PlayAffect: A Developer API for Creating Affective Video Games Using Physiological and Behavioral Measures

Dept. of CIS - Senior Design 2012-2013

Dr. Norm Badler
badler@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Johnathan Mell
jmell@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Kyle P. Vincent
kylev@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Chris Chike
chike@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

ABSTRACT

Herein is proposed the creation of an Application Programming Interface (API) for the Unity 3 and 4 video game development engine that not only reads behavioral measures from traditional video game input devices (such as if there has been an increase in mouse movements and clicks) but also takes into account physiological measures from biometric devices (such as an increase in respiratory rate).

The API parses these inputs based on study results that correlated player performance and engagement with physiological signs across several different game genres. Through the use of several rudimentary machine learning algorithms, raw physiological data is transformed into data relevant to a developer, including player engagement. The results of these calculations allow a game designer to have powerful tools for detecting when players experience certain emotions, and allow for the design of affective games.

Furthermore, the API also exposes the raw data to developers wishing to propose and utilize their own learning algorithms, to allow for a rich development environment for developers of all skill levels. These development tools will enrich the game experience for the player, as well as prepare designers for the use of the next wave of non-traditional input hardware.

This report serves to illustrate the current status of the API. A brief overview of the significance of galvonic skin response (GSR), heart rate (HR), and respiratory rate (RR) in detecting player performance and engagement will be followed by a discussion of the API itself and the design choices therein.

1. INTRODUCTION

The field of video gaming has evolved significantly over its forty-year history. Originating as a niche market novelty item, video games are now accepted as a major form of media and entertainment, and they create revenue rivaling the film industry. Hardware interfaces between the human player and the video game platform have undergone significant design improvements during the industry’s history, however today’s modern player still has a surprisingly limited number of ways of using given input methods to interact with the video game (e.g. mouse clicks, keyboard presses, or in the case of dedicated consoles, motion detection and pointing). This limitation of interactivity between the player and the video game consequently constrains the level of uniqueness of each individual player’s video game experience.

Improving the uniqueness of a player’s individual gaming experience will be a key milestone in the next evolution of modern gameplay. When video games are played, emotions from the player are inherently involved (e.g. surprise, fear, nervousness, anger, etc.). Achieving a means to incorporate those emotions into gameplay will effectively expand the players’ experiences in many ways. Specifically:

1. Game difficulty could increase/decrease based upon the player’s emotional arousal and the type of arousal.
2. Avatars depicted in an online multiplayer game could change depending on the mood of the player (e.g. Player 1 could see that Player 2 is angry).
3. Therapeutic features regarding mood could be including in gameplay (e.g. the game can modify itself in real time, to purposely make the player angry).

Recent improvements to sensor technology in the commercial markets have created an opportunistic segue for the further evolution of game development. The logical next step is now to bring affective computing to video gaming. With affective computing, game developers can use the newfound wealth of physiological and behavioral data gathered about a player in real-time to expand the scope and interactivity of games.

The research herein has been divided into two distinct phases. Phase I comprises a study to determine the correlations of a variety of potential physiological inputs to that of logged data of gameplay events experienced in a subset of industry-standard genres. Phase IIA consists of the implementation of information gleaned from Phase I into an API designed for Unity 3 and Unity 4. This API allows the data to be parsed into a useable format; subsequently, it can be used to record emotions and modify play experiences of players in real-time. Phase IIA demonstrates the functionality of the API using pre-generated user input. The culmination of
this work is in Phase IIB, wherein a customizable API development kit is constructed, using proprietary biosensor hardware and custom software. This customizable development package enables the collection of user input in real-time and demonstrates a mechanism for producing player-customized changes in gameplay. Affective gaming is thusly enabled in a realistic and commercially practical manner.

This paper explores the concept of real-time biometric data integration into Unity 3 and Unity 4 and demonstrates its feasibility and implementation. An outline of the model and implementation of the research study is provided, and detailed information is included to demonstrate how the study’s results relate to the creation of the aforementioned Unity 3 and Unity 4 affective gaming API. The API is discussed in detail, as are its tripartite design goals of usability, extensibility, and compatibility. The correlations found between the measured physiological inputs and emotional state are analyzed, as are the initial results from the API-included classifying algorithms. Finally, avenues for future improvement are outlined.

2. RELATED WORK

Video games, like most other entertainment devices, are meant to evoke an emotive response from the participant. Within the existing video game industry, a continuing goal is to blur the line between computer and user and allow game developers finer control over the responses of their players.

One way in which game developers attempt to interact with their audience is by implementing biofeedback systems. Broadly defined, any I/O device could be considered biofeedback after all, a keyboard requires actuation of the fingers to communicate data. Kiel Gilleade et al thus define a narrower term: affective gaming [2]. Affective gaming means that “the computer is an active intelligent participant in the biofeedback loop”. Games can become affected not only by the collection and use of explicit inputs from players (e.g. mouse clicks, keyboard typing), but also from the collection and use of unconscious emotive and physiological responses of the players. By collecting and using this type of user feedback, real-time changes in the game itself can then reinforce (or discourage) behavior from the player. These player behavioral changes then in turn can modify the game further, completing a loop that brings players closer to a truly affective environment. Even a small change in a game’s behavior can cause a statistically significant deviation in the emotion of the player.1

To achieve this level of interaction and feedback from game to player and back again, it is necessary to detect, record, and decode vast amounts of physiological data about a participant. Niklas Ravaja et al have succeeded in this task, measuring psychophysiological emotional reactions triggered by video game events [5]. Their study provides the following insights:

1. It is possible to determine whether or not an in-game event triggers an emotional response.

2. These in-game events can be compared, in order to determine which of the events best represent the emotional response.

3. Game designers can use information such as decreasing recovery time in successive emotional responses to determine how frequently an in-game event should appear.

The research discussed herein is not directly focused on how game designers should design their games, rather it is focused on how these emotional responses can be detected and provided to game designers in such a way that designers have a commercially practical mechanism to create affective games. While the idea of connecting physical response to emotional state is certainly not new, computing and hardware capabilities have only recently evolved into a form whereby user data can be collected and collated cheaply, unobtrusively, and in a real-time environment. For example, Microsoft’s recent Kinect® device can now be used to measure heart rate wirelessly, from a distance of several feet, and future developments to allow detection of pupil dilation are likely.

Inferring data from physiological sources is difficult [1]. Even so, achieving even a crude representation of affective interfacing with computers can be a great leap forward in Human Computer Interfacing (HCI) and will ultimately result in more detailed and robust interaction with all computer programs, including gaming [4]. Because of these various factors, the thrust of this project is to create a functional and commercially feasible Application Programming Interface (API) that allows for efficient programming of affective games, given a set of real-time logged physiological and behavioral data.

Previous explorations of the concept of affective gaming have not addressed the implementation challenges from the perspective of a game developer. Prior research within the affective gaming field has typically focused entirely on the data gathering aspects, sometimes focusing on a single physiological input and single game genre3. Additionally, most prior research has been embodied as contrived laboratory simulations which are difficult to apply to existing industry games and genres [6]. Sheirer is able to show a notable correlation between blood pressure and galvanic skin response (GSR) and user frustration using automatic pattern recognition with hidden Markov models [6]. Sheirer’s results are, however, confined to that single, low-valence, high-arousal emotion, and the research is somewhat restricted by the need to focus on precise synchronization between biosensors and in-game events [6].

In order to feasibly implement an affective gaming experience, a mechanism must exist whereby a game developer can efficiently customize a game based on the developer’s choice of what constitutes significant user data criteria for a given application. Prior research to-date has not addressed this need, nor explored its feasibility. In order to ascertain

1Ongoing unpublished research, Prof. Keisha Cutright, The Wharton School, indicates that slightly changing the physics of a gameplay object to seem “unnatural” and frustrating to use, causes significant (p = .08) departure from players’ feelings of control over seemingly unrelated events in their day-to-day lives.

2Ongoing research, Prof. Robert Morley, Washington University in St. Louis.

3Dr. Norm Badler, SEAS, University of Pennsylvania conducted an unpublished study that attempted to record stress levels for players engaging in a helicopter simulation. Unfortunately, such a simulation proved stressful for all participants, regardless of success at the task, so the data were difficult correlate to success.
the practicality of creating such a mechanism, aspects of Sheirer’s study are emulated within the present research, in order to determine correlation of certain physiological flags to the emotive state of the player. Heart Rate and Galvanic Skin Response (GSR) have proven to be reliable indicators, and the present research thus focuses on these two types of data, and explores a means to utilize this data effectively.4

3. BACKGROUND RESEARCH AND RELEVANCE

In order to initially assess the feasibility of creating a practical API for use in affective game development, Phase I of the present research focused on a background study of the general viability of user data extracted from various available biometric devices. This background study included a review of which game genres most practically lend themselves to affective gaming modifications, which types of biometric data were most feasible for the research, and a brief assessment of the correlation between preferred types of biometric data extracted from a game player and the player’s corresponding game performance metrics. The gaming industry self-identifies commercially available games into various genres. These genres are outlined in the following table. The Phase I research incorporated the use of two games, “Amnesia: The Dark Descent” by Frictional Games of Sweden, and “StepMania” an open source clone of a popular commercial game (“Dance Dance Revolution”) distributed under the MIT License. “Amnesia: The Dark Descent” is a survival horror video game and StepMania is a rhythm/music video game. These two genres were chosen due to the availability of games that have built-in data logging features, as well as the diametrically opposed themes of these two genres. The biometric data types selected for testing were Heart Rate (HR), Galvanic Skin Response (GSR), and Respiratory Rate (RR). These data types were selected based on their usage in prior research within the affective gaming field of study, as well as the availability of appropriate sensor hardware. Data were recorded using a biometric reader (Thought Technology Ltd.’s FlexComp Infiniti Model SA7550) with three types of readings being acquired; heart rate, respiratory rate and skin conductance, along with the simultaneous recording of key in-game events to derive correlations. The results of this research study were used to extrapolate the correlations between in-game events and a players’ physiologic state at the time of each event. The experimental setup for this background research component involved gathering test subjects for sessions of approximately 45 minutes each. Each subject was asked to sign a statement indicating that the risks and rewards of the study had been adequately explained to them. Subsequently each subject filled out a short questionnaire. The questionnaire was used to gather basic demographics information, including age and preferred game genres. The full survey is included as Appendix A. This information was used to aid in a qualitative assessment of the results. The subjects were connected to a GSR sensor on their right hand, a respiratory sensor around their chest, and an EKG sensor attached to their arms. They were asked to remain quietly seated for a period no shorter than five minutes, during which physiological data was gathered to from a baseline. After that, the subjects were introduced to the basic controls of the game that they would be playing, and then the subjects commenced their game play. After either a period of 25-30 minutes (for Amnesia), or having completed all songs (StepMania), the participants were stopped and data collection was terminated. As mentioned previously, both Amnesia and StepMania have built-in data logging features, thus as the data was collected from the biometric devices, corresponding in-game events were also synchronously logged. Thus, ranges of readings from each device were obtained, and correlations of events from each genre were matched to values recorded by the biometric devices. Using this data collection methodology, general trends in the physiological measures of HR, GSR, and RR were observed. An analysis of the results obtained for both StepMania and Amnesia is included herein.

3.1 StepMania Analysis

StepMania is a rhythm-type game in which players are asked to press certain keys on the keyboard that correspond to onscreen timing indicators and musical accompaniment. The experimental setup was such that for each participant three sets of files were collected. The first file is the player questionnaire. Due to the limited initial sample size, the questionnaire is not utilized in the data analysis segment, and is rather being utilized only for a qualitative review. The second file is a log file that includes significant events in the game. These significant events included performance statistics for StepMania. The third file includes physiological data about the participant sampled at a rate of 8Hz. In the analysis of StepMania, each player’s performance on each song is rated as either ‘poor’, ‘good’, or ‘great’. Players receive a ‘poor’ grade if they receive a score of less than 50%. They receive a ‘good’ grade if they scored between 50% and 75%. Finally, they receive a ‘great’ grade if they

Table 1: Game Genres Considered in this Work

<table>
<thead>
<tr>
<th>Game Genres</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adventure</td>
<td>Zork, Myst, The Elder Scrolls</td>
</tr>
<tr>
<td>Role-Playing</td>
<td>Final Fantasy, Mass Effect</td>
</tr>
<tr>
<td>Educational</td>
<td>JumpStart, Reader Rabbit</td>
</tr>
<tr>
<td>Racing</td>
<td>Mario Kart, Crazy Taxi</td>
</tr>
<tr>
<td>Sports</td>
<td>FIFA, Pong</td>
</tr>
<tr>
<td>Real-Time Strategy</td>
<td>Starcraft, Age of Empires</td>
</tr>
<tr>
<td>Turn-Based Strategy</td>
<td>Reversi, Sid Meier’s Civilization</td>
</tr>
<tr>
<td>Fighting</td>
<td>Mortal Kombat, Soul Caliber</td>
</tr>
<tr>
<td>Survival Horror</td>
<td>Alan Wake, Amnesia: The Dark Descent</td>
</tr>
<tr>
<td>Puzzle</td>
<td>Portal, Tetris</td>
</tr>
<tr>
<td>Rhythm</td>
<td>Guitar Hero, Rock Band</td>
</tr>
<tr>
<td>Shooter</td>
<td>Halo, Borderlands</td>
</tr>
</tbody>
</table>

In addition to GSR and electrocardiography data (EKG), respiration rate and behavioral variables are targets for future work. Other physiological variables as discussed by Kramer [3] are either unreliable in changing environmental conditions (blinking, pupil dilation) or are too hard to practically measure, even for the technology of today (brain metabolism).
scored over 75%. Several hypotheses were then tested regarding the physiological metrics. The hypotheses for GSR are listed below.

- H0: Difficulty and performance statistics have no effect on the mean of the physiological measure of GSR.
- H1: GSR is elevated during periods of concentration characterized by more difficult songs, with each difficulty level representing a higher GSR than the previous.
- H2: GSR is elevated during periods of lower performance, with GSR increasing as performance decreases.

By calculating the percent elevation of GSR over the baseline values, a tabulation of average GSR values for the duration of a song is compiled, with each song being rated with both an inherent difficulty (easy, medium, hard, or expert) and a performance rating (poor, good, or great). Users play in a single session a set of three easy songs, five medium songs, five hard songs, and three expert songs. The GSRs across difficulties are averaged together for a single subject (e.g., a single averaged value is determined for the three easy songs per participant).

Since the variance of the subject set is indeterminate, a Student’s t-distribution is used to determine significance in hypothesis testing. For the population mean, a value of 0 is used in comparing easy songs to the baseline (no increase). Then, the sample mean of the easy set is compared to the medium songs, the sample mean of the medium songs is compared to the hard songs, and the sample mean of the hard songs is compared to the expert songs. Additionally, a comparison is made between hard and easy songs, and between expert and both easy and medium songs. Thus, it can be determined if the increases in GSR are strictly increasing as difficulty increased (compared to each previous level). For all tests, n=5. Due to the small sample size, the assumption is made that the original baseline results follow a normal distribution, though this constraint could be relaxed as sample size is increased.

To compare across performance the entire set of all songs is taken, and the analysis compares ‘great’ results to the baseline. This sample has a size n = 27. The analysis then compared, in the manner described above, the ‘good’ results to the ‘great’, and the ‘poor’ results to the ‘good’. These sizes were n = 17 and n = 36, respectively. Due to corruption of data, the sizes for GSR data sets only are lower (at n = 27, n = 8, n = 29 for ‘great’, ‘good’, and ‘poor’ respectively).

The results were analyzed with significances levels α = .1 and α = .05. As can be seen, the analysis indicates strong evidence that GSR is helpful in indicating the stress brought on by playing any type of song. Somewhat more surprisingly though, at the 10% significance level, the analysis does not show additional support for H1 when the difficulty of songs is raised from each previous level. While it is clear that GSR increases when the game is played, there is no significant difference between GSRs of participants completing ‘easy’ songs versus ‘medium’ ones. However, significance reappears when comparing across larger jumps—that is, comparing GSR response between players differing by two or more difficulty levels is significant at the 10% level. In fact, when comparing between ‘expert’ and ‘easy’ difficulties, there is significance at the 5% level.

<table>
<thead>
<tr>
<th>Condition</th>
<th>p-value</th>
<th>α = .1</th>
<th>α = .05</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy over None</td>
<td>0.0225</td>
<td>Accept H1</td>
<td>Accept H1</td>
</tr>
<tr>
<td>Medium over Easy</td>
<td>0.1314</td>
<td>Reject H1</td>
<td>Reject H1</td>
</tr>
<tr>
<td>Hard over Easy</td>
<td>0.1047</td>
<td>Reject H1</td>
<td>Reject H1</td>
</tr>
<tr>
<td>Expert over Hard</td>
<td>0.3340</td>
<td>Reject H1</td>
<td>Reject H1</td>
</tr>
<tr>
<td>Hard over Easy</td>
<td>0.0555</td>
<td>Accept H1</td>
<td>Reject H1</td>
</tr>
<tr>
<td>Expert over Easy</td>
<td>0.0413</td>
<td>Accept H1</td>
<td>Accept H1</td>
</tr>
<tr>
<td>Expert over Medium</td>
<td>0.0721</td>
<td>Accept H1</td>
<td>Reject H1</td>
</tr>
<tr>
<td>Great over None</td>
<td>0.0069</td>
<td>Accept H2</td>
<td>Accept H2</td>
</tr>
<tr>
<td>Good over Great</td>
<td>0.0366</td>
<td>Accept H2</td>
<td>Accept H2</td>
</tr>
<tr>
<td>Poor over Good</td>
<td>0.1344</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
</tbody>
</table>

### Table 2: GSR Results for StepMania

As the performance of the player drops however, we successfully detect a corresponding increase in skin conductiv-

ity when they perform at the ‘good’ level. However, there is no significant difference between a player performing at the ‘good’ level and one performing at the ‘poor’ level. This data is useful in predicting when players have achieved a degree of stress brought on by poor performance.

The results for heart rate were calculated identically to those of skin conductance. The hypotheses and results can be shown here.

- H0: Difficulty and performance statistics have no effect on the mean of the physiological measure of HR.
- H1: HR is elevated during periods of concentration characterized by more difficult songs, with each difficulty level representing a higher HR than the previous.
- H2: HR is elevated during periods of lower performance, with HR increasing as performance decreases.

<table>
<thead>
<tr>
<th>Condition</th>
<th>p-value</th>
<th>α = .1</th>
<th>α = .05</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy over None</td>
<td>0.1459</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Medium over Easy</td>
<td>0.3260</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Hard over Easy</td>
<td>0.3359</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Expert over Hard</td>
<td>0.4313</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Hard over Easy</td>
<td>0.2504</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Expert over Easy</td>
<td>0.2276</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Expert over Medium</td>
<td>0.2997</td>
<td>Reject H2</td>
<td>Reject H2</td>
</tr>
<tr>
<td>Great over None</td>
<td>0.0004</td>
<td>Accept H2</td>
<td>Accept H2</td>
</tr>
<tr>
<td>Good over Great</td>
<td>0.0498</td>
<td>Accept H2</td>
<td>Accept H2</td>
</tr>
<tr>
<td>Poor over Good</td>
<td>0.3293</td>
<td>Reject H12</td>
<td>Reject H2</td>
</tr>
</tbody>
</table>

### Table 3: Average HR Results for StepMania

The results for heart rate are very similar to those of GSR, with a few differences. First, while heart rate is helpful in detecting frustration associated with poor performance, it is not as good an indicator of general concentration. Even at significance bounds of 10%, heart rate did not rise significantly with difficulty. However, player emotive response to performance again was detected, though the distinction between ‘good’ and ‘poor’ was still indistinguishable. These results mirror those of GSR and will add confirmation to calculations performed with only one metric.

Analysis of the respiration rate indicates a low correlation. GSR seems to be the best measure for measuring difficulty in task performance in rhythm games. The similar results
between GSR and HR informed the decision to use a naïve Bayes classifier as one possible learning algorithm offered to developers.

3.2 Amnesia Analysis

Amnesia is a survival horror game in which players are subjected to frequent surprise scares and a foreboding atmosphere while adventuring. In addition to physiological data collected during the Amnesia study, events are recorded in the internal game log. These ‘events’ are what are considered to be in-game occurrences that are of interest to the game programmer. Amnesia’s events correspond to scary elements, surprising elements, and plot related elements (such as when the player makes progress in the game).

Data analysis of Amnesia has been focused on spike responses to certain events in order to determine if a user is feeling the fear response physiologically. Unfortunately, GSR is particularly ill-suited for such a task, since skin conductance takes several seconds to produce noticeable change. It was hoped that heart-rate and breathing-rate would be more informative, but analysis seems to indicate that they are held fairly constant.

This lack of results is likely due to the immersive nature of Amnesia, to which the environment of the laboratory is not particularly suited. Further analysis should focus on aggregating player progress over an extended period. It is also likely that behavioral variables (such as when a player chooses to flee or stay in the game) would be more conducive to player analysis in a survival/horror genre.

3.3 Relevant Conclusions from Phase I Background Research

Generally, the preliminary Phase I research provided the following results, which aided the research path selected for Phase II:

- The rhythm/music genre, and specifically the game ‘StepMania’, lends itself well as a tool in affective gaming research because of its built-in data logging features and because of the fine granularity of user emotive response to specific short-duration events within the game.

- The adventure/horror genre, and specifically the game ‘Amnesia’, does not lend itself well as a tool in affective gaming research because the users emotive response is not sufficiently correlated with short-duration events (such as fear events).

- Heart Rate and Galvanic Skin Response are more reliable than Respiratory Rate in predicting an individual players game performance.

- Using a combination of regression and naïve Bayes techniques, it is possible to distinguish good and bad game performance based on measured biometric data.

- Game developers will likely wish to implement affective gameplay responses based on polarized (yes/no) outcomes to user biofeedback, and therefore naïve Bayes analysis lends itself well to implementing an affective gameplay API.

4. SYSTEM MODEL AND IMPLEMENTATION

Broadly speaking, the results from the initial study confirm the hypotheses that physiological metrics can be used to capture simple facts about player engagement and frustration levels. The proposed API is designed to accomplish the task of easily transforming developer goals regarding the emotive state of a player into a customized experience that causes a game to change and adapt in real-time. A graphical representation of a basic affective gaming model is shown below:

1. Desired Emotion?

2. Initialize game with default parameters

3. User plays game

4. Sensors gather data

5. Do sensor readings indicate desired emotion?

6a. Parameters are not changed

6b. Modify game parameters

1. The genre-specific model is selected, which contains information for translating raw data into usable variables relating to player emotions.

2. The game is started.

3. The participant plays the game.

4. The biosensors that are hooked up to the participant gather data.

5. Emotional variables are determined and are passed to the game program.

6. (a) If the emotional triggers are what the game designer wanted at this time, the game continues with no change. (b) If not, the game parameters change to hopefully induce these responses.
Using this basic affective gaming methodology, physiological responses (how the body reacts) gathered from the biometric sensors as well as behavioral responses (how the user acts) can be combined to create a set of emotional targets, or “AdaptiveFlags” that the developer wishes to monitor. Specifically, the game begins with initial parameters, and the developer is then free to poll these targets at key moments in game play. Depending on the polled value of these targets, the developer will then either modify the game parameters or not. Presumptively, if the emotional target indicates that the user is feeling very disengaged as well as very frustrated, the developer will choose to alter the game parameters such that the game is easier and thus more rewarding for the player. This cycle of adjustment allows developers to circumvent the well-known difficulty problem in gaming too difficult and the game is frustrating, too easy and it is boring. By adapting in real-time, the developer can guarantee that the game difficulty is optimized for each player. In conjunction with Gillett’s et al findings [2], the adaptive gaming API can therefore fulfill the tripartite functions of assistance, challenge, and emotion.

4.1 Phase II: Design of the Affective Gaming API

The emphasis of Phase II research was to develop an expandable, and developer-customizable affective gaming API. The end product of this research, dubbed “PlayAffect”, is a stand-alone affective gaming API which is designed as two discrete and symbiotic parts. The first part is a C++ executable that interacts directly with the hardware itself to gather physiological data from the sensors. The C++ executable program is referred to as the ‘server’. The C++ based program is not only the interface to the biosensor device, but also preprocesses the data to remove outliers. The server is currently configured to work with the FlexComp Infiniti Model SA7550, which can support up to eight different sensor types, including GSR, heart rate, and respiratory rate. The server also handles measuring a sensor baseline upon request from a ‘client’ program. At a default rate of 5 Hz, a data packet is sent through a pipeline to the client program, containing the current value measured by each connected device, the identity of each device, and a timestamp for latency corrections.

The second part of the API is a C# based program referred to as the client. The client runs locally within the Unity engine. Upon receipt of the raw data, the client parses the data through one of several algorithms as selected by the developer. The current algorithms include a simple regression analysis which measures an increase over the baseline values for a particular sensor, as well as a naïve Bayes classifier. The classifier was included since it was concluded from Phase I research that developers are more likely to desire simple binary answers (e.g., is the user frustrated or not?), and that these answers were likely to be answerable by a combination of biological factors, including GSR and HR. The client program is designed to be expandable in this regard. Further algorithms are includable and definable by the developer. A model of this architecture follows:

The decision to separate the processing into a server and client side supports multiple design goals. First, this two-sided division allows the preprocessing to be done in parallel to the C# data analysis, which not only improve performance, but also allows the data collection and preprocessing to be done on a location remote to the machine actually running the game software. Indeed, support for distributed systems is one improvement that future game releases are likely to support. Secondly, the server-client division allows a set of C++ server libraries to be compiled per the developer’s particular hardware needs. Though the system used herein used the FlexComp Infiniti Model SA7550, further expansions will almost certainly exploit different hardware. As such, the only modifications required for future hardware expansions are to the server-side architecture, thus allowing the bulk of the C# library to remain untouched yet still functional as expansions occur.

5. RESULTS AND PERFORMANCE

The effectiveness of the Phase II affective gaming API itself is measured from a game developers perspective. The real measure of any software extension or library is its popularity among end-user developers, and since this API has yet to be deployed, such figures would be mere projections. However, the Unity-side software was designed with several core aspects in mind; aspects that it shares with other popular Unity plugins. These aspects include:

- Usability—PlayAffect is designed to be bundled with a set of suggested genre-specific AdaptiveContexts, which contain prepackaged behavioral programming suggestions as well as pre-calibrated physiological sensor packages.
- Extensibility—PlayAffect allows for the easy creation and easy modification of AdaptiveContexts, allowing developers to better customize the system to their needs,
and allowing researchers to add more genre-specific packages as needed.

- Compatibility—PlayAffect was designed to supplement existing scripting practices and to compile existing items (such as trigger togglers) into a more organized and understandable format. Further, the separated server/client architecture allows for hardware to be switched out for different models with ease.

It should be noted that the API was initially coded for the Unity 3 game engine, but has been recently upgraded to Unity 4 to support namespaces and a more robust set of .NET 2.0 features. The Bayes classifier uses a custom subset of the .NET 2.0 System.Data libraries to allow for data table analysis.

A copy of the source code of PlayAffect is available upon request from the researchers.5

6. ETHICAL CONSIDERATIONS

In designing PlayAffect, ethical considerations come into play in two areas. Firstly, in gathering the baseline and genre-specific data from participants in Phase I, there are privacy concerns for the individual game participants. Post-deployment, ethical considerations continue as the exposure of the raw data to the developers allows them to potentially collect personal data about a player’s physiology and then compromise the player’s privacy. During the execution of Phase I, participants were given a unique identifier to link them to their data, but all specific identification information was stripped. Demographic information (such as age and gender) was collected, but was again connected only to the anonymous identifier. These steps, in addition to the signed statement participants submitted, allowed the study to proceed within the boundaries set by the University of Pennsylvania Institutional Review Board.

There is, however, no such guarantee for future data collection by independent developers. Though most data that is currently collected about users (including demographic and behavioral data) is often sanitized and held within secured databases, no official protocols exist for the further management of said data. Users are given only marginal notice through hastily accepted End-User License Agreements (EULAs) that such data is even collected. It is hoped that as physiological sensors become more prevalent within the industry, the same care is taken with the new types of data that is then collected. Such data is invaluable to both researchers and industry experts in aggregate, but attention must be given to the possibility of it being linked to specific users.

Even if data collection is held to the highest standards of privacy control however, there are yet further dangers with overreliance on such privacy measures. Many physiological metrics are at least semi-voluntary, and can be easily altered by the game players themselves. For example, simple steps performed by a game player, such as wiping ones hands, or force-breathing, could greatly skew the biological results interpreted by the API and, thus could greatly affect the interpretation of the player’s actual physiological state. Within the context of a game, the most like result of data manipulation by players is the possibility of cheating within the game, or the possibility of a non-optimal affective gaming outcome. Should the technology spread outside the entertainment realm, however, attention should be given to the validity of the incoming data, as well as the possibility of external attacks on databases. In cases where physiological data is used for identification purposes, black hat data mining techniques are a real danger. PlayAffect was designed to maximize programmer flexibility by exposing raw physiological data to programmers that desire it, but this flexibility will allow for application of the API outside its intended use.

7. CONCLUSIONS AND FUTURE WORK

The goals of the Phase II research, and the development of PlayAffect as a feasible and flexible affective gaming API, were simultaneously to create an API that could be widely distributed among game developers and also to endow the API with a set of experimentally informed models that could be used to jumpstart the development process. To that end, the Phase I study confirms initial hypotheses that GSR and HR both increase during periods of increased gaming concentration and engagement, as well as during periods of higher frustration as indicated by decreased game performance. Though data was also collected regarding respiration rate, due to the semi-voluntary nature of the physical process and the limited data set of subjects, limited correlation to respiration rate could be found. It is anticipated that future work will explore this and other as-of-yet unexplored physiological measures such as neural activity, iris dilation, or facial tics.

The specific design parameters for the PlayAffect library called to attention several end-user considerations, namely usability, extensibility, and compatibility. In addition to providing developers with an initial set of design recommendations for each game genre tested, the current PlayAffect version release also allows for considerable customization. By allowing developers to specify their own physiological and behavioral variables to monitor, it is hoped that further optimization and suggestions may be compiled into the PlayAffect library. Further, modifications are not limited to variable recommendations, but may also include more advanced machine learning algorithms beyond the simple regression and naïve Bayes inclusions here. Additionally, due to the separated client/server architecture, it is assured that changing hardware specifications will only require modifications to one end of the data pipeline, allowing the bulk of the PlayAffect library to remain untouched. Finally, the learning curve for the software is specifically designed to be gentle, allowing those already familiar with Unity-specific in-game triggering systems to quickly adapt to the PlayAffect interface. The software provides both demonstration files as well as an extensible interface to facilitate this transition for the developer.

It is the purpose of this software to be fully extensible, and thus, continuously refinaable. Although PlayAffect now offers a range of solutions for common developer problems (as well as those that are expected to become common with the proliferation of hardware sensors in gaming), optimizations and additions are welcomed by the researchers. Future work should consider these algorithmic improvements of equal importance to the gathering of more genre-specific data vis-à-vis Phase I. Further data should also include support for a more diverse range of sensors. Continued research into the underlying physiological features of the two emotive responses of engagement and frustration should strive

5Contact Johnathan Mell at jmell@seas.upenn.edu
to discern the two emotions more reliably, as both are positively correlated with heart rate and GSR. At the current development stage, it is anticipated that the current version of PlayAffect answers many design problems for game developers while also integrating new ideas of physiological integration into the arena of adaptive, affective computing.

8. REFERENCES


APPENDIX

A. QUESTIONNAIRE

<table>
<thead>
<tr>
<th>Questions</th>
<th>Possible Answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is your lab ID?</td>
<td>(Provided by testers)</td>
</tr>
<tr>
<td>Gender?*</td>
<td>Male or Female</td>
</tr>
<tr>
<td>Age?</td>
<td>Integer Answer</td>
</tr>
<tr>
<td>How would you rank your video game playing frequency?*</td>
<td>Very Often (weekly or more), Somewhat Often (monthly), Rarely (occasionally), Never</td>
</tr>
<tr>
<td>How would you rank your video game playing proficiency?**</td>
<td>Advanced, Average, Novice, I don’t play video games.</td>
</tr>
<tr>
<td>What gaming devices do you use to play video games?**</td>
<td>Xbox/Xbox 360, Playstation 1/2/3, Nintendo Wii/U, Nintendo DS/3DS, PC, XBox Kinect, PSP/Vita, iPhone/Android</td>
</tr>
<tr>
<td>What gaming genres do you regularly play?***</td>
<td>Adventure, Educational, Fighting, Puzzle, Racing, Real-Time Strategy, Rhythm/Music, Role-Playing, Shooter, Sports, Survival Horror, Turn-Based Strategy</td>
</tr>
</tbody>
</table>

*Only One Answer

**Multiple Answers Acceptable

Table 4: Question List from Questionnaire