R Barnholt, and Roger M Enoka. 2001. Activation of the arousal response and impairment of performance increase with anxiety and stressor intensity. *Journal of applied physiology* 91, 5 (2001), 2093–2101.
23. Nishkam Ravi, Nikhil Dandekar, Preetham Mysore, and Michael L Littman. 2005. Activity recognition from accelerometer data. In *AAAI*, Vol. 5. 1541–1546.
24. Ian H Robertson, Tom Manly, Jackie Andrade, Bart T Baddeley, and Jenny Yiend. 1997. Oops!': performance correlates of everyday attentional failures in traumatic brain injured and normal subjects. *Neuropsychologia* 35, 6 (1997), 747–758.

25. Marten K Scheffers and Michael GH Coles. 2000. Performance monitoring in a confusing world: error-related brain activity, judgments of response accuracy, and types of errors. *Journal of Experimental Psychology: Human Perception and Performance* 26, 1 (2000), 141.

26. Tom Sharma, Nandita; Gedeon. 2013. Optimal Time Segments for Stress Detection. In *Machine Learning and Data Mining in Pattern Recognition*, Petra Perner (Ed.). Lecture Notes in Computer Science, Vol. 7988. Springer Berlin Heidelberg, 421–433.

27. J Ridley Stroop. 1935. Studies of interference in serial verbal reactions. *Journal of experimental psychology* 18, 6 (1935), 643.

28. Chi Vi and Sriram Subramanian. 2012. Detecting Error-related Negativity for Interaction Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 493–502.

29. Chi Thanh Vi, Izdihar Jamil, David Coyle, and Sriram Subramanian. 2014. Error Related Negativity in Observing Interactive Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3787–3796.

30. Daryl Weir, Henning Pohl, Simon Rogers, Keith Vertanen, and Per Ola Kristensson. 2014. Uncertain text entry on mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2307–2316.