A Toolkit for Exploring Affective Interface Adaptation in Videogames

Kiel Mark Gilleade, Jen Allanson

Computing Department
Lancaster University
Lancaster
LA1 4YR, UK
{gilleade, allanson}@comp.lancs.ac.uk

Abstract

From its humble beginnings back in the early 1960's the videogame has become one of the most successful form of HCI to date. However if we look more closely at the interactions between the game and gamer it becomes evident little has changed since the advent of *SpaceWar*[1] back in 1961. These interactions are for the most part static and thus predictable, given a particular set of circumstances a game will always react in one particular manner despite anything the player may actually do. Because of this the expected lifespan of a videogame is inherently dependant on the choices the videogame provides; once all possible avenues have been explored the game loses its appeal. In this paper we focus on adapting techniques used in the field of Affective Computing to solve this stagnation in the videogames market. We describe the development of a software development kit (SDK) that allows the interactions between man and machine to become dynamic entities during play by means of monitoring the player's physiological condition.

1 Introduction

After the release of *Pong*[1] by Atari back in 1972 the videogame established itself as one of the most popular forms of HCI. In spite of this, research into this particular field of HCI has been relatively slow and only recently has any academic entity attempted to analyse the interactions between the game and gamer. Such examples include AffQuake[2], a modified version of id Software's *Quake II*[3] that used the player's level of arousal to control the games avatar and a variety of affective videogames used to treat neurological problems[4].

However the projects undertaken so far have a very limited area of applicability. The problem inherit in the majority of videogames is that the interactions between the game and gamer are always fixed, for every action the game/player initiates there is always a predetermined reaction by the player/game. Because of this it's relatively easy to predict what will happen given an action initiated by the player or the videogame, which becomes even easier after the player finishes the game. Once all the possible avenues that are open to the player have been explored the game loses its appeal because they're unable to change.

1.1 Interactions as a Dynamic Entity

In order to rectify this problem we believe a videogame must be capable of dynamically changing the ways in which it reacts to the player's presence in the gaming environment. One approach to achieving this is by using the psychological status of the player as a point of reference. Therefore when the player's state of mind changes so do the interactions between the game and the gamer, thus preventing the player from accurately predicting what a given action or reaction will bring about in the videogame. Our premise for the work described here is that by using the player's psychological status, as governed by their level of arousal, to control how the gaming environment reacts to the player's presence, the videogame can become affective. This should in turn enhance the game's appeal and potentially increase its overall life span.

2 Developing an Affective Videogame

To put this theory to the test we decided to carry out some preliminary investigations with a view to creating a SDK that could be used by games developers to create affective videogames.

2.1 Suitable Genres

Phase 1 of this project involved the study of the relationships governing the interactions between game and gamer in order to decide which genre of videogame would be suitable for affective interface adaptation. We investigated several of the most common[1] genres of videogame including action, adventure, analytical (puzzle and strategy), role playing games (RPG), simulation and sports and concluded that only those genres which already required a high level of involvement on the player's part would be suitable for this project.

The reasoning behind this is that when developing an affective application of this type we are trying to enhance the interactions that already exist between man and machine, we cannot overtly change the pace of these interactions otherwise we change the nature of the system the user interacts with. And only those genres that require a high level of interaction (player is inundated with objects of interest) are going to be able to invoke a psychological response (whether positive or negative) that we can use to manipulate the gaming environment with.

Suitable genres for incorporation include both the action and adventure genres. Due to their reliance on bombarding the player with a wealth of events which require the player's full and undivided attention in order to prevent failure e.g. a wave of ravaging monsters. The sport genre was also incorporated because the majority of real world sports are highly physical, which is a quality that is inherited by their video game counterparts and as such they operate much like action games. Only the puzzle side of the analytical genre was chosen for incorporation because in the majority of cases with games such as *Tetris*[5] the player needs to respond in rapid succession in order to win. Strategy games like *StarCraft*[6] are often long winded and actions whether initiated by the computer or the player can sometimes take several hours to execute e.g. building a suitable sized force to storm the enemies' base. Because of this the player's level of arousal is unlikely to change by any significant value if at all for the duration of play. The same goes for RPG and simulation games, the level of interaction is often so slow an emotional response is unlikely to occur. Hence such genres are not suitable for incorporation into our programming platform.

[1] Hybrid genres were not analysed e.g. survival-horror.

Using this information we decided to build an affective programming platform for the following videogame genres, action, adventure, puzzle, and sports.

2.2 *Manipulating the Gaming Environment*

Phase 2 of this project involved the identification and isolation of the chosen genres' mutable gaming components. Each component must be capable of being manipulated within the context of the videogame so not to make the player overtly aware any change has been made when one is instrumented. For example a game cannot simply change the stage the player is interacting with because of the way they reacted, as that would spoil the games continuity. However the games difficulty settings could be changed because the resultant effect would blend more transparently within the context of the videogame. An investigation into the chosen genres revealed the following exploitable gaming components to be:

- *Difficulty:* controls the games difficulty setting.
- *Story:* controls the strength of the drama involved in the story line.
- *Music:* controls the music tempo and style.

With these components identified we could begin to develop counter responses to the changes in the player's emotional state.

2.3 *Responding to Change*

With the identification of the components mentioned above we could move onto phase 3, deciding how we were going to respond to the changes in the players psychological condition during play.

To begin with we needed to decide on how the player's psychological status would be assessed. Videogame developers are often forced to push the hardware their games are run on to the very limit in order to stay competitive. Therefore a complex analysis of the players bio signs in order to assess there psychological condition is not a possibility. Consequently we decided to opt for a simple class based psychological assessment. A physiological aspect (heartbeat rate) of the player is grouped into several classes; where each class represents a different psychological state the player is in. The supported states were as follows, bored (negative state, decreased heart rate), tired, content, excited and ecstatic (positive state, increased heart rate). When the player enters a given class a new set of interactions (unique to that class) will take place in order to prevent the game from losing its appeal. For each class present we allocated a selection of environment changes the game would be able to implement based upon the components it is allowed to manipulate.

With that aside we had to formulate how the gaming experience was going to be enhanced given the player's current psychological state and by how much. In order to do this we needed to take into account what we wanted the game to do given a positive or negative reaction by the player. When the player responds negatively (decreased heart rate) it is reasonable to assume that the player is dissatisfied with the current state of play. Play would therefore need to be invigorated in order to encourage the gamer to play a more active role in the game. This can be achieved by increasing the game's difficulty and overall tempo, as the resultant effect would require the player to interact more with the game in order to remain in play. When the player is responding positively (increased heart rate) to the game we need to retain the current level of interaction to appease the player but we must not over stimulate the player otherwise they may burn out. Therefore during this period we need to increase the level of variety in the gaming environment and slow the overall tempo of the game down using the components available. This can be achieved by changing the games music style, enemy variation and AI capabilities. The scale of the environment change would depend on the degree the player is feeling in either direction e.g. if they're only slightly bored with the current state of play the game tempo is only slightly increased.

3 Implementation and Results

Based upon this analysis we created a software development kit (SDK) for prototyping affective videogames, which we call the Intelligent[2] Gaming System (IGS)[7]. An electrocardiograph (ECG) is employed to assess a player's psychological status. ECG data can be streamed in real time to a prototype videogame developed using the SDK.

In our experimental set up we provided 31 pre-determined game responses to a range of psychological states we envisaged a player might experience during game play for an action-based videogame. Our prototype videogame required the elimination of multiple targets within a set time frame; failure to destroy a target before it escaped the player's field of view would result in the player's health being depleted and eventually death.

The initial evaluation was undertaken by a group of 8 people (2 female, 6 male) aged between 21 and 38. Of the 8, 6 frequently played videogames and 5 of these indicated that they preferred playing action games to any other game genre. For each player a resting baseline ECG measurement was taken. From the baseline we calculated the boundary conditions for each psychological state (bored, tired, etc). This data was then used by the IGS-enabled affective version of the same action game. Within these boundaries streaming ECG data from the player caused the game play to change.

Rather than carrying out an analysis of arousal statistics we wanted to focus on the *subjective* experience. Consequently, for this preliminary study we provided a questionnaire in order to ascertain whether subjects preferred the affective version of the game to the control version (non-affective). All 8 volunteers indicated that they did. In addition the 6 serious gamers indicated that they felt the games life span would be improved by affective means. Of the remaining 2 subjects one was unsure and the other believed it wouldn't affect the games life span directly.

4 Conclusions and the Future

Through the pilot study described we have demonstrated the principle of affective gaming. We have chosen to do this by providing a platform for the prototyping of ECG-responsive videogames. The responses of our small group of volunteers to an affective adventure game indicate the potentially compelling affect of such adaptive technologies. In addition the gamers among our group agreed with our assertion that this technology has the potential to extend the life span of the game.

The IGS programming platform is still at a developmental stage and we can expect it to be some time yet before the videogame industry is likely to take notice of the benefits of affective videogames. This paper has shown that although there is evidence suggesting affective interface

[2] Machine intelligence can be characterised as the ability to react in light of ongoing events.

adaptation can lead to an improvement in the gaming experience, further exploration and analysis is required. We hope to carry out further explorations ourselves in the future, and have made the IGS SDK available online in the hope that others will be compelled to use it for their own experiments.

It may be possible to provide a programming platform that allows for responses more in line with the games context, and thus increasing the effect affective interface adaptation would have on the overall gaming experience. For example while playing the psychological horror game *Silent Hill 2*[8], the system could monitor the player's psychological condition and adjust the 'scary' factor according to how the player reacts to the gaming environment created. We could even go as far as recording the player's psychological reactions during key events in the game in order to adjust the next event to something that would have a much greater impact. Such use in hybrid genres such as survival horror could create immensely entertaining environments as the game and gamer are brought closer together.

Although the research described here is relatively limited, its implications can be very far reaching within the domain of videogame research, and the theory behind it can be applied to many other aspects of HCI. By studying how man and machine interact we can identify the relationships that govern those interactions and thus manipulate to make products much easier to use, more fun to interact with or anything else we wish within the context of the devices application. Therefore we feel that the lesson learned from this exploration of affective interface adaptation will be of interest to designers and developers of compelling interactive interfaces in other application domains.

References

1. Herman, L., Horwitz, J., Kent, S. & Miller, S. (2001). History of Videogames. Retrieved January 2003 from http://gamespot.com/gamespot/features/video/hov/

2. AffQuake. Retrieved March 2002 from http://affect.media.mit.edu/AC_listings.html

3. Quake II. Retrieved August 2002 from http://www.planetquake.com/quake2/

4. NASA. (2000). Video Games May Lead to Better Health Through New NASA Technology. Retrieved May 2002 from http://www.gooddealgames.com/articles/NASA_and_Games.html

5. Tetris. Retrieved January 2003 from http://www.tetris.com

6. Starcraft. Retrieved January 2003 from http://www.blizzard.com/starcraft/

7. Gilleade, K. (2002). IGS Programming Platform. Retrieved October 2002 from http://www.comp.lancs.ac.uk/computing/users/gilleade/

8. Silent Hill 2. Retrieved December 2002 from http://www.silenthill2.de