Using Physiological Sensors to Detect Levels of User Frustration Induced by System Delays

Brandon Taylor
Carnegie Mellon University
bttaylor@cs.cmu.edu

Anind Dey
Carnegie Mellon University
anind@cs.cmu.edu

Daniel Siewiorek
Carnegie Mellon University
dps@cs.cmu.edu

Asim Smailagic
Carnegie Mellon University
asim@cs.cmu.edu

ABSTRACT

In mobile computing, varying access to resources makes it difficult for developers to ensure that satisfactory system response times will be maintained at all times. Wearable physiological sensors offer a way to dynamically detect user frustration in response to increased system delays. However, most prior efforts have focused on *binary* classifiers designed to detect the presence or absence of a *task-specific* stimulus. In this paper, we make two contributions. Our first contribution is in identifying the use of variable length system response delays, a universal and *task-independent* feature of computing, as a stimulus for driving different levels of frustration. By doing so, we are able to make our second and primary contribution, which is the development of models that predict *multiple levels* of user frustration from psycho-physiological responses caused by system response delays. We investigate how incorporating different sensor features, application settings, and timing constraints impact the performance of our models. We demonstrate that our models of physiological responses can be used to classify *five* levels of frustration in near real-time with over 80% accuracy, which is comparable to the accuracy of binary classifiers.

Author Keywords

Sensors; Emotion Recognition; System Response Delays

ACM Classification Keywords

H.5.m Information interfaces and presentation (e.g. HCI): Miscellaneous

INTRODUCTION

In mobile computing, application designers face a number of challenges to provide a consistently satisfactory user experience [11, 36]. A key challenge is that designers cannot guarantee the exact performance of an application because they cannot know what resources will be available at a given time. As resources, such as network bandwidth, vary over time, the application performance varies and can lead to a *frustrating* user experience when the bandwidth falls below an acceptable level.

To address user frustration caused by system delays, we can either try to predict what the available resources will be and how they will vary or we can try to detect frustration as it occurs. Without a service-level agreement that guarantees a certain quality of service, which most consumers do not have, it would be extremely challenging to predict resource availability to ensure optimal response times. Instead, in this paper, we focus on real-time detection of frustration.

It has long been established that mental states are often linked with automatic, physiological responses [32]. As physiological sensors continue to be integrated into phones and other wearable devices, it creates an opportunity for developers who are interested in understanding or mitigating real-time user frustration in mobile computing. However, to date, most efforts at detecting frustration from physiological responses rely on stimuli and sensors that cannot generally be applied in a mobile setting [19, 21, 31, 42]. Additionally, these models use binary classifiers to detect the presence or absence of a frustrating stimulus, thus limiting the models' utility when faced with multiple levels or sources of frustration [3, 4, 31].

In this paper, we demonstrate that introducing variable length delays in system response time provides researchers with a generalizable, granular control that can be used to induce varying levels of frustration. We then build machine learning models that use wearable physiological sensors to detect five levels of frustration, a granularity often reported in human factors research. We discuss methods for optimizing these models and demonstrate that they can detect levels of frustration at accuracies comparable to other models that only treat frustration as a binary state (i.e., frustrated or not frustrated).

We will begin with a discussion of relevant background material, including a theoretical definition of frustration, empirical evidence connecting system delays to frustration, sources of delays, and previous efforts to detect frustration. We use this to motivate our design of an approach to detecting frustration that can apply to mobile computing. We will then explain our experimental design and the process we used to analyze our data. Next, we will present the results of our study and provide detailed analysis of how

UbiComp '15, September 7-11, 2015, Osaka, Japan.
Copyright 2015 © ACM 978-1-4503-3574-4/15/09…$15.00.
http://dx.doi.org/10.1145/2750858.2805847

various model parameters impact the detection of frustration. Finally, we will discuss the implications of this research and our goals for future work.

BACKGROUND

Detecting frustration is not a novel idea. However, to date, most research on frustration detection has focused on goal-driven factors that induce frustration or systems that detect frustration as a binary phenomenon. There are issues with both of these approaches. Goal-driven frustration, like that caused by struggling to learn calculus, will not generalize to commonplace frustration induced by system performance in mobile applications. Binary approaches are too coarse-grained to accurately capture how multiple factors such as time pressure and task criticality can combine to intensify user frustration.

In the remainder of this section we define what we mean by frustration in the context of the mobile computing environment. We will then discuss human factors research that supports our approach for inducing frustration by inserting response delays into interactive systems. We will discuss other research aimed at frustration detection and examine the limitations of their approaches for mobile computing environments. All together, this section highlights the value of our contribution: a near real-time method for detecting ranges of frustration responses caused by system response delays.

What is Frustration?

Frustration, like many emotional states, is far easier to experience and intuitively understand than to precisely define. Psychologists have spent considerable effort developing theories of frustration to understand and explain exactly what people mean when they say they feel frustrated. In this paper, we use the viewpoint expressed by Rosenzweig that frustration is "*the occurrence of an obstacle that prevented the satisfaction of a need*" [25].

In the context of a specific computer application, users typically have a well-understood need. Obstacles to fulfilling these needs can derive from a wide range of sources. For example, a researcher writing a paper may be frustrated by the difficulty of finding the right words (a goal-driven factor), figuring out how to set his text to the desired format (a design-driven factor), by a slow computer (a system-driven factor), or by some combination of these and other factors.

How a user responds to frustration is influenced by more than just the degree of frustration they experience. For example, imagine two students dealing with a slow Internet connection. The student doing research for a critical presentation is likely to feel a higher level of frustration than one browsing the Internet during a coffee break. Both students have objectives (preparing a presentation, being entertained) that qualify as needs in their respective contexts, so both are likely to be frustrated by the slow connection. However, the relative importance of their tasks will likely result in very different responses.

The Pleasure-Arousal-Dominance (PAD) emotional state model [28] seeks to explain these differences using variations along three dimensions rather than precisely distinguish between discrete emotions. Frustration, in this sense, is not a binary state that a user is either in or not. It can have multiple, concurrent sources and varying context-specific effects.

Note however, that we are not seeking to pinpoint an absolute emotional state. We use the term frustration as a shorthand expression for negative affect experienced as a result of an obstacle to a goal. Any one individual's response to an obstacle is likely differ from another person's and may well differ from their own response to a different source of frustration.

System Response Delays and Frustration

Decades of human factors research have examined how delays in system response impact the user experience. Numerous studies have confirmed that task performance and user satisfaction decrease as system response times increase on tasks ranging from simple data entry to code debugging to problem solving [6, 9, 14, 15, 23]. More recent studies have demonstrated a negative correlation between system response time and user satisfaction in web page and browser-based applications [17, 30]. In fact, while factors such as personality type have been shown to influence the level of reported frustration [13, 15], system response time was found to be the most important factor related to user satisfaction with computing systems [35].

Thus, prior research confirms the intuitive idea that slower, less responsive applications are frustrating and that the perceived level of frustration scales with the length of system response delays regardless of the particular application context. Additionally, work by Sheirer et al. [38] demonstrated that introducing random delays into mouse movement responses was effective as a frustration condition. However, they only performed a binary analysis, comparing instances with a delay to instances with no delay, and did not explore the impact of variable delay lengths. Building on this foundation, we hypothesized that introducing variable system response delays would allow researchers to induce varying levels of frustration.

If this hypothesis holds true, it offers two benefits. First, by detecting multiple levels of frustration it aligns our model with the PAD theory of frustration and allows for the relative comparison of frustration across different contexts. Secondly, since system response delay is a universal and well-defined phenomenon, studying delay-induced frustration can inform developers about the impact of network conditions or algorithmic choices [1, 41].

Sources of System Response Delays

Today, when system response delays are discussed, most people immediately think of an experience with a slow internet connection. However, network bandwidth limits are by no means the only source of system response delays. Much of the human factors research on the topic stems from a time when access to remote mainframes was the primary computing bottleneck [20]. Limited processing power still impacts system response delays when dealing with computational complex applications like speech recognition [26].

The fundamental constraints on resources in mobile computing often exacerbate these delays [36]. Infrastructure such as cell networks introduce additional lag into network communications [2]. Battery limits have resulted in a wide range of energy saving techniques that often come with a tradeoff in response time [24]. In order to properly evaluate the cost of these trade-offs to the user, methods for detecting user frustration as a result of such delays need to be explored.

Detecting Frustration

Research into frustration detection has encompassed a wide variety of sensing modalities and use contexts. Here we will highlight a subset of this research and consider its applicability to frustration detection in a mobile computing environment.

Education and tutoring systems are one context that has received a lot of attention in frustration research. Here, the goal is that one day tutoring systems will be able to detect frustration and intervene so that students "persevere and remain motivated through failure" [5]. Kapoor et al. [18] lay out a convincing case for the utility of frustration intervention in tutoring systems. Their work uses a multimodal system (eye tracking, posture chair, pressure mouse, skin conductance, game state) for predicting frustration, which is determined when users' click a button indicating they are frustrated. Another work involving a different tutoring system and slightly different sensor sets foregoes the direct analysis of frustration detection and evaluates only the system's impact on learning [8]. While these efforts offer compelling use-case scenarios for frustration detection, the methodologies they use rely on frustration induced by learning, which does not generalize to the detection of frustration in other contexts.

Outside of education, other researchers have relied on behavioral observations. One study used application usage statistics in a programming environment to predict frustration, and then relied on observers to recognize and record when novice programmers appeared frustrated [34]. Another study used speech patterns to classify emotional state and similarly employed observers to determine when a user sounded frustrated [4]. Belle et al. used physiological sensors to distinguish between a baseline (rest) condition and a state of frustration [3]. While they achieved promising results on a small dataset, they discussed the utility of distinguishing *levels* of frustration as important future work.

Frustration Detection in a Mobile Computing Context

One study that avoided many of the complications of incorporating additional sensors was conducted by Gao et al. [12]. In this work, users played a smartphone game and variations in touch characteristics (e.g., pressure) measured by the touchscreen were used to distinguish various emotional states including frustration. While their results were certainly promising, their models performed best when the emotional response was treated as a binary classification, either a high or low state of arousal. Even so, exploring the impact that touch characteristics could add in addition to physiological responses would certainly be worthwhile in future work.

Taken as a whole, we observed three general issues that limit the applicability of existing frustration detection research in a mobile environment. The first is that many approaches rely on sensing modalities that do not easily lend themselves to a mobile environment. Methods that rely on measures of posture, mouse gripping pressure, and even camera-based estimates of facial expression do not easily translate to a mobile setting [8, 19, 21, 29, 31, 43]. Secondly, many studies rely on highly goal-driven inducers of frustration such as learning, which are not present in all applications. We propose that frustration induced by delays in system response can be introduced more broadly, even in conjunction with other frustrating stimuli, to better develop models of physiological response to frustration. Finally, most efforts to detect frustration have treated it as a binary state. Frustration exists on a continuum and methods for inducing and detecting multiple levels of frustration need to be explored. This is especially true for the implementation of optimization techniques when there is a trade-off between system response times and other factors like rates of energy use.

EXPERIMENTAL DESIGN

Our ultimate goal is to work towards a system of real-time frustration detection for use in mobile computing contexts. Towards this end, the three primary limitations we identified in previous research were: 1) sensor sets that do not lend themselves to mobile platforms 2) the use of goal-driven sources of frustration that cannot easily be introduced into other applications and 3) the use of binary frustration conditions that do not adequately capture the continuum of frustration.

Task: Playing Breakout

In an effort to address these three issues, we chose an interactive game, Breakout, as our application testbed. Gaming applications have been used in previous frustration detection studies [12, 38] and offer several practical advantages. The gameplay is engaging, yet simple enough that additional sources of frustration (e.g., due to learning or planning strategies) are minimized. The interface also requires continual interaction in a way that ensures increases in system response delays make gameplay more difficult. While the gameplay's high level of interactivity may overemphasize the impact of system response delays relative to more static applications (e.g. web browsing), it also ensures more exposure to the frustration stimuli within the study timeframe.

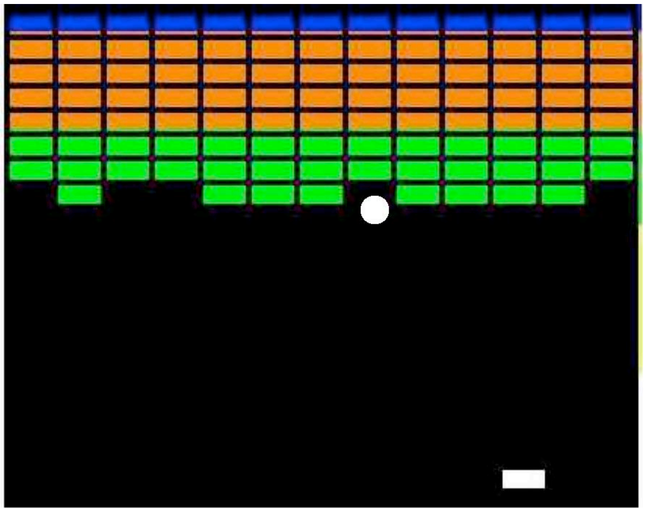

Figure 1. Screenshot of the Breakout interface. The player drags the paddle along the bottom of the screen in order to deflect the white ball back up into the colored bricks.

An open source version of the game Breakout for the Android operating system was modified for our experiment (see Figure 1) and run on a Samsung Galaxy Tab 10.1 touchscreen tablet. During gameplay, users control the position of the white paddle at the bottom of the screen via direct touch. The objective of the game is to manipulate the paddle so it deflects a ball into a grid of blocks at the top of the screen. Points are gained as blocks are removed when hit by the ball. Additional performance metrics such as paddle hits and misses were also recorded.

The game was modified so that we could introduce additional latency into the system, between when the Android operating system detected a touch event and when that touch event was reported to the game play engine. We were not able to reduce latency beyond the intrinsic floor set by the tablet hardware and operating system.

We also built in two distinct gameplay modes, each with two difficulty settings. In the first gameplay mode, *speed mode*, adjusting the difficulty modified how quickly the ball moved (14 or 28 pixels per refresh). In the second gameplay mode, *size mode*, the size of the paddle decreased (from 102 pixels to 51 pixels wide) when the difficulty was increased. Both gameplay modes were calibrated such that the easier difficulty settings had identical ball speeds and paddle sizes. The gameplay mode and difficulty settings represent standard settings that exist in versions of Breakout. They also provide a reference to examine how the additional latencies impact frustration and gameplay performance relative to standard game features.

Method

We recruited 24 participants (13 female) ranging in age from 20 to 41 (mean 28.1). Participants were equipped with a variety of physiological sensors in our lab, and then asked to play the game Breakout on an Android touchscreen mobile tablet for 90-seconds to familiarize themselves with the gameplay. The timeframe was determined by the average completion time a round by pilot study participants. Participants were asked if they were comfortable with the gameplay and sensors and then, after confirming this, began the first round of the Breakout game, comprised of twelve 90-second sessions with randomized levels of difficulty and latency. After the first round of 12 Breakout sessions, participants completed a series of Elementary Cognitive Tasks (ECTs) [7] on a separate laptop to obtain data to validate the collected physiological response measures. Finally, participants played a second round of twelve sessions of Breakout with an *alternate gameplay mode*.

During Breakout gameplay, the 12 conditions consisted of a difficulty setting (harder or easier) and an introduced latency (0, 25, 50, 100, 200, or 400 ms) and were presented in a randomized order. The gameplay mode was consistent throughout each round of 12 Breakout sessions, but alternated between the rounds so that each subject experienced both the *speed mode* and *size mode*. After each 90-second session of gameplay, subjects were presented with six questions from the NASA-TLX (Task Load Index). Included in these questions was an assessment of frustration (5 point Likert-scale: *Very Low, Low*, *Medium*, *High*, *Very High*) that we used as ground truth for our models.

Physiological Sensors

Subjects were equipped with three different sensor devices: a BodyMedia armband, a wireless heart rate detector, and a finger clip skin conductivity sensor. These devices were chosen both because users can easily wear them when mobile, and because they capture physiological responses that have been shown to be useful in detecting frustration or negative affect [16].

BodyMedia Armband

The Body Media Sensewear Pro3 armband was placed on the subjects' right upper arm. It senses a number of physiological responses including heat flux, skin temperature, electrodermal activity (EDA), and electrocardiogram (ECG) data, all at a rate of 32Hz.

Zephyr Bioharness BT

The Zephyr Bioharness BT chestband was placed around the subjects' chest. The Zephyr provided two separate data streams. The first included data such as heart rate, breathing rate and skin temperature, sampled at 1Hz. A second data set reported lower level breathing data and R-R intervals from the ECG, reported at approximately 18 Hz. Subjects were asked to wear the chestband directly against their skin if they were comfortable doing so. Two participants opted to wear the harness over their shirt.

Lightstone Fingertip Sensor

The Lightstone sensor was worn on the index, middle, and ring fingers of the subjects' non-dominant hand. Electrodermal activity, heart rate data, and a sensor gain value were recorded at 32 Hz.

Validation of the Physiological Response Measures

To validate that our approach could adequately model psycho-physiological responses to mental states, we replicated the procedure described by Haapalainen et al. [16]. Participants completed sets of 5 different ECTs. The tasks themselves are well established in psychology studies and are designed with parameters that can be adjusted to reliably vary the task difficulty [7, 27]. These parameters had been previously verified with subjective assessments and time-to-completion measures [16]. Using the physiological responses recorded during these ECTs, we built models to detect whether the participant was engaged in a more or less cognitively demanding trial. The results we obtained, an average of 91% classification accuracy, were in line the previously reported findings of 81.1%. Thus, we reasoned that that insofar as psycho-physiological responses to frustration are similar to responses induced by mental workload, our sensor set and classification technique are appropriate.

DATA ANALYSIS

While our primary goal was simply to detect frustration from physiological responses, we also sought to examine the relative efficacy of various sensors and specific physiological responses for sensing levels of frustration. The exact sensors and features we used in any particular model will be discussed in the description of the model. Here we describe our general data analysis process.

Preprocessing

Prior to any analysis, we noticed occasional timestamp errors and removed any data points with timestamps that did not match the recorded study times. We also filtered out any heart rate measurements that fell outside of the range of 35-155 beats per minute.

Features

A total of 12 physiological measures were recorded from the set of physiological sensors we used. From these data we took the mean, median and variance values across the specified time windows to create 36 features.

Sensor Errors

Throughout the study, there were some errors in the collection of the physiological data. The BodyMedia armband collected no data for one subject. Two subjects opted against wearing the Zephyr heart rate monitor underneath their shirts, thus impacting the ECG data. The heart rate monitor failed to record any data for one subject. We also observed higher variability in some of the sensor data during the first rounds of gameplay for the first two subjects. From the third subject on, we adjusted our calibration and familiarization routine so that the sensors stabilized before any gameplay rounds. The software logging tool for the Lightstone finger tip sensor had a systematic timeout issue: after about an hour of operation, recorded data values often dropped across all subjects.

To account for these errors, each sensor source was given a quality flag for each gameplay segment. As we tested models using different features, the data set was expanded or contracted according to the quality flags of the desired features. Thus, for models using BodyMedia data, one subject is excluded, whereas models relying on Zephyr ECG data would exclude three subjects. Models using both sensors would exclude all four subjects.

RESULTS

The analysis of our results will be presented in several parts. We will begin by verifying our hypothesis that introducing variable system response delays induces varying levels of frustration. We will also explore the relationships between various game settings and gameplay performance across latencies to examine other factors that may impact frustration. Next, we will present the results of a simple model based on the procedure we presented in our validation process. While this model's results were encouraging, it left a number of questions that would need to be answered for such a system to work in a mobile setting:

- Can we reduce the complexity of the psych-physiological sensing system while maintaining accuracy?
- Do delays in physiological responses limit the responsiveness of our classifier?
- Can we further increase accuracy by including application settings in the model?

After examining each of these aspects individually, we apply our findings to a new model that achieves a classification accuracy of over 80% for classifying 5 levels of frustration.

Assessing Frustration

The first thing we examined was the relationship between reported levels of frustration and the amount of additional latency introduced into gameplay. As seen in Figure 2, there is a clear monotonic relationship between the two. This means that our hypothesis that we can induce multiple levels of frustration by varying system response time is confirmed.

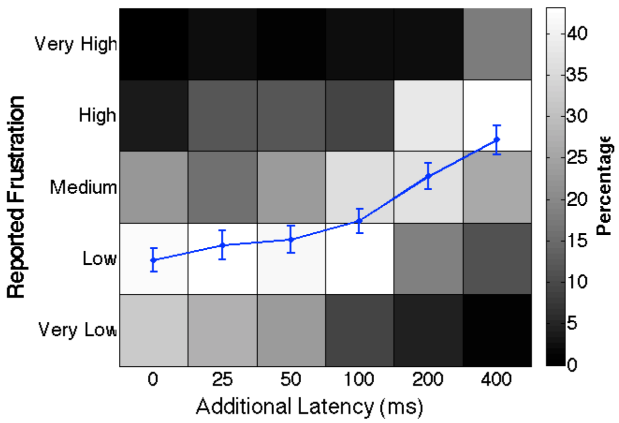

Figure 2. Average level of reported frustration by additional latency. Error bars represent the confidence interval at each latency. The surface shades represent the percentage of surveys reporting each level of frustration at a given level latency.

As we examine the data more closely, though, it is clear that latency was not the only source of frustration. As described in the Experimental Design section, we implemented two different gameplay modes (*speed mode* and *size mode*) that exhibited different effects in the higher difficulty setting (though they were identical in the standard, lower difficulty mode). We expected that the different gameplay modes would impact frustration, as well as the user performance in the game as shown in Figures 3 and 4, respectively. Overall, the *speed mode* induced a significantly higher level of frustration than gameplay with the default ball speed (p < .000001, using paired 2 tailed t-test between sets). However, the smaller paddle size in the *size mode* did not induce significant differences than standard gameplay.

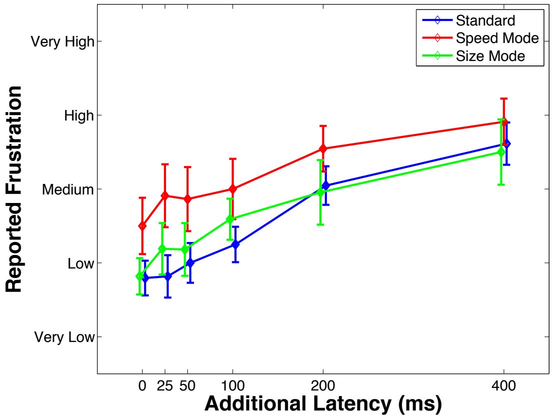

Figure 3. Reported frustration given different gameplay settings across the various additional latencies.

While the aggregate trends between reported frustration and gameplay performance across latencies look similar, we see discrepancies when we look at the different gameplay modes. Figure 4 presents a measure of user performance in the form of the percentage of the time the user was able to make contact with the ball before it went off screen. By this metric of performance, users performed significantly worse (p < .00001) using the small paddle in the difficult *size mode* relative to the *standard mode*. Despite this performance degradation, the reported frustration was not significantly different between these modes. However, the increased ball speed in the difficult *speed mode* resulted in both worse performance and higher reported frustration.

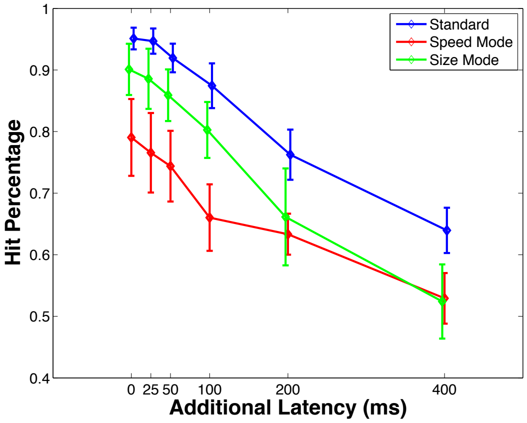

Figure 4. The rate at which subjects were able to hit the ball given different gameplay settings across the various additional latencies. Both modes resulted in significantly different performances across difficulty settings.

From this, we can conclude that merely looking at how well a user is performing in their task does not directly inform us to how frustrated they are. For example, with 400ms latency, there is a clear difference between the hit percentage in small paddle mode and the standard mode (Figure 4), but this performance difference is not reflected in the level of frustration users report (Figure 3). From this we can see that frustration is not just a matter of how well one performs a task. Instead, as one would expect, system delays and other interface features (e.g., paddle size, ball speed) combine to impact a user's performance and experience. Thus, as will be explored in the 'Model with Application Settings' section, we expect that application settings are important features in modeling frustration.

Naïve Bayesian Models of Multi-level Frustration

Throughout this work, all models were developed following the procedure used by Haapalainen et al. to model levels of cognitive load [16]. We built individually trained Naïve Bayesian models to classify each users' reported level of frustration. The data was labeled by the frustration reported during the survey following each round of gameplay. Since we were not developing a population model, we used a leave-one-out training and testing method to mitigate against limited data samples. All results are presented as an average performance of the 24 individual models trained and tested on a single individual. Thus, following our approach would require the collection of new data and the training of a new model to predict frustration for any given individual.

Prior to any parameter optimization, the data sets, labeled once per 90-second gameplay round, were broken into 15-second non-overlapping windows to ensure that each model had multiple instances of each frustration class. The 36 features previously described, were calculated for each 15-second segment and were all used in the model. Across all 15-second segments, the models correctly classified the frustration level 48% (1293 of 2718) of the time. This is considerably higher than the 20% accuracy that would be expected for a uniformly distributed five-class model and the 32% accuracy that would be obtained from always predicting the most likely class (low frustration). At first glance, these results appear unimpressive compared to previous works that reported accuracies in the 80-90% range [3, 4, 18, 31]. However, it is important to note that all of these previous studies treated frustration as a *binary* state. For comparison's sake, we can evaluate our models' ability to detect whether or not the user is quite frustrated (treating reports of *High* or *Very High* frustration as one state and reports of *Medium* or lower frustration as another). If we evaluate our model in this way, it achieves 75% accuracy (2033 of 2718).

We will now turn our attention to aspects of our models that can be optimized for better overall performance.

Device Comparisons

While we found the results of our initial models to be encouraging, a system requiring a user to wear three different devices to collect data for 36 unique features is unlikely to be accepted outside of a laboratory in the near future. We thus turned our attention to the contributions made to our models by different feature sets.

Our original Naïve Bayes model was built using 36 data features from 3 different devices (BodyMedia armband, Lightstone fingertip sensor and Zephyr chestband). However, since the Zephyr Heart Rate Monitor reported different sets of data at different sampling rates, we will treat it as two separate devices in the following discussion. With four devices, there are 15 unique combinations that can be made (see the bottom half of Figure 5 for the combinations). For each combination, we created a data set using only the features provided by the included devices. We then followed the same procedure as in the previous section to create and evaluate models using those limited feature sets. The aggregate classification accuracy for each data set is reported in top half of Figure 5.

The results from the model using the complete dataset, as described in the Data Analysis section, can be seen in the first feature set combination in Figure 5. The fact that several feature set combinations outperform the complete dataset indicates that some amount of overfitting may be occurring. The best performing combination, correctly classifying the frustration level 56.5% of the time, was the data set using features from the BodyMedia armband and the Lightstone fingertip devices.

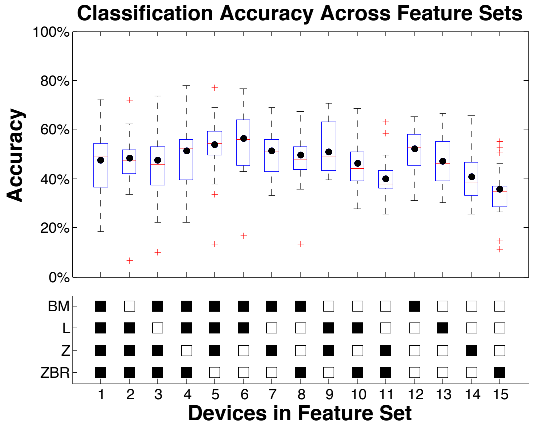

Figure 5. The classification accuracies for models built with different combinations of device data. The lower chart indicates which devices' data are included in the models (BM-Body Media armband, L- Lightstone Fingertip sensor, Z-Zephyr Heartrate data (1Hz), ZBR- Zephyr other data (18Hz) The upper chart consists of boxplots with a dot at the mean indicating the distribution of classification results for the individual models for given a device set.

Time Effects

One difficulty in classifying continuous phenomena in near real-time lies in identify appropriate sampling sizes and rates. It is known that different physiological responses operate on time scales ranging from a fraction of a second to several minutes [22, 33]. Emotions and engagement can also vary on different timescales. Consequently, the time-windows over which data features are calculated and labels are generated may play a relevant role in the performance of the classification system.

In our initial analysis, data from the 90-second gameplay rounds were divided into non-overlapping 15-second data segments to avoid any singular classes in our data set. However, there was no empirical reason to subdivide the dataset into 15-second windows. In fact, the time window over which data is sampled acts as an inherent delay in the system's ability to classify new physiological reactions. To examine how our system could perform over shorter timeframes we recalculated the feature set using non-overlapping windows of 1, 2, 3, 5, and 10-seconds, in addition to our initial 15-second windows. To avoid issues with limited data samples, we removed data from the Zephyr heart-rate monitor that was reported on a 1-Hz basis (Thus, this set matches feature set number 4 as presented in Figure 5). The average classification accuracies across all users are plotted across the various window sizes in Figure 6. As can be seen, the classification rates are higher and relatively consistent for time windows under 5 seconds.

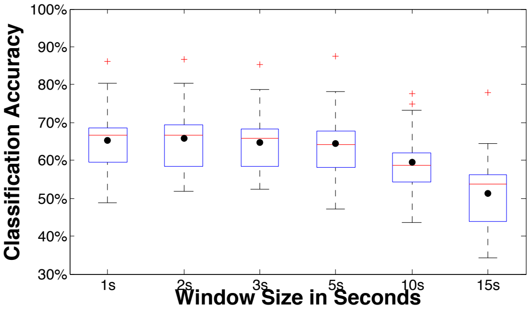

Figure 6. Boxplots of classification rates across users when features are calculated with different window sizes. The models used here incorporate data from Feature Set 4 (see Figure 5). The rates seem relatively stable for windows <= 5 seconds.

Having demonstrated that a window size of no more than five seconds provides better classification results, we next explored how the relative ordering of the windows impacted classification results. Given that the physiological responses being measured take some finite amount of time to be initiated after exposure to a stimulus, we anticipated that there might be a noticeable lag in classification accuracy. This does, in fact, seem to be the case as seen in Figure 7. Running the models on consecutive 5-second windows results in a noticeably lower average classification rate in the first 30 seconds of a trial. Furthermore, if the first 30 seconds of data represents a physiological response to previous stimulus (due to lag), then models trained with this data would be expected to have degraded classification performance.

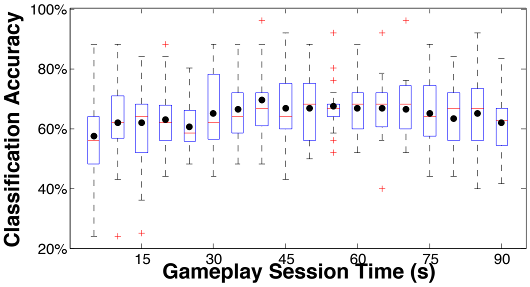

Figure 7. Boxplots of the classification rate for each 5-second window cross the 90-second session. The circle represents the average accuracy across all subjects.

Taking into account both of these results (smaller window sizes improve classification performance, and the first 30 seconds of a session has worse performance), we developed another model using 5-second time windows to calculate features but only training on data from the final 60 seconds of each session (and not training on the initial 30 seconds). This led to reduced classification accuracy over the first 30 seconds, as would be expected. Over the final 60 seconds, however, the classification rate was boosted to 67.44% with less variation in classification performance. Figure 8 shows the classification rate by window, for the new model.

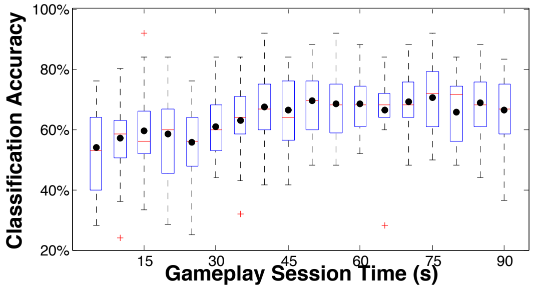

Figure 8. Boxplots of the classification rates for each 5-second window across the 90-second session. The circle represents the average classification accuracy across all subjects. The model was trained on only the last 60 seconds of data from any given session.

These results are presented again in a confusion matrix, in Table 1. If we examine the model's ability to distinguish between higher and lower levels of frustration, the model now has an accuracy of 86% (treating reports of *High* or *Very High* frustration as one state and reports of *Medium* or lower frustration as another). If we accept predictions within 1 level of the reported level of frustration (thus for a reported frustration of *Very Low*, a prediction of *Very Low* or *Low* is considered correct), the model provides a correct classification 88.2% of the time.

	Very Low	**Low**	**Medium**	**High**	**Very High**
Very Low	550 (65%)	100	81	70	16
Low	159	1315 (76%)	325	157	33
Medium	82	203	935 (61%)	93	16
High	26	113	177	712 (67%)	45
Very High	23	9	18	24	154 (58%)

Table 1. The aggregate frustration confusion matrix for the Naïve Bayesian model using the features with greater than 1Hz sampling rate (27 features) and 5-second windows. The columns represent the reported level of frustration and the rows indicated the model's prediction. The count of correct prediction is indication along the highlighted diagonal. Values in parentheses indicate the percentage of prediction (percentages sum to 100% along the columns.

It is worth noting that a fundamental limitation of the subjective surveys we used to provide ground truth for our models is that they require active engagement and take a non-trivial amount of time to complete. This methodology simply provides no way to record 'ground truth' levels of frustration in real-time. One possible reason that the 'Very High' level of reported frustration was relatively poorly classified is that subjects often disengaged with the task when exposed to the highest latency. Subjects who laugh and/or stop trying during high latency gameplay sessions may still have reported their frustration to be 'Very High', but it is unlikely that their physiological responses were consistent.

Model with Application Settings

Given the effects that different gameplay modes had on the reported level of frustrations across latencies, it is reasonable to assume that a model built with information about these settings would outperform a model without this information. To explore precisely how the gameplay settings impact frustration classification, we repeated the feature set and timing parameter adjustments that we reported previously, on models that included the difficulty setting and gameplay mode as additional features.

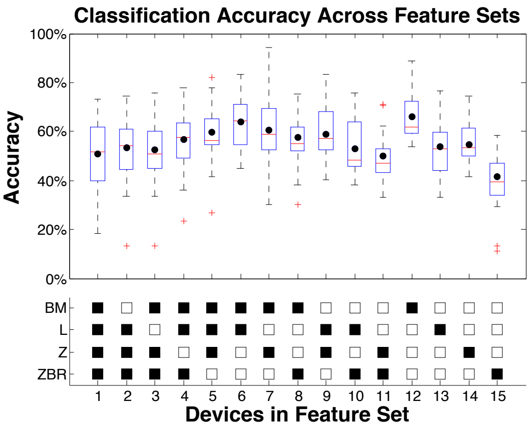

Figure 9. The classification accuracies for models using different feature sets when game settings (difficulty and mode) are included as features. Using the Body Media device alone (12 total features) produces the highest average classification rate.

As expected, the models with information about the gameplay settings outperformed the models without across every feature set with an average improvement of 7.8% (compare Figures 9 and 5). Interestingly, when gameplay settings were included the feature set with the highest accuracy, 66.1%, included only features from the BodyMedia armband (column 12 in Figure 9). As for timing parameters, these models performed the best with 1-second windows when allowing for a 20-second lag by training only on the final 70 seconds of each session. For the following analysis, these timing parameters and only the BodyMedia armband features and gameplay settings are included in the models.

Sensor Features

As described in our validation section, we followed a previous established approach to broadly survey a variety of commercially available physiological sensors. We chose to explore commercially available sensors with the hopes of finding a set of existing sensors that could be immediately used for further studies into the viability of frustration detection in a mobile platform. While we must leave a deeper exploration into optimizing the of feature selection beyond the by-device sets we previously examine, Figure 10 gives some sense of the variations in physiological response being modeled.

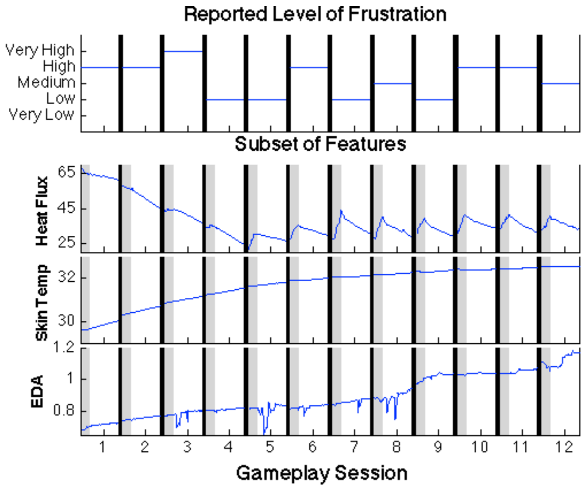

Figure 10. Reported levels of frustration and a subset of mean values over a 1-second time windows of data signals recorded by the Body Media Arm Band. The sessions are presented in recorded order with each gameplay session lasts 90 seconds. The gray bars represent the 20 seconds of data left out of training sets. The black bars represent gaps in time during which subjects filled out surveys.

The 12 gameplay sessions presented in Figure 10 represent data collected during user 4's set of games in *speed mode*. The top plot reports the level of frustration reported immediately after each round of game play.

Note that these samples are not representative of the all users' physiological responses. User 4 was selected because they had the highest standard deviation amongst reported levels of frustration. The three features presented were chosen because they had the highest degree of correlation with the reported levels frustration for the *speed mode* gameplay subset of User 4.

It is interesting to note the early peaking of Skin Temperature that takes place in the first 20 seconds (represented by the gray bars of Figure 10) of several rounds of gameplay that corresponds to 20-second lag time we estimated for these models. However, we caution against reading too much into the feature values themselves as individual responses vary.

Final Model Performance

Following the same process we described before, we trained and tested the models using 14 total features. These features consisted of 12 physiological features measured by the Body Media armband (the mean, median, and variance of the skin temperature, heat flux, EDA, and ECG signals) and two gameplay settings (difficulty and game mode). We found the best model performance using 1-second time windows for calculating physiological features and leaving the first 20-seconds of each trial out of the training set.

Putting all this together gives us the results shown in Figure 11. This model correctly classified the reported frustration

77% of the time. Once the trial hit the 20-second mark, the five class classification accuracy averaged 80.3%.

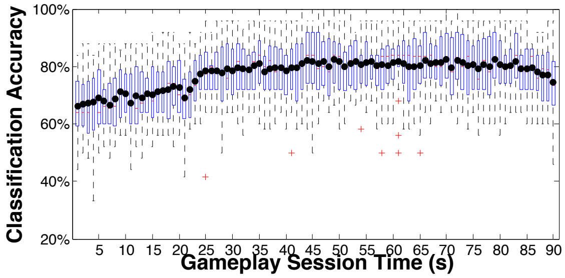

Figure 11. Boxplots of the classification rates for each 1-second window across the 90-second using only the 12 Body Media features and Game Settings. Trained on only the last 70 seconds of data. Classification accuracy averages over 80% for the final 70 seconds. (77% for all 90 seconds)

DISCUSSION

Frustration, in the context of mobile computing applications is directly connected to the user's ability to achieve their intended goal with that application. Any number of factors, from application aesthetics to responsiveness, can combine to increase or mitigate a user's frustration. A large part of what application designers do is optimizing design and performance to make their products more appealing and less frustrating. In real-world settings, though, computing resources fluctuate making performance and frustration unpredictable.

Our work demonstrates that controlling system response delays can induce varying levels of user frustration. This is useful in two ways. First, whereas previous research used delays as a frustration stimulus [38], we demonstrated that controlling the length of delays induces varying *levels* of frustration. This is more representative of how frustration is described and can help researchers build more sophisticated and accurate models of frustration. Secondly, since variable system delays are often a user-observable effect of underlying resource limits in mobile computing [37], this can better inform designers of how real-world conditions may affect user satisfaction. For example, researchers who measure the delays caused by network conditions [41] or algorithmic choices [1] can use explorations of frustration induced by system response delays to better understand the impact of their work on the user.

A second important contribution of this work is that it demonstrates that physiological responses to delays can be modeled to detect multiple levels of frustration without a severe reduction in accuracy. This is an important differentiation from past work that has treated frustration as a binary value. The fact that we could achieve 80% classification accuracy for 5 levels of frustration (see Figure 11) using commercially available physiological sensors in even a relatively simple context is an encouraging step. However, we acknowledge that there is much more work that needs to be done.

There are also significant limitations to the approach that we took. The models we built are not population models; thus, supervised training would be required for any additional user. For tasks that induce a wide range of frustration or simply take a long time to complete, acquiring ground truth estimates of frustration will likely require much more sophistication than a post-hoc survey. Using a system such as we propose in a truly mobile environment would also require an understanding of how factors like physical exertion and external sources of distraction impact physiological responses. There is also significant work to be done exploring other classification models and physiological response features.

CONCLUSIONS

Noting the well-established inverse correlation between system response times and user satisfaction, we hypothesized that system response delays would be a reliable source of user frustration. Our user study confirmed this and demonstrated that in addition to inducing frustration, varying the length of system delays induces different levels of frustration. This was confirmed by self-reports of frustration and by detectible differences in physiological responses.

Having established the connection between system response delays and induced levels of frustration, we then developed individually trained models to detect the users' level of frustration from their physiological responses. We built on previous research by using a small set of wearable devices that can be used in a mobile setting, by inducing frustration in an application-independent manner, and by treating frustration as a multi-class (5-class) phenomenon rather than as a binary state. We then sought to reduce the number of physiological sensors needed and to optimize the timing parameters in our models. Ultimately, we were able to classify 5 levels of frustration induced by delays with 80% accuracy, using the physiological sensors on a BodyMedia armband, known application settings, and empirically determined physiological response times. This accuracy is comparable to what has been achieved by binary classification of frustration.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation under grant IIS-1065336 and the NSF Quality of Life Technology Engineering Research Center under grant EEC-0540865. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and should not be attributed to Carnegie Mellon University or the funding sources. Special thanks to Himanshu Zade for his help.

REFERENCES

1. Abe, Y., Geambasu, R., Joshi, K., Lagar-Cavilla, H.A., and Satyanarayanan, M. vTube: Efficient streaming of

virtual appliances over last-mile networks. *Proc. Symposium Cloud Computing*, (2013), 1-16.

2. Akamai. State of the Internet, First Quarter 2015. http://www.akamai.com/dl/content/q1-2015-soti-report.pdf
3. Belle, A., Ji, S.Y, Ansari, S., Hakimzadeh, R., Ward, K., and Najarian, K. Frustration detection with electrocardiograph signal using wavelet transform. *IEEE Biosciences*, (2010), 91-94.
4. Boril, H., Sadjadi, S. O., Kleinschmidt, T., and Hansen, J. Analysis and detection of cognitive load and frustration in drivers' speech. *Proc. INTERSPEECH*, (2010), 502-505.
5. Burleson, W., and Picard, R. Affective agents: sustaining motivation to learn through failure and a state of "stuck." In *Workshop on Social and Emotional Intelligence in Learning Environments,* (2004).
6. Butler, T. Computer response time and user performance. In *Proc. SIGCHI Conference on Human Factors in Com. Sys.,*(1983), 58–62.
7. Carroll, J.B. *Human Cognitive Abilities: A Survey of Factor-Analytic Studies,* Cambridge University Press, Cambridge, UK, (1993).
8. D'Mello, S. K., Craig, S. D., Gholson, B., Franklin, S., Picard, R., and Graesser, A.C. Integrating affect sensors in an intelligent tutoring system. In *Affective Interactions: The Computer in the Affective Loop Workshop*, (2005), 7-13.
9. Dannenbring, G. L. The effect of computer response time on user performance and satisfaction: A preliminary investigation. *Behavior Research Methods & Instrumentation* 15, 2 (1983), 213–216.
10. Ekman, P., Levenson, R.W., and Friesen, W.V. Autonomic nervous system activity distinguishes between emotions. *Science,* 221 (4616), (1983), 1208-1210.
11. Forman, G. H., and Zahorjan, J. The challenges of mobile computing. *Computer* 27, 4 (1994), 38–47.
12. Gao, Y., Bianchi-Berthouze, N., and Meng, H. What does touch tell us about emotions in touchscreen-based gameplay? *ACM Trans. Comput.-Hum. Interact.* 19, 4 (2012), 1-30.
13. Glass, D., Snyder, M.L., and Hollis, J.F. Time urgency and the type A coronary-prone behavior pattern 1. *J Applied Soc Psych* 4, 2 (1974), 125-140.
14. Grossberg, M. An experiment on problem solving with delayed computer responses. *IEEE Trans. On Systems, Man and Cybernetics*, (1976), 219–222.
15. Guynes, J. Impact of system response time on state anxiety. *Com. of the ACM* 31, 3 (1988), 342–347.
16. Haapalainen, E., & Kim, S., Forlizzi, J.F., and Dey, A.K. Psycho-physiological measures for assessing cognitive load. In *Proc. UbiComp,* (2010), 301–310.
17. Hoxmeier, J., and DiCesare, C. System response time and user satisfaction : an experimental study of browser-based applications. *AMCIS 2000 Proceedings*, (2000): 347.
18. Kapoor, A., Burleson, W., and Picard, R. W. Automatic prediction of frustration. *International Journal of Human-Computer Studies* 65, 8 (2007), 724–736.
19. Karg, M., Samadani, A.-A., Gorbet, R., Kuhnlenz, K., Hoey, J., and Kulic, D. Body movements for affective expression: a survey of automatic recognition and generation. *IEEE Trans. Affective Computing,* 4 (2013), 341-359.
20. Kemeny J.G., and Kurtz, T.E. The Dartmouth Time-Sharing Computing System. Final Report. (1967).
21. Kleinsmith, A., and Bianchi-Berthouze, N. Affective body expression perception and recognition: a survey. *IEEE Trans. Affective Computing, 4 (2013), 14-33.*
22. Klingner, J., Kumar, R., and Hanrahan, P. Measuring the task-evoked pupillary response with a remote eye tracker. *ETRA '08: Proc. Symp. on Eye Tracking Res. & Apps*, (2008), 69–72.
23. Kuhmann, W. Experimental investigation of stress-inducing properties of system response times. *Ergonomics*, *32*(3), (1989), 271–80.
24. Kumar, K., Liu, J., Lu, Y.H. and Bhargava, B. A survey of computation offloading for mobile systems. *Mobile Networks and Apps* 18, 1 (2013), 129-140.
25. Lawson, R. *Frustration: The Development of a Scientific Concept.* Macmillan, New York, NY, USA, 1965.
26. Lei, X., Senior, A., Gruenstein, A., and Sorensen, J. Accurate and compact large vocabulary speech recognition on mobile devices. In *Proc. Interspeech,* (2013), 662-665.
27. McGrew, K.S. CHC theory and the human cognitive abilities project: Standing on the shoulders of the giants of psychometric intelligence research. *Intelligence* 37*,* 1 (2009), 1-10.
28. Mehrabian, A. *Basic Dimensions for a General Psychological Theory: Implications for Personality, Social, Environmental, and Developmental Studies.* Oelgeschlager, Gunn & Hain, (1980).
29. Mota, S., and Picard, R. Automated posture analysis for detecting learner ' s interest level. In *Com. Vis.and Pattern Rec Workshop,* IEEE (2003).
30. Nah, F. A study on tolerable waiting time: how long are Web users willing to wait? *Behaviour & Info. Tech.*, (2004), 37–41.

31. Qi, Y., Reynolds, C., and Picard, R. The Bayes point machine for computer-user frustration detection via pressuremouse. In *Proc. Workshop Perceptive UI*, (2001), 1-5.

32. Quigley, K.S., and Barrett, L.F. Is there consistency and specificity of autonomic changes during emotional episodes? guidance from the conceptual act theory and psychophysiology. *Biol Psych,* 98 (2014), 82-94.

33. Reimer, B., Mehler, B., Coughlin, J. F., Godfrey, K. M., and Tan, C. An on-road assessment of the impact of cognitive workload on physiological arousal in young adult drivers. In *Proc. AutomotiveUI*, (2009), 115-118.

34. Rodrigo, M. M. T., and Baker, R. S. Coarse-grained detection of student frustration in an introductory programming course. In *Proc of 5th Intl. workshop Com. Edu. Res*, (2009), 75-80.

35. Rushinek, A., and Rushinek, S. F. What makes users happy? *Com. of the ACM* 29, 7 (1986), 594–598.

36. Satyanarayanan, M. Fundamental challenges in mobile computing, *Proc. ACM Symp. Principles of Distributed Computing,* (1996), 1-7.

37. Satyanarayanan, M., Bahl, P., Caceres, R., and Davies, N. The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Comp* 8, 4 (2009), 14-23.

38. Scheirer, J., Fernandez, R., Klein, J., and Picard, R. Frustrating the user on purpose: a step toward building an affective computer. *Interaction with Computers 14, 2* (2002), 93–118.

39. Schalkwyk, J., Beeferman, D., Beaufays, F., Byrne, B., Chelba, C., Cohen, M., Garret, M., and Strope, B. Google search by voice: a case study. *Advances in Speech Recognition: Mobile Environments, Call Centers and Clinics*. Springer, (2010), 61-90.

40. Smailagic, A., Siewiorek, D.P., Rudnicky, A., Chakravarthula, S.N., Kar, A., Jagdale, N., Gautam, S., Vijayaraghavan, R., and Jagtap, S. Emotion recognition modulating the behavior of intelligent systems, In *Proc. IEEE Int. Sym. On Multimedia*, (2013), 378-383.

41. Tolia, N., Andersen, D., and Satyanarayanan, M. Quantifying interactive user experience on thin clients. *Computer* 39, 3 (2006), 46-52.

42. Xue, J., Cui, X., Daggett, G., Marcheret, E., and Zhou, B. Towards high performance LVCSR in speech-to-speech translation system on smart phones. In *Proc. Interspeech,* (2012), 2861-2864.

43. Zeng, Z., Pantic, M., Roisman, G. I., and Huang, T. S. A survey of affect recognition methods: audio, visual, and spontaneous expressions. *IEEE Trans. on Pattern Analysis and Machine Intel*. 31, 1 (2009), 39–58.