Towards Tailoring the Emotional Experience in a Physical Wii Game through Artificial Neural Networks based on Physiology and Gesture Data

IT University of Copenhagen
Media Technology and Games, Technology
August 3rd, 2009

Dajana Dimovska
dajana@itu.dk

Sebbe Selvig
sebbe@itu.dk

Patrick Jarnfelt
pjarnfelt@itu.dk

Supervisor: Georgios N. Yannakakis – yannakakis@itu.dk

ABSTRACT

In this study an artificial neural network model of players' emotion preferences while playing a physical Wii game was constructed. Ultimately the purpose of the emotion model is to recognize players' emotions and entertainment while playing a Wii game. Further on, it was investigated and proposed that the model could be used to tailor the player experience with a two-fold purpose for the player: 1) to increase player satisfaction in real-time, and 2) to involve the player in selected emotional experiences. The proposed technology can be used to assist game designers to automate parts of the game design, balance the gameplay, and create Wii games with adaptive emotion experiences. The model is trained on data derived from the player-Wii interaction, which included physiological response (Blood Volume Pulse, Heart Rate, and Skin Conductance), Wii Remote gesture and game data. In this study the developed emotion model proved to achieve the highest classification accuracy of 83.97%, 87.58%, 82.96%, 77.02%, and 86.11% for the emotions relaxation, boredom, excitement, frustration, and fun. However, to use the model for tailoring the player experience the model had to be trained with controllable game factors, decreasing the performance to 78.42%, 73.29%, 75.18%, 74.24%, and 82.83% for the emotions relaxation, boredom, excitement, frustration, and fun respectively. Furthermore, the restriction of input data to Wii Remote specific features –which yielded 81.41%, 72.12%, 71.85%, and 71.97% for the emotions relaxation, boredom, excitement, and frustration respectively – and the usability of real-time application of the model are discussed.

CONTENTS

1 INTRODUCTION

The industry standard for making games is a team of game designers, artists and level designers, amongst others, who realize their vision and creativity into a game. Today people, regardless of their age, play games mostly for the experience and the emotions that the games create (Lazzaro, 2004). Therefore, game developers strive to create games with gameplay or stories that evoke various emotions. For example, the Japanese sci-fi game series *Final Fantasy* (Square Enix, 2009) uses a linear storytelling like in films to evoke emotions. *The Sims* (Electronics Arts, 2009) on the other hand is a game, where the players control the life of their character and insert their own emotional story in the game. However, there are different player types (Bateman, et al., 2006) with various playing styles that might experience the game in another way than the game developers had envisioned. This is not necessarily a problem, since the games are built of many factors that the player can find interesting no matter what.

The interest among game technology studies has increased for creating adaptive emotional experiences for the players through intuitive models that can automate the game design and the balancing of the game features. Having such models will assist the game designers, and give the possibility for exploring new game design conventions. Game studies have already tried to build these types of models for example in (Yannakakis, et al., 2007c)and (Gilleade, et al., 2004) models that can adapt based on the player's emotion, are proposed.

Affective computing is the study that investigates various techniques and approaches that can be used for recognizing emotions with computer. This study suggests techniques for developing an emotion model that can recognize the player's emotional state or their perceived level of entertainment by using the player's physiology and body movement responses. Further on, the emotion model can be used to tailor the player experience in video games towards a selected emotional experience.

Video games today are experiencing many novelties in particular within the Human Computer Interaction (HCI) area. The input devices like the mouse, keyboard, and game-pad used to be the standard HCI interfaces used in games, but recently this have been changing gradually because new gaming platforms and consoles with novel interfaces have been introduced. For example, the *iPhone* (Apple, 2007) is a multimedia smartphone that instead of a keyboard has a multi-touch screen that the user can use to, but not limited to, play games on. Another example is the *Nintendo Wii* game console (Nintendo, 2006a) which instead of a game-pad, uses Wii Remote that bases the game interaction primarily on motion sensing, gesture recognition, and pointing features. Therefore, the games developed for the *Nintendo Wii* that are played with the Wii Remote involve the player's hands and body in physical activity. This makes the *Wii* an interesting platform for developing adaptive games that use the player's physiology and body movement responses as indication for when and what to change in the game.

The goal of this project is to develop a computational affect recognition model that can recognize the player's emotions: frustration, excitement, boredom, relaxation, and her level of "fun", by using the player's physiological signals and Wii Remote acceleration data, and based on the model's outcome, tailor the gameplay towards a selected emotional state, normally defined by a game designer. Having this technology gives the possibility to use emotions as game mechanics and to create physical games suited for physical therapy that have automated gameplay that decreases and increases the players' activity.

1.1 Motivation

Artificial intelligence (AI) is extensively used in almost all computer games. One of the first video games that implemented AI to control enemies was *Qwak* (Woodhouse, 1989), which was a duck hunting game and *Pursuit* (Atari, 1975), a combat aircraft simulator revolving around dogfights (Wolf, 2007). The introduction of games that could be played by a single player – as compared to the first games like *Spacewar* and *Pong* – required balancing and tweaking the games' difficulty level (Wolf, 2007). As a result, many games to date give a choice to the players to adjust the difficulty by choosing between game modes, which usually range from easy to hard. This idea of user adjusted difficulty was replaced in some games (Max Payne, 2001) by implementing auto-adjusting features instead that based on the player's performance would adapt different factors in the game. One way to do this is to base the game's adaptive system on the low level features of the player e.g. their score. This could, for instance be done by comparing the player's score to the AI opponent's score – if the player is leading, the game could be made harder, and vice versa if she is falling behind.

Recent research in affective computing and games has suggested the use of affective or entertainment models that utilize physiology input from the player to adjust the game accordingly (Yannakakis, et al., 2007c), (Gilleade, et al., 2004), and (Tijs, et al., 2008). This approach to adaptive games has a commercial potential today, one example is Nintendo's announcement of their development of the Wii Vitality Sensor extension for the Wii-platform at E3 2009 (Nintendo, 2009), that measures the BVP through the thumb. The Vitality Sensor is currently in development and is not yet available for the public.

Numerous attempts have been made for creating games based on player's physiological signals. For example, few commercial games have been released by *WildDivine* (WildDivine, 2001) – a company that focuses on developing anti-stress and meditation training games – which use physiology and are not related to action games, as seen in the rest of the game market. *WildDivine* uses the player's physiological signals to investigate her physiological state and develop games that base the gameplay on it. Nintendo released a game based on physiology, called Tetris 64 (aka. Bio Tetris, released in Japan 1998 for the Nintendo 64 console (SETA, 1998)), which was designed to adapt the speed (challenge) of the game according to the player's Hear Rate (HR) calculated from the Blood Volume Pulse (BVP) from a sensor attached to the lobule on the ear.

AIIDE '07 (AIIDE, 2007) organized workshops that investigated emotions in games and showcased several related games. At the conference, a Pacman clone (Cowley, et al., 2007) was shown that tried to classify what type of a player was playing the game, based on what decisions were made during gameplay. While this setup attempted to classify the players, it could not tell if the players were having fun or not. A solution to this problem has been proposed and developed by Yannakakis et al. (Yannakakis, et al., 2007c), who use the player's HR to model the player's feeling of fun while playing a physical active game. The games used in their study were *Bug Smasher* and *Space Invaders*, created on interactive tiles described in *Playware technology for physically activating play* (Lund, et al., 2005). The model was created using a machine learning approach to approximate the player's feeling of fun and is proved to be successful in predicting the level of fun with 76% accuracy.

With these technologies at hand, the next step is to utilise them in real-time prediction of the player's emotional state and adjust the game accordingly. This study proposes a method for achieving this goal.

1.2 Short Overview of the Approach

The goal of this study is to develop a model that can recognize players' emotions while playing a physical Wii game, and ultimately tailor the gameplay with a two-fold purpose: 1) have the players express a will to play the game more; not in the sense of addiction but because the game simply is fun and entertaining to play, and 2) to involve them in a selected emotional experience while playing.

The purpose is to combine research in various emotion related areas, AI, and user study experiments to develop an emotion model that can estimate the player's emotions in a physical Wii game. The approach in this study performs the following five phases:

1. Theory phase – investigation of various research areas is done:
    a. What emotions are and how to measure them in games
    b. What machine learning approaches have been used to create entertainment or emotion models for games
    c. Which game factors contribute to the player experience in games
2. Method phase – based on the research from the theory phase Methodology is formed that states all the decisions and details about the design and implementation of the experiment performed in this study
3. Design and Implementation phase – the following parts of the experiment are designed and implemented:
    1. Wii physical game
    2. User study experiment that would collect player, game, Wii Remote, and physiology data, needed for implementing the emotion model
    3. Emotion model
4. Results phase – results and evaluation of the emotion model performance and potential for tailoring the player experience
5. Conclusion phase – conclusion and future work is discussed

2 THEORY

In this part of the thesis, all relevant theory and related work will be presented. The areas covered are, firstly, theory about emotion and how to identify them. Secondly, research about how physiology and body movement in games relates to emotions is discussed. Lastly, theory about gameplay factors that influence player experience and theory of Artificial Neural Networks (ANN) are presented.

2.1 Identifying Emotions

In this section, we define the psychological and terminological meaning of the words “emotion” and “affect” and the difference between emotions, feelings and mood.
Many different disciplines study the concept of emotions and their role in different processes. The main focus in this section is on the psychology study of emotions, which examines emotions as mental processes and behaviour, and explores the fundamental physiological and neurological processes that the emotions trigger. The purpose of the study is to understand what is happening to us when we are in an emotional state.

2.1.1 Defining Emotions

There are numerous proposed emotion theories that explain what emotions are and how they arise (Mandler, 2001) and (Keltner, et al., 2000). The pioneer work in this area was done by Ancient Greek philosophers, as Plato and Aristotle and continued with famous philosophers from the renaissance, as René Descartes, Baruch Spinoza and others. In the end of the 19th century the psychologists W. James in (James, 1884) and C. Lange in (Lange, 1887) proposed similar emotion theories, which are some of the first scientific emotion theories, supported by empirical research (Mandler, 2001). Ever since the study of emotions has expanded, it has occupied many scientists from different fields in psychology, physiology, and the alike. As a consequence many theories today examine emotions from different perspectives.
Some of the famous emotion theories are: a) James-Lange theory, b) Cannon-Bard theory, c) Schachter-Singer theory, and d) Vascular theory. J. Wilson in (Wilson, 2003) gives good overview of the four mentioned theories.

Based on observiation of the many emotion theories Picard in (Picard, 1997) proposes that emotions can be examined in terms of: a) cognitive (mental component) and b) physical (bodily component). The cognitive component is the mental appraisal of a situation that leads to a certain emotional experience. It focuses on understanding the situations that elicit emotional episodes, for example: “I have a date tonight; therefore, I am excited.” The physical component is the physiological response that co-occurs with an emotion, for example that could be changes in the facial expressions, body movement, HR, temperature, respiration , etc.

The main argument that separates the different emotion theories is the question of whether or not the cognitive processes are involved in the emotion definition. For example, Aristotle has very cognitive definition about emotions, his argument is that the thought comes first and then the feeling comes next.

Aristotle's definition:

"Emotions are the things on account of which the ones altered differ with respect to their judgments, and are accompanied by pleasure and pain: such are anger, pity, fear, and all similar emotions and their contraries". (Leighton, 1982)

Other researchers like W. James (James, 1884) and Lange (Lange, 1887) define emotions as consequential outcome from the perception and interpretation of physical changes in the body. This means when the organism counters certain stimuli, first it responds and then it experiences emotions. For example W. James argues:

"*We see a bear and run; then our fear is a perception of our bodies as we run*." (James, 1884)

While some argue that the physical and cognitive emotion components function independently much of the time, Scherer (Scherer, 2005) argues that the special nature of emotion consists of the coordination and synchronization of both components during an emotion episode, driven by appraisal.
Scherer's definition of emotions is the following:

"*Emotion is defined as an episode of interrelated, synchronized changes in the states of all or most of the five organismic subsystems in response to the evaluation of an external or internal stimulus event as relevant to major concerns of the organism.*" (Scherer, 2005)

In this definition, Scherer argues that an emotional episode can be elicited by stimulus events, such as natural phenomena, behaviour of other people or animals that have significance to the subject or the subject's own behaviour. He also argues that during an emotional episode the subject experiences changes in both the cognitive and the physical components. Scherer defines the cognitive and physical emotion components as the respective states of the five organismic subsystems: 1) information processing, 2) support, 3) executive, 4) action and 5) monitor subsystem. To better understand the relationships between the five organismic subsystems and the components of emotion see Table 1.

Table 1. Relationship between organismic subsystems and the functions and components of emotion (Scherer, 2005)

Emotion Function	Organismic Subsystems and Major Sabstrata	Emotion Component
Evaluation of objects and events	Information processing (CNS)	Cognitive component (appraisal)
System regulation	Support (CNS, ANS, NES)	Neurophysiological component (bodily symptoms)
Preparation and direction action	Executive (CNS)	Motivational component (action tendencies)
Communication of reaction and behavioural intention	Action (SNS)	Motor expression component (facial and vocal expression)
Monitoring of internal state and organism – environment interaction	Monitor (CNS)	Subjective feeling component (emotional experience)

Note: CNS = central nervous system, NES = neuro-endocrine system, ANS = autonomic nervous system, SNS = somatic nervous system

In general this study uses Scherer's observations and definition of emotions. Conversely, the word "affect" or "affective" refers to the experience of feeling or emotion. The adjectives "affective" and "emotional" describe both physical and cognitive components of emotion.

2.1.2 Difference between Emotions, Mood, and Preferences

Based on Scherer's observation in (Scherer, 2005), the difference between emotions and other affective phenomena mainly lies in their properties; arousal, response, rapidity of change, intensity, duration, and behavioural impact. In this section, the properties for emotions are defined and then compared to the properties of preferences and moods like in Scherer's study.

Emotions, in general, are elicited by stimulus events that are major concerns and relevant to the organism. For example, we do not usually get emotional about people we do not know or situations that do not concern us. The appraisal of a stimulus arouses appropriate emotions that are followed by appropriate cognitive and physiological responses, which makes it possible to measure emotions empirically.
Emotions and their emotional response patterns are likely to change rapidly. This is due to their dependence on stimulus events and their appraisal, which often changes rapidly because of new information or re-evaluations. Another emotion property is behavioural impact, which means that emotions generate behaviour and physical expressions, often interrupting ongoing behaviour sequences and generating new goals and plans. For example, you are at home relaxing and watching TV and you receive a phone call with bad news about your friend; immediately you will stop doing what you did, your body language will change and you will start thinking about how to help your friend. Based on the fact that emotions have high behavioural impact and can rapidly change the organism's state, emotions are characterized to have a relatively high intensity. Conversely, the duration of the emotions is relatively short since they arouse massive and rapid response, which wastes great amount of resources in the organism.

Preferences are relatively stable evaluation in form of liking and disliking or preferring one over other object or stimuli. Compared to emotions, preferences are very low influenced by current events, needs, or goals and have lower intensity and lower duration. Another interesting difference is that preferences generate unspecific positive and negative feelings with low behavioural impact, which are independent of the organism's current goals and needs and are based on genetic or learned preferences.

Mood is a relatively long lasting diffuse affective state that affects the experience and behaviour of a person. Moods differ from emotions, because they are less likely to be aroused by a particular stimulus or event, but very often they can be triggered by a non-obvious cause that is linked to a certain event. Other differences are that moods have a lower intensity, but a longer duration and they generate a very low response. However, like emotions, moods can have high behavioural impact on the organism.

Table 2 gives a visual representation of the comparison done in this section.

Table 2. Differences between emotions, preferences and moods (Scherer, 2005)

Design features **Type of affect**	**Event focus**	**Transactional appraisal**	**Rapidity of change**	**Behavioral impact**	**Intensity**	**Duration**
Emotions	VH	VH	VH	VH	H	L
Preferences	VL	M	VL	M	L	M
Moods	L	L	M	H	M	H

Note: VL = very low, L = Low, M = medium, H = high, VH = very high

2.2 Measuring Emotions

In recent years, the interest in emotion research has grown rapidly and has advanced into measuring emotions (Rosenberg, et al., 2000). Emotions are complex processes and arouse numerous changes and responses among behavioural, physiological, and subjective systems of the body; therefore, it is difficult to measure them precisely. Scherer argues that in an ideal world of science, one should measure: 1) the continuous changes in the appraisal processes in the central nervous system (CNS)[1] , 2) the response pattern generated in the neuroendocrine[2], the autonomic (ANS)[3], and the somatic nervous system (SNS)[4], 3) the motivational and behaviour changes produced by the appraisal results, 4) the patterns of facial and vocal expressions and the body movements, and 5) the nature of the subjectively experienced feelings (Scherer, 2005). Such an experiment has never been performed because of its complexity. In recent years there have been many observations that suggest there is no standard rule of which response should be tracked in order to measure emotions, although there are some methodology steps that need to be considered (Rosenberg, et al., 2000).

---

[1] The central nervous system (CNS) coordinates the activity of all body parts in the organism. The brain and the spinal cord compose the CNS. (Wilson, 2003)

[2] The neuroendocrine system is the interaction between the nervous system and the hormones of the endocrine glands. (Wilson, 2003)

[3] The autonomic nervous system (ANS) controls internal organs functions below the level of consciousness. The ANS affects heart rate, digestion, respiration rate, salivation, perspiration, diameter of the pupils, urination, and sexual arousal. (Wilson, 2003)

[4] The somatic nervous system (SNS) is the voluntary control of body movements, which helps keep the body in touch with its surroundings (e.g., touch, hearing, and sight). (Wilson, 2003)

1. The first step in measuring emotions is to elicit them first. This can be done with different techniques. For example, anything in our daily life can elicit emotions, like interpersonal interaction, emotional stories or statements, films, games and other media or situations. Some of these techniques elicit emotions easier than other, which can depend on many factors and on the individual sensitivity.
2. The second step is to choose measures and use them to infer the elicited emotions. Since there are individual differences in the response system sensitivity for each person, it is recommended to use multiple measures from which some could be used for verification and some as dependent variables. The most widely studied methods for measuring emotional response are (Rosenberg, et al., 2000):
   - Subjective experience (inferring emotions from subjective self reports – asking people what they feel)
   - Physiological measures (inferring emotions from physiological responses - the activity in the ANS, the CNS, and the endocrine system)
   - Behavioural/expressive measures (inferring emotions from vocal, facial expressions, and body movement)

These emotion measuring methods have a common problem when recognizing emotions. As mentioned, this is mainly because different individuals may express the same type of emotions differently. The patterns in expression can be "person-dependent" and vary in many ways. This can be due to many factors, such as temperament, personality, gender, context, social and cultural expectations, and other factors (Picard, 1997). For example, when nervous, some people have sweaty feet and others sweaty hands.

In the following section the three measures stated and the problems when recognizing emotions are discussed in details.

2.2.1 Inferring Emotions from Subjective Self Reports

As mentioned in *Defining Emotions* (section 2.1.1), emotions have subjective cognitive representation that reflects the experience of the mental and bodily changes. The only way that the subjective cognitive representation can be accessed is through asking the individual to report that experience. In general, researches use two methods to measure emotions through subjective self reports: 1) free response measurement and 2) forced choice measurement (Scherer, 2005).

Free respond measurement of emotions includes a free-response format questionnaire that the test subjects answer with free choice of labels and expressions to explain their own perception of the emotional experience. The advantage of this method is the test subjects are most likely to report the most accurate and specific description of their emotional experience. However, this method has disadvantages as well because test subjects might have problems coming up with appropriate labels. Another disadvantage is the fact that it is very hard to analyze and convert the free response data into quantitative and statistical data because of the difficulty of categorizing the answers into emotion categories. To ease this process the *Geneva Affect Label Coder* (GALC) is proposed in (Scherer, 2005), which is an Excel macro parser program that can recognize 36 affective states distinguished by words in natural languages.

The second method, forced choice measurement, includes a questionnaire with a standardized list of emotion labels with different kinds of fixed-response answering formats. In this method, the test subjects choose from suggested emotion label alternatives to express their emotional experience. Unlike the free respond method, the advantage of this method is that the collected data can easily be converted into statistical data that quantifies and measures the emotional experience. However, the accuracy of this method is questionable because the given emotion alternatives are not always representing the exact category that the test subject would choose to use. Another disadvantage is the possibility that the test subject might not be familiar with the label that the researcher uses as an expression for a certain emotion. There are two main methods used for measuring emotions through forced-choice self-reports: a) the discrete emotions approach and b) the dimensional approach (Scherer, 2005).

The discrete emotions approach is based on categorizing emotions and labelling them in terms of the semantic field for emotions in natural languages. The questionnaire includes chosen emotion labels that the test subject should for example: rate – how intense she felt them (usually on a 3-5 point scale), mark – the emotions that she experienced (Scherer, 2005), or by preference – choosing which emotional experience had a larger impact on a given emotion compared to another experience. For expressing preferences two examples are the *Two Alternative Forced Choice* (2-AFC) and the *Four Alternative Forced Choice* (4-AFC) survey method (Yannakakis, et al., 2007c). The 2-AFC forces the test subject to choose which of two experiences induced an emotion more than the other. This can lead to noise, if the test subject is undecided. To solve this problem, the 4-AFC method can be used that in addition to 2-AFC gives the option of choosing *both* or *neither* for the emotional experiences.

An example of the dimensional approach is the *Affect Grid* (Russell, et al., 1989), which is a graphical representation using a 2D grid (see Figure 1) that shows emotions as the relation between the arousal and the valance felt during an emotional episode.

The procedure for using this method is the following:

"*The test subject is given specific instructions, such as "Please rate how you are feeling right now." And then she is asked to place one checkmark somewhere in the grid. The pleasure-displeasure (P) score is taken as the number of the square checked, with squares numbered along the horizontal dimension, counting 1 to 9 starting at the left. The arousal-sleepiness (A) score is taken as the number of the square checked, with squares numbered along the vertical dimension, counting 1 to 9 starting at the bottom.*" (Russell, et al., 1989)

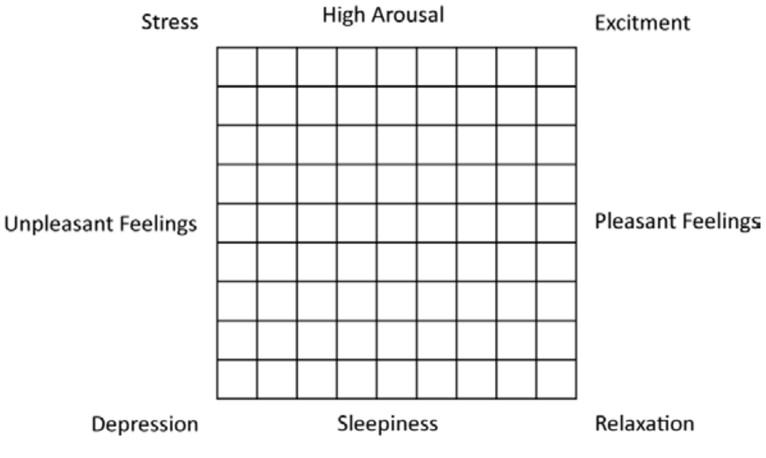


Figure 1. The Affect Grid (Russell, et al., 1989)

More detail about the relation between the arousal and the valance to emotions is explained in the following section.

2.2.2 Inferring Emotions from Physiological Response

Measuring and finding patterns in the physiological signals from the ANS, CNS, and the endocrine system are often used to infer emotions. The changes in the mentioned physiological systems happen in different time frames; therefore different physiological responses are more or less suitable for studying different physiological processes and emotions. For example, the responses from CNS (brain response i.e. event related potentials (ERP)) occur in milliseconds, while the responses from ANS (like the HR, skin conductance (SC) and others) generally are slower and occur after a few seconds (Hugdahl, 1995) (see Figure 2).

One of the methods for measuring the response from CNS is electroencephalography (EEG), which uses electrodes placed on the scalp to detect the electrical activity of the neurons in the brain. EEG has good temporal resolution, but it has very limited spatial resolution, which means that it tracks activity fast but it does not provide information about which part of the brain is active. This makes it hard to analyse data and infer significance from it (Rosenberg, et al., 2000). Positron emission tomography (PET) and magnetic resonance imaging (MRI) are other CNS response measures that provide good spatial resolution; however, they are very costly methods. In addition, PET has poor temporal resolution and is a risky technique. It is performed by injecting the participant with a radioactive isotope that is used for detecting which part of the brain is active because the isotope metabolizes in the active areas of the brain. MRI is more accessible than PET and risk free but its drawbacks is the noisy environment that it creates and therefore it is not very useful for studying brain changes during emotional episodes (Rosenberg, et al., 2000).

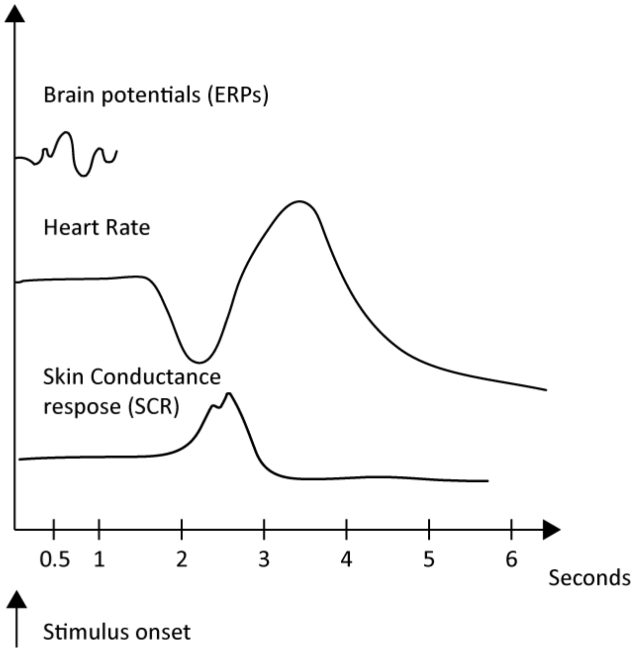


Figure 2. The duration of various psychophysiological measures (Hugdahl, 1995)

The ANS is the most widely studied physiological response system for measuring emotions. Numerous researchers (Lang (Lang, 1995), Schachter (Schachter, 1964), and Scherer (Scherer, 1993), among others) argue that physiological changes like for example increasing HR and sweaty hands (increasing SC) co-occur with emotions and measuring them can indentify emotions accurately. For example, the sweat glands in the palms are stimulated by nerves, which are activated during states of arousal and surprise, and HR responds to stress and emotion intensity (Hugdahl, 1995). HR and SC are the most common ANS measures, but some studies have measured other responses including skin temperature, respiratory depth, etc.

Osgood et al. (Osgood, et al., 1957) and Russell (Russell, 1980) discuss that emotions can be distributed along a bipolar dimension of affective valence – positive (pleasure) and negative (displeasure) emotions – and affective arousal – high arousing (activation) and low arousing (deactivation) emotions. This is because arousal and valance can be the cause or symptom for emotions. Figure 3 is an example of how the emotions are distributed among the valence and arousal dimensions. The research by Hugdahl (Hugdahl, 1995) and Ikehara et al. (Ikehara, et al., 2005) argue that physiological measurers can be used to indicate the amount of valence and arousal responses from the organism. For example, the SC measures arousal, and HR measures emotion intensity and combined with body temperature, it can indicate valence. Taking into account that each emotion is correlated to an amount of affect valence and arousal, physiological signals can be converted into valance and arousal and then classified into emotion (Mandryk, et al., 2006b).

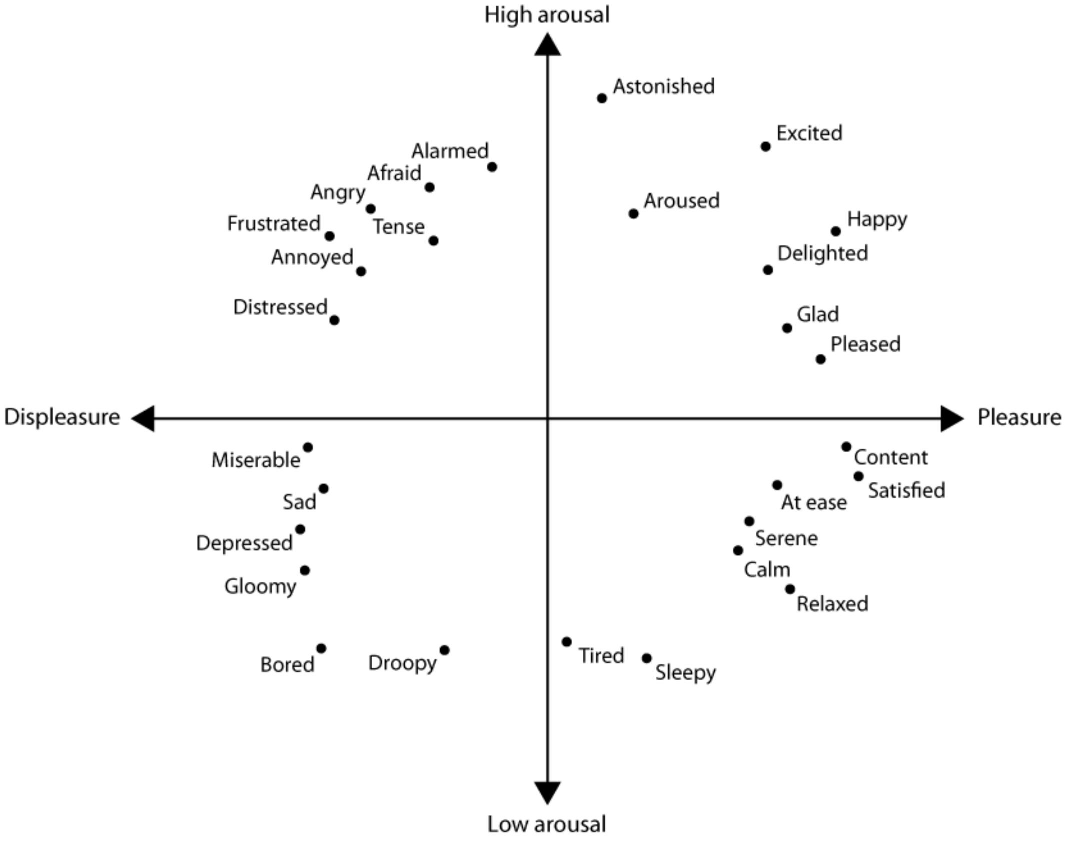


Figure 3. Emotions distributed along valence and arousal (Russell, 1980)

When using physiological data sources, there are methodological issues that must be addressed including the effort to control factors, such as: movement, drugs, general health status, and age, which can influence and bias the data (Picard, 1997). There are theoretical limitations to inferring significance from physiological data that need to be considered as well. For example, Cacioppo et al. (Cacioppo, et al., 1990) investigate the inference of emotions from physiological responses and conclude that the results are not always consistent. This could be because of noisy signals, occurrence of feelings that do not show significant physiological change, or occurrence of physiological responses similar to those in an emotional state without a relation to any emotion as well as the fact that each individual express emotion through physiology in different but related ways. A solution to this problem is to build an emotion recognition system that can generalize over different individuals by finding patterns in the physiological signals to predict emotions for each individual (Picard, 1997).

2.2.2.1 Physiology Sensors

As mentioned above, a great range of physiological signals can be used to build an affect recognizer i.e. an emotion model. In general, the tools for measuring any physiological response are specialized sensors like Galvin Skin Conductor (GSR) for measuring SC, Photoplethysmographic (PPG) for measuring BVP, Electrocardiogram (EKG) for detecting electrical activity from the heart muscle. Yannakakis et al. built an entertainment model that proved success by using the following sensors and metrics: Polar s610i pulse watch for measuring HR, PPG for BVP, and GSR for SC (Yannakakis, et al., 2007a), (Yannakakis, et al., 2007c). Mandryk et al. on the other hand used GSR and the additional sensors: Electrocardiography (EKG) to detect the heart muscles activity and Electromyography (EMG) to detect the face muscles tension (facial expressions) (Mandryk, et al., 2006b).

Galvin Skin Response (GSR) Sensor

GSR measures the SC with electrodes (small metal plates that apply a safe, imperceptibly level of voltage across the skin), which are usually attached to the subject's fingertips. Figure 4 is an example of a GSR sensor.

Figure 4. The Galvanic Skin Response Sensor (Picard, 2009)

The SC is a function of the sweat gland activity and the skin's pore size. This is because the sweat gland activity increases the SC. Lang (Lang, 1995) argues that the change in SC is linearly correlated to arousal, which means that it is a good indicator of emotional responses as well as cognitive activity. For example, when a subject is highly aroused (angry, excited), there is a fast increase in the SC (a period of seconds).

In Figure 5, an example of a SC signal for anger and grief is presented. The baseline of the SC varies for many reasons, including gender, diet, skin type and situation (Picard, 1997).

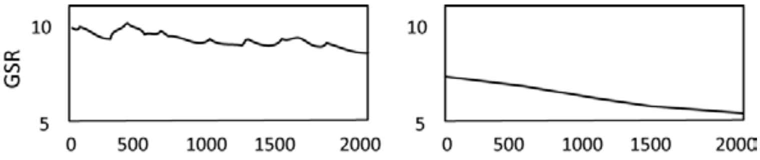


Figure 5. SC signal measured from an actress expressing anger and grief (Picard, 1997)

The Photoplethysmographic (PPG) Sensor

The BVP signal is an indication of the blood flow. The PPG sensor uses photoplethysmographic process, which applies a light source and measures the light reflected by the skin to detect the blood pressure in the extremities of the blood vessels in the body. This is possible because the heart contractions (heart beats) are forcing blood through the peripheral vessels and fill the vessels under the light source. In that way, the amount of light reflected is measured in the sensor. A PPG sensor is shown in Figure 6.

Figure 6. The Photoplethysmographic Sensor (Picard, 2009)

When the BVP signal is plotted, it has a very distinct form of peaks and valleys reflecting every heart beat (as seen in Figure 7). By knowing the timestamp for each heart beat, it is possible to calculate the HR, the algorithm is explained in *Specific Blood Volume Pulse and Heart Rate Features* (section 6.1.2.1). The interval between two heart beats is called the *inter-beat interval*, often referred to as a RR-interval. From the BVP signal, the inter-beat amplitude can be calculated. The inter-beat amplitude is the difference from a peak indicating a beat to the bottom of the inter-beat valley between two beats. In Figure 7, a BVP signal segment is shown with descriptions of the two BVP measurements.

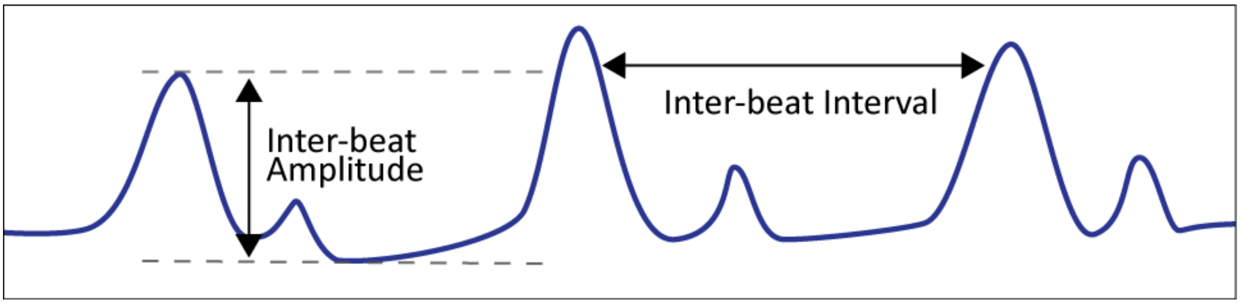


Figure 7. BVP signal segment showing where amplitude and inter-beat interval is calculated

Like SC, the BVP measurements can display changes in arousal. However, the increase in the BVP amplitude indicates decreased arousal and greater blood flow to the fingertips (Picard, 1997). When the HR has been calculated from the BVP, it can be used to infer emotion activity and can be used to infer affect arousal and valence (Mandryk, et al., 2006b). Figure 8 shows a BVP signal for anger and grief.

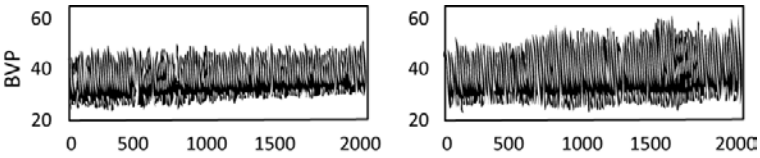


Figure 8. BVP signal measured from an actress expressing anger and grief (Picard, 1997)

2.2.3 Inferring Emotions from Behavioural Expressions

Behavioural expression in this study refers to the responses from the body motor system (facial expression, vocal expression, body movement and posture), which are elicited from emotions. In (Picard, 2009), Picard calls this influence of emotions, sentic modulation. Many studies have been investigating the relationship between emotions and the sentic modulations, such as vocal expression ((Scherer, 1986) and (Picard, 1997)), facial expression ((Manstead, 1988), (Picard, 1997), and (Dimberg, 1988)) and body movement and posture ((Bull, 1987) and (Bernhardt, et al., 2007)). The purpose of these studies is to find patterns of change in the sentic modulations that communicate emotions.

When investigating the influence of emotions on vocal expression, it is the vocal waveform and patterns of change in the intensity (loudness), frequency (pitch), quality (phonation type), and other related features that are analysed (Rosenberg, et al., 2000). However, to date more refined research has been done on finding correlation between emotions and facial expressions.

Commonly used facial expression measurement methods are: observer judgment, componential coding schemes, and electrophysiological recordings (Rosenberg, et al., 2000). Observer judgment is a method based on recording subjects' facial expression during an emotional episode and then analyzing the recording. This method is easy but it has a lot of drawbacks including naïve and biased judgment. On the other hand, the componential coding schemes method includes a developed coding system that follow a set of procedures for detecting certain facial expressions (Rosenberg, et al., 2000). The last mentioned method proposes to measure facial expression through electrophysiological recordings, which is done with EMG, as seen in (Mandryk, et al., 2006b). EMG measures the activity and contractions in the facial muscles, which can give an approximate "picture" of the face movement. By analysing the face muscles movement, researchers have defined independent expressive muscles, such as the muscle of attention, muscle of joy, muscle of lust, muscle of sadness (Picard, 1997).

Compared to voice and facial expression, body movement and posture responses have not received as much attention when studying emotions, because body movements and postures are complex and can vary individually. However, in recent years the psychology studies have been investigating the body movement and posture responses and have shown that they are good indicators for certain emotion states such as anger, boredom, interest, and excitement (Bianchi-Berthouze, et al., 2008). The following three techniques: video annotation, motion capture, and observer ratings, are the most commonly used techniques for analysing body movement. Here are two concrete examples of studies that are investigating the relation between player experience, emotions, and body movement:

- Berthouze uses the video annotation and observer ratings as well as immersion questionnaire techniques, and proves that body movement in game interaction is related to the player's fun, emotional and social experience and therefore it can be used to measure and adapt the player experience (Bianchi-Berthouze, 2008a).

- Robinson presents a framework for analyzing non-stylised (i.e. door knocking) motion to detect implicitly communicated emotion. It uses motion capture to analyze the body movement data (Bernhardt, et al., 2007).

Motion capture techniques capture the body movement data and transform it into a digital signal. Further on many features from this signal can be calculated that can be used for building automated models that can recognize patterns in body movement.

2.2.4 Body Movement in Video Games

The idea of incorporating body movement into video games is not unfamiliar today. One of the first commercially available video gaming systems that incorporated physical interaction was the *Super Nintendo Entertainment System* (SNES) (Nintendo, 1990). Albeit, these early developments were not so physically challenging, since they only incorporated a light gun the player used to aim and shoot objects on the screen. More recent development has facilitated more physical interaction in video games, referred to as *kinetic video games*. In (Parker, 2003), a kinetic video game is described as:

"*A game that requires a computer to mediate game play, and that has as a critical aspect of its interface, the input of information concerning the overall physical activity of the player. Activity is not defined by motions that specifically manipulate computer input devices (keyboard, mouse, etc.) but movement of body parts that are interpreted by the computer to have specified meanings.*" (Parker, 2003)

A few novel interfaces exist, like guitars, bongo drums and maracas that facilitate to some extend body movement. However, these interfaces are binary in structure, which means that they do not employ any spatial information and could easily be replaced by keyboards, although with "not-as-fun" interaction. Titles like *Dance Dance Revolution* (DDR) (Konami, 1998) and the more recent video game console add-on, *EyeToy* (PlayStation, 2003), have had great market success in that time and were some of the few real kinetic video games on the market.

In DDR, players stand on a "dance pad" and step on coloured arrows, laid out in a cross, moving their feet to musical and visual cues on the screen (see Figure 9). Players score points based on how well they time their dance movements to the patterns that are presented to them.

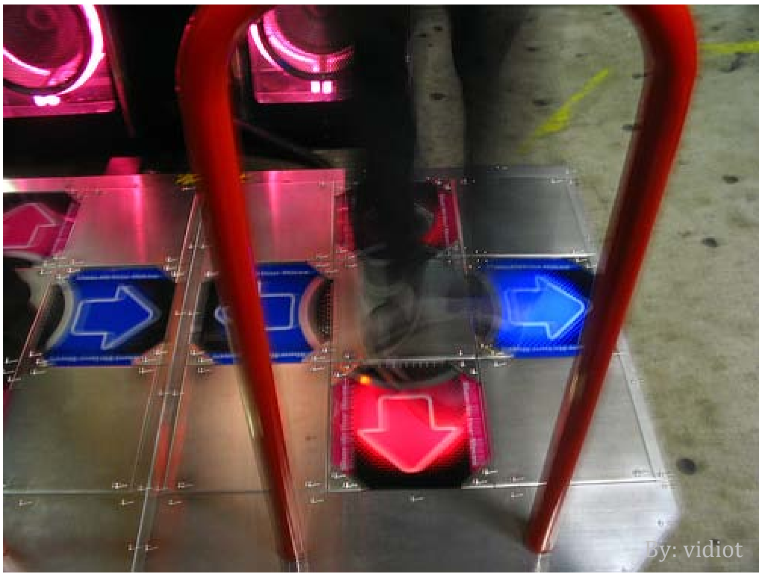


Figure 9. The Dance Dance Revolution game with a player, playing really fast

The *EyeToy* add-on to the *PlayStation 2* gaming console used a low-cost video camera and microphone to record players to present them with a mirrored image of themselves (see Figure 10). The game characters and events are superimposed onto this image and based on various computer vision algorithms the player could interact with these. Because of the limited technology behind the *EyeToy*, the games could not utilize any postures or body movements as spatial information e.g. dynamic 3D gestures like a bowling throw would not be able to be recognized, as only 2D movement is available for the camera.

Figure 10. In-game view of an EyeToy game where the player is seen mirrored in the game.

In September 2006, Nintendo released their latest video game console, the *Wii* (Nintendo, 2006a). The controller released with it is the Nintendo Wii Remote (also known as the Wii Remote or Wiimote), that revolutionized the gaming industry with kinetic video games. The Wii Remote has an ADXL330, three-axis accelerometer, built in, as well as an infrared camera to track infrared light from various sources .i.e. the Nintendo Sensor Bar. The infrared light tracking can then be used to approximate distance to sensor bar and the orientation of the Wii Remote. The accelerometer in the Wii Remote can capture the force in three dimensions with g-forces ranging between ±3 *g*.

Figure 11. The Nintendo Wii Remote

Since 2006 a large variety of novel applications for the Wii Remote have been developed. The first game developed for the *Nintendo Wii* was *WiiSports* (Nintendo, 2006b), which includes different sports games such as boxing, bowling, and tennis. In these games, the in-game characters are controlled by making the motion associated with the sport i.e. swinging the Wii Remote as if it was a tennis racket. The details of the motion processing algorithms used in *Wii* games are not publically available and not many games utilize true 3D motion tracking and gesture recognition. It appear to be that most of the existing *Wii* games use static information obtained from the Wii Remote and not acceleration data obtained over time. However, the software company AiLive Inc. (AiLive, 2000) has developed a 3D motion recognition middleware for the Wii Remote that "*uses state-of-the-art machine learning technology to completely automate the process of creating motion recognizers*".

2.2.4.1 Inferring Emotions from Physical Video Games

Recognizing 3D motion and gestures allows for better emotion recognition, since more information is available, e.g. the accuracy or the power of the gesture performed. But what does the force of a baseball bat swing say about the emotion of the player? The study in (Pasch, et al., 2008) tries to identify players' movement patterns while playing *Wii* games. The results shows that players had two motivations for playing *Wii* games, namely; "Achieving" and "Relaxing". In addition, they concluded that players employed two related strategies when playing *Wii* games, namely; "Game" and "Simulation". When players are "gaming", they are trying to achieve the highest score possible by only performing the necessary movements required by the game i.e. small, quick and precise movements. However, when players are in the "simulation" state, they want to immerse into the game and their movement pattern changes to more simulation of the real-life movements like swinging a tennis racket with big movements. Indeed, Juul, game theorist and ludologist[5], in his work (Juul, 2005) makes the distinction between game and play. He states that in games there exists two motivations, you can either "play" or "game". In English, there is no distinction between playing and "gaming" (since this word does not exist as a verb in the English language); however in Scandinavian languages the difference is clear. "Lege" is when one is just "playing around" or as they in the aforementioned study are referring to as "simulating" and "spille" is when one is "gaming the game" (spille spillet) i.e. when one is playing

---

[5] Ludology is the discipline that studies games, play, toys and videogames. http://www.ludology.org/

the game *and* adhering to the rules and trying to achieve the goal in the game. A qualitative observation of this can be seen in one of the player responses to the aforementioned study:

"As you play and play you start to realize that you don't really need to swing and it's just a small movement that you need to make - so I tend to play more technically rather than emotionally. […] When I am playing to relax and I play baseball, I swing like I would with a real baseball bat. But if I am playing to beat somebody else then I do what I need to do, to do the movements." (Pasch, et al., 2008)

This does not depict the player's emotion, but gives an indication of engagement or immersion. In (Bianchi-Berthouze, et al., 2007) results also show that an increase in body movement in video games results in an increase in the player's engagement level. Berthouze (Bianchi-Berthouze, 2008a) identifies five types of body movements exerted in physical video games as shown in Table 3.

Table 3. Different types of body movement (Bianchi-Berthouze, 2008b)

Body Movement	Description	Lazzaro's Factors
Task control	necessary to control the game	Hard fun
Task facilitating	facilitating the control of the task but not required to play the game	Hard fun
Task(role)-related	typical of the role defined by the game scenario even if interfering with game	Easy fun
Emotional expressions	expressing affective states related to or induced by the game experience	Emotional factor
Social behaviour	supporting social interaction	Social Factor

Their conclusion, is when the game controllers afford task-related types of body movement, the player has a more affective experience. When the controllers do not afford body movements that are natural to the game scenario, they observed a lack of movements other than those necessary to facilitate the control of the game, in contrast when the controllers do afford body movements that is natural to the scenario they observe movements that are related to enjoyment.

The analysis in this section is an indication that player's body movement data can be used for inferring player's preferences and emotions, and therefore, be used as features for the emotion model.

2.2.5 Recognizing Emotions with Affective Computing

Affective computing is a study, which develops Artificial Intelligent (AI) models and systems that can analyse and understand human emotions, as well as express them to the AI's physical surroundings.

Analysing patterns in human emotion expressions with machine learning methods can be used to do objective evaluation of subjective experiences (Picard, 1997). This is especially interesting for the entertainment industry (Mandryk, et al., 2006b). Mandryk et al. have developed a methodology for objective evaluation of entertainment technology that uses fuzzy logic techniques to create a continuous emotion model. The model is proven to provide more choice and robustness for the evaluators than a subjective self report method for recognizing emotions.

The model uses physiological feedback (GSR, EKG, and EMG) as the input for evaluating the player's emotions during play.

The American company EmSense (EmSense, 2004) developed a commercial sensor-laden headset that uses physiology and brainwave (EEG and other sensors) measurement technology to measure the consumers' responses to media. The technology monitors the brain waves, breathing rate, head motion, HR, blink rate and skin temperature and uses custom build mathematical models to indicate, if a person is engaged or excited (Greene, 2007).

However, numerous researches have been done on recognizing emotions with AI models that are very closely related to the study in this paper. One of them is the study of McQuiggan et al. (McQuiggan, et al., 2006), which uses SC and BVP signals to construct an emotion model with the following three machine learning techniques: induced decision trees, naïve Bayes, and Bayesian network. Their study investigates which one of these models is the most accurate in mapping the player's physiological data to emotions. The purpose is to choose the most accurate model and use it to improve engagement, especially in educational games (McQuiggan, et al., 2006). They have gained high success rates when evaluating the models, as high as 90% accuracy with the induced decision tree model.

Another related study is the research from Yannakakis et al. ((Yannakakis, et al., 2007c) and (Yannakakis, et al., 2007a)), in which an ANN model is developed that can capture player's level of reported "fun", while playing physical active games built on BVP and SC signal input. This model proved success with 76% classification accuracy in recognizing the level of "fun" felt.

As mentioned in section *Inferring Emotions from Physiological Response* (section 2.2.2), sensors like PPG and GSR are widely used, but other ways of detecting emotions have also been tried. The study in (Sykes, et al., 2003) investigates to what extent the pressure on the gamepad buttons can be mapped to the player's arousal. With the introduction of the *Wii*, and soon followed by Microsoft's *Project Natal* (Microsoft, 2009) that introduces a play experience completely free from any input device only using body movement for input, pressure on gamepad buttons will be unnecessary. Therefore, development of method to capture players' affective state through body motion as the one proposed in *Detecting Affect from Non-Stylised Body Motions* (Bernhardt, et al., 2007) could have a commercial success in games for platforms such as *Wii* or *Natal*.

Because of its popularity and commercial potential the concept of detecting emotions from players' body movement is investigated in this study. In addition, detecting emotions from physiological SC and BVP signals, because of its success rate and commercial use today, is investigated. The proposed model in this study follows the experimental methodology proposed by Yannakakis et al. in (Yannakakis, et al., 2007c) and constructs a model for capturing players' reported entertainment and emotions in *Wii* games using HR, SC, BVP, and acceleration data acquired from the Wii Remote. The reason for following the ANN approach in (Yannakakis, et al., 2007c) compared to the, at first glance, better decision tree approach in (McQuiggan, et al., 2006), is that the ANN approximates a continuous function, whereas the decision trees only classify the input.

2.3 Game Factors for Affecting Player Experience

A game is made up of many game design factors (such as story, sound, challenge, goal, etc.), which have significant influence on the player experience. The emotion model developed in this study is used for automated tailoring of the player's game experience towards a selected emotional state. To achieve this, controllable game factors have to be defined in the developed test-bed game, which can be manipulated based on the emotion model outcome.

Researchers have been studying games with the purpose to define the main game factors that make games fun or influence the experience. For example, Malone in his study (Malone, 1981) identifies three game factors that influence the player's engagement and experience in games:

- Challenge: how hard/challenging the goal in the game is
- Curiosity: how predictable the game states are
- Fantasy: what mental images of physical objects and social situations are generated

Malone's definition of challenge is "*in order for a computer game to be challenging, it must provide a goal whose attainment is uncertain.*" (Malone, 1981)

With this statement, Malone argues that games are enjoyable and engaging if they provide appropriate goals without an obvious solution. The goals are the objectives in the game and the challenge refers to what the players has to do, by using their skills, to reach them. If it is designed well, the goals can keep the attention and the interest of the player. Malone suggests guidelines for designing the appropriate goals, for example providing obvious clear goals (preferably by the use of story or visual effects) or complex environment, where players are easily able to generate goals with the appropriate difficulty. The difficulty level of reaching the goals in a game influences the player experience. For example, if the challenge in the game is too easy or too hard for the player, it can be a boring or a frustrating experience for her (Bateman, et al., 2006).

Malone defines the curiosity factor as "*curiosity is the motivation to learn […] computer games can evoke a learner's curiosity by providing environments that have an optimal level of informational complexity. They should be novel and surprising, but not completely incomprehensible.*" (Malone, 1981)

Malone states that curiosity refers to the presentation of the content and the environment in the game. Designing optimal curiosity means the player can understand the environment and learn what to expect from it, but sometimes these expectations are broken down and surprise the player. Malone discusses two types of curiosity in games that affects the player experience: sensory and cognitive curiosity. Sensory curiosity in a game is the changes and patterns in light, graphics, sound, and other sensory stimuli, which can be used as decoration to enhance the fantasy, to reward the player, and to convey information. The design and presentation of these sensory stimuli in a game environment affect the player experience (Malone, 1981). For example, in a game like *Yoshi's Story* (Nintendo, 1998), the player is taught that falling is fatal when a sound of rushing air is heard, if this indication suddenly changes, it can elicit frustration and confusion for the player (Bateman, et al., 2006). Cognitive curiosity is the factor that motivates players to create logical meaning and structure (completeness, consistency, and parsimony) from their knowledge about the game. Engaging a player in a game can be done by presenting just enough information in the game, thus she feels that her knowledge is incomplete and thereby she is interested to play and learn more of the game. Therefore, certain decisions about how and what information to be presented (new or repeated information) can affect the player's curiosity and experience.

Malone defines the fantasy factor as "f*antasies often make computer games more interesting. In general, games that include fantasy show or evoke images of physical objects or social situations not actually present.*" (Malone, 1981)

The fantasy factor is the story and the setting in a game, which can be communicated through the game mechanics as well as the visuals and the sounds in the game world. Fantasies have emotional aspects that affect the player experience. For example, different fantasies appeal to different types of players (Bateman, et al., 2006).

According to a study by Lazzaro (Lazzaro, 2004) there are four game factors that affect the player's experience (these factors are not explained in details since they are similar to the Malone's three factors):

- Hard fun (related to Malone's challenge element)
- Easy fun (related to Malone's curiosity)
- Altered states (the player's perception and behaviour in the game which produces emotions and other internal sensations – this element is closely related to Malone's fantasy)
- Socialization (the people factor)

Lazzaro have discovered these factors by observing the emotions that test subjects expressed through facial gestures, body language, and verbal comments while playing games (Lazzaro, 2004). Lazzaro argues that each of the four factors is the reason why people play games and each one of them affects the player experience. Furthermore, from her observation she concludes the following: hard fun frequently generates emotions and experiences of "Frustration" and "Personal Triumph", easy fun generates "Wonder", "Awe", and "Mystery", altered states generates "Excitement" and "Relief" from personal thoughts and feelings, and the people factor generates "Amusement", "Pleasure", and "Pride".

2.4 Machine Learning

When developing an emotion model, various concepts and theories can be used. Some of these approaches have been discussed in *Recognizing Emotions with Affective Computing* (section 2.2.5), where related work has attempted to classify emotions using decision trees, fuzzy logics, and Bayesian networks. In the methodology followed in this study, ANNs have been proved successful at approximating complex continuous functions for "fun". To get an in-depth knowledge of ANNs – so they can be used for the emotion model – they will be discussed and explained in the following sections.

2.4.1 Artificial Neural Networks (ANN)

The human brain is very good at pattern recognition and generalization of existing knowledge and to use it in new situations. Following this fact, a lot of research has gone into emulating the human brain in a digital system to enable AI to better handle new situations and problems (Buckland, 2002a). The human brain is multithreaded as many different inputs (senses) are handled at the same time, but also because a lot of parallel processes are running to solve the same problem. For example, the visual cortex handles the input from the eyes; it recognizes important parts and is able to recognize known objects in new environments that has never seen before. The process takes about 150ms (Thorpe, et al., 1996), which is very fast compared to the amount of data that is being processed. In digital systems, it is very interesting to develop ANNs as they have the potential of being executed on multiple computers with multiple CPUs, and thereby making it possible to execute huge complex neural networks with the speed of the human brain. The reason that would be possible is that ANNs consists of numerous small units called neurons that could very-well be distributed onto multiple systems and CPU's. A neuron is simply a small unit that will pass on a value based on the input it receives (Buckland, 2002c). For complex problems, one of the main advantages of ANNs is their execution speed. The investigation and theory of ANNs presented in the following sections is based on the ANN literature by Buckland (Buckland, 2002b) and Lippmann (Lippmann, 1987).

Figure 12. Illustration of connected neurons in the brain

2.4.1.1 *Single Layer Perceptrons*

To get a better understanding of how ANNs work, we need to understand the basic principles of neuron. The inner-workings of the neuron in the human brain is not completely known, and most likely it functions nothing like the digital equivalent. However, the digital neuron contains an activation function that will convert the input signals to an output and pass it on to all connected neurons. This idea is named a *perceptron* by Rosenblatt (Rosenblatt, 1961). A perceptron can have multiple inputs, each of which is weighted with a *connection weight*. This makes it possible to modulate the input before it is passed through an activation function, giving the inputs different priorities. The perceptron shown in Figure 13 has five inputs with associated weights that are passed through the activation function, in this case a step function. The output will depend on what the inputs are and return either 0 or 1 which is the nature of a step function.

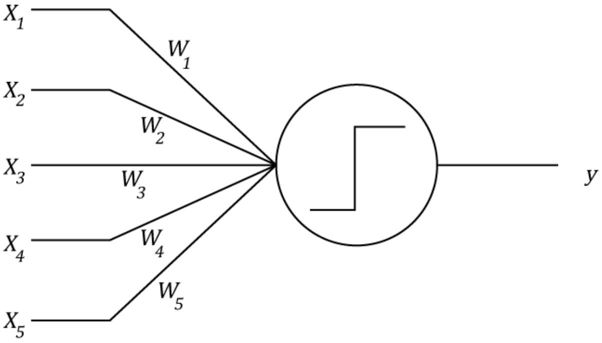


Figure 13. Perceptron with weighted inputs (w_n), and an output (y) (Buckland, 2002b)

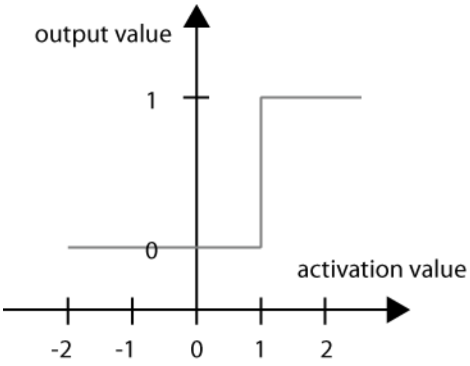


Figure 14. Step activation function (Buckland, 2002b)

The inputs are weighted and summed and are passed to the activation function using the following method, where n is the number of inputs, x is the input value, and w is the associated weight:

$$y = f\left(\sum_{i=1}^{n} x_i * w_i\right)$$

A simple perceptron can solve simple linear separable problems like the *AND* or *OR* operator. A simple perceptron would be able to solve the problem using two inputs (true or false) and one output (true or false) using a step function as the activation function.

A simple ANN could be manually configured to give the right output based on the input. However, as the network increases in size or the problems become more complex, the ANN can be trained using a training algorithm to approximate the connection weights automatically. When the ANN is trained, the weights are gradually adjusted to approach the best solution. To determine how much the weights should be adjusted, the Delta Rule can be used to calculate the network gradient:

$$\Delta w_{ji} = \alpha\left(t_j - y_j\right) * g'\left(h_j\right) * x_i$$

Where α is the learning rate, t is the desired output, y is the actual output, $g'(h_j)$ is the activation functions derivative, and finally x is the input .When the weight-change has been calculated, it needs to be applied to the current weight:

$$w_{ji} = w_{ji} + \Delta w_{ji}$$

Depending on the learning rate, the process has to be repeated many times and will at some point converge, if the problem is linearly separable.

2.4.1.2 Multi-Layered Perceptrons

For the ANN to find solutions to complex non-linearly separable problems, the simple single layer perceptron needs to be extended with more neurons and layers known as hidden layers. This enables the ANN to approximate complex functions with more or less precision depending on the size of the network i.e. how many hidden layers, how many neurons there is in each layer, and how it is all connected internally. As seen in Figure 15, the input layer will receive the input and pass it on to the following layer. Based on the connection weights and the activation functions, the signal will be modulated through all the consecutive layers until the signal finally is received by the output layer.

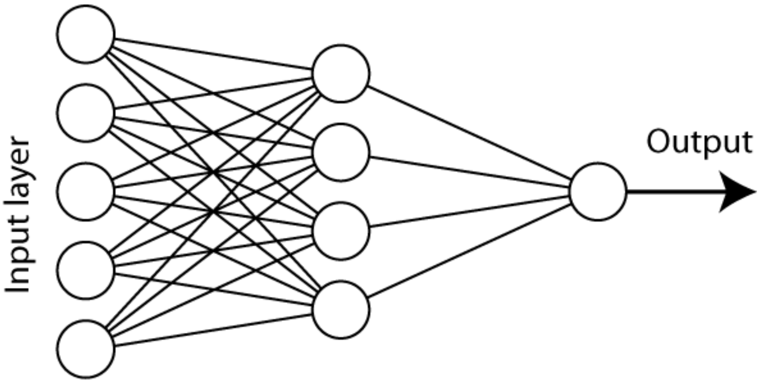


Figure 15. A simple multi-layered neural network (Buckland, 2002b)

One of the most used activation functions for multi-layered perceptrons is the sigmoid function (Haykin, 2008):

$$f(x) = \frac{1}{1+e^{-x}}$$

The sigmoid is a continuous function that can either go from 0 to 1 or the bipolar version if the output should be able to be negative in the range from -1 to 1.

$$f(x) = \frac{2}{1+e^{-x}} - 1$$

Shown in Figure 16 and Figure 17 is the sigmoid activation function and the bipolar sigmoid function respectively.

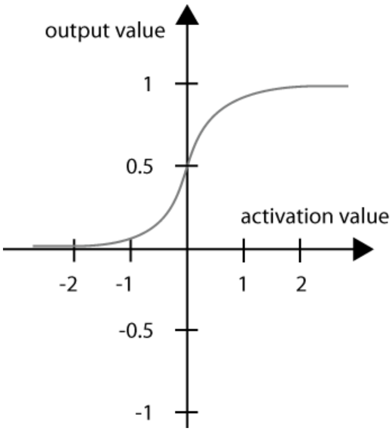


Figure 16. Sigmoid logistic function (Buckland, 2002b)

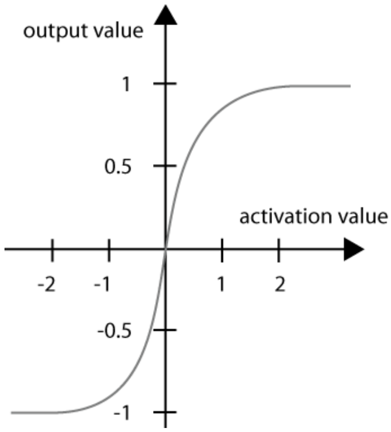


Figure 17. Bipolar sigmoid function (Buckland, 2002b)

The derivative of the sigmoid function that is used when training is:

$$f'(x) = s(x)(1 - s(x))$$

The training of a multi-layered ANN is a bit more complex, as the error has to be passed around the network and gradually adjust all the weights. This can be achieved through a back-propagation algorithm that will propagate the error from the output and backward through the network to the input layer, hence the name back-propagation. To adjust the connection weights the following steps must be calculated recursively:

$$w_{ji}(t+1) = w_{ji}(t) + \ \eta\delta_j x_i$$

In this equation, $w_{ji}(t)$ is the weight from a hidden node or an input node to the current node at time t, x_i is either the output from node i or an input value, η is the learning rate, and δ_j is the node gradient (Lippmann, 1987). The nodes error gradient is calculated in two different ways depending on whether it is a node in a hidden layer or in the output layer. If it is an output node the following formula is used:

$$\delta_j = y_j\left(1 - y_j\right)(d_j - y_j)$$

Where d is the desired output from the node and y is the actual output. When inside a hidden layer another formula is used:

$$\delta_j = x_j\left(1 - x_j\right)\sum_k \delta_k w_{jk}$$

Where k is the number of posterior neurons connected to the neuron.

2.4.1.3 Training

The procedure for training an ANN is to firstly initialize the network with random weights. Depending on the network, these values can be greater or smaller and both positive and negative or positive only i.e. [-5; 5] or [0; 1].

To perform the actual training, a training set is needed. The training set should contain the inputs and the correct outputs for each of the corresponding inputs. If the network is to be used as a classifier, usually all inputs except the class the input belongs to is set to zero and the correct class output will be set to one. Then, for each input-output pair in the training set the network result is calculated and compared with the desired output. Lastly, the error will be back-propagated through the network. The process is repeated over and over again until a solution is found. When using a training set to train the network, it is called a training epoch, where all training samples has been used, this again is used over and over.

Besides back-propagation mentioned earlier, other ways exist to train the ANN, for example artificial evolution algorithms.

2.4.2 Artificial Evolution

Artificial evolution is achieved with Genetic Algorithms (GAs). GAs are commonly used for finding solutions for optimization problems or search problems. GAs is especially useful when the best behaviour is not known in advance, but needs to be guessed. Only requirement is the perfect behaviour can be recognized when it is found. The investigation and theory presented in this section is based on (Buckland, 2002b).

GAs are inspired from the nature when two humans, animals, or plants exchange DNA to form a new specimen. In nature, the DNA is made up of genes and is the building blocks of the specimen. When a new descendant is created, it is a mix of the parents' DNA strings. The parents – according to Charles Darwin – is selected through natural selection e.g. survival of the fittest. In such a way that the best specimens will survive and evolve in the "best" direction and the least fitted (those that die from disease, unable to gather food, etc.) will die.

In computers, the concept is the same. The digital DNA is the building blocks for an algorithm. When the algorithm is to be trained, a population of chromosomes is created and competes against each other to exchange DNA (The size of the population can vary and there is no golden rule). Depending on the utilized selection method, the best chromosome from a population is selected using a fitness function; the fitness is usually how well a chromosome performs when calculating solutions from an unseen input dataset. Different methods for selecting chromosomes exist, where the two most used methods are the *Rank* and *Roulette Wheel* selection methods. The *Rank* selection method orders the population after fitness score and selects the N best chromosomes, this will effectively delete all the worst performing chromosomes. To uphold the population size after the selection, new random chromosomes is introduced to the population. The *Roulette Wheel* selection method selects chromosomes with probabilities that is proportional to their fitness i.e. the fitter the chromosome, the more probability it has of being selected. After selection the number of chromosomes in the population will have decreased, as with the *Rank* selection, there has to be added new random chromosomes to the population. Another selection method exists; namely *Elite* selection. *Elite* selection ranks the population in the same manner as the *Rank* selection method, the main difference is that the best performing chromosomes are cloned before any crossover or mutation is performed. This will increase the size of the population, that later has to be reduced. The fact that no chromosome are deleted or altered directly prevents the population to lose any of the best performing chromosomes.

When selection has been performed, new offspring need to be added to the population. To do this, crossover is applied. There exist different strategies on how to perform crossover on two chromosomes to form offspring. A small selection is seen in Figure 18. The three generic are: *Single-Point* , *Two-Point*, and *Uniform* crossover. The *Single-Point* crossover method will pick a random splitting point in the chromosome and merge the first section of the chromosome with the second part of the other chromosome and form an offspring and vice versa with the opposite sections for another offspring. The *Two-Point* crossover method is similar to *Single-Point*, in contrast using two splitting points opposed to one. The *Uniform* crossover method mixes the genes the most of the three, as multiple splitting points are selected. When the crossover has been performed the parents of the new offspring will be removed from the population. This is different from the *Elite* selection that does not remove any parents, when the crossover has been performed.

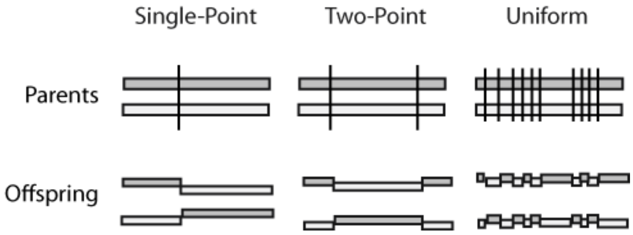


Figure 18. Crossover strategies for genetic algorithms (Champandard, 2004)

For the algorithm to find new solutions, besides those that can be formed of endless combinations of chromosomes, a mutation step is added. To mutate a chromosome, a random gene will be modified, it can be multiplied or added with a given value. In each generation, only a small group of chromosomes will be mutated dictated by the mutation rate.

When using *Elite* selection, the population will have grown over its allowed population size, after crossover and mutation. To decide which chromosomes should be kept and which should be removed; all the chromosomes have to be evaluated and have their fitness determined. With the chromosome performance known; 75% of the initial population size will be selected to go on to the next training epoch (75% is selected for example only and is not a fixed amount). To grow the population to the intended size, random chromosomes are added, and thereby might add new or better genes to the population.

2.4.2.1 Evolving Neural Networks

GAs can be used to evolve the network to converge towards a solution. When ANNs are evolved the connection weights is the gene making up a chromosome and is an abstract representation of the ANN.

Mutating chromosomes for ANNs is straight forward by using the method explained earlier, but Montana et al. (Monatna, et al., 1989) suggests calculating which nodes in the ANN is the weakest, and then mutating those. A weak node is a node, where outgoing connections is weighted zero (or close to zero) in the posterior nodes. However, this is not necessary to find a good solution.

2.4.3 Feature Selection

ANNs operate using various features describing the state of the problem; by using the features the solution can be calculated. In a complex world scenario the ANN designer has no problem in finding 10, 20, or 100 features (or more) that can be used to describe the state. But the more inputs to the network, the slower it converges, and it might perform badly using some of the chosen inputs. To automate the task of selecting the best performing features for the neural network, a feature selection algorithm can be used. In literature there exists numerous ways to select which features to include in the feature subset for optimal classification accuracy (Zongker, et al., 1996): *n Best Feature Selection* (nBest)*, Plus l - Take r Away* (PTA), *Sequential Forward Selection* (SFS) and *Sequential Forward Floating Selection* (SFFS), among others.

The nBest algorithm tests the features one by one and determines their performance and chooses the *n* best features for the subset. For obvious reasons, the selected feature subset cannot be assumed to be the best as the selected features are not necessarily optimal when combined. This can be done with a much smarter but more computational expensive algorithm, namely the SFS algorithm.

The SFS is designed to find the smallest possible feature subset, while maintaining high performance. The algorithm is a bottom up search, where one feature is added at a time to the current feature subset. The current subset together with each of the remaining features is evaluated and the best feature is selected and added to the current subset. This is repeated until the added feature yields lower or equal validation performance compared to the previous feature subset performance. To minimize the risk of ending at a local minimum, it can be useful to allow the newly added feature to decrease in performance to check if an additional feature could perform better than the feature subset prior to performance decrease.

The PTA algorithm is, to a large extend, manual in the same sense as nBest as the designer has to decide on two parameters prior to execution: the number of inclusion steps: *l*, and the number of exclusion steps: *r*. The algorithm will iterate over the features in the same way as SFS including features that improve the classification accuracy. When *l* features has been added, the algorithm will reverse and iteratively remove the *r* worst features that decrease, the performance for the selected feature subset.

The PTA algorithm can be automated so that the parameters do not need to be set in advance but can be selected during execution. That algorithm is SFFS and uses the full functionality of SFS to find the smallest feature subset without compromising on performance. When the SFS algorithm is not able to find any new features that will improve the performance, SFFS will start to remove features to find any features that can be removed, without decreasing, or maybe even increase performance. When all has been tried to be removed it will start over and try to add new features to see if it can find a new combination of features to improve performance.

For comparison the different selection method performances is shown in Figure 19.

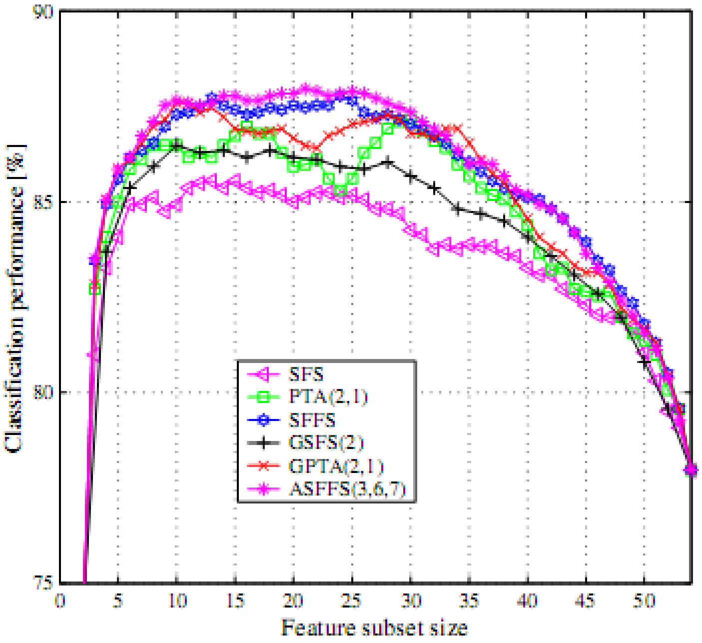


Figure 19. Feature selection algorithm performance (Pernkopf, et al., 2001)

2.4.4 Preference Learning

The task of learning a model based on various user preferences is a difficult problem. The idea is to let users try different configurations of a system in pairs and let them express, which configuration they liked more than the other. Because the preferences are not an absolute value but a relative relation between two options, the ANN cannot be trained used normal back-propagation algorithm. This is, why it is suggested to use GAs to search for a solution (Yannakakis, et al., 2007c).

When training the model by using GAs, the network is presented pair-wise with two inputs and a preference for either of the two inputs. The results for the network for the two inputs are then compared, and the networks performance is evaluated on how well its results correspond to the preference for the input pair. The overall performance of the ANN is how many percent of the presented preference pairs that was calculated to match the preference.

3 METHODOLOGY

Details about the methodology used in this study are explained in this section. To build the proposed computational model for recognizing emotion, the following five steps are implemented:

3.1 Step 1

Based on the research done on emotions in *Identifying Emotions* (section 2.1) and *Measuring Emotions* (section 2.2) features from the player's physiological signals (SC, BVP and HR) and body motion data (Wii Remote) are used to infer emotions from. This is because the related work done in affective computing, as seen in *Recognizing Emotions with Affective Computing* (section 2.2.5), showed that using these features is successful in inferring emotions. The emotions that the proposed model recognizes are frustration, excitement, relaxation and boredom. The reason is the different properties of these emotions, which according to the *Affect Grid* (Russell, et al., 1989) represent each area of the arousal and the valance properties of emotions as depicted in Figure 20 (more details about the Affect Grid are presented in *Inferring Emotions from Physiological Response* (section 2.2.2)).

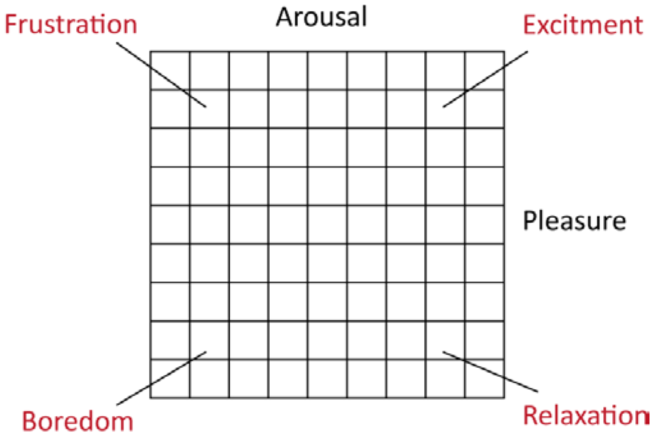


Figure 20. Frustration, excitement, relaxation and boredom representation on the Affect Grid

3.2 Step 2

To tailor the player experience, game design factors needed to be defined and automated to map to the player's emotional state. Based on the research in *Game Factors for Affecting Player Experience* (section 2.3), challenge and curiosity were chosen to be the controllable factors in the test-bed game. These two factors were used to generate different variations and experiences of the test-bed game, *Wiizards* (the design and description of the game is explained in *Wiizards – The Test-bed Game* (section 4)). The challenge and curiosity were quantified into *low*, *medium,* and *high*, and different combinations of these controllable features were used to generate nine variants of the game i.e. one variation could be "*low* curiosity" and "*high* challenge" together. Further details about the quantifying process and implementation of these two factors in the game can be found in Game Factors Implementation (section 4.4).

3.3 Step 3

A user study was designed that uses the *Wii* test-bed game and a survey to collect the following data: a) player's physiological signals from the GSR and BVP sensors, b) player's motion acceleration data from the Wii Remote, c) player's interaction data from the game, and d) user reported pair-wise preference data of the different emotions for each of the game variants. The test scenario included the following procedure: I) the test subject had the GSR and BVP sensors attached and was given Wii Remote to play the test-bed game, then II) the test subject was asked to play two game variants, and finally III) she was asked to evaluate which of the two game variants played were more exciting, boring, frustrating, and relaxing. Further details about the user study can be found in *User Study* (section 5).

3.4 Step 4

To get the best possible features the physiological signals were processed to remove noise and cropped to fit each play session down to the second. The BVP signal was used to calculate the HR that in turn was used to calculate additional features related to the HR. A total of 61 features were calculated for the physiological signals and the Wii Remote acceleration data. More details on the signal processing and extracted features can be found in *Signal Processing* (section 6.1.1).

3.5 Step 5

Machine learning (evolving ANN) was used to learn the association between features of the physiological signal data (SC, BVP and HR) and selected emotion preferences that players reported while playing the test-bed game variants. In addition to the physiological input to the ANN model the player's Wii Remote acceleration data together with certain elements of the player's playing specifications (e.g. average response time), were used. Further on, features were picked (using feature selection) in order to help the model achieve higher performance in capturing the emotion. Further details about the implementation and training of the ANN emotion model can be found in *Emotion Model Learning* (section 6.3).

3.6 Step 6

The ANN trained emotion model is proven to be successful in capturing the player's emotions, therefore the last step in this study is a discussion about how this model can be used to tailor the player experience in real-time. The proposal is to automate the controllable game features – challenge and curiosity – and use them to change the player's emotional experience. The model should predict the outcome of minute changes to the game features, and thereby make it possible to push the player experience towards a specific emotion. Further details about the discussion on tailoring the player's emotional experience can be found in *Tailoring Player Experience* (section 7.7).

4 WIIZARDS – THE TEST-BED GAME

A test-bed Wii game titled *Wiizards* was developed and used to gather user preferences for constructing an emotion model. In the following section details about the game design and implementation are explained.

4.1 Game Description

Wiizards is a single player magic fighting game, in which the player embodies the role of a wizard and duels against an enemy wizard. As described in *Inferring Emotions from Physical Video Games* (section 2.2.4.1), when video games afford body movement, it in effect affords players' to engage more in the game. Therefore, *Wiizards* is designed with focus on a novel gestural interface, where the Wii Remote represents a magic wand that the player uses to "perform magic". The "perform magic" mechanic is tied up to the player's interaction with the Wii Remote, by doing certain physical movements (gestures). The purpose of the game is to win all duels against the opponent player, which resembles the goals of a typical fighting game. To win the duel, the player needs to strategically use attack spells that will drain the opponent wizard's health and use shield spells to protect her self when being attacked by the opponent. Furthermore, the spells cost mana to use and thereby limiting the over-usage of spells and adds additional strategic elements to the game. An in-game screenshot can be seen in Figure 21.

Figure 21. In-game screenshot of the *Wiizards* game after the opponent has thrown a fireball at the player

4.2 Gameplay Details

The *Wiizards* game revolves around creating magic spells. The player creates spells by performing the magic gestures. There are three magic spells in the game. A defence spell that, when activated, will create a *bubble shield* that is active in a short period of time, but will deflect all attack spells thrown at the player. The two attack spells are a *fireball* attack and a *lighting* attack. The activated *lightning* hits the opponent instantaneously, but must be held active by *focusing* as a result of holding the magic wand completely steady, thereby gradually draining the opponent's health. The spell caster can *focus* the *lightning* for as long as she has mana left (the mana slowly drains over time while *focusing* the *lightning*). However, the one hit by *lightning* has the ability to *break out* of the *lightning* by furiously shaking the magic wand until the spell is broken. The *fireball* on the other hand costs less mana and cannot be broken out of, but is less harmful than the *lighting*. Furthermore the *fireball* has a traverse time giving the opportunity for the opponent to activate a shield spell.

Each player has a mana and a health bar showing the remaining amount of health and mana. Each spell has different mana costs and the available mana depicts what spells can be activated. When a spell is selected, it can be held for as long as the player wishes, until she gets hit by the opponent's attack, or until she wishes to *cast* it. Mana restores gradually by itself, only while the player does nothing and does not have any spells selected.

The gameplay is in real-time and therefore the mana plays an important role in balancing the intervals between throwing spells in the game.

4.3 Gesture Mechanics

As mentioned in the previous section, there are different magic spells that corresponds to a distinct movement patterns (gestures) performed with the Wii Remote. But to understand the gestures, it is imperative to have an understanding of the Wii Remotes axes orientation. In Figure 22 the orientation of the various axes can be seen.

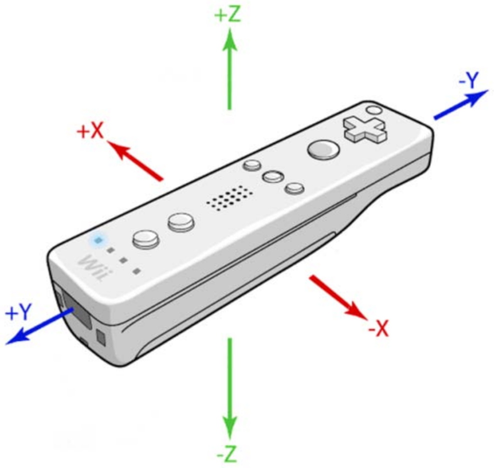


Figure 22. The orientation of the various axes for the Wii Remote

The various gestures have different movement patterns in the three dimensional space. However; the movement patterns are limited to two dimensions exclusively in order for the gestures to be uncomplicated for the player to remember and perform. Furthermore, all the gestures except the "cast" gesture are performed in the XZ-plane, which makes it easy to visualize for the player and conceptualize for the designer (as they can be shown in two-dimensions). In the figures below, the conceptual images of the gestures can be seen. These images are together with a short descriptive text presented to the players in the tutorial mode of the game.

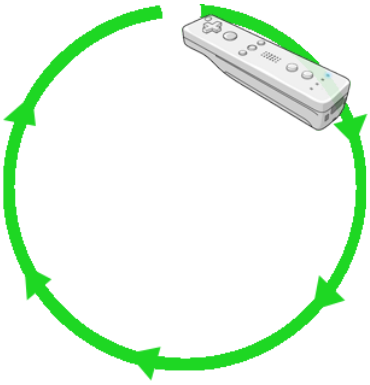


Figure 23. Defence gesture

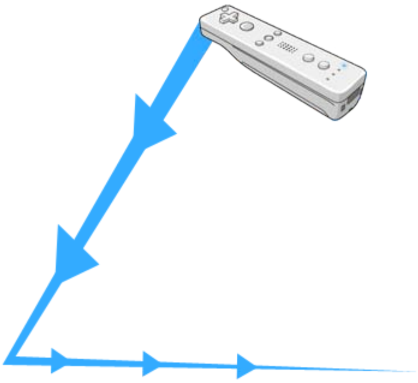


Figure 24. Attack 1 gesture

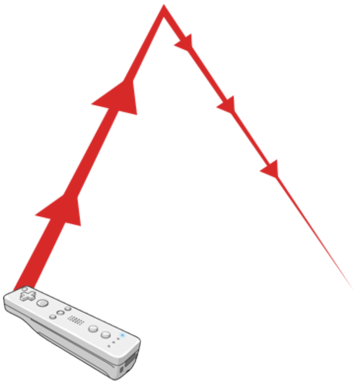


Figure 25. Attack 2 gesture

The defence gesture, which corresponds to the *shield* spell in the game, is performed by holding the Wii Remote straight at 12 o'clock and then moving it clock-wise, returning to 12 o'clock as seen in Figure 23.

The First attack gesture, which corresponds to the *lightning* spell in the game, is performed by starting with the Wii Remote at North-East and then moving it swiftly to South-West immediately followed by a movement to South-East as seen in Figure 24.

The second attack gesture, which corresponds to the *fireball* spell in the game, is performed by holding the Wii Remote in South-West, then moving to North and then down to South-East as seen in Figure 25.

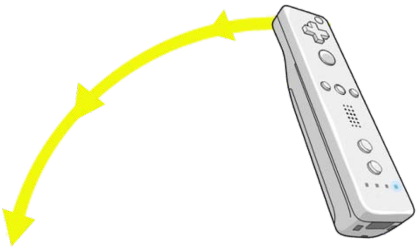


Figure 26. The flick gesture

Furthermore, it is mentioned that the player needs to activate the spell by "casting it". The cast gesture is done by making a "flicking" gesture as if one was to cast the line of a fishing pole. This gesture can be seen in Figure 26 and is utilizing the YZ-plane, as the motion is performed forward and downwards.

4.4 Game Factors Implementation

The purpose of implementing game factors that can be controlled in the game is two-fold. First, the game factors provide different game variations that can be played and compared to collect preference data. Second, when the emotion model is created, the controllable game factors can be changed based on the model's emotion prediction to optimize or minimize a given emotion. As stated in *Game Factors for Affecting Player Experience* (section 2.3), various game factors influence and engage the player differently in the game, for instance if the challenge in the game is too easy or too hard it can be a boring or a frustrating experience for her (Bateman, et al., 2006). Cognitive curiosity can engage a player in a game by presenting enough information in the game and thus she feels that her knowledge is incomplete and, therefore, she is interested to play and learn more of the game. Based on that research the two game factors – curiosity and challenge – were chosen as the controllable game factors in *Wiizards*. These factors were then quantified and normalized to the range [0, 1].

The first game factor, *curiosity*, depicts how much the player can predict the spells used by the opponent. This is achieved by implementing *action pairs* that will contain two spells that the opponent can select. The possible action pairs are all the pair-wise combinations of the spells *lightning*, *fireball*, and *shield*. An example of an action pair could be: $a_1 = \{shield, fireball\}$. When an action pair is selected by the opponent, both of the contained spells will be executed in turn before a new action pair can be selected. In this way, the player will have the chance to anticipate the next spell, if the same action pair has been executed before. The curiosity factor is representing the optimal entropy of the executed action pairs, where low curiosity represents low entropy and high curiosity represents high entropy. The entropy for each available action pair– if this action pair would be selected – is calculated. The action pair that matches the optimal entropy the best will be the one selected.

An example of an opponent's action sequence could be: given a low *curiosity factor* that would dictate that only two action pairs from all action pairs available could be selected, e.g. $a_1 = \{lightning, shield\}$ and $a_2 = \{fireball, fireball\}$. The player would be able to figure out the system as the resulting action sequence used by the opponent would be $\{lightning, shield, fireball, fireball\}$ and repeated until the end of the game.

To calculate the entropy the following equation is used:

$$H = -\sum_{i=1}^{n} p(x_i) * \log_{10}(p(x_i))$$

Where *n* is the number of action pairs, $p(x_i) = {}^{v_i}/_{V}$ is the number of times the specific pair has been used v_i out of the total number of executed action pairs, *V*.

The second game factor, *challenge*, is adjusting how fast the AI is playing. If the challenge is set to *high* the AI will play fast with small time spans in between spells and vice versa for *low* challenge making the AI slow and in turn make it easier for the player to attack the AI in between spells.

By using the two chosen game factors there can be created infinitely many variations of the test-bed game, which can create many different game experiences. As stated in the *Methodology* (section 3), the game factors were quantified in three steps: *low*, *medium*, and *high*. Using these three steps for the two game factors, the nine different game variants were generated from the test-bed game.

The test-bed game was developed and used in the user study to automatically generate all possible combinations of game variants and go through the combinations, such that each test subject played a new unplayed combination.

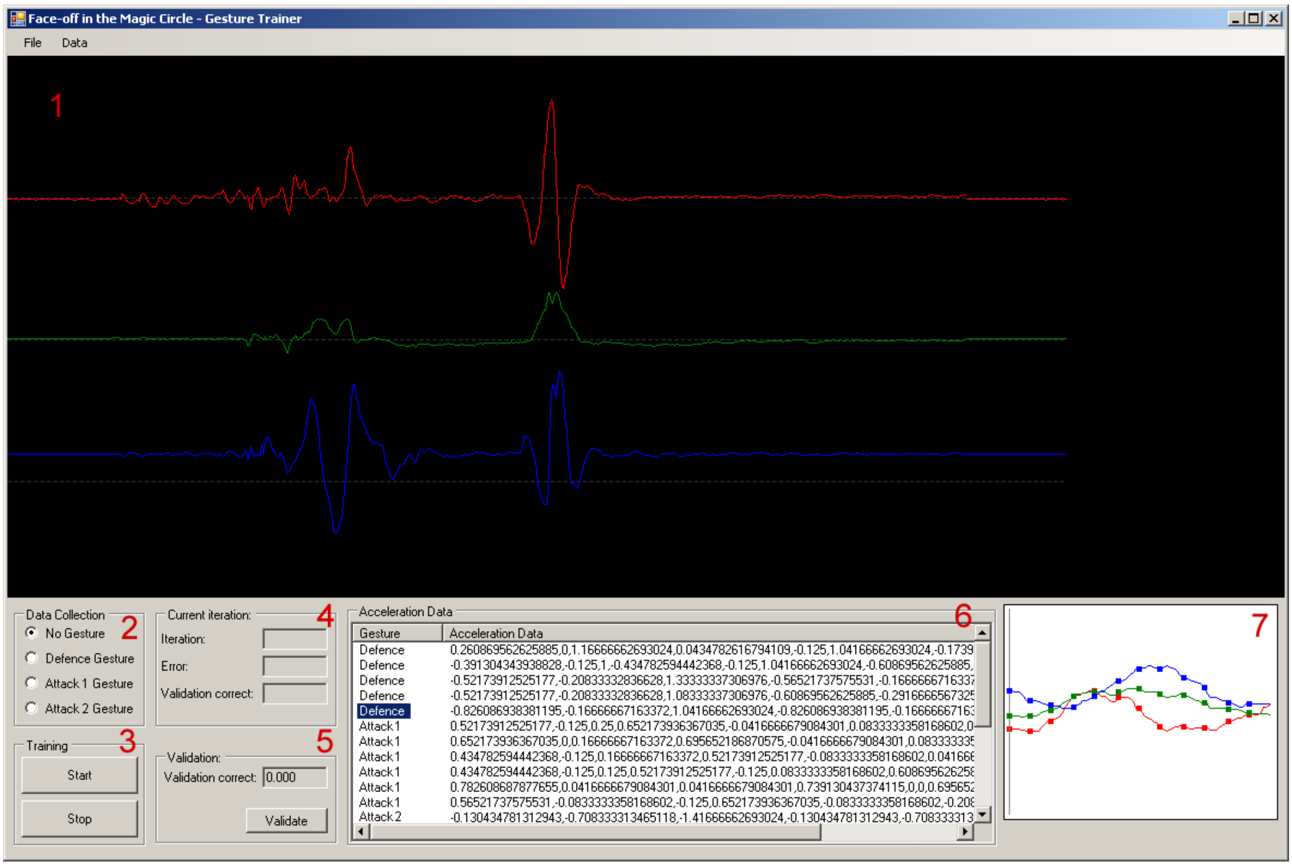
The game variants were logged to file together with all of the other data collected while the test subject played the game.

4.5 Gesture Recognition using ANN

To infer emotions from body movements, we stated in *Inferring Emotions from Physical Video Games* (section 2.2.4.1) that we needed to recognize gestures. By recognizing 3D dynamic gestures we obtain a great deal of spatial information that can be used in our emotion model learning algorithm. Approaches for gesture recognition by applying computer vision have been studied. However studies in gesture recognition with the Wii Remote are sparse. Two examples of a Wii Remote gesture recognizer is the *Wiigee* library (Schlömer, et al., 2008) and the commercially available product *LiveMove* (AiLive, 2006), however the gesture recognition algorithm in this study is followed from (Wiggins, 2008) that uses an ANN with back-propagation which achieves higher classification accuracy than the *Wiigee* library and is not proprietary like *LiveMove*.

4.5.1 Method

As mentioned in *Gesture Mechanics* (section 4.3), three gestures were needed to be learned; one defence gesture and two attack gestures. A gesture trainer program was developed to collect Wii Remote data and it eventually creates and trains a gesture recognizer. The program was created to relieve the work of eventual gesture scheme changes. Furthermore, having a program that could collect and save training data reduced future work when new approaches were tried.



Data Collection
● No Gesture
○ Defence Gesture
○ Attack 1 Gesture
○ Attack 2 Gesture

Validation correct: 0.000

Gesture	Acceleration Data
Defence	0.260869562625885,0,1.16666662693024,0.0434782616794109,-0.125,1.04166662693024,-0.1739
Defence	-0.391304343938828,-0.125,1,-0.434782594442368,-0.125,1.04166662693024,-0.608695626258 85,
Defence	-0.5217391252 5177,-0.20833332836628,1.33333337306976,-0.565217375755 31,-0.1666666716337
Defence	-0.5217391252 5177,-0.20833332836628,1.08333337306976,-0.608695626258 85,-0.291666656732
Defence	-0.826086938381195,-0.16666667163372,1.04166662693024,-0.826086938381195,-0.1666666716
Attack1	0.5217391252 5177,-0.125,0.25,0.652173936367035,-0.0416666679084301,0.0833333358168602,0
Attack1	0.652173936367035,0,0.16666667163372,0.695652186870575,-0.0416666679084301,0.08333333
Attack1	0.434782594442368,-0.125,0.16666667163372,0.5217391252 5177,-0.0833333358168602,0.04166
Attack1	0.434782594442368,-0.125,0.125,0.5217391252 5177,-0.125,0.0833333358168602,0.60869562625
Attack1	0.782608687877655,0.0416666679084301,0.0416666679084301,0.739130437374115,0,0,0.69565
Attack1	0.565217375755 31,-0.0833333358168602,-0.125,0.652173936367035,-0.0833333358168602,-0.20
Attack2	-0.130434781312943,-0.708333313465118,-1.41666662693024,-0.130434781312943,-0.708333313

Figure 27. Gesture trainer program

The gesture trainer program can be seen in Figure 27, where a continuous stream of x, y, and z acceleration values are shown. The display, seen in *1* in Figure 27 , works like the cardiac monitors seen in hospitals, where a continuous stream, loops to the beginning after it reaches the end and then overwrites the old values. Raw data is collected for the training algorithm by choosing the gesture to collect data for in *2* and then perform the gesture while pressing and holding the B-button on the Wii Remote. When the gesture has been performed, the B-button is released and the data is saved to a list and presented in the *Acceleration Data* list as seen in *6*. The collected acceleration values for the gestures can be examined by selecting it in the *Acceleration Data* list; where after the data and the sampling (more on this in the Data Preparation) of the data is presented as a graph in the graph display seen in *7*. When enough data has been collected for each gesture the training data can either be saved to a text file for later use, or the network training can be commenced by pressing the start button in *3*. While the training algorithm is running, the basic training information, i.e. current iteration and current training error, is presented to user. The training data is divided into two sets; a training set (2/3) and a validation set (1/3). While the network is training, the validation accuracy on the unseen validation set is presented to the user in *4*. Whenever a network is trained, it can be validated on new data (if new is added to the list) or on the entire list used for training by pressing the validate button in *5.*

For constructing the gesture recognizer that can classify gestures based on Wii Remote acceleration data, a multilayered ANN was implemented. Gestures of arbitrary length were collected for later processing.

4.5.2 Data Preparation

As presented in *Inferring Emotions from Physical Video Games* (section 2.2.4.1), to optimize the player's affective experience the game controller needs to afford body movement that are natural to the game scenario. The purpose of the gesture recognizer was, therefore, to have an interface to the game that felt intuitive, unobtrusive, and natural, when playing the game i.e. engaging in a magic duel. As seen in wizard duels in movies or other fictional work, the wizards are doing rapid and continuous movements with their wand to create spells. This is the behaviour that is implemented in this study by allowing continuous and real-time recognition of gestures. However, this introduces the problem of users varying the speed and duration of the gesture. The defence gesture, which is a circular movement, could be done in a quick movement, where only the wrist is involved, or big and powerful movement, where the shoulder is used. To recognize affect, the differentiation between these two different movement strategies is wanted. However as a pattern recognition problem this introduces the question of the input signal window size.

For the training, the data was collected by pressing and releasing the B-button on the Wii Remote, indicating the start and the end of the gesture. This made it easy for the algorithm to "guess" the duration (start and end) as these were clearly marked. However, when the actual real time recognition was done, no such indications were made. The model would receive a continuous stream of data and should then realize when a gesture was made. A proposal to do this is to have a threshold on movement that would mark the beginning and the end of a gesture capture. However, because of the nature of the Wii Remote, this is not possible since motion in the same direction of gravity would induce "free fall"-like state and thereby give zero in acceleration. This would effectively end the gesture capture, since there would be no more movement even though this state could be in the middle of a gesture. This issue has also been addressed in (Boehm, et al., 1994) where the solution was to have different intervals that should be considered. In our case the input buffer was divided into 3 intervals and then the recognizer would continuously try to recognize the gestures based on these intervals.

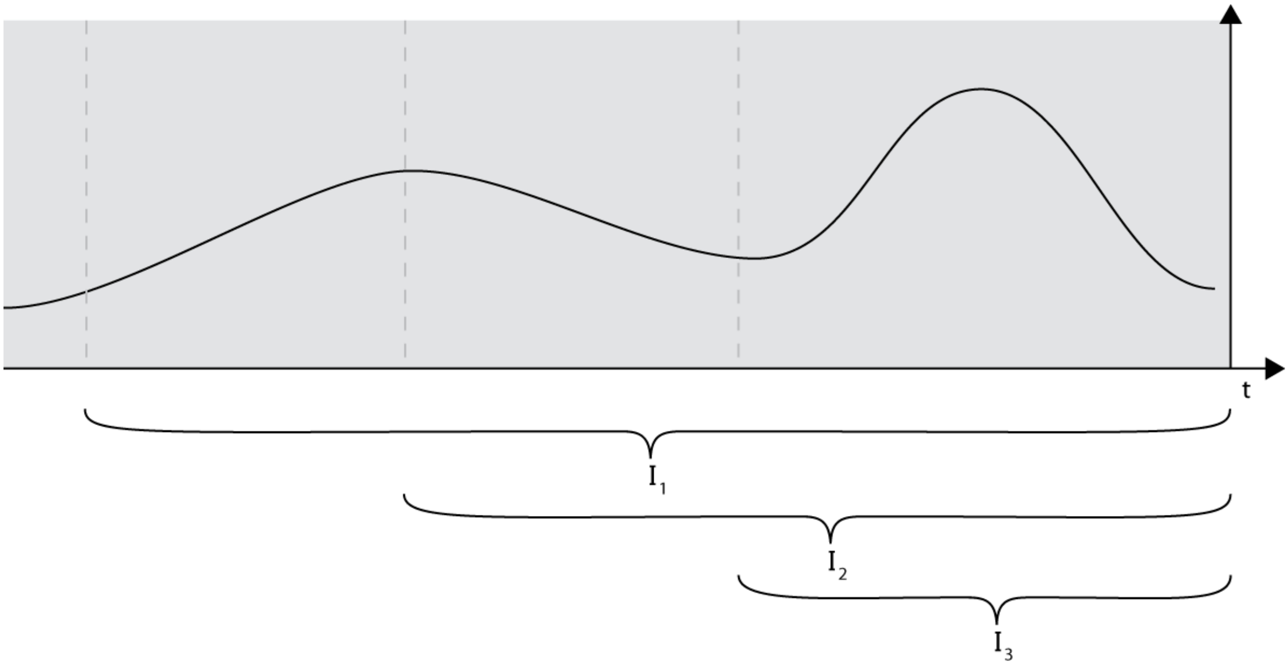


Figure 28. Wii Remote gesture recognition buffer. Wii Remote signal shown in the top used to fill three buffers of varying size (I_1, I_2, and I_3) (Boehm, et al., 1994).

In Figure 28 a snapshot of a continuous stream can be seen. The signal is cut into three different intervals and is then passed to the sampling algorithm three times; once for each interval. The first interval is regarding the previous 100 frames, the second interval regards the previous 75 frames, and the third regards the previous 50 frames. The signal would then look very different dependent on the interval length. This can be seen in Figure 29, where the three different intervals can be seen.

The sampling method would then sample the given signal at 12 equally spaced points for each axis and in turn the samples would be scaled to [0.0; 1.0]. The 12 samples for each axis would together with the length of the gesture be used as the input for the network.

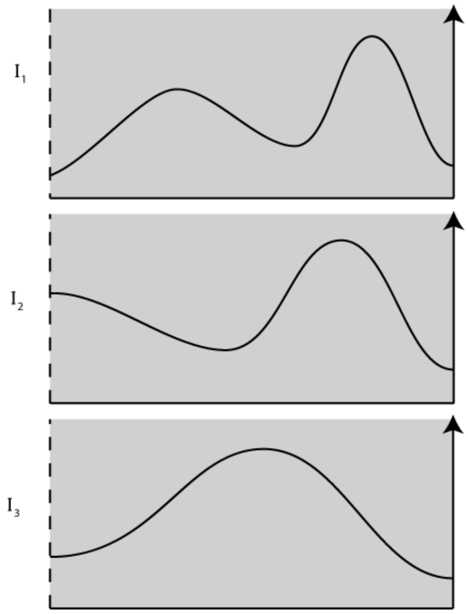


Figure 29. Results of dividing the Wii Remote signal into three buffers of different size (Boehm, et al., 1994)

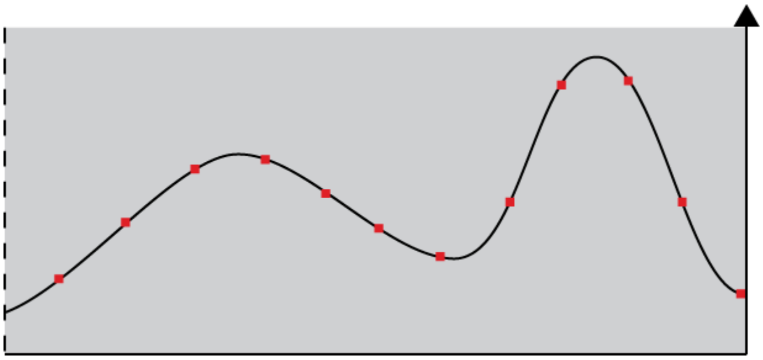


Figure 30. Twelve samples on a given input signal

4.5.3 Network Architecture

The ANN was constructed with one hidden layer containing 12 hidden neurons. The input vector consisted of 37 inputs: 12 for each axis and one extra for the gesture duration. The output layer consisted of three neurons; one for each gesture it had to classify. For the activation function, the popular sigmoid function was utilized.

4.5.4 Training

Because of the complex and dynamic gestures, this classification problem needed a large training data set (in our case 120 samples). To get a more generalizing model, noise was injected into the training set as “non gestures”. The training data was divided into two sets, a training set and validation sets. The training was done using back-propagation and was terminated when the training had a sufficiently low training error or when the validation accuracy on the unseen data began to drop, which corresponds to the early termination technique used in network training to avoid over-fitting (Gurney, 1997). The results obtained from the training were very promising. The trained ANN had a success of 87%.

5 USER STUDY

A user study was conducted in order to gather the data required for training the emotion model. The user study included a subjective self report questionnaire with the purpose to collect emotion data from the test subjects. In the following sections the experiment procedure, experiment setup, and all relevant user study details are presented.

5.1 Emotion Questionnaire Design

The first step in the user study was to design an appropriate format for the questionnaire, so that the test subjects' self reports about the emotions experienced in the game can be used for training the emotion model.

Inspired by the experiment in (Yannakakis, et al., 2007c), this study uses the 4-AFC questionnaire format, with the purpose to provide standard emotion labels for the test subjects and obtain their emotion preference for the two games. Humans' definition of emotions and the emotions' intensity is subjective and therefore hard to generalize (see *Inferring Emotions from Subjective Self Reports* (section 2.2.1)). The 4-AFC format was chosen to minimize the assumption of the test subjects' definition of emotions and the emotions' intensity to provide reasonable comparison between the answers from the different test subjects. Another reason for choosing the 4-AFC format was because the study in (Yannakakis, et al., 2007c) propose that the alternative forced choice format is preferred over other approaches (e.g. ranking format) when developing an emotion model that relates the test subject's emotion preferences to physiological features.

The questionnaire provided the following alternatives to choose from (where *X* is a variable for the type of emotion):

a) Was Game A more *X* than Game B
b) Was Game B more *X* than Game A
c) Both Game A and Game B are equally *X*
d) Neither Game A or Game B is *X*

The chosen alternatives a) and b) were considered valid data and were used as training data for the emotion model. However, the alternatives c) and b) were given as a choice because of the following two reasons:

- To give more expressive freedom to the test subject. This was done in order to reduce noise in the questionnaire data because as it is already seen in *Inferring Emotions from Subjective Self Reports* (section 2.2.1), if the test subject is forced to select which game elicited the highest level of a given emotion, then the result could be less accurate
- To point out "problems" in the game design. In this context game design "problems" refers to the fact that the variations of the test-bed games are not distinct enough from each other or poorly designed and therefore, the test subject might not have an emotion preference between the two given games

A concrete example of an answered questionnaire can be seen in Table 4. However, the final version of the questionnaire was presented in a digital form and integrated in the test-bed game with a purpose to rule out any biasing introduced from an interviewer (Mandryk, et al., 2006a).

Table 4. An example of answered 4-AFC questionnaire (the test subject has to put a checkmark on the appropriate answer)

Question	Game A	Game B	Both equally	Neither
Which game was more fun	✓			
Which game was more relaxing		✓		
Which game was more frustrating				✓
Which game was more exciting			✓	
Which game was more boring				✓

Further on, another questionnaire was designed with a purpose to evaluate the test subject's general feeling of emotions and level of "fun" in the test-bed game. This questionnaire included the ranking self report format, explained in *Inferring Emotions from Subjective Self Reports* (section 2.2.1), which was used for the test subjects to rank the emotions and the "fun" experienced after playing each game on a scale from 1 to 5 (see Table 4). The results from this questionnaire and the evaluation of the overall test subjects' game experience are discussed in *User Study Evaluation* (section 5.5).

Table 5. An example of an answered ranking questionnaire (the test subject had to put a checkmark on the appropriate answer)

Question	not at all – 1	slightly – 2	moderately – 3	fairly – 4	extremely – 5
I was having fun			✓		
I felt relaxed		✓			
I felt frustration	✓				
I felt excitement		✓			
I felt bored	✓				

5.2 Experiment Procedure

In this section, details about the experiment procedure are discussed. The experiment was designed with the purpose to optimize the input for the machine learning algorithm, as well as to minimize the test subject's stress and discomfort, which could give noisy and incorrect physiological data (Picard, 1997). To get consistent and accurate data, each of the test subjects followed the same procedure for the experiment. In general, the test subjects played two game pairs i.e. four games in total. However, few of the test subjects played four game pairs. The experiment procedure was performed in nine steps:

1. *Demographic questionnaire* – the test subject was asked to answer a demographic questionnaire that can be found in Appendix A – Demographic Questionnaire.
2. *Setting up physiology sensors* – the sensors were attached on the finger tips of the test subject's inactive hand and were activated to record the subject's physiological activity during play.
3. *Tutorial game* – Before playing the test-bed game the test subject had to complete a tutorial, which started with a teaching session on how to perform magic spells in the game, followed by a 90 seconds practice game in the same form as the actual test game.
4. *One minute rest* –The test subject had to relax for one minute before playing each game.
5. *Play Game A* – The test subject was asked to press a button on the Wii Remote to start the game. In this way, the test subject would not get stressed to begin the game. The total playing time was limited to 90 seconds for all test subjects.
6. *Rate Game A* – The test subject was asked to complete the rating questionnaire that can be found in Appendix B – 4-AFC and rating questionnaires.
7. *One minute rest* – This step is the same as step 4.
8. *Play Game B* – This step was the same as step 5 except that the test subject was playing another variation of the test-bed game, than Game A.
9. *Rate Game B* – The test subject was asked to complete the rating questionnaire that can be found in Appendix B – 4-AFC and rating questionnaires.
10. *Compare Game A and B* – The test subject was asked to complete the emotion questionnaire that can be found in Appendix B.
11. *Repeat step 4-8* – with new game variations.

In *step 1*, the test subject was introduced to the experiment observer – which was one of the authors of this study – and was asked to answer an anonymous demographic questionnaire. The demographic questionnaire was designed to keep track of the test subject's age, physical condition and gamer profile, since the research in *Inferring Emotions from Physiological Response* (section 2.2.2) suggested that the physiology data can vary depending on these attributes. Therefore, the aim was to perform the experiment on a group of test subjects that have similar age and life style.

In *step 2*, the observer helped the test subject to attach the physiology sensors on the hand, which remained inactive during game play. The rest of the needed data – motion acceleration data from the Wii Remote and the test subject's interaction data from the game – were gathered by game log files that were recorded during play.

Further on, in *step 3*, the test subject went through tutorial instructions about how to play the game and played one training game. This was done because *Wiizards* is not intuitive like other *Wii* games (tennis, bowling, etc.) and requires players to learn and to remember the specific gestures before they could play the game at all. The experiment observer was present in the experiment room at this stage and assisted the test subject in the learning of the different gestures. However, when the training session was over the observer left the room leaving the test subject alone to go through the rest of the procedure in order to keep the experimental effect at a minimum (Mandryk, et al., 2006a).

Measuring emotions from physiology efficiently requires that the test subject's physiological signals are at baseline level when she starts playing the test-bed game. Therefore, in *step 4* and *step 6*, the test subject was instructed to sit down and relax for one minute before playing the next game. Based on the research done in (Mandryk, et al., 2006b) and (Yannakakis, et al., 2007c), one minute was found to be enough to allow the physiological signals to return to their baseline level.

In *step 5* and *step 7*, the test subject played two different variations of the test-bed game. At that time the test subject was alone in a closed room concentrating only on the game without being interrupted. The study in (Yannakakis, et al., 2007c), did an experiment to test if the order of the games played has a significant effect on the player experience. The result from their experiment did not find any significant order effect; therefore, the experiment in this study was designed with the assumption that there will be no order effect. In order to collect acceptable amount of preference data for training the emotion model, all nine test-bed game variants had to be compared with each other at least once. Applying the following combination formula (which does not consider the order effect and where n is the number of game variations and r represents the number of games in one combination i.e. pair):

$$\frac{n!}{r!\,(n-r)!}$$

It was estimated that the acceptable number of preference data is 36. This means that, if each test subject played one test-bed game pair, a minimum of 36 test subjects were needed. However, considering the nature of the 4-AFC questionnaire format, as explained earlier in this chapter, if the test subject could not prefer one game over the other for a given emotion, the test subject's preference data could not be used in training the emotion model. To overcome a potential low number of valid preference pairs, a choice was made to double the amount of preference answers by asking each test subject to repeat the experiment from *step 4* to *step 8*. In this way 36 test subjects would generate 72 preference data in total.

5.3 Experiment Setup

The experiment was performed in a room at Virum Skole – a public school located in Virum, north of Copenhagen. The room was customized for the experiment purpose. The test subjects were pupils who were familiar with the school environment and the chosen experiment room. To keep the situation as neutral as possible, each test subject was alone in the room while playing the test-bed games to rule out the feeling of being watched. This particular room and set up was chosen to avoid the impact of unfamiliar surroundings and to keep discomfort for the test subjects to a minimum, thus their physiology and self reported data quality is increased (Yannakakis, et al., 2007b).

In order to make the game experience complete the game was set up on a projector that projected the game on a white wall, and speakers for the audio. The experiment room was dimly lit to make the projected image as clear as possible without making the surrounding imperceptible.

The physiology sensors attached on the test subjects that was used to track their physiological responses was the *Infiniti ProComp* encoder (Technology, 2009) with SC and BVP biofeedback. The encoder was equipped with *Tele-Infiniti* (Technology, 2009) – a wireless component that uses Bluetooth to connect to a PC – providing full freedom of movement for the test subjects.

Figure 31 represents a visual example of the experiment room setup.

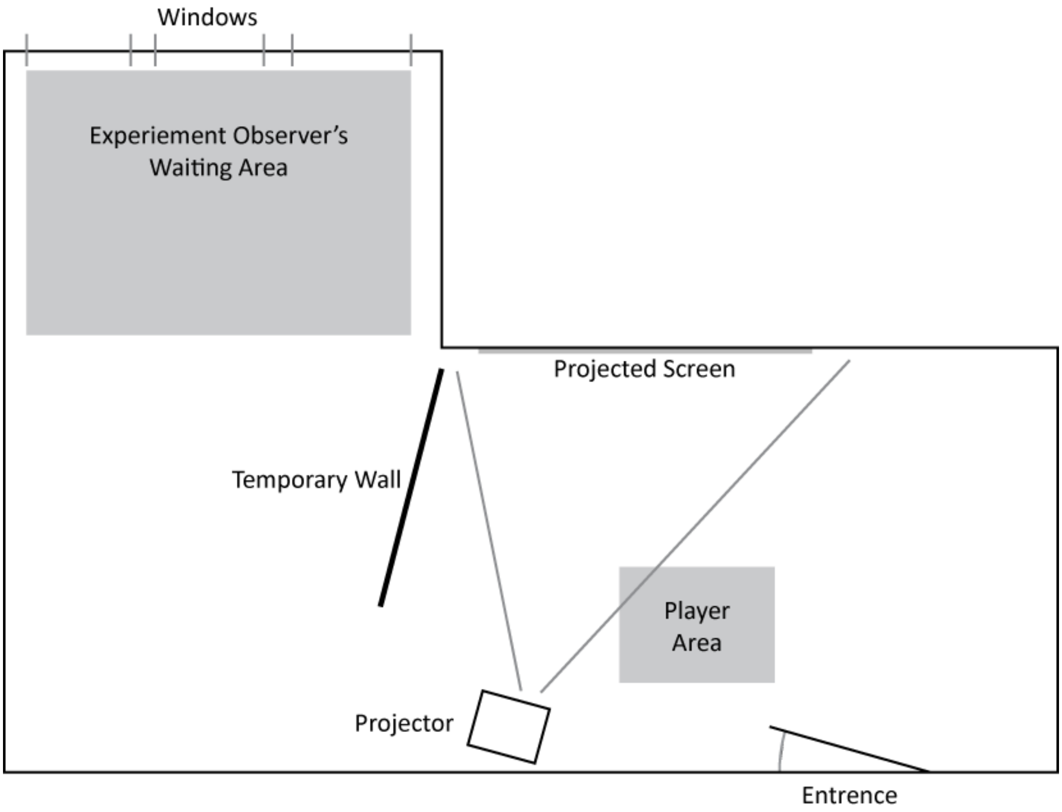


Figure 31. Test environment at Virum Skole

5.4 Test Subjects

The user study was performed on teenagers with similar demographical data. Since their first language was Danish the questionnaires discussed in *Emotion Questionnaire Design* (section 5.1) were translated into Danish and integrated into the game. In Appendix A – Demographic Questionnaire and Appendix B – 4-AFC and rating questionnaires, screen shots from the implemented final questionnaire screens can be found.

The preliminary estimate was to perform the experiment on 36 test subjects in order to collect 72 emotion preference data. However, because of various issues this number was not reached. The total number of test subjects reached was 29 test subjects where four were allowed to perform the experiment twice. This resulted in a total number of 66 game preference comparisons for each of the investigated emotions. In (Picard, et al., 2001), it was found that there can be great differences in physiological signals from day-to-day when the same emotion is experienced. This is not the case when experiments are performed in the same day. Therefore, in order to get valid data a single test subject was tested only the same day.

From the 29 test subjects, 35% were girls and the remaining 65% were boys and all pupils were from 7th and 9th grade. Their age ranged from 13 to 16 years (the average age was 14.5 with the standard deviation $\sigma = 1.24$). The experiment was performed during their normal school hours and took between 20-30 minutes per person in total.

From the demographic data, it was assessed that all the test subjects were physically active in their daily life by doing sports or fitness 2-3 times a week. Their average BMI was 20 ($\sigma = 2.4$), which according to the (NCHS, 2000) for children between 13 and 16 is considered normal weight (see Figure 32).

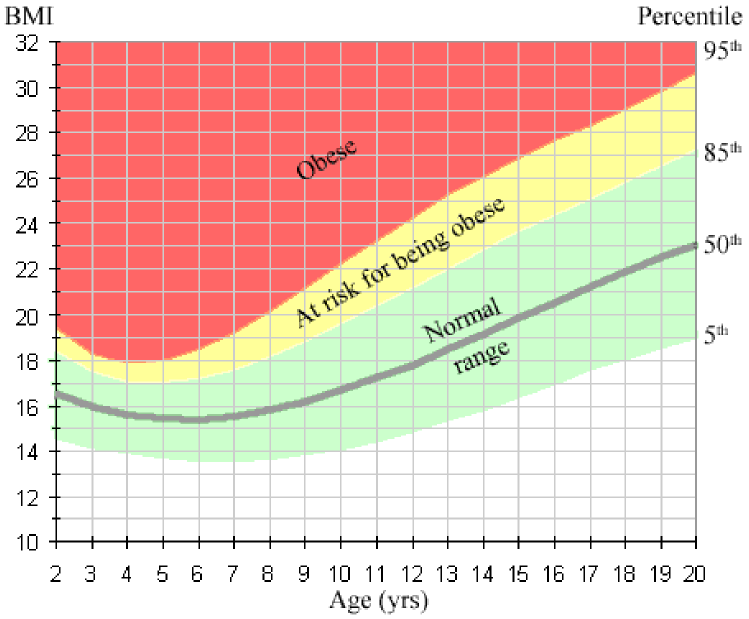


Figure 32. Body mass index (BMI) for boys, ages 2 to 20 (Health, 2008)

5.5 User Study Evaluation

As mentioned in *Emotion Questionnaire Design* (section 5.1), a self report questionnaire for ranking emotions and “fun” was designed to evaluate the test subject’s general feeling of emotions and level of “fun” in the test-bed game. The reason to develop the questionnaire was two-fold: 1) to get an indication of the test subject’s preference consistency, and 2) to evaluate of the game design of the test-bed game.

The game design need to be evaluated because in perfect conditions all game variations can evoke all levels of emotions uniformly through their gameplay. This is in order for the emotion model learning to have a more complete affective representation to generalize over. The other reason for including the rating questionnaire was to check the test subject’s preference consistency. A test subject’s preference is said to be consistent when the test subject’s rating for Game A and Game B corresponds to the preference i.e. when Game A is chosen as the most relaxing compared to Game B, the rating should also have higher rating for relaxation in Game A than in Game B. However, when a preference was chosen and the ratings for the two games were equal, the preference was still considered consistent, because the games could have a minuscule difference in the level of felt emotion (e.g. the difference between 4.2 and 4.1, would not be apparent in the rating). As stated in *Emotion Questionnaire Design* (section 5.1), preferences were considered invalid when the test subject’s choice was “Both Game A and Game B is equally X” or “Neither Game A or Game B is X”. This meant that many pairs were disregarded. However, as stated in (Yannakakis, et al., 2007) the 2-AFC could introduce noise in the form of player inconsistency. For example, if there is no clearly perceived difference in games played and a choice is still forced, the data could be random. Therefore, introducing 4-AFC eliminates much data that otherwise might not have been useful. The total amount of test pairs played was 66, but because of the elimination of the invalid data, this number decreased to: 37, 35, 28, 30, and 35 for relaxation, frustration, excitement, boredom, and fun respectively.

As mentioned in *Experiment Procedure* (section 5.2), nine game variants of the test-bed game were created. A test subject would play two game variants to compare these. It was also stated that all pair-wise combinations should be played at least once, meaning that 36 game pairs were created. Figure 33 illustrates all the game variants and their pair-wise combinations. Furthermore, the game pairs that only have one game factor that vary are highlighted in red.

To get an indication of the validity of the game factors, the following test hypothesis was formulated: “the game variants do not have clear discernable differences”. To reject the hypothesis, the test data was divided into two sets; valid and invalid sets. The invalid set was then investigated to see if the change of one of the game factors in the second game had no impact on test subject’s ability to express their emotion preference. This was done by inspecting the game pairs where either *curiosity* or *challenge* did not vary in both games (red lines in Figure 33).

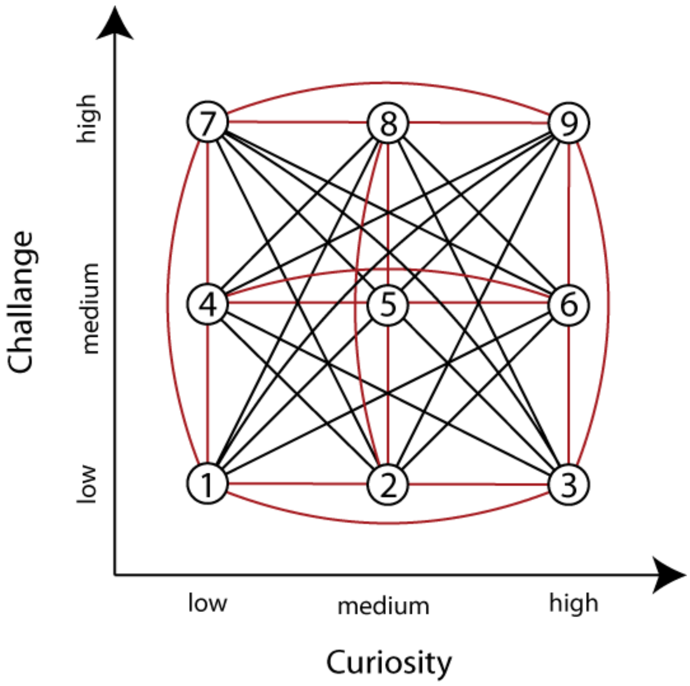


Figure 33. All the game variants (circles) and all the pair-wise combinations (lines). Game pairs only varying in one game factor is connected with red lines and game pairs varying in both game factors connected with black lines.

The result of this investigation has been summed in Figure 34. As it can be seen in Figure 33, half of the game pair combinations have only one game factor changing from Game A compared to Game B; these are referred to as *Single Game Factor*. The same distribution should be apparent in the pairs of invalid preferences (which are the black and the red segments in the pie charts in Figure 34) in order to conclude that challenge and curiosity variants are discernable. The observation in Figure 34 shows that all of the emotions except *Frustration* fulfil this requirement. A small difference can be seen in the pie chart for *Frustration* that has 7.6 % more invalid preferences in the *Single Game Factor* pairs than in the *Double Game Factor* pairs. To investigate, if one of the game factors had a flaw in the design, the *Invalid Single Game Factor* pairs were further analysed. The *Single Game Factor* pairs were then divided into each game factor i.e. curiosity and challenge (horizontal or vertical red lines in Figure 33), and the same process was applied. In order to conclude if the challenge and the curiosity factors are successfully designed for evoking different player experiences the number of invalid preferences should be equal for each factor. The test showed that there was an equal amount of invalid data for each game factor. This means that the test does not show any perceivable problems with the design of the game factors.

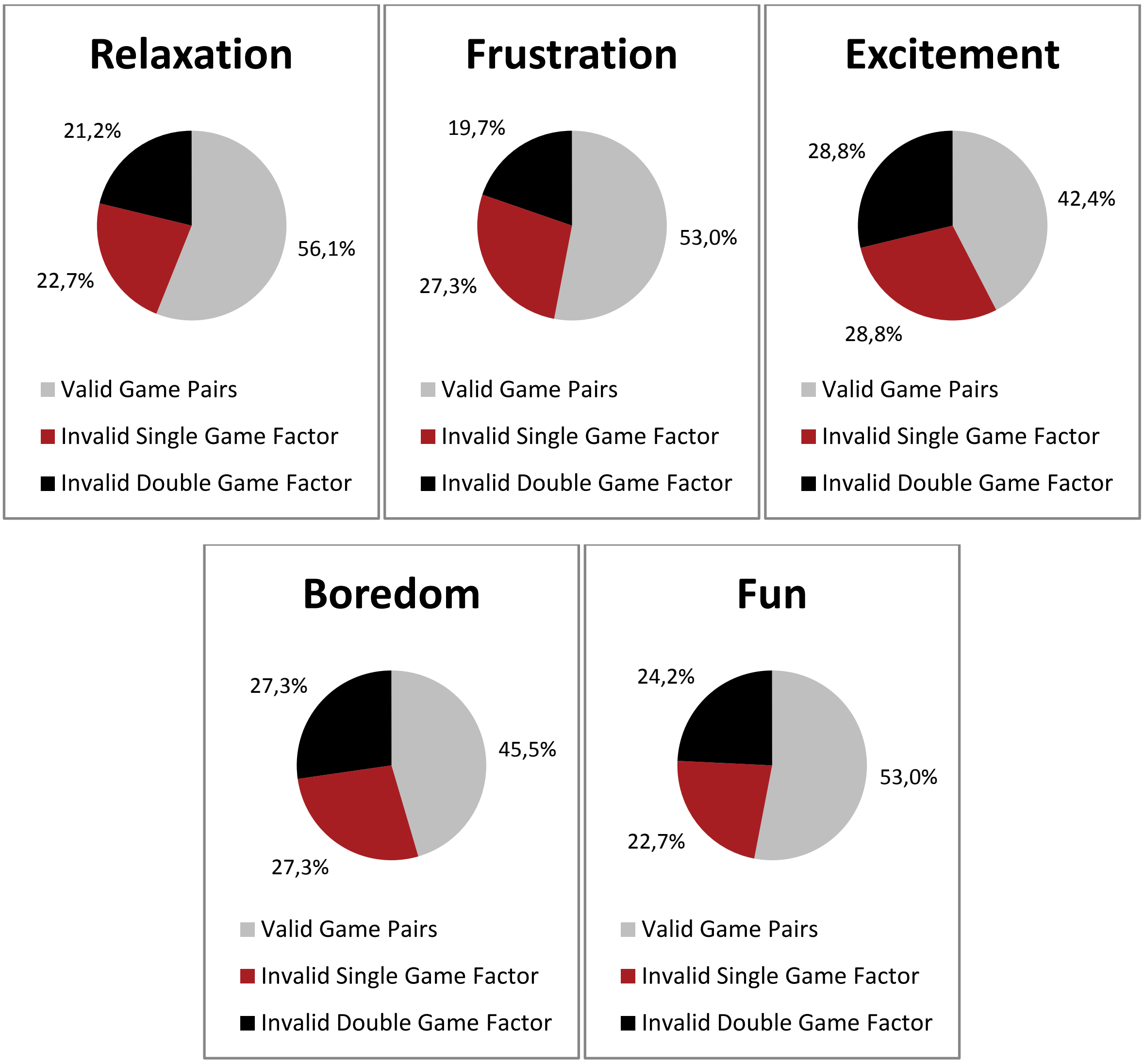


Figure 34. Pie charts showing the percentage of valid games compared to invalid. Invalid games are divided into game pairs with only one game factor changing (Invalid Single Game Factor) and with both game factors changing (Invalid Double Game Factor)

From Figure 34, it can also be seen that only approximately 50% of the test subjects expressed valid preferences. This is a rather low amount and limits the number of game pairs that could be used for training. However, since the invalid data would have been included if the 2-AFC method in contrast to the 4-AFC method, it proved to be a good decision to follow the 4-AFC method, where otherwise much randomness would have been introduced.

Another reason, for the test subject's to express ratings for each emotion was to get an evaluation of the game design. To achieve this, histograms of the rankings were created and can be seen in Figure 35. Here it can be seen that the game did not induce much frustration or excitement, where the game scored low rating, with an average of 2.125 (σ = 0.94) and 1.6 (σ = 0.98) out of 5. Optimally the game would induce all levels of each emotion, meaning that the standard deviation would be high, however as seen in the figure, this is only the case in relaxation (average = 2.78, σ = 1.07) and boredom (average = 2.15, σ = 1.14). However, the game scored a decent average score on "fun" with 2.83 (σ = 0.92). However, it is apparent from the figures, that the game did not induce much excitement and frustration and that the game design might be flawed in that sense. A preliminary test was done before the user study to ensure that the test subjects could differentiate the various game variations; this test however did not present any problems, which could be because the game was not tested on the correct target group. The reason for this could be because the preliminary test was executed on students from the IT University.

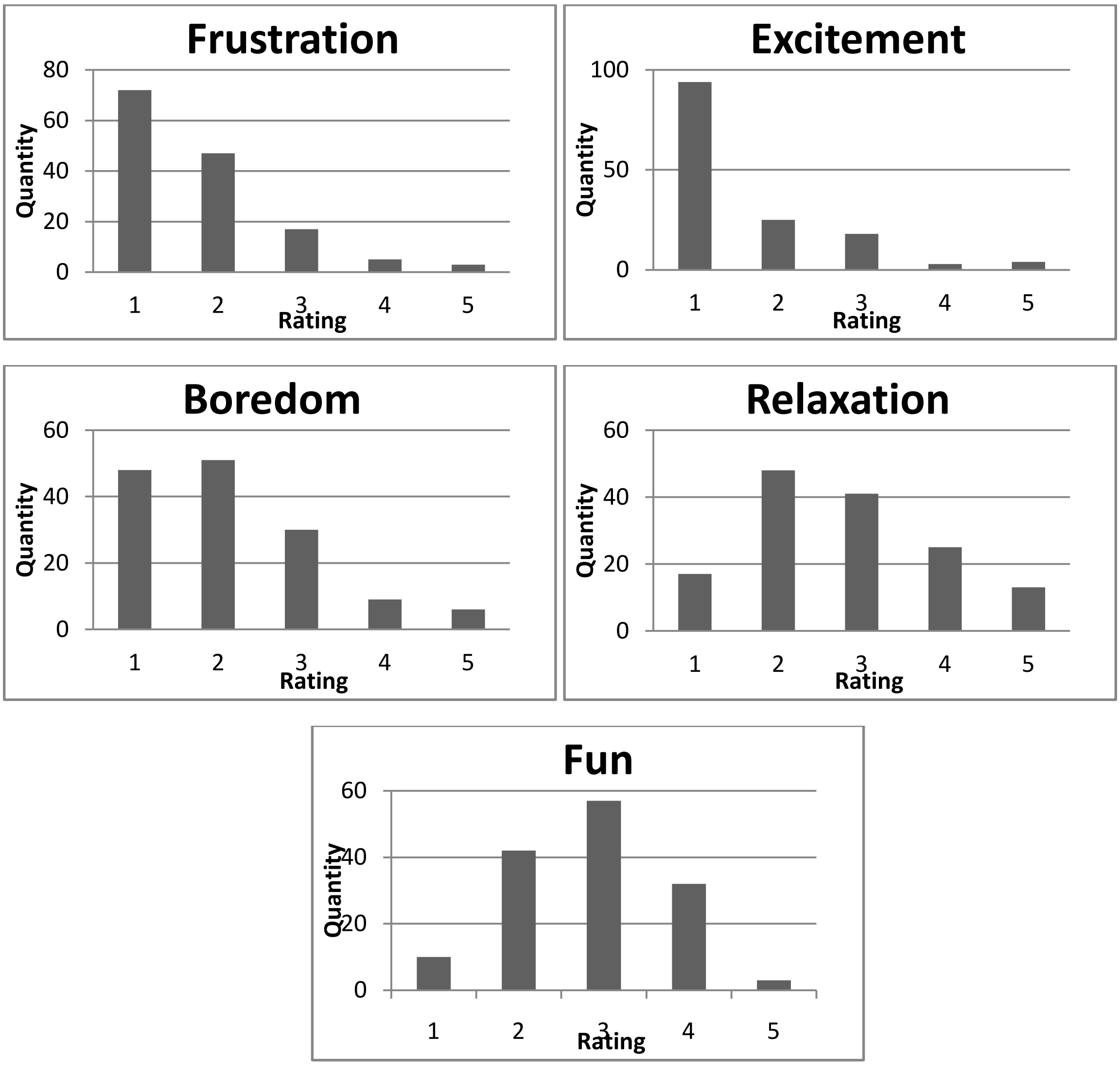


Figure 35. Histogram of emotion rating collected after playing each test game

6 IMPLEMENTATION OF THE EMOTION MODEL

The emotion model in this study is based on input from physiology sensors, the Wii Remote, and game events. Before they can be used for building the model, the physiology signals has to be noise reduced and have features calculated that represent specific features. In the following sections, firstly signal processing methods will be presented to reduce noise. Secondly the signal feature equations will be presented.

6.1 Data Acquisition

This section discusses the feature extraction process of the BVP, SC, Wii acceleration input signals and player interaction data. Further on, a statistical analysis of the calculated features is presented to investigate the significance of the given features.

6.1.1 Signal Processing

Even with the most carefully planned and well executed user study, it is nearly impossible to avoid all noise in the physiology input signals (Picard, 1997). When dealing with human test subjects and physiological measurements there will almost always be introduced some noise to the input data. In this section detail on what methods and algorithms were used in this study to remove any noise from the input signals is presented.

6.1.1.1 Blood Volume Pulse Signal Processing

The player's HR response was calculated from the BVP signal obtained from the PPG sensor. The PPG sensor measures how much light is absorbed in the finger from the light produced by the sensor. Noise can be introduced when the test subject moves her hand attached with sensors too much or when she clenches her fist. This is because additional light from the surrounding is let into the thumb beneath the sensor. Therefore, in the user study setup the lighting of the experiment environment was optimized for the PPG sensor by having the milieu as dark as possible to make sure to reduce the external light coming into the sensor.

The raw signals from the Infiniti encoder, recorded through the wireless Tele-Infiniti hardware was encoded in an undocumented format that needed to be decoded (see *Appendix C – Infiniti Encoder through the Tele-Infiniti* for more information). The signal was successfully decoded, but the units were unable to be converted without having more knowledge of the sensors – which were unfortunately not available. Because it was impossible to convert the units into the standard BVP units, mmHg and the standard SC units, micro-Siemens, the values in this study will be in the range of [500; -3000] for BVP and [0; -3500] for SC. This is why the values on the Y-axis on the various graphs in this study are different from what is expected. The assumption is made that as long as the informal units is used both for training and for future prediction of physiology, there will be no problem as the ANN is unaware of any units and is only concerned about relations between the input values. Furthermore, the signals is not tried to be interpreted with an existing system further reducing the need of converting the signals to formal units.

The raw decoded BVP signal received from the encoder is shown in Figure 37. As seen on this figure the signal is over-all rather good and stable, but around the 50 sec mark some noise can be seen in the form of extreme values, this is the first thing that needed to be removed before calculating BVP related features. The algorithm used to remove peaks follows this procedure:

1. looks at the detected beat,
2. calculates the average of the four surrounding beats' amplitude in the BVP signal, the two preceding beats and the two subsequent beats (see Figure 36),
3. If the difference between the value in the BVP signal for the current beat and the calculated average value differs more than a given amount (500 unites was found to be a good threshold), then
4. the average value is used to cut off the peak.

An example of the noise reduced BVP signal from Figure 37 is shown in Figure 38.

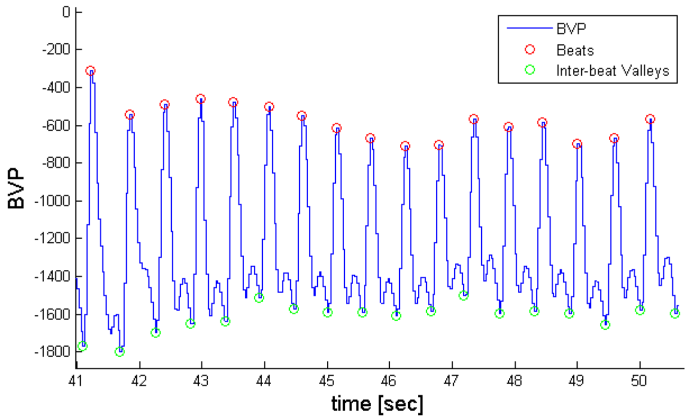


Figure 36. Image representation of the noise removal procedure. The green circle marks the detected peak to be processed and the read circles mark the surrounding peaks

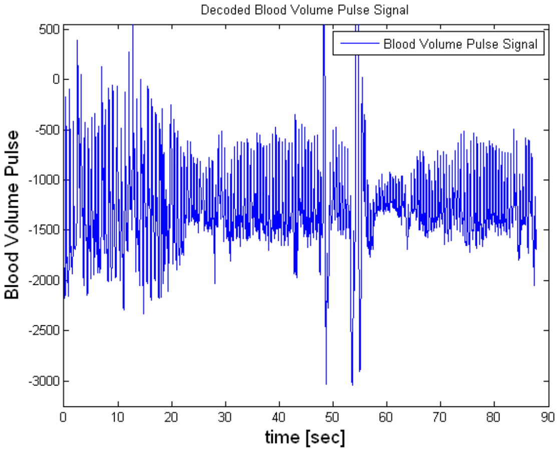


Figure 37. Decoded BVP signal received from the Infiniti encoder. The signal is still very noisy especially around 50 seconds into the game

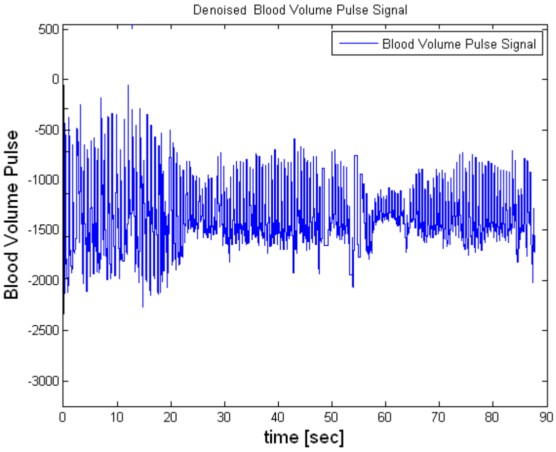


Figure 38. Noise reduced signal with extreme peaks removed

6.1.1.2 Skin Conductance Signal Processing

Noise removal on SC signals is easier than on the BVP or other fluctuating signals since there is no rapid change in the signal. The change in SC is slow and gradual; however when noise is present, sudden peaks in the signal are perceptible. The algorithm used for removing the noise (high peaks) in the SC signal follows this procedure:

1. looks at each sample consecutively using a moving average window (window size: 15 samples) , which calculates the average of the 15 samples,
2. if the difference between the sample under investigation and the moving average is bigger than the threshold (SC threshold was found to be good at 100 unites), then
3. the outlier values were transformed to the windows average.

Graphical representation of the raw and noise reduced SC signal can be seen in Figure 39.

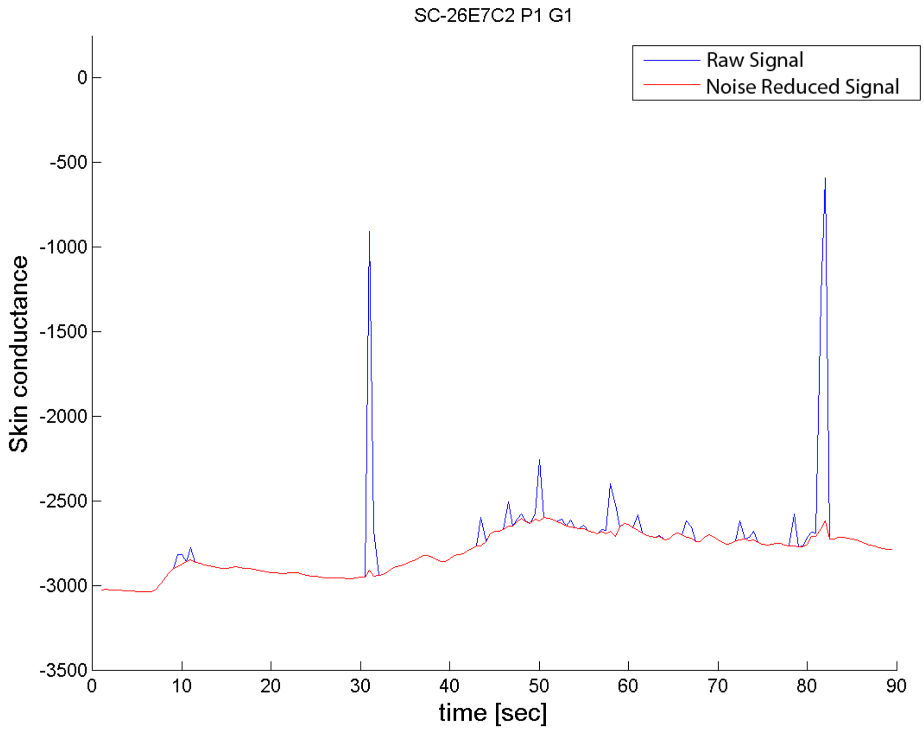


Figure 39. Raw signal (blue) noise reduced using a moving average window. The resulting signal (red) is plotted on top

6.1.2 Signal Features

As discussed in *Recognizing Emotions with Affective Computing* (section 2.2.5), there are many physiological measurements that could provide useful information for emotion recognition; however, because the common use and success in related studies the physiology measurements from the BVP and the SC signal were chosen. To infer emotions from body movement and postures, as described in *Body Movement in Video Games* (section 2.2.4), the Wii Remote's acceleration signal is also processed and used in this study.

The time frame for playing one test-bed game variant was 90 seconds. This time frame was furthermore estimated to be sufficient to use for calculating different statistical features instead of using the momentary value of a signal (Yannakakis, et al., 2007c). The different formulas will use h for referring to HR, b for BVP, s for SC and lastly x is used to refer to a generic signal.

The following section gives an overview of the different formulas used in this study to extract features from the input signals. Picard et al. (Picard, et al., 2001) suggests features that can be calculated for both the BVP and the HR signals (these can be seen in the list below as 1-5). And the rest of the signal feature extraction formulas used (presented in the list below 6-12), are suggested by Yannakakis et al. (Yannakakis, et al., 2007c).

In the following section a list of the common features for the physiology signals and their mathematical formulas are presented next.

1. The signal mean value:

$$E\{x\} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

2. The standard deviation of the signal:

$$\sigma\{x\} = \left(\frac{1}{N-1}\sum_{n=1}^{N}(x_n - x_{mean})^2\right)^{\frac{1}{2}}$$

3. The mean absolute value of the first differences, giving an indication of how linear the signal is:

$$\delta_{|1|}^{x} = \frac{1}{N-1}\sum_{i=1}^{N-1}|x_{i+1} - x_i|$$

   As the equation shows, the result is calculating how much each adjacent value in the signal is differentiating on average.
4. The mean absolute value of the second differences:

$$\delta_{|2|}^{x} = \frac{1}{N-2}\sum_{i=1}^{N-2}|x_{i+2} - x_i|$$

   This equation is the same as for the first difference, but calculates the average difference of two values separated by one value in between.
5. The average acceleration of the signal was calculated by taking the mean of the first differences:

$$E\{x\} = \frac{1}{N-1}\sum_{i=1}^{N}(x_{i+1} - x_i)$$

   The acceleration of the signal is the same as calculating the first differences. However, the values must not be absolute numeric values as this would loose information if the signal is accelerating or decelerating. It is important to note that a linear signal of any steepness will have zero in acceleration.
6. The initial signal value:

$$x_{in} = x_1$$

7. The last signal value at the end of a play session:

$$x_{last} = x_N$$

8. The maximum value recorded:

$$max\,\{x\} = \max_{1\leq i\leq N} x_i$$

9. The minimum value recorded:

$$min\,\{x\} = \min_{1\leq i\leq N} x_i$$

10. Difference between the maximum and minimum values:

$$D^x = max\,\{x\} - \ min\,\{x\}$$

11. Time of maximum value:

$$t^x_{max} = t(max\,\{x\})$$

12. Time of minimum value:

$$t^x_{min} = t(min\,\{x\})$$

A full list of all the features calculated can be seen in *Appendix D – Full* Feature List.

Further on, other specific features related to the BVP and HR signals, as well as the Wii Remote features were calculated. These features are discussed in the following two sections.

6.1.2.1 Specific Blood Volume Pulse and Heart Rate Features

As mentioned in *Recognizing Emotions with Affective Computing* (section 2.2.5), the BVP signal is an often used feature for inferring emotions.

To calculate specific BVP features the signal needs to be processed. Firstly, the peaks in the BVP signal indicating heart beats must be detected. For this a hill climbing algorithm was used to detect peaks. However, the algorithm has to be modified so not all peaks are detected. This is because the BVP signal contains small peaks in between peaks that indicate heart beats, for more information on the BVP signal refer to the description in *The Photoplethysmographic (PPG) Sensor* section. To overcome the small peaks in the inter-beat valley a minimum sample span between beats was introduced and thereby skipping inter-beat valley peaks there could possibly double the calculated HR. The sample span was calculated by empirically setting HR_{max} to 150bpm (beats per minute) as a higher HR is unrealistic in a *Wii* game. To calculate the span the following equation was used:

$$samplespan = \frac{samples/s}{HR_{max}\ /\ 60s}$$

where the sample rate is 32 samples/s, giving a span of 12 samples.

Using the beat information obtained from the BVP signal the current HR could be calculated by extrapolating the beats to approximate the bpm for each time step. The two steps performed by the algorithm are:

1. Divide every second into four equal sized buckets and count how many beats there is in each
2. Calculate the average number of beats in each bucket, the average is multiplied by 4 and again by 60 seconds to get the HR for a given second

The implementation of the algorithm was done by Daniel T. Kaplan in Matlab (Kaplan, 2003) based on the research by Ron Berger et al. (Berger, et al., 1986). Figure 40 shows the detected beats marked with red circles. The HR calculated by the algorithm was validated by comparing the calculated HR of six subjects to the HR from a Polar Pulse Monitor (CS600).

From the beat information, also the inter-beat intervals or RR-intervals were derived to calculate additional features.

The resulting signal can be seen in Figure 41 where the green plot shows the calculated HR at each time step. The moving average is using a window size of five, smoothing it cutting off high and low peaks (red plot). To make the signal more fluent and disregard small changes in the HR, a step-wise average were calculated (blue connected-circle plot).

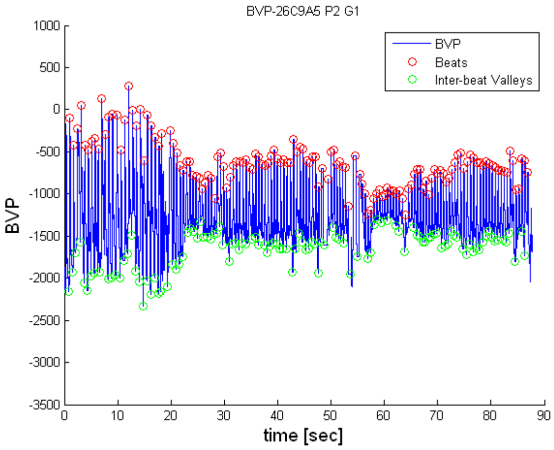


Figure 40. The noise reduced BVP signal with heart beat peaks marked with red, and the inter-beat valleys marked with green

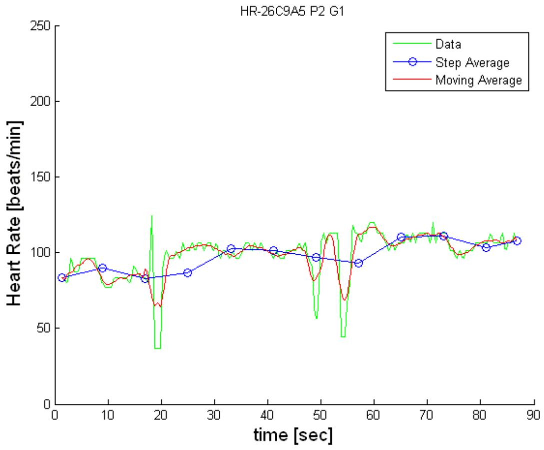


Figure 41. The final HR signal showing beats per minute based on the heart beats found in the BVP signal

For the BVP signal the Picard's first and second differences has proven to be good measures for emotions (Picard, et al., 2001) and is used for the ANN solution described in (Yannakakis, et al., 2007a). Besides the Picard features and the generic features presented in the previous section, other BVP specific features are calculated. For example, an analysis of the Heart Rate Variability (HRV) time and frequency domain for the RR-intervals is performed. The HRV energy values are calculated as the integral of the power of each energy band calculated from the power spectrum produced by Fast Fourier Transformation. The energy values are calculated for four BVP signal frequency bands, namely: High Frequency (HF), Low Frequency (LF), Very Low Frequency (VLF) and Ultra Low Frequency (ULF) (Yannakakis, et al., 2007a).

Here is a list of the additional BVP signal specific features calculated in this study:

1. From the HRV-time domain the standard deviation of the RR-intervals was calculated with the following formula:

$$\sigma\{RR\} = \left(\frac{1}{N-1}\sum_{i=1}^{N}(RR_i - RR_{mean})^2\right)^{\frac{1}{2}}$$

2. The fraction of deviating RR-intervals will also be calculated by counting the intervals that grows more than 50 ms in duration compared to the previous interval, pRR_{50}.
3. The last time-domain feature calculated is the root-mean square of successive differences or RR-intervals, $RMSSD_{RR}$.
$$RMSSD_{RR} = \frac{1}{N}\sum_{i=1}^{N-1}(RR_{i+1} - RR_i)^2$$
4. Energy for *ULF* [0 Hz; 0.0033 Hz]
5. Energy for *VLF* [0.0033 Hz; 0.04 Hz]
6. Energy for *LF* [0.04 Hz; 0.15 Hz]
7. Energy for *HF* [0.15 Hz;0.4 Hz]

Further more, one additional feature calculated for the HR signal is:

1. Seconds between minimum and maximum HR occurrence:

$$D_t^h = t_{min}^h - t_{max}^h$$

6.1.3 Wii Remote and Game Features

As seen in *Inferring Emotions from Physical Video Games* (section 2.2.4.1), studies indicated that there are two playing strategies in physical games; namely "Achieving" and "Relaxing". This meant that the player's behavioural expression would change dependant on the playing strategy, where small precise movement would be elicited when the player was trying to "achieve" and big movements that simulated the movements in game scenario would be elicited when the player was "relaxing". To get an indication of these strategies a measure of "power" (p) is proposed in this study as the aggregated absolute values of all the axes: $|a_x| + |a_y| + |a_z|$. The "power" should give an indication of how much the Wii Remote is moved over time and based on this signal, the minimum, $min\ \{p\}$, maximum, $max\ \{p\}$, and mean, $E\ \{p\}$ "power" values were calculated. Since the developed test game utilizes a gesture based input, many features based on the performed gestures were also calculated:

1. Total idle time i.e. the time when the Wii Remote was not moved around in the game (white areas in Figure 42).
2. Average "power" of the flick gestures, $E\{P^f\}$, is measured by summing the data from 10 samples from before the gesture was recognized. The sample size of 10 is derived from the game logic that uses 10 samples to determine the flick gesture.
3. Average "power" just after the flick, $E\{P^{pf}\}$, is measured by summing the data from the following 100 samples.
4. Average post flick idle time, $E\{t^{pfIdle}\}$. The post flick idle time is for how much time the Wii Remote is kept steady after a successful flick.
5. The average "power" of all gestures independently, $E\{P^g\}$.

Furthermore, following features for the raw Wii Remote signal the minimum, maximum, and mean values for each axis $\{x, y, z\}$ were calculated.

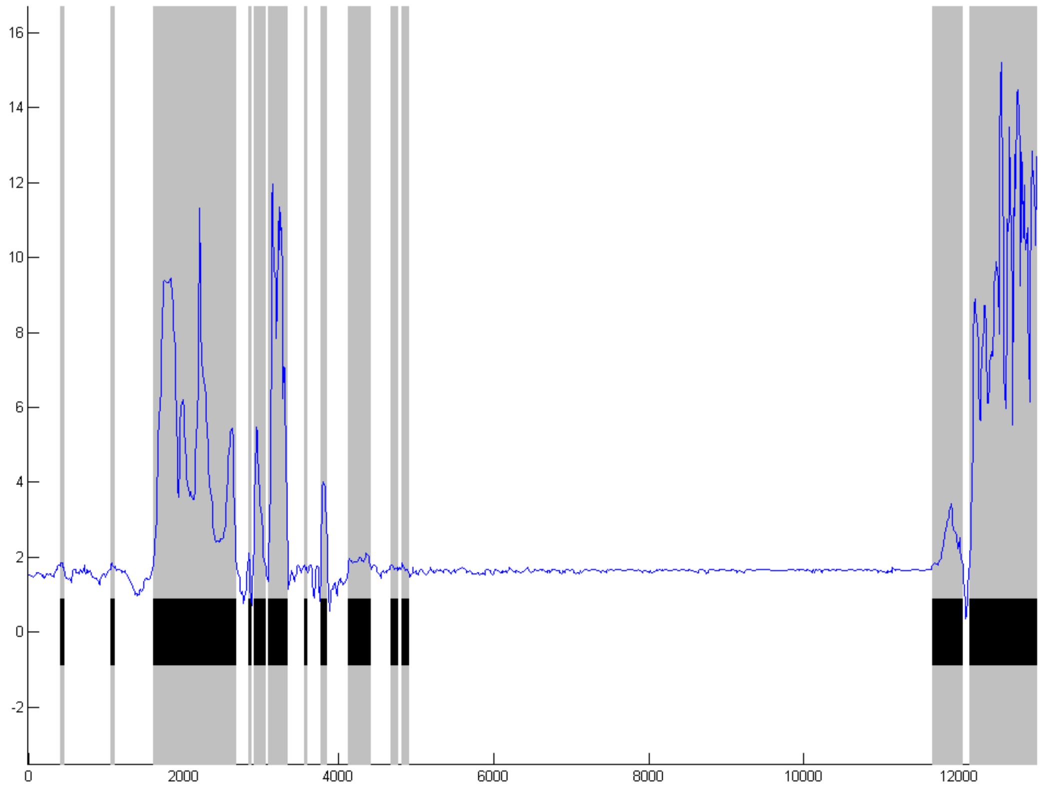


Figure 42. Graph showing acceleration "power". Gray areas is areas considered to be active. The threshold is for active periods emerically set to 1.8

Furthermore, features related to game events were also calculated:

1. Entropy of used spells, $H\{spell\}$
2. Number of used spells, $nSpell$
3. Total milliseconds the player has been under lightning attack, t^{lb}
4. Average milliseconds time the player has been under lightning attack, $E\{t^{lb}\}$
5. The power used to break out of a lightning attack, P^{lb}
6. Successful defensive moves, $nDefence$
7. Average reaction time from being hit with a lightning to the first motion $E\{t^{reac}\}$

6.2 Statistical Analysis

In (Yannakakis, et al., 2007c) a method to investigate statistically significant correlations between the test subjects' reported emotion and the input features was presented, which is followed in this section. For that purpose a null hypothesis was formed:

"The correlation between the test subjects' reported emotion and the input features in regard to the different games played is a result of randomness. The test statistic is obtained through:"
(Yannakakis, et al., 2007c)

$$H_0 : c(\vec{z}) = \frac{1}{N}\sum_{i=11}^{N} z_i$$

Where N is the total number of game pairs and is dependent on the emotion in question where the subject reported a clear preference i.e. the first two choices in the 4-AFC. In our case N for relaxation was: 37, frustration: 35, excitement: 28, boredom: 30 and fun: 35. If the subject chooses the game with higher feature values as the game that elicited more emotion (i.e. higher feature = more relaxing) then $z_i = 1$ and if the subject chooses he game with lower feature values as the that elicited more emotion (i.e. lower feature = more relaxing) then $z_i = -1$ in the game pair i.

Table 6 presents the 14 features with highest $c(\vec{z})$ and their corresponding p-values between reported emotion preference and selected features for each of the emotions. The p-values in Table 6 give indication of the probability that the particular features have a correlation to the preference. For example, the features with lower p-values, close to zero, are significantly correlated to the preference. It is interesting to see that the features with the highest significance for every emotion (including fun) are the features extracted from the acceleration data from the Wii Remote. The minimum motion in x, $min\{a_x\}$, is highly significant with p-values lower than 0.00005 (rounded to 0.0000 in the table) for four of the emotions; namely, relaxation, excitement, boredom and fun. Maximum and minimum in all the axes seem to have high correlation to all the perceived emotions as well. Since most of the features have p-values below 0.05, the null hypothesis was rejected.

In Table 6, it can also be seen that frustration does not have as many Wii Remote features as the other emotions. Based on the related work, the hypothesis in this study is that frustrated subjects elicit more powerful motion in the game. However, powerful motion is not what is measured from the Wii Remote. Features corresponding to "power" have in the previous section been stated to be the aggregated absolute values of all the axes: $|a_x| + |a_y| + |a_z|$. Because of the nature of the Wii Remote, this is unfortunately not the "power" that was sought after in this study. Small and quick movements would produce large fluctuations while the big powerful movements would not.

Another observation that can be drawn from Table 6, is that when the average BVP, $E\{b\}$ is high, the reported boredom and relaxation is high, opposed to the reported frustration and excitement where the average BVP, $E\{b\}$ is low.

Table 6. The 14 highest correlation coefficients and their corresponding p-values between reported emotion preference and selected features. The p-values have been rounded to 4 digit precision

Relaxation			Frustration			Excitement			Boredom			Fun		
Feature	$c(\vec{z})$	*p-value*	Feature	$c(\vec{z})$	*p-value*	Feature	$c(\vec{z})$	*p-value*	Feature	$c(\vec{z})$	*p-value*	Feature	$c(\vec{z})$	*p-value*
$min\{a_x\}$	-0.56	**0.0000**	$min\{a_x\}$	-0.53	**0.0000**	$min\{a_x\}$	-0.39	**0.0000**	$min\{a_x\}$	-0.45	**0.0000**	$min\{a_y\}$	-0.50	**0.0000**
$max\{a_y\}$	-0.41	**0.0000**	$min\{a_z\}$	-0.47	**0.0000**	$max\{a_y\}$	-0.33	**0.0000**	$max\{a_y\}$	-0.36	**0.0000**	$min\{a_x\}$	-0.50	**0.0000**
$max\{a_z\}$	-0.41	**0.0000**	$min\{a_x\}$	-0.38	**0.0000**	$min\{a_z\}$	-0.33	**0.0000**	$max\{a_z\}$	-0.30	**0.0002**	$max\{a_y\}$	-0.35	**0.0001**
$min\{a_z\}$	-0.38	**0.0000**	$min\{a_z\}$	-0.26	**0.0030**	$max\{a_x\}$	-0.27	**0.0005**	$min\{a_z\}$	-0.21	**0.0081**	$max\{a_z\}$	-0.38	**0.0000**
$max\{a_x\}$	-0.35	**0.0001**	$nSpell$	-0.26	**0.0030**	$\delta^s_{\|2\|}$	0.18	**0.0063**	$max\{a_x\}$	-0.18	**0.0214**	$max\{a_x\}$	-0.32	**0.0003**
$nDefence$	-0.29	**0.0013**	t^h_{min}	-0.20	**0.0205**	$max\{a_z\}$	-0.18	**0.0178**	$nDefence$	-0.18	**0.0214**	$\delta^s_{\|1\|}$	0.26	**0.0009**
$t^{reaction}$	-0.29	**0.0013**	s_{last}	0.14	**0.0448**	$\delta^s_{\|1\|}$	0.15	**0.0178**	$\delta^s_{\|1\|}$	-0.18	**0.0214**	$\delta^s_{\|2\|}$	0.23	**0.0030**
$E\{b\}$	0.22	**0.0038**	$\delta^s_{\|2\|}$	0.14	**0.0448**	t^{idle}	0.15	**0.0178**	$\delta^s_{\|2\|}$	-0.18	**0.0214**	t^h_{min}	-0.20	**0.0205**
$nSpell$	-0.20	**0.0235**	$min\{h\}$	-0.17	**0.0448**	s_{last}	0.12	**0.0436**	s_{in}	0.12	**0.0494**	D^s	0.17	**0.0205**
$t^{lighningBreak}$	-0.20	**0.0235**	$E\{b\}$	-0.17	**0.0448**	$t^{reaction}$	0.12	**0.0436**	$E\{IBamp\}$	-0.15	**0.0494**	$E\{IBamp\}$	0.14	**0.0448**
D^s	0.17	**0.0235**	$nBeats$	0.14	**0.0448**	$nDefence$	-0.12	0.0925	$E\{b\}$	0.12	**0.0494**	$\delta^b_{\|1\|}$	0.11	0.0877
$\delta^s_{\|2\|}$	-0.17	**0.0494**	$E\{h\}$	0.14	**0.0448**	$E\{s\}$	0.09	0.0925	t^h_{min}	0.12	**0.0494**	$\delta^b_{\|2\|}$	0.11	0.0877
$min\{P\}$	-0.17	**0.0494**	$\sigma\{RR\}$	-0.14	0.0877	t^s_{min}	0.09	0.0925	$nSpell$	-0.15	**0.0494**	$hann\{b\}$	0.11	0.0877
$max\{P\}$	-0.17	**0.0494**	$\delta^s_{\|1\|}$	0.11	0.0877	$E\{b\}$	0.09	0.0925	$min\{h\}$	0.12	**0.0494**	$E\{s\}$	0.11	0.0877

6.3 Emotion Model Learning

The statistical analysis showed that there was a significant correlation between many of the features and the subjects' reported emotion preferences; most notably the Wii Remote features had high significance. In this section, a non-linear approach for predicting the subjects' emotion preferences is examined.

To construct an emotion model that predicts the subject's reported emotion preference, a machine learning algorithm is used to train an ANN with a minimal feature subset selected by a feature selection algorithm. The assumption is that the player's emotion value y, which is a response to the game variant, is an unknown function of individual features which can be learned by machine learning (Yannakakis, et al., 2007c). Given that both physiology signal data can be noisy and that the player's self reported preference is person-dependent; we believe that ANNs should demonstrate a good generalizing function approximation. Feedforward multilayered ANN for learning the relation between the selected features (ANN inputs) and the emotion value y (ANN output) is therefore presented in this section. Since the output value y is not directly known, normal ANN training algorithms like back-propagation cannot be used. Learning is achieved through artificial evolution instead. For the ANN, a sigmoid function with the alpha value at 2 is utilized for each neuron. The neuron connection weights are initialized to values between 0 and 1, randomly from a uniform distribution, and all input values are normalized to the range [0, 1] before they are fed into the network. There are no clear rules for network topology and architecture, therefore the architecture determined in (Yannakakis, et al., 2007c) – consisting of two hidden layers with five hidden neurons in each – is followed in this study. Their choice was based on an evaluative comparison of the performance of ANN architectures of up to two hidden layers containing 30 hidden neurons each.

6.3.1 Feature Selection

In this experiment the SFS method was utilized since it has been applied successfully in a wide variety of feature selection problems – especially in this kind of classification problem (Yannakakis, et al., 2007b). In order to evaluate the performance of each feature subset, the performance of the model was evaluated using threefold cross-validation, where the available data is divided into three equal parts. Two of which is used for training and the last one used for validation. The parts are then rearranged and the model is validated again, until every part has been used for both training and validation. The model's overall performance is the average of the three independent cross-validation runs.

6.3.2 Evolutionary Algorithm

For developing the emotion model learning algorithm the procedure presented in *Preference Learning* (section 2.4.4) is followed. For this a generational GA is utilized, which uses an evaluation function that measures the difference between the subjects' reported preferences of emotion and the model's output value y.

A population size of 100 networks was initialized randomly with random connection weights. In an attempt to optimize training time without losing accuracy, a population size of 100 was determined based on an informal evaluative comparison with a population size of 1000. The GA chromosome is a vector of ANN connection weights. Then for each generation a four step process occurs:

6.3.2.1 *Step 1 - evaluation*

Each member *i* of the population is given two n-tuple input values corresponding to the current feature subset for the current emotion. The ANN receives player input values for A (preferred game) and input values for B (non-preferred game). The member thereafter returns two output values for that emotion, $y_{j,A}$ and $y_{j,B}$ respectively. When the two ANN output values are consistent with the reported emotion preference of subject *j*, i.e. when $y_{j,A} > y_{j,B}$, it is said that the member output is in "agreement" with the subject. In the opposite case it is said that the member is in "disagreement" with the subject.

The member is then evaluated using the fitness function f_i presented in (Yannakakis, et al., 2007c):

$$f_i = \frac{1}{2N} \sum_{j=1}^{N} \{ g(d_j^{AB}, \epsilon) \}$$

Where N is the number of training pairs and d_j^{AB} is the difference in output from preferred/non-preferred:

$$d_j^{AB} = y_{j,A} - y_{j,B}$$

$g(d_j, p)$ is the sigmoid function:

$$g(d_j, p) = \frac{1}{1 + e^{-pd_j}}$$

Where $\epsilon = 30$ if there is an agreement and $\epsilon = 5$ if there is a disagreement.

6.3.2.2 *Step 2 - crossover*

The population reproduces offspring with a crossover rate of 0.75, exceeding the original population size of *N*. New offspring are evaluated with the fitness function f_i presented in step 1.

6.3.2.3 *Step 3 - mutation*

When mutation happens, 25% of the population is cloned to create new offspring. Then the mutation happens to one random gene in each of the offspring's genome.

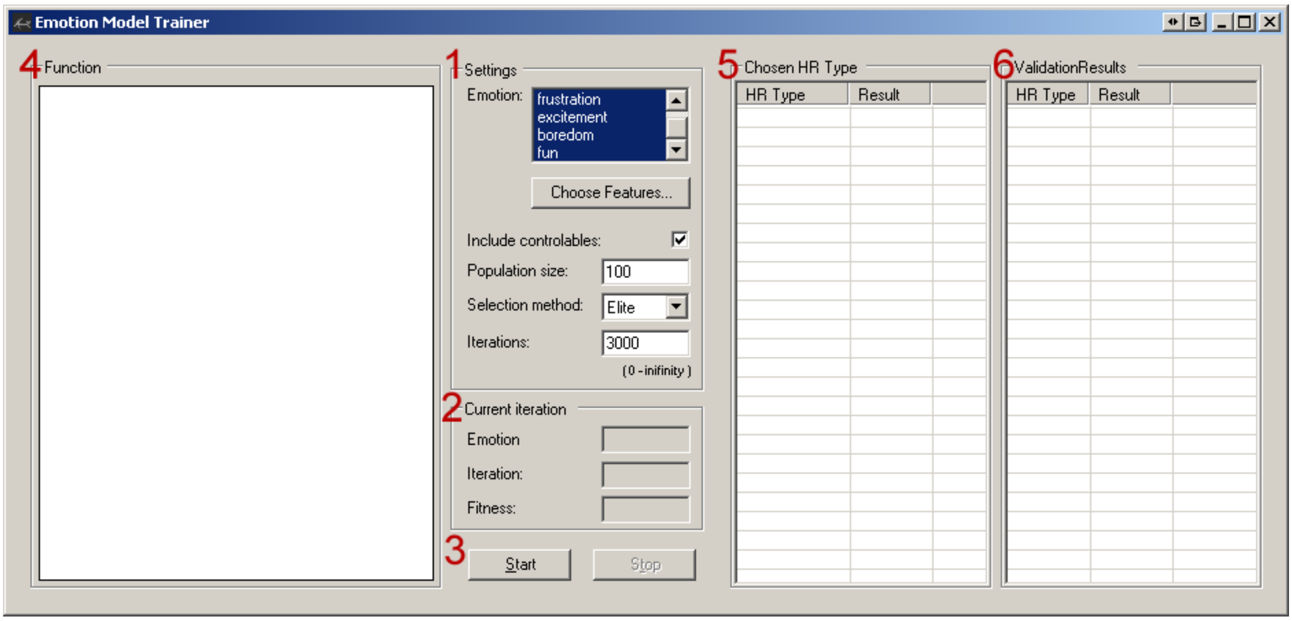
6.3.2.4 *Step 4 - selection*

An elite selection is used as the selection method. The population is ordered by fitness score and the 80% best performing members of the population are selected. Then new random members are introduced to the population ensuring that the population size of N is upheld.

When the population reaches a fitness of above 0.9 or when an adequate amount of generations, g have been completed (in this study g = 3,000) the algorithm is terminated and the next feature subset search begins.

6.3.3 Implementation

A C# application named *Emotion Model Trainer* was implemented to train the emotion model and can be seen in Figure 43. The application follows the method presented in the previous section, but gives the possibility for interaction and experimentation without having to recompile. Furthermore it also gives feedback on the current process of the algorithm if one wishes to follow along.



Emotion: frustration, excitement, boredom, fun
Include controlables: ☑
Population size: 100
Selection method: Elite
Iterations: 3000
(0 - inifinity)

Figure 43. The C# implementation of the emotion model learning algorithm

The user can in *1*, in Figure 43, choose which emotions the application should train. The population size, selection method and maximum generation iterations, together with the preference of whether or not the controllable game factors should be included can also be set here. Furthermore, it was possible to specify which features should be used by the trainer. This was useful when the model should be trained with Wii restricted features or physiology restricted features. When the user has pressed the start button in *3*, the application begins the SFS method by iterating over each feature three times (for cross-validation). The current iteration information; the emotion trained, iteration number, and the fitness can be seen in *2*. As the iteration progresses, the fitness is increasing. This can be seen in the graph in *4*. As the application steps through all features, it shows the validation performance of each feature in the list “Validation Results”, as seen in *6*. Then the best feature is selected and added to the current subset, which is shown in *5* and the features are then iterated again together with the new subset and shown in *6* once more. In *6* every feature has three entries; one for each independent cross validation run.

The input data is read from file; one file for each type of sensor (GSR, PPG, and the Wii Remote) containing all the calculated features as well as the preference expressed by the test subject.

While the program is searching for solutions several output files are generated containing data describing the run. Firstly a file containing the ANN configuration settings is generated stating the different setting the user has selected in *1*. Every time a feature is tested the result is saved to file (one file per emotion) giving the possibility to backtrack and verify the process. Lastly, the best performing ANN for each emotion is serialized and saved for later execution.

7 RESULTS

In this section, the results for the emotion model are presented. Firstly, the emotion model's classification accuracy using a single feature will be presented, and then the classification accuracy with the SFS method will be presented and evaluated. Lastly, the potential for tailoring player experience by using this emotion model is discussed. An important source of error noticed during the analysis of the results must be noted here. It was discovered that the Danish translation of the word "excitement" used in the user study was not sufficient because the word "ophidset" used for "excitement" does not have the exact same meaning as excitement in English, even though it is the translation found in the dictionary. An informal survey was done later to investigate the Danish interpretation of "ophidset", which clearly indicated that this word has a negative connotation and is much more related to anger or frustration than the English counterpart.

7.1 Single Feature Performance

The purpose of this section is to investigate a single feature's classification accuracy to get an indication of the model's performance. The performance of the model is given by the average classification accuracy of threefold cross-validation. The classification accuracy for each feature is presented in Table 7 where the features are ranked by performance.

The table shows that, for the emotion relaxation, the highest performing feature is the player's reaction time. This can be because players that are very relaxed are most likely to have slow reaction times. On the other hand the features for boredom and excitement are less obvious. These emotions have the mean inter-beat amplitude and the absolute second difference on SC respectively. For the last emotion frustration, the time for the lowest HR and the mean of the raw BVP signal features are chosen as the highest.

Table 7. Validation performance P and its respective standard deviation ($\sigma\{P\}$) of the five highest features for all emotions

Relaxation

Feature	Validation - $P(\%)$	$\sigma\{P\}$
$t^{reaction}$	70.08	10.22
$\delta^{s}_{\|2\|}$	67.31	9.31
$E\{b\}$	67.09	22.54
HF	65.17	10.66
VLF	64.95	8.84

Boredom

Feature	Validation - $P(\%)$	$\sigma\{P\}$
$E\{IBamp\}$	78.18	13.73
$\delta^{s}_{\|2\|}$	68.48	7.34
$t^{max\,\{h\}}$	65.45	6.56
s_{in}	63.03	23.38
$E\{x\}$	62.42	20.25

Excitement

Feature	Validation - $P(\%)$	$\sigma\{P\}$
$\delta^{s}_{\|2\|}$	78.15	11.67
$t^{min\,\{s\}}$	68.52	16.03
$t^{min\,\{h\}}$	67.78	1.92
t^{idle}	67.41	12.24
$max\,\{P\}$	64.44	3.85

Frustration

Feature	Validation - $P(\%)$	$\sigma\{P\}$
$t^{\min\,\{h\}}$	71.72	8.74
$E\{b\}$	71.21	17.91
$p^{gesture}$	68.94	16.75
$E\{P^{f}\}$	65.40	10.28
VLF	65.15	17.05

It is apparent that a single statistical feature to successfully predict emotion preference is relatively high. The best features yield cross-validation performances of 70.08%, 78.18%, 78.15%, and 71.72% for relaxation, boredom, excitement and frustration respectively.

With more features added to the model, it should be able to achieve even higher validation performance.

7.2 Multiple Feature Performance

To achieve higher validation performance more features were taken into account when training the model. One of the strengths of an ANN approach is that it is good at modelling complex functions. Therefore, training it with more features, higher performance can be achieved. As mentioned before, for the purpose of selecting features, the SFS method is utilized as it is relatively successful in picking good features.

It has been proposed to stop the SFS method when the added feature yields lower or equal validation performance to ensure minimal subset of features chosen. If the algorithm is stopped when the added feature yield a lower or equal performance, this might result in a local minimum as seen in Table 8. For example, the first feature for boredom yields a performance of 78.18% and the second feature yields 77.88% which is lower than the first one; if the SFS method is not stopped, the third (81.52%) and the fourth (87.58%) feature would not have been discovered. Optimally, all combinations of features should be tried, but because the training time grows rapidly with the number of features, the model learning would be too time consuming (a selection of five features took 12 hours to complete for all emotions). In this study, the ANN algorithm was given a slack on 5 percentage point, meaning that it would terminate when an added feature yielded 5 percentage points lower in performance than the previous feature subset. For comparison, the performance of random networks has been included. It would be expected for the random networks to perform at 50% but in this problem their performance is very poor showing that the features, found by the ANN to be good indicators is not random. It is also important to notice that none of the features with highest correlation (see Table 6) is selected for the best inputs for the ANN.

Humans display emotions differently, which is evident in the results shown in Table 8. For instance, relaxation and excitement is inferred from the body movement and the SC features, whereas boredom and frustration is inferred from the BVP features. More specifically, the relaxation is very apparent in the reaction time of the subject and the difference in the time for the minimum and maximum SC, and boredom is very apparent in the BVP as these features are chosen with the highest classification accuracy. It has been discussed in *Measuring Emotions* (section 2.2) that SC is a good measure and indicator for arousal. In the *Affect Grid* two of the emotions located in the extremes of the arousal axis is relaxation and excitement and held together with the classification accuracy and selected features, it is apparent that SC is in fact a good measurement for arousal. To differentiate between negative and positive feelings (valence) the BVP signal features is picked for a successful classification. For boredom, the SC and the BVP signal were selected, which is consistent with the research in the area. The accuracy of 77.02% seen in Table 8 for the feeling of frustration compared to the other emotions is much lower. This is most likely because the test-bed game *Wiizards* does not elicit the feeling of frustration to a high enough degree to build a consistent model from.

Table 8. Validation performance $P(\%)$ of a random network and the SFS selection method. The random network's performance is the average performance of three independent runs initialized with random weights. The random network's input vector is the best performing feature subset from the SFS method

Relaxation		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ t^{reaction} \}$	70.09	42.97
$\{ t^{reaction}, D_t^s \}$	83.33	
$\{ t^{reaction}, D_t^s, D^s \}$	**83.97**	
$\{ t^{reaction}, D_t^s, D^s, max\{s\}\}$	83.97	
$\{ t^{reaction}, D_t^s, D^s, max\{s\}, min\{P\} \}$	78.41	

Boredom		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ E\{IBamp\} \}$	78.18	35.56
$\{ E\{IBamp\}, \sigma\{b\} \}$	77.88	
$\{ E\{IBamp\}, \sigma\{b\}, E\{b\} \}$	81.52	
$\{ E\{IBamp\}, \sigma\{b\}, E\{b\}, \sigma\{s\} \}$	**87.58**	
$\{ E\{IBamp\}, \sigma\{b\}, E\{b\}, \sigma\{s\}, s_{in} \}$	77.58	

Excitement		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ \delta_1^s \}$	78.15	37.46
$\{ \delta_1^s, nDefence \}$	75.19	
$\{ \delta_1^s, nDefence, min\{P\} \}$	75.19	
$\{ \delta_1^s, nDefence, min\{P\}, s_{in} \}$	**82.96**	
$\{ \delta_1^s, nDefence, min\{P\}, s_{in}, \sigma\{b\} \}$	78.51	
$\{\delta_1^s, nDefence, min\{P\}, s_{in}, \sigma\{b\}, \delta_{\|2\|}^b\}$	78.89	
$\{ \delta_1^s, nDefence, min\{P\}, s_{in}, \sigma\{b\}, \delta_{\|2\|}^b, P^{gesture} \}$	77.78	
$\{ \delta_1^s, nDefence, min\{P\}, s_{in}, \sigma\{b\}, \delta_{\|2\|}^b, P^{gesture}, RMSSD_{RR} \}$	67.78	

Frustration		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ t^{min\{h\}} \}$	71.72	44.54
$\{ t^{min\{h\}}, nGesture \}$	73.99	
$\{ t^{min\{h\}}, nGesture, E\{IBamp\} \}$	**77.02**	
$\{ t^{min\{h\}}, nGesture, E\{IBamp\}, \delta_1^s \}$	71.72	

The trained emotion model can be used for tailoring the game experience, which is discussed in *Tailoring Player Experience* (section7.7).

7.3 Wii Remote Features

Because of the widely available Wii Remote, an investigation of whether a subset consisting solely of the Wii Remote and game features could achieve high classification accuracy is relevant. As it can be seen in Table 9 the model performs poorer than with the full subset; however, it classifies relaxation with a high accuracy of 81.41%.

Table 9. Validation performance $P(\%)$ of a random network and the SFS selection method with subsets restricted exclusively to Wii Remote specific features for all emotions

Excitement		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ H\{spell\} \}$	67.78	36.46
$\{ H\{spell\}, E\{a_x\} \}$	**71.85**	
$\{ H\{spell\}, E\{a_x\}, E\{P\} \}$	63.33	

Frustration		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ E\{P\} \}$	57.58	39.13
$\{ E\{P\}, max\,\{P\} \}$	**71.97**	
$\{ E\{P\}, ma\,x\{P\}, E\{P^{lb}\} \}$	63.13	

Relaxation		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ t^{reac} \}$	72.65	41.56
$\{ t^{reac}, min\{a_z\} \}$	72.65	
$\{ t^{reac}, min\{a_z\}, H\{spell\} \}$	**81.41**	
$\{ t^{reac}, min\{a_z\}, H\{spell\}, E\{P^f\} \}$	81.41	
$\left\{ \begin{matrix} t^{reac}, min\{a_z\}, H\{spell\}, E\{P^f\}, \\ t^{lb} \end{matrix} \right\}$	78.63	
$\left\{ \begin{matrix} t^{reac}, min\{a_z\}, H\{spell\}, E\{P^f\}, \\ t^{lb}, max\{x\} \end{matrix} \right\}$	65.38	

Boredom		Random network
Feature subset	$P(\%)$	$P(\%)$
$\{ nGestures \}$	53.33	40.64
$\{ nGestures, E\{P^{pf}\} \}$	62.73	
$\{ nGestures, E\{P^{pf}\}, H\{spell\} \}$	65.15	
$\left\{ \begin{matrix} nGestures, E\{P^{pf}\}, H\{spell\}, \\ E\{t^{lb} \end{matrix} \right\}$	68.78	
$\left\{ \begin{matrix} nGestures, E\{P^{pf}\}, H\{spell\}, \\ E\{t^{lb}\}, max\{a_x\} \end{matrix} \right\}$	**72.12**	
$\left\{ \begin{matrix} nGestures, E\{P^{pf}\}, H\{spell\}, \\ E\{t^{lb}\}, max\{a_x\}\ , E\{z\} \end{matrix} \right\}$	56.06	

In the study (Pasch, et al., 2008), two player movement strategies in physical *Wii* games were found; namely, "Achieving" and "Relaxing". A reason for the poorer performance achieved from only Wii features, could be that important distinction between big motion (relaxing) and small precise motion (achieving) is not made. Because of the limitations of the accelerometers in the Wii Remote, it simply might not be the best sensor to infer body movements from. An informal empirical test was done to evaluate the difference between the big and small movements mentioned, which indicated that small precise movement is easier to distinguish from big movements in the Wii Remote signal. However, big movements are not easily distinguishable from "no movement", since the most decisive factor in the movements is the sudden stops in the motions which are mostly seen in the small precise movement strategies. However, as seen in Table 9 the accuracy of inferring relaxation from Wii Remote features is relatively high. Another problem with the Wii Remote is the acceleration sensors are cheap; giving imprecise body movement data. Furthermore, the motion is limited to the hand the Wii Remote is in, possibly leaving out important information about body movements and posture.

However, with classification accuracies of 71.85%, 71.97%, 81.41%, and 72.12% for the respective emotions the commercial utilization of this technique is promising.

7.4 Physiology Features

There has not been found a comparative study of affect recognition with Wii movement versus physiological signals in the research done for this study; therefore, a comparison is done here. To perform the comparison the training algorithm was completed two times; once with features restricted to physiology, and once with Wii Remote features only. Table 10 shows the validation performance for the highest performing feature subsets for every emotion.

Table 10. Validation performance $P(\%)$ of the emotion model with subsets restricted to physiological features for every emotion

Relaxation		Boredom	
Feature subset	$P(\%)$	Feature subset	$P(\%)$
$\{\ t^{\max\{h\}}, ULF, \sigma\{b\}, \sigma\{RR\}, \sigma\{s\}\ \}$	75.64	$\{\ E\{IBamp\}, E\{b\}, \sigma\{s\}\ \}$	83.64

Excitement		Frustration	
Feature subset	$P(\%)$	Feature subset	$P(\%)$
$\begin{Bmatrix} s_{last},\ \sigma\{RR\}, hann\{b\}, \\ max\{h\}, min\{s\}, LF, E\{s\} \end{Bmatrix}$	78.51	$\{\ D_t^{\max\{s\}}, t^{\max\{h\}}, LF, t^{\min\{s\}}\ \}$	74.54

As seen in Table 11, the physiology features is not clearly better than Wii Remote features. More specifically, half of the features were higher with Wii Remote features and half were higher with physiology restricted features. Relaxation was performing approximately 6 percentage points better with Wii restricted features than with physiology restricted features. The keen reader will recognize that the emotion model for boredom is performing relatively well when features are restricted to physiology features compared to Wii restricted features, which was also the case when all features were available. Even though all features were available, the selection method still only chose physiology features, which is why boredom still has high accuracy in this comparison.

Table 11. Showing the highest classification accuracies $P(\%)$ for Wii Remote feature subset, and physiology feature subset

	Wii Remote	Physiology
Relaxation	81.41	75.64
Boredom	72.12	83.64
Excitement	71.85	78.51
Frustration	71.97	68.93

7.5 Feature Subset Performance

The emotion model was trained three times, once using all available features, once using only Wii Remote features, and lastly using only physiology features. The comparison can be seen in Table 12.

The emotion model results have best performance when the emotion is predicted with all features available. Relaxation is classified best using Wii Remote features, and is only improved slightly when combined with physiology. Boredom is classified accurately with all features, as well as when only physiology features are available. Whereas excitement is improved a lot with full feature set compared to only Wii Remote or physiological feature subsets. Lastly, the feeling of frustration is hard to classify in any regard. This might be because of the available features, or simply the fact that the test-bed game did not elicit enough frustration.

Table 12. Showing the highest classification accuracies $P(\%)$ for all features, Wii Remote features, and physiology features

	All	Wii Remote	Physiology
Relaxation	83.97	81.41	75.64
Boredom	87.58	72.12	83.64
Excitement	82.96	71.85	78.51
Frustration	77.02	71.97	68.93

7.6 Entertainment Model Performance

When talking about game design many use the expression “fun” and that a game has to be “fun” (Koster, 2004), (Bateman, et al., 2006), (Lazzaro, 2004), and (Malone, 1981). However, the feeling of fun is very complex and made up of many different aspects and human emotions, in addition the feeling of fun” can be very subjective and personal. To get an indication if the player is having fun and to optimize “fun” while maintaining the correct emotional state in the game, an entertainment model that recognizes “fun” in the game could be helpful for a game designer.

The methodology and the techniques used to build the emotion model in this study, were loosely based on the entertainment model developed in (Yannakakis, et al., 2007c) and (Yannakakis, et al., 2007a). Although the emotion model is meant to predict emotional states, it can also be trained using “fun” preferences. Their model of entertainment used only HR features and performed with 76% accuracy, showing that it is indeed possible to predict levels of “fun” in a game. The feature subset they ended up with was $\{R, E\{h\}\}$; where R is the correlation coefficient between HR recordings and the time in which data were recorded and $E\{h\}$ being the average HR.

Using the same methodology as for the emotions, the model in this study was trained for the feeling of fun. The difference from the model in (Yannakakis, et al., 2007c) and the model in this study is the implementation of the SFS algorithm, which was modified to include more features as described in *Feature Selection* (section 2.4.3). The reason for having more features was because a minimal subset size was not forced. The best performance of 86.11 % were recorded using all features and selecting a combination of SC, BVP and Wii Remote features as seen in Table 13. For comparison the model was trained given only physiological and Wii Remote features, and as expected they performed worse, but still reasonably well.

Table 13. The selected feature subset in three independent runs with a different set of available features, namely all, physiology only, and Wii Remote only

All		Physiology		Wii Remote	
Feature subset	$P(\%)$	Feature subset	$P(\%)$		
$\{\ \delta_{\vert 1\vert}^{b}, nDefence, D^{s}\ \}$	86.11	$\{\ \delta_{\vert 2\vert}^{s}, E\{IBamp\}\ \}$	74.49	$\{\ min\{a_y\}, t^{lb}, P^{gesture}\ \}$	77.02

The entertainment model can furthermore be used for tailoring the game experience; this is discussed in the following section.

7.7 Tailoring Player Experience

In game development, the possibility to tailor the players' game experience has always been investigated. The first attempts to tailor the player experience began with simple game mechanics that could adapt to the skill level of the player, as mentioned in *Motivation* (section 1.1). In recent years a lot of effort has gone into investigating emotions for use in affective computing. These emotion models for interpreting the player's emotions can in turn be used to tailor the game experience to adjust the emotions experienced by the player. This could enable more comprehensive game experiences with a larger impact on the player, compared to the techniques used today, where the task of invoking emotions in the player lies solely in the hands of the designer at design-time.

To tailor the experience in real-time, the emotion model needs to be aware of the controllable game factors in the game as well as the consequence of adjusting them. To meet this requirement, it is necessary to include the controllable game factors (in our case *curiosity* and *challenge*) in the input vector for the ANN and train the model including them. Including extra features in the ANN comes at a price, not only does the input grow in size it can also be harder for the model to generalize and incorporate the factors into the model. The training of the model was completed using the same subset configuration mentioned earlier (all, Wii Remote, and physiology features only) and can be seen in Table 14 for comparison. As it can be seen it has a rather large impact on performance when the game factors are included in the input. For example, the feeling of boredom with the highest accuracy drops by 16.32% from 87.58% to 73.29%. It is obvious from the results that the possibility to predict emotional states based on controllable game factors comes at a cost.

Table 14. Highest classification accuracies $P(\%)$ compared to the model that includes the controllable game factors

	Without Controllables - $P(\%)$			With Controllables - $P(\%)$		
	All	Wii	Physiology	All	Wii	Physiology
Relaxation	83.97	81.41	75.64	78.42	73.29	72.43
Boredom	87.58	72.12	83.64	73.29	68.79	71.71
Excitement	82.96	71.85	75.92	75.18	75.18	70.74
Frustration	77.02	71.97	68.93	74.24	68.18	72.42
Fun	86.11	77.53	82.83	82.83	68.94	74.24

The model is successful in predicting the player's level of "fun" both with and without controllable game factors; the emotions on the other hand, suffer from a performance decrease when the controllable game factors are added.

With a trained classification model, it is possible to calculate the network gradient to determine how the *challenge* and *curiosity* levels should be adjusted to change the emotions experienced in the game. A visualization of this can be seen in Figure 44, where a random subject's (subject 23) input data is used in the classification model. In this test game, the subject was playing with the controllable game factors at, *challenge*: 0.3 and *curiosity*: 0.3. This is depicted in the figure with a square. In order to increase e.g. the relaxation experience, the gradient would reveal that the game should increase challenge and curiosity.

The graphs appear to be a map of each emotion, where it could be assumed that the white areas would be the optimal emotion areas; however it is worth to note that the emotion map is not static. These maps were created from a fixed set of features (the chosen feature subsets from the previous section), and changing curiosity or challenge would evidently change the player's emotional state and therefore, also her physiological state and movement behaviour. These maps can therefore only be used for small changes and must be re-evaluated continuously.

From Figure 44, it can be seen that the model for excitement and frustration is very similar. This might be because the model is inaccurate in predicting frustration levels, as stated in *Multiple Feature Performance* (section 7.2). Another reason could be that the questionnaire question about excitement was translated into Danish incorrectly to "ophidset", which was found to have more of a negative than positive connotation and is much more related to anger or frustration.

The *boredom* map in Figure 44, shows that the game can be boring despite challenge being high, but also when challenge is low and curiosity is high. The lower area of the boredom map shows that boredom is high, which evidently also where frustration is high. This could be because of the test subject's wrongful interpretation of boredom, where in this study the definition of the highest boredom is "like staring into a wall". However the findings suggest that test subject would believe that a game is boring when the game is too hard for their skill i.e. when the game is frustrating. Batemann and Boon argue that boredom will be felt when the challenge is too low for the player's skill, whereas frustration will be felt when the challenge is too high for the player's skill (Bateman, et al., 2006). As it can be seen in our graph of frustration this does not fit with our findings. One of the reasons for the outcome of our findings not matching literature could be because curiosity corresponds to the perceived challenge of the game. When there is low curiosity, the AI is predictable (only choosing few spell combinations), whereas when the curiosity is high, the AI is completely unpredictable (choosing from all spell combinations), making the AI more challenging. Another problem could be because the test game does not induce frustration and therefore cannot be used to build a full model of frustration.

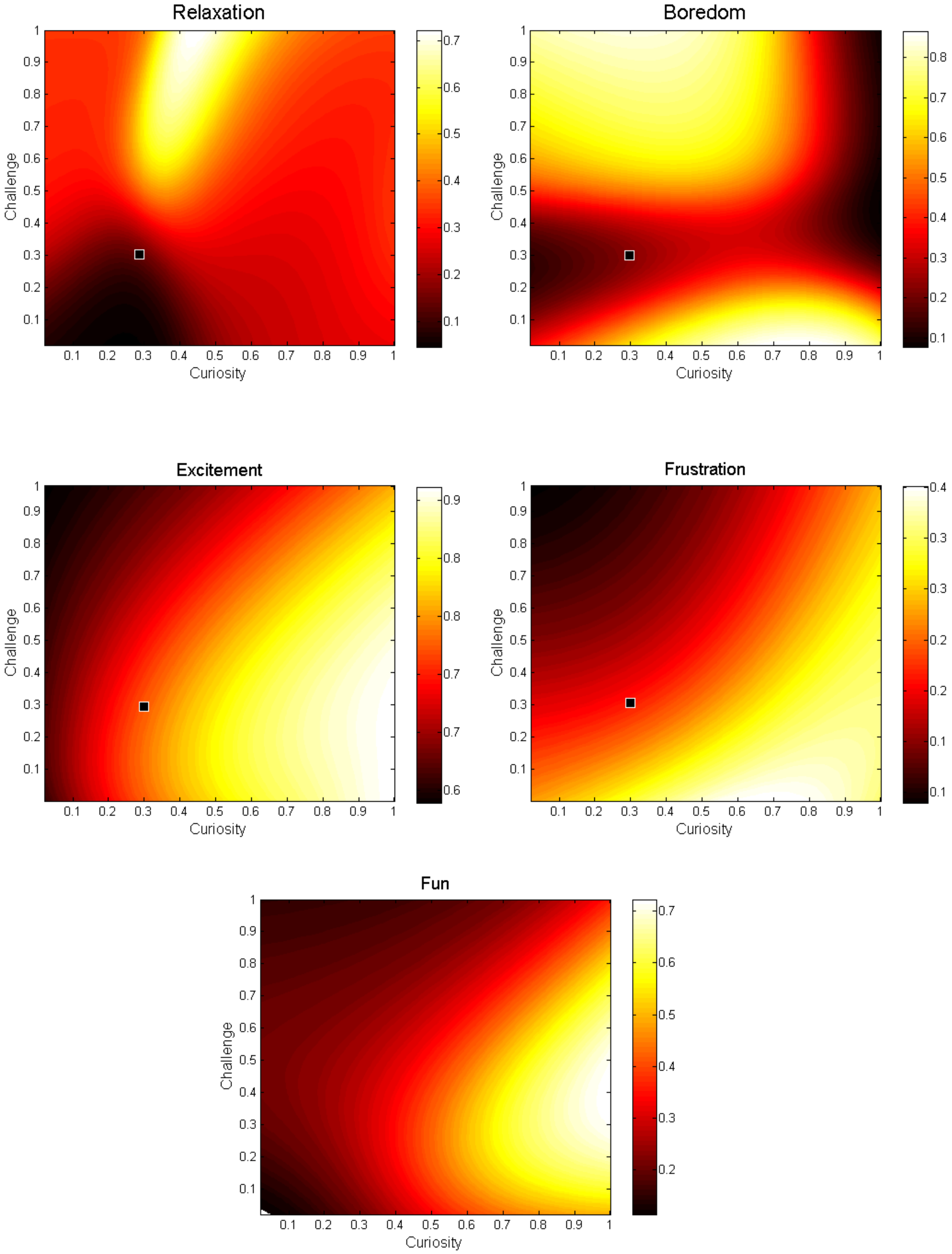


Figure 44. Graphs showing the level of relaxation, boredom, excitement, and frustration experienced in the game. The graphs are made from a fixed set of input features (features selected by SFS and presented in Table 8), and the two controllable game factors: *challenge* and *curiosity*. The black square depicts the player's current game variant (*curiosity*: 0.3 and *challenge*: 0.3)

8 DISCUSSION AND CONCLUSION

In this study the possibility of creating an emotion model to recognize emotions and entertainment in a Wii game was investigated. An ANN emotion recognizer was developed that could achieve a single feature performance of 78.18% for the feeling of boredom. Given a full feature subset the algorithm was able to achieve 83.97%, 87.58%, 82.96%, 77.02%, and 86.11% for the emotions relaxation, boredom, excitement, frustration, and fun. Adding controllable game factors to the model, the performance decreased a small amount and was at 78.42%, 73.29%, 75.18%, 74.24%, and 82.83% for the emotions relaxation, boredom, excitement, frustration, and fun respectively. Using the model with controllable game factors, it was shown that the automated affect recognizer could be used to tailor player experience by calculating the gradients of the *curiosity* and *challenge* parameters in the model for the emotion in interest and ultimately make game design decisions based on these. Furthermore, it was shown that restricting the features to Wii Remote only, the highest classification accuracy achieved was 75.18%. This result is interesting for the game industry as well as the academia as it can be utilized with no need for invasive physiology sensors. However, as physiology sensors are beginning to become more available in games (Nintendo, 2009) it is not unimportant to regard physiological features as well, since the model can achieve higher classification accuracy.

Although the emotion model presented in this study achieves high performance levels, it could achieve better classification if it was trained for use with a specific player. This could be achieved through an introductory part in the game where the player's physiological response would be recorded and analysed, and used to adjust the existing emotion model to achieve more accurate predictions. To get more knowledge about the player it could be relevant to use online learning of the model to continuously adjust the model. In this context the meaning of online learning is a model that can be retrained after it has been developed and shipped to the end user. For this strategy to be successful, further research is needed to ensure that the model can overcome day-by-day changes in the player's emotional states and physiological responses to emotions (Picard, et al., 2001).

Another aspect that has not been tested is how well the model will perform in real-time. We propose that the model is used with a moving window of 90 seconds, however to decrease the size of the moving window the model's validity has to be verified. In the research section, it is found that SC is slower than e.g. responses in the brain, but still fast enough for real-time use, as a normal human SC response time to a stimulus is 2-10 seconds. With such low response time, it should be possible to have the model perform in real-time given that the right features are selected from the signals. The main obstacle for creating a real-time model with a small time window is to let the players express their emotion state without distracting them from what is happening in the game during physiology data acquisition. This was not a problem in our method using the 90 seconds window, as it fits with one round of gameplay. However, if the game was to be trained on 10 second interval data, the test subjects would have to be presented with a questionnaire after 10 seconds of gameplay, which could ultimately lead to some annoyance in the test subject and thereby bias the training data.

The translation of excitement into the Danish word "ophidset" was, as mentioned earlier, correct in regard to the dictionary but had a different connotation that was more negative than the English version. This has shifted the emotion dimension that is classified from the physiology signals. This is not a big problem in the sense that we are not trying to plot the players in an emotion grid like for instance the Affect Grid. In that case, it would have been of the utmost importance that the emotion labels had the correct "distances" from the affective arousal and valance axes. To improve the methodology in this study, especially if the model is to be expanded to infer other emotion labels than those expressed directly by the subject, a more precise word for excitement must be found. The word "begejstret" is proposed to use as this word has a more positive connotation, and therefore matches the location of excitement in the Affect Grid better.

Finally, using this study's emotion model for tailoring the game experience the designer will get a solid tool to design the game in new and interesting ways, and could be used to create the so called *rollercoaster effect* as seen in recent blockbuster titles (*Gears of War* and *Half-Life 2*) (Hong, 2008). The rollercoaster effect is when the player experiences a lot of in-game stress followed by cool down periods repeated over and over again during the game. The same effect could be achieved in the emotion space and push users into new emotion states after each other to create much deeper and more interesting games.

The task of moving the player around in the emotion space (emotion state traversal) is assumed to be a difficult task. The emotions classified in the emotion model depend on each other, and it is therefore not possible to adjust the emotions separately without affecting the other emotions. An example could be that the game tries to optimize for frustration, but induces boredom at the same time. In this case there would be needed to set an emotion threshold for how much the other emotions are allowed to increase, when the emotion that is sought to be optimized is increased. The four emotions chosen for this study are located in the extreme corners of the Affect Grid; it can therefore – to a certain degree – be assumed that the optimal level for two emotions cannot be reached at the same time. This could be possible for emotions and complex emotions of enjoyment like "fun". In this case a specific player might find maximum excitement to be the most fun and another player might think that frustration is the most fun. This is of course only possible because the feeling of "fun" is a complex feeling composed by many different feelings as well as cognitive processes of what one perceives as fun.

Following the methodology proposed in (Yannakakis, et al., 2007), it has been shown that it can be used to generalize to other game scenarios and platforms, and can be implemented using other inputs than strictly physiology input. Therefore, the model could have commercial potential, as it can be implemented using current technology.

This study is the next step towards an emotion model used to tailor the player experience by using the Wii Remote for pervasive and unobtrusive input for the game, and seconding as an input for recognizing emotion to drive the player through an emotional rollercoaster to give the ultimate feeling of fun.

9 BIBLIOGRAPHY

AIIDE The Third Artificial Intelligence for Interactive Digital Entertainment Conference [Online] // AAAI. - 2007. - 14 July 2009. - http://www.aaai.org/Conferences/AIIDE/aiide07.php.

AiLive [Online] // AiLive. - 2000. - 7 July 2009. - http://www.ailive.net/.

AiLive LiveMove White Paper [Report]. - 2006.

Apple iPhone [Online] // Apple. - Apple Inc., 2007. - 24 July 2009. - http://www.apple.com/iphone/.

Atari Pursuit [Online] // Atari. - 1975. - 13 July 2009. - http://www.atari.com/.

Bateman Chris and Boon Richard 21st Century Game Design [Book]. - Hingham, MA, USA : Charles River Media, Inc., 2006. - pp. 53-78. - 1-58450-429-3.

Berger Ronald D. [et al.] An Efficient Algorithm for Spectral Analysis of Heart Rate Variability [Article] // IEEE Transactions On Biomedical Engineering. - September 1986. - 9 : Vols. BME-33. - pp. 900-904.

Bernhardt Daniel. and Robinson Peter. Detecting affect from non-stylised body motions [Book Section] // Affective Computing and Intelligent Interaction / ed. A. Paiva R. Prada, and R.W. Picard. - [s.l.] : Springer Berlin / Heidelberg, 2007. - Vol. 4738. - ISBN: 978-3-540-74888-5 .

Bianchi-Berthouze Nadia [et al.] On posture as a modality for expressing and recognizing emotions [Conference] // Emotion in HCI: Joint Proceedings of the 2005, 2006, and 2007 International Workshops. / ed. Christian Peter Russell Beale, Elizabeth Crane, Lesley Axelrod, Gerred Blyth. - [s.l.] : Fraunhofer IRB Verlag Stuttgart, 2008. - pp. 74-81. - ISBN: 978-3-8167-7540-9.

Bianchi-Berthouze Nadia Body movement as a means to modulate engagement in computer games // Workshop on Whole body Interaction II: The future of the human body, in conjunction with HCI'08. - 2008a.

Bianchi-Berthouze Nadia Body Movement as a Modality for Supporting Positive Experience in HCI [Conference] // CHI 2008. - Florence, Italy : CHI, 2008b.

Bianchi-Berthouze Nadia, Woong Kim Whan and Patel Darshak Does Body Movement Engage You More in Digital Game Play? and Why? [Book Section] // Lecture Notes in Computer Science. - 31-32 Alfred Place, London WC1E7DP, UK : Springer Berlin / Heidelberg, 2007. - Vol. 4738. - ISBN: 978-3-540-74888-5.

Boehm Klaus, Broll Wolfgang and Sokolewicz Michael Dynamic gesture recognition using neural networks; a fundament for advanced interaction construction [Conference] // Proceedings of the SPIE---The International Society for Optical Engineering. - 1994. - pp. 336-46.

Buckland Mat A Biological Neural Network—The Brain [Book Section] // AI Techniques for Game Programming. - Cincinnati, Ohio : Premier Press, 2002a. - ISBN: 1-931841-08-X.

Buckland Mat AI Techniques for Game Programming [Book]. - Cincinnati, Ohio : Premier Press, 2002b. - ISBN: 1-931841-08-X.

Buckland Mat The Digital Version [Book Section] // AI Techniques for Game Programming. - Cincinnati, Ohio : Premier Press, 2002c. - ISBN: 1-931841-08-X.

Bull Peter. E. Posture and gesture [Book]. - Oxford : Pergamon Press, 1987. - ISBN: 0080313329.

Cacioppo John. T. and Tassinary Louis. G. Inferring Psychological Significance from Physiological Signals [Journal]. - [s.l.] : American Psychologist, 1990. - 45. - pp. 16-28.

Champandard Alex J. AI Game Development [Book]. - Indianapolis : New Riders, 2004.

Cowley Ben [et al.] Data-Driven Decision Theory for Player Analysis in Pacman [Conference] // AAAI Technical Report - Proceedings of the Optimising Player Satisfaction workshop. - [s.l.] : AIIDE '07, 2007. - pp. 25-30.

Dimberg U. Facial Expressions and emotional reactions: A psychobiological analysis of human social behaviour [Book Section] // Social Psychophysiology and Emotions / book auth. Wagner H. L.. - [s.l.] : John Wiley & Sons, 1988. - 6. - ISBN-10: 0471912662.

Electronics Arts The Sims 2 [Online] // The Sims 2. - Electronics Arts Inc., 2009. - 24 July 2009. - http://thesims2.ea.com/about/index.php?pid=pcgames.

EmSense EmSense Technology [Online] // EmSense. - 2004. - 8 July 2009. - http://www.emsense.com/technology.php.

Gilleade Kiel M and Dix Alan Using Frustration in the Design of Adaptive Videogames [Conference] // Proceedings of ACE 2004. - Singapore : ACM Press, 2004. - pp. 228-232.

Greene Kate Brain Sensor for Market Research [Online] // Technology Review. - MIT, 7 December 2007. - 8 July 2009. - http://www.technologyreview.com/business/19833/page1/.

Gurney Kevin An introduction to neural networks [Book]. - [s.l.] : CRC Press, 1997. - pp. 83-84. - 9781857285031.

Haykin Simon S. Neural Networks and Learning Machines [Book]. - [s.l.] : Prentice Hall, 2008. - 3 : p. 14. - 9780131471399.

Health Body mass index (BMI) for boys, ages 2 to 20 [Online] // Health. - 2008. - 12 July 2009. - http://www.health.com/health/library/mdp/0,,zm2274,00.html.

Hong Tim Shoot to Thrill; Bio-sensory Reactions to 3D Shooting Games [Online] // Gamasutra. - 2 December 2008. - 24 July 2009. - http://www.gamasutra.com/view/feature/3868/shoot_to_thrill_biosensory_.php.

Hugdahl Kenneth Psychophysiology: The Mind-Body Perspective [Book]. - [s.l.] : Harvard Univesity Press, 1995. - ISBN 0-674-72207-8.

Ikehara Curtis S. and Crosby Martha E. Assessing Cognitive Load with Physiological Sensors [Conference]. - [s.l.] : Hawaii International Conference on System Sciences, 2005.

James William What is an emotion? [Article] // Mind. - 1884. - Vol. 9. - pp. 188-205.

Juul Jesper Half Real: Video Games between Real Rules and Fictional Worlds [Book]. - [s.l.] : MIT Press, 2005. - ISBN-10: 0-262-10110-6.

Kaplan Daniel T. Adaptive Filters Applied to Heart ECG [Online] // Adaptive Filters. - RICE: Office for the Vice Provost for Information Technology, 18 1 2003. - 07 07 2009. - http://www.owlnet.rice.edu/~elec301/Projects02/adaptiveFilters/.

Keltner Dacher and Ekman Paul Emotion [Journal] // Encyclopedia of psychology . - [s.l.] : Oxford University Press, 2000. - Vol. 3.

Konami Konami [Online] // Dance Dance Revolution. - 1998. - 7 July 2009. - http://www.konami.com/games/ddr/.

Koster Raph A Theory of Fun for Game Design [Book]. - [s.l.] : Paraglyph Press, 2004.

Lang Peter J. The emotion probe: Studies of motivation and attention [Journal] // American Psychologist. - 1995. - pp. 372 - 385.

Lange C. Ueber Gemuthsbewgungen [Book]. - 1887.

Lazzaro Nicole Why We Play Games: Four Keys to More Emotion Without Story. - Oakland : [s.n.], 8 March 2004.

Leighton Stephen Aristotle and Emotions [Journal] // Phronesis. - 1982. - pp. 144-173.

Lippmann Richard P. An Introduction to Computing with Neural Nets [Article] // ASSP Magazine, IEEE. - 4 1987. - 2 : Vol. 4. - pp. 4-22.

Lund Henrik H., Klitbo Thomas and Jessen Carsten Playware technology for physically activating play [Journal] // Artifical Life and Robotics Journal 9(4). - 2005. - pp. 165-174.

Malone Thomas W. What makes computer games fun? [Journal]. - [s.l.] : Byte 6, 1981. - pp. 258-277 .

Mandler George Emotion: History of the Concept [Journal]. - [s.l.] : Elsevier Science Ltd., 2001. - International Encyclopedia of the Social & Behavioral Science : Vol. 7. - pp. 4437-4440. - ISBN 0-08-43076-7.

Mandryk Regan L., Inkpen Kori M. and Calvert Thomas W. Using Psychophysiological Techniques to Measure User Experience with Entertainment Technologies [Journal] // Behaviour and Information Technology (Special Issue on User Experience). - 2006a. - pp. 141-158.

Mandryk Regan L., Stella Atkins and Inkpen Kori M. A Continuous and Objective Evaluation of Emotional Experience with Interactive Play Environments [Conference] // CHI 2006 Proceedings - Novel Methods: Emotions, Gestures, Events. - Montréal, QC, Canada : [s.n.], 2006b. - pp. 1027-1036.

Manstead A. S. R. The role of facial movement in emotion [Book Section] // Social Psychophysiology: Theory and Clinical Applications. / book auth. Wagner Hugh L.. - [s.l.] : John Wiley & Sons, 1988. - ISBN-10: 0471912662.

McQuiggan Scott W., Lee Sunyoung and Lester James C. Predicting User Physiological Response for Interactive Environments: An Inductive Approach [Report]. - [s.l.] : American Association for Artificial Intelligence, 2006.

Microsoft Project Natal [Online] // XBox. - 2009. - 8 7 2009. - http://www.xbox.com/en-US/live/projectnatal/.

Monatna David J. and Lawrence Davis Training Feedforward Neural Networks Using Genetic Algorithms [Conference] // Eleventh International Joint Conference on Artificial Intelligence, Vol. 1. - Cambridge, MA : [s.n.], 1989. - pp. 762-767.

NCHS 2000 CDC Growth Charts: United States [Online] // National Center for health Statistics. - 2000. - 12 July 2009. - http://www.cdc.gov/growthcharts/.

Nintendo Nintendo :: E3 2009 :: Press Release. - 02 06 2009.

Nintendo Press Release [Online] // Nintendo :: E3 2009. - 2 July 2009. - 21 July 2009. - http://e3.nintendo.com/pressrelease/.

Nintendo Super Nintendo Entertainment System. - 1990.

Nintendo Virtual Console Games [Online] // Nintendo. - 1998. - 7 July 2009. - http://www.nintendo.com/wii/virtualconsole/games/detail/yhT-qheyIGv54kkcrdMc3vYLWL_UNr9e.

Nintendo Wii [Online] // Nintendo. - 2006a. - 7 July 2009. - http://www.nintendo.com/wii.

Nintendo Wii Sports [Online] // Nintendo. - 2006b. - 7 July 2009. - http://www.nintendo.com/games/detail/1OTtO06SP7M52gi5m8pD6CnahbW8CzxE.

Osgood Charles E., Suci George J. and Tannenbaum Percy H The measurement of meaning [Book]. - [s.l.] : University of Illinois Press, 1957. - ISBN-10: 0252745396.

Parker J.R. Human Motion as Input and Control in Kinetic Games [Report] / University of Calgary . - 2500 University Dr. NW, Calgary, Alberta, Canada : [s.n.], 2003.

Pasch Marco [et al.] Motivations, Strategies, and Movement Patterns of Video Gamers Playing Nintendo Wii Boxing [Conference] // Procedings of the Facial and Bodily Expressions for Control and Adaptation of Games. - [s.l.] : 2008 IEEE International Conferenceon Automatic Face and Gesture Recognition, 2008.

Pernkopf Franz and O'leary Paul Features Selection for Classification Using Genetic Algorithms with a Novel Encoding [Book Section] // Computer Analysis of Images and Patterns. - Berlin / Heidelberg : Springer, 2001. - Vol. 2124/2001. - 978-3-540-42513-7.

Picard R. W. Affective Computing [Book]. - Cambridge, MA : MIT Press, 1997. - 0262661152.

Picard Rosalind W. Research on Sensing Human Affect [Online] // Affective Computing. - 2009. - 29 June 2009. - http://affect.media.mit.edu/areas.php?id=sensing.

Picard Rosalind W., Vyzas Elias and Healey Jennifer Toward Machine Emotional Intelligence: Analysis of Affective Physiological State [Article] // IEEE Transactions on Pattern Analysis and Machine Intelligence. - 2001. - 10 : Vol. 23.

PlayStation EyeToy [Online] // PlayStation. - 2003. - 7 July 2009. - http://www.us.playstation.com/PS2/Games/EyeToy_Play/ogs/.

Rosenberg E. L. and Ekman P. Emotion: method and study [Article] // Encycllopedia of Psychology. - [s.l.] : Oxford Univestity Press, 2000. - Vol. 3. - pp. 171-175. - ISBN 1-55798-187-6.

Rosenblatt Frank Principles of Neurodynamics: Perceptrons and The Theory of Brain Mechanisms [Book]. - Washington DC : Spartan Books, 1961.

Russell James A. A Circumplex Model of Affect [Journal]. - [s.l.] : Journal of Personality and Social Psychology, 1980. - 39. - pp. 1161–1178.

Russell James A., Weiss Anna and Mendelsohn Gerald A. Affect Grid: A Single-Item Scale of Pleasure and Arousal [Journal] // Journal of Personality and Social Psychology. - 1989. - 3 : Vol. 57. - pp. 493-502.

Schachter Stanley The interaction of cognitive and physiological determinants of emotional state [Journal] // Advances in Experimental Psychology. - New York : Academic Press, 1964. - Vol. 1. - pp. 49-80.

Scherer Klaus R. Neuroscience projections to current debates in emotion psychology [Journal] // Cognition and emotion. - [s.l.] : Psychology Press, 1993. - Vol. 7. - pp. 1-41. - ISSN 0269-9931.

Scherer Klaus R. Vocal Affect Expression A Review and a Model for Future Research [Journal] // Psychological Bulletin. - [s.l.] : American Psychological Association, 1986. - Vol. 99. - pp. 143-165.

Scherer Klaus R. What are emotions? And how can they be measured [Journal] // Social Science Information. - 2005. - Vol 44 – no 4. - pp. 695-729.

Schlömer Thomas [et al.] Gesture Recognition with a Wii Controller [Conference] // TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction. - [s.l.] : ACM, 2008. - pp. 11-14.

SETA Nintendo 64 Tetris 64 [Online] // NCSX. - SETA, 1998. - 14 July 2009. - http://www.ncsx.com/www/ncs110998/tetris_64.htm.

Square Enix FINAL FANTASY [Online] // Square Enix. - SQUARE ENIX CO., 2009. - 24 July 2009. - http://www.square-enix.com/na/title/finalfantasy/.

Sykes Jonathan and Brown Simon Affective Gaming: Measuring emotion through the gamepad [Journal] // CHI. - 2003. - pp. 732-733.

Technology Thought Physiology Suite [Online] // Thought Technology. - 2009. - 12 July 2009. - http://www.thoughttechnology.com/index.html.

Thorpe Simon, Fize Denis and Marlot Catherine Speed Of Processing In The Human Visual System [Journal] // Nature. - 1996. - Vol. 381. - pp. 520-522.

Tijs Tim J. W., Brokken Dirk and IJsselsteijn Wijnand A. Dynamic Game Balancing by Recognizing Affect [Book Section] // Fun and Games. - [s.l.] : Springer Berlin / Heidelberg, 2008. - Vol. Volume 5294. - ISBN 978-3-540-88321-0.

Wiggins Henry O. Gesture Recognition of Nintendo Wiimote Input Using an Artificial Neural Network [Report] / University of British Columbia . - [s.l.] : Cognitive Systems, 2008.

WildDivine WildDivine [Online] // WildDivine. - WildDivine, 2001. - 14 July 2009. - http://www.wilddivine.com/.

Wilson Josephine F. Biological Foundations of Human Behavior [Book]. - [s.l.] : Wadsworth Publishing, 2003. - ISBN-10: 0155074865.

Wolf Mark J. P. The Video Game Explosion: A History from PONG to PlayStation and Beyond [Book]. - [s.l.] : Greenwood, 2007. - p. 18.

Woodhouse Jamie Qwak [Online] // BBC Micro. - 1989. - 13 July 2009. - http://www.qwak.co.uk/.

Yannakakis Georgios N. and Hallam John Entertainment Modeling in Physical Play through Physiology beyond Heart-Rate [Conference] // Proceedings of the Int. Conf. on Affective Computing and Intelligent Interaction, Lecture Notes in Computer Science, vol. 4738. - Lisbon : Springer-Verlag, 2007a. - pp. 256-267.

Yannakakis Georgios N. and Hallam John Game and Player Feature Selection for Entertainment Capture [Conference] // Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Games. - Hawaii, USA : IEEE, 2007b. - pp. 244-251.

Yannakakis Georgios. N., Hallam John and Lund Henrik H. Entertainment Capture through Heart Rate Activity in Physical Interactive Playgrounds [Article] // User Modeling and User-Adapted Interaction. - Feburary 2007c. - Special Issue on Affective Modeling and Adaptation. - 1-2 : Vol. 18. - pp. 207-243.

Yannakakis Georgios. N., Hallam John and Lund Henrik H. Entertainment Capture through Heart Rate Activity in Physical Interactive Playgrounds [Article] // User Modeling and User-Adapted Interaction. - Feburary 2007. - Special Issue on Affective Modeling and Adaptation. - 1-2 : Vol. 18. - pp. 207-243.

Zongker Douglas and Jain Anil Algorithms for Feature Selection: An Evaluation [Conference] // Proceedings of the 13th International Conference on Pattern Recognition, Vol. 2. - Washington, DC, USA : IEEE Computer Society, 1996. - 0-8186-7282-X.

APPENDIX A – DEMOGRAPHIC QUESTIONNAIRE

Demographic questionnaire in English

1. Player Code:
2. Age:
3. Sex:
4. Height:
5. Weight:
6. BMI:
7. Do you practise any sports? If yes, what sports? How often?
8. I. Do you play video games?
   a) Yes b) No
   II. How often do you play video games?
   a) Almost everyday b) Few times a week d) Few times a month c) Less
9. I. Have you played Wii games?
   a) Yes b) No
   II. How often do you play video games?
   a) Almost everyday b) Few times a week d) Few times a month c) Less

Demographic questionnaire in Danish

1. Spiller Kode:
2. Alder:
3. Køn:
4. Højt:
5. Vægt:
6. BMI:
7. Dyrker du nogle sportsgrene? Hvis ja, hvilke sprtsgrene og hvor ofte?
8. I. Spiller du computerspil eller konsol?
   a) Ja b) Nej
   II. Hvor ofte spiller du?
   a) Hver dag b) Et par gange om ugen d) Et par gange om måneden c) Sjældent
9. I. Har du spillet Wii?
   a) Ja b) Nej
   II. Hvor ofter spiller du Wii?
   a) Hver dag b) Et par gange om ugen d) Et par gange om måneden c) Sjældent

APPENDIX B – 4-AFC AND RATING QUESTIONNAIRES

4-AFC questionnaire

The following image is a screen shot from the 4-AFC questionnaire translated in Danish.

Vælg hvilket svar der bedst passer til udsagnet herunder:

Jeg var afslappet	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Jeg var frustreret	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Jeg var ophidset	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Jeg kedede mig	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Jeg havde det sjovt	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Hvor udfordrende var spillet?	A ☑	B ☐	Lige meget ☐	Hverken eller ☐
Hvor forudsigeligt var spillet?	A ☑	B ☐	Lige meget ☐	Hverken eller ☐

Rating questionnaire

The following image is a screen shot from the rating questionnaire translated in Danish.

Vælg et tal mellem 1 og 5 der bedst beskriver følgende udsagn, hvor 1 er lavest og 5 er højest

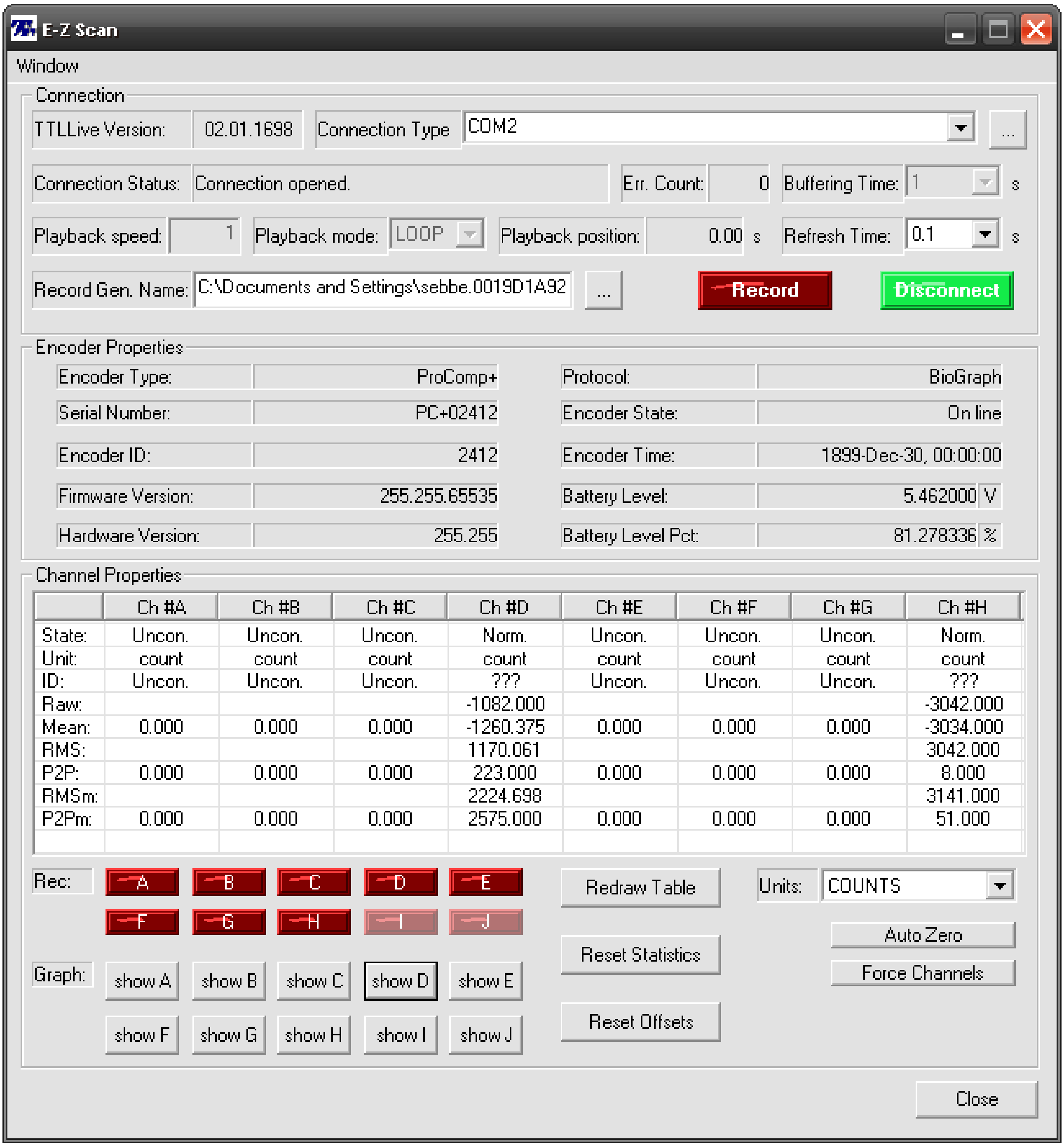
Jeg var afslappet	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Jeg var frustreret	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Jeg var ophidset	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Jeg kedede mig	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Jeg havde det sjovt	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Hvor udfordrende var spillet?	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐
Hvor forudsigeligt var spillet?	1 ☑ 2 ☐ 3 ☐ 4 ☐ 5 ☐

APPENDIX C – INFINITI ENCODER THROUGH THE TELE-INFINITI

To measure the player's physiology the Infiniti Encoder and sensors from Through Technology is used. The encoder has the possibility to record the signals to a CompactFlash (CF) card for later processing. If this option is used, that card should not be larger than 2GB as the FAT16 file system cannot handle anything larger and will result in the encoder being unable to record data. Because we aim at a real-time solution another approach can be used to get the signals, namely through the Tele-Infiniti device.

The Tele-Infiniti device ships with a small package of example software called the TTLAPI (Thought Technology Limited Application Programming Interface). The package contains examples on how to connect to the encoder using various programming languages like C/C++, VBA and Matlab code. The API can also be used to process recorded signal files stored on the CF card.

The Tele-Infiniti uses a wireless Bluetooth connection to transfer live signals to the listening computer. To receive signals the computer has to be connected and have the TTLAPI example program E-Z Scan running (see Figure 45). The application supports recording of all eight channels. The channels on the encoder differ in sampling rate; from 256samples/sec (channels C to J) up to 2048samples/sec (channels A and B). The sensors for BVP and SC were connected to channel D and H respectively. The program can show real-time graphs (see Figure 46 and Figure 47) of the signals and record the signals to disc for later use.



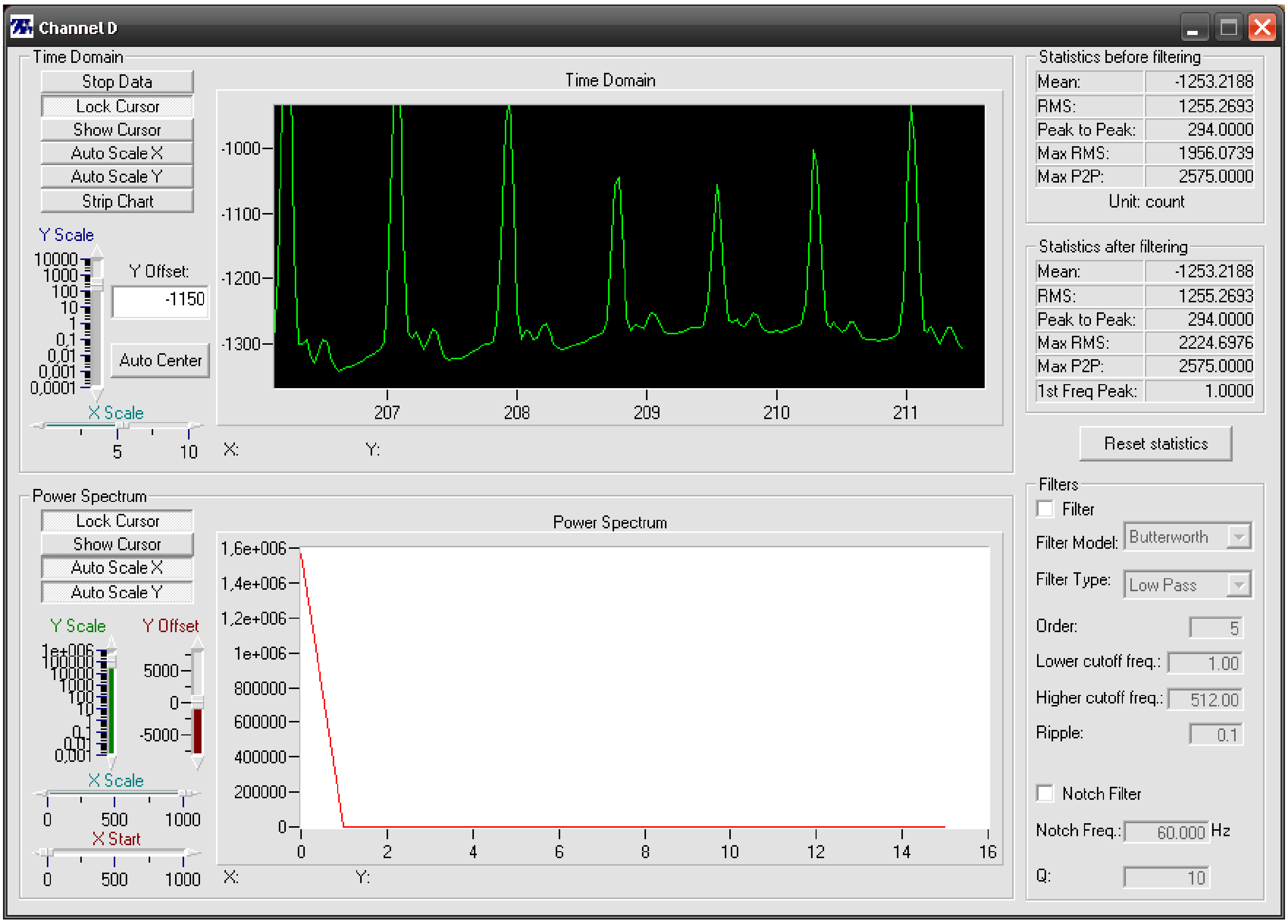
TTLLive Version: 02.01.1698
Connection Type: COM2
Connection Status: Connection opened.
Err. Count: 0
Buffering Time: 1 s
Playback speed: 1
Playback mode: LOOP
Playback position: 0.00 s
Refresh Time: 0.1 s
Record Gen. Name: C:\Documents and Settings\sebbe.0019D1A92

Encoder Type:	ProComp+	Protocol:	BioGraph
Serial Number:	PC+02412	Encoder State:	On line
Encoder ID:	2412	Encoder Time:	1899-Dec-30, 00:00:00
Firmware Version:	255.255.65535	Battery Level:	5.462000 V
Hardware Version:	255.255	Battery Level Pct:	81.278336 %

	Ch #A	Ch #B	Ch #C	Ch #D	Ch #E	Ch #F	Ch #G	Ch #H
State:	Uncon.	Uncon.	Uncon.	Norm.	Uncon.	Uncon.	Uncon.	Norm.
Unit:	count	count	count	count	count	count	count	count
ID:	Uncon.	Uncon.	Uncon.	???	Uncon.	Uncon.	Uncon.	???
Raw:				-1082.000				-3042.000
Mean:	0.000	0.000	0.000	-1260.375	0.000	0.000	0.000	-3034.000
RMS:				1170.061				3042.000
P2P:	0.000	0.000	0.000	223.000	0.000	0.000	0.000	8.000
RMSm:				2224.698				3141.000
P2Pm:	0.000	0.000	0.000	2575.000	0.000	0.000	0.000	51.000

Units: COUNTS

Figure 45. Screen dump of Thought Technology's E-Z Scan application for monitoring and recording live physiology signals



Y Offset: -1150

Statistics before filtering

Mean:	-1253.2188
RMS:	1255.2693
Peak to Peak:	294.0000
Max RMS:	1956.0739
Max P2P:	2575.0000

Unit: count

Statistics after filtering

Mean:	-1253.2188
RMS:	1255.2693
Peak to Peak:	294.0000
Max RMS:	2224.6976
Max P2P:	2575.0000
1st Freq Peak:	1.0000

Filters

☐ Filter

Filter Model: Butterworth

Filter Type: Low Pass

Order: 5

Lower cutoff freq.: 1.00

Higher cutoff freq.: 512.00

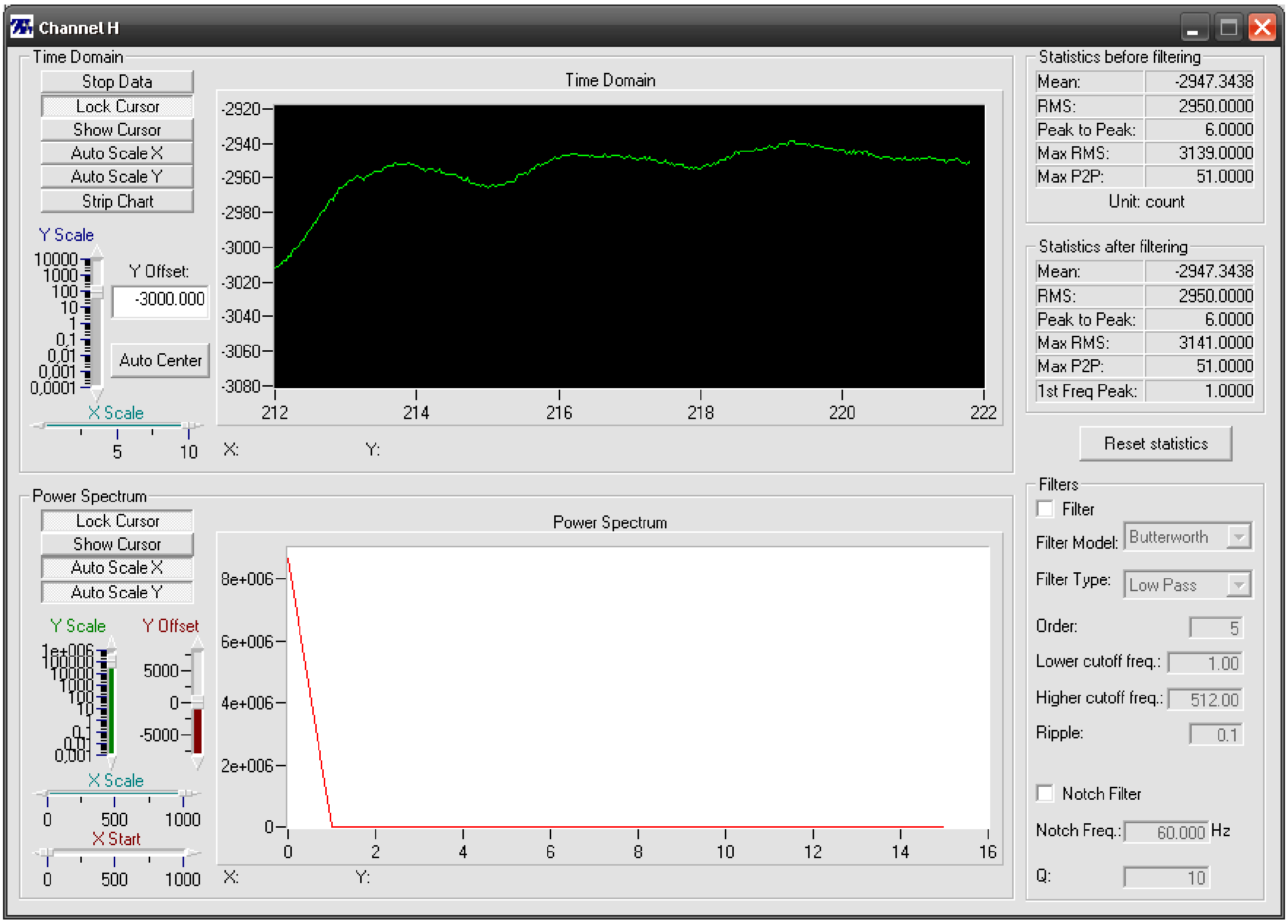
Ripple: 0.1

☐ Notch Filter

Notch Freq.: 60.000 Hz

Q: 10

Figure 46. The BVP graph window showing the live signal from the encoder



Y Offset: -3000.000

Statistics before filtering

Mean:	-2947.3438
RMS:	2950.0000
Peak to Peak:	6.0000
Max RMS:	3139.0000
Max P2P:	51.0000

Unit: count

Statistics after filtering

Mean:	-2947.3438
RMS:	2950.0000
Peak to Peak:	6.0000
Max RMS:	3141.0000
Max P2P:	51.0000
1st Freq Peak:	1.0000

Filters

☐ Filter

Filter Model: Butterworth

Filter Type: Low Pass

Order: 5

Lower cutoff freq.: 1.00

Higher cutoff freq.: 512.00

Ripple: 0.1

☐ Notch Filter

Notch Freq.: 60.000 Hz

Q: 10

Figure 47. The graph window showing the live skin conductance signal

The format the EZ-Scan application stores the signal data is undocumented and therefore the format had to be heuristically decoded into a format that makes sense in regard to the way physiological signals is normally looks like.

The channel specifications stated that the channels used where at 256samples/sec but our investigation indicates that a total of 512 values were recorded per second. This was measured by letting the application record for 10 seconds and divide with the number of recorded values. As it can be seen in Figure 48 the BVP signal looks nothing like a real BVP signal. If one looks close enough a pattern arises in the signal indicating that the signal is interweaved many times with small delays.

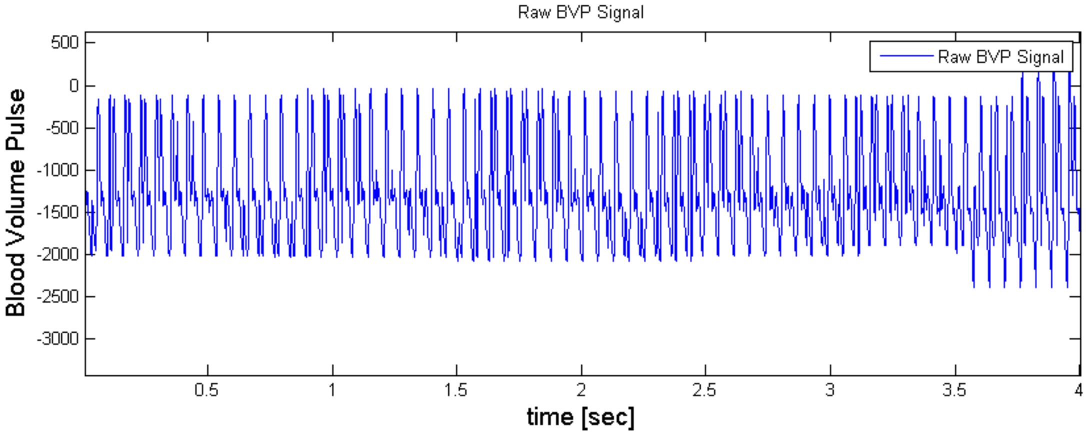


Figure 48. The raw BVP signal recording using E-Z Scan and visualized in Matlab

When analysed in a spreadsheet it became clear that the pattern were indeed interweaved signals. The values were grouped in blocks of 32 values where one value would change repeating all the same values that was found in the previous group. An example of this behaviour is visualised in Figure 49 where all values are the same in one line as in the previous except for one value.

X_{1}	X_{2}	X_{3}	X_{4}	X_{5}	X_{6}	X_{7}	X_{8}	X_{9}	X_{10}	X_{12}	X_{13}	X_{11}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}	X_{20}	X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}	X_{28}	X_{29}	X_{30}	X_{31}	X_{32}
X_{33}	X_{2}	X_{3}	X_{4}	X_{5}	X_{6}	X_{7}	X_{8}	X_{9}	X_{10}	X_{12}	X_{13}	X_{11}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}	X_{20}	X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}	X_{28}	X_{29}	X_{30}	X_{31}	X_{32}
X_{33}	X_{34}	X_{3}	X_{4}	X_{5}	X_{6}	X_{7}	X_{8}	X_{9}	X_{10}	X_{12}	X_{13}	X_{11}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}	X_{20}	X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}	X_{28}	X_{29}	X_{30}	X_{31}	X_{32}
X_{33}	X_{34}	X_{35}	X_{4}	X_{5}	X_{6}	X_{7}	X_{8}	X_{9}	X_{10}	X_{12}	X_{13}	X_{11}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}	X_{20}	X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}	X_{28}	X_{29}	X_{30}	X_{31}	X_{32}

Figure 49. Examples of the encoded signal, with repeating values

We put together a small algorithm that could iterate the signal while detecting changes and only copy changes to a new array. The new array size indicated that only 32 samples were recorded per second.

```
    while (i < length(bvpSignal)-32)
      if (bvpSignal(i) ~= bvpSignal(i+32))
        decodedSignal(j) = bvpSignal(i);
        j = j + 1;
      end
      i = i + 1;
    end
```

Figure 50. Matlab algorithm to decode the signals

After completed decoding the signal shown in Figure 48 is similar to the signal that can be seen in the E-Z Scan application's graph view of the live signal. We did not succeed in converting the signal into meaningful values like mmHg for BVP and micro-Siemens for SC. The assumption was made that the values are ok, as long as the emotion model is trained with "weird" values and later use "weird" values for real-time emotion prediction.

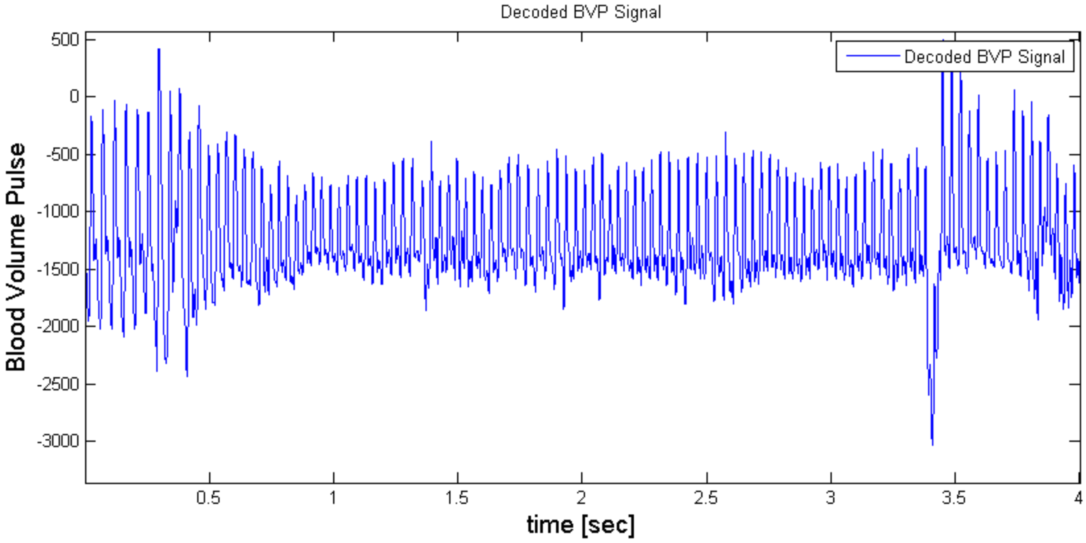


Figure 51. The decoded BVP signal visualized in Matlab

APPENDIX D – FULL FEATURE LIST

BVP and HR Features								
	Relaxation		Frustration		Excitement		Boredom	
Feature	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*
$E\{b\}$	0.22	**0.0038**	-0.17	**0.0448**	-0.06	0.2858	0.12	**0.0494**
$\sigma\{b\}$	0.05	0.2557	0.11	0.0877	-0.03	0.4253	-0.09	0.1808
$E\{IBamp\}$	0.02	0.3714	0.01	0.3679	0.03	0.2858	-0.15	**0.0494**
$nBeats$	-0.05	0.3714	0.14	**0.0448**	0.00	0.4253	0.00	0.4278
$E\{h\}$	0.02	0.3714	0.14	**0.0448**	0.00	0.4253	0.03	0.2923
$max\{h\}$	0.14	0.0494	-0.02	0.5000	0.00	0.4253	0.06	0.1808
$min\{h\}$	0.02	0.3714	-0.17	**0.0448**	-0.03	0.4253	0.12	**0.0494**
D^h	0.08	0.1620	0.08	0.1553	0.00	0.4253	-0.09	0.1808
t^h_{min}	0.05	0.2557	-0.20	**0.0205**	-0.06	0.2858	0.12	**0.0494**
t^h_{min}	0.10	0.0939	-0.05	0.3679	0.00	0.4253	0.12	**0.0494**
D^h_t	-0.05	0.3714	0.11	0.0877	0.06	0.1725	-0.03	0.4278
δ^b_1	-0.05	0.3714	-0.11	0.1553	0.06	0.1725	0.09	0.1002
δ^b_2	-0.05	0.3714	-0.08	0.2498	0.06	0.1725	0.03	0.2923
$\delta^b_{\lvert 1 \rvert}$	-0.02	0.5000	0.11	0.0877	0.09	0.0925	-0.09	0.1808
$\delta^b_{\lvert 2 \rvert}$	-0.02	0.5000	0.11	0.0877	0.09	0.0925	-0.09	0.1808
$hann\{b\}$	0.05	0.2557	-0.02	0.5000	0.00	0.4253	0.03	0.2923
$E\{b\}$	0.05	0.2557	0.02	0.3679	0.09	0.0925	0.03	0.2923
ULF	0.02	0.3714	0.02	0.3679	-0.09	0.1725	-0.06	0.2923
VLF	0.05	0.2557	-0.05	0.3679	-0.03	0.4253	-0.06	0.2923
LF	-0.04	0.3714	-0.02	0.5000	0.03	0.2858	-0.03	0.4278
HF	0.17	0.0235	-0.05	0.3679	-0.09	0.1725	-0.03	0.4278
$\sigma\{RR\}$	0.05	0.2557	-0.14	0.0877	-0.06	0.2858	0.03	0.2923
pRR_{50}	-0.02	0.5000	0.08	0.1553	0.03	0.2858	0.00	0.4278
$RMSSD_{RR}$	0.08	0.1620	-0.11	0.1553	-0.09	0.1725	0.00	0.4278

BVP and HR Features						
	Fun		Challenge		Predictability	
Feature	$c(\overrightarrow{z})$	*p-value*	$c(\overrightarrow{z})$	*p-value*	$c(\overrightarrow{z})$	*p-value*
$E\{b\}$	0.02	0.3679	-0.14	0.1163	0.14	**0.0307**
$\sigma\{b\}$	0.08	0.1553	0.02	0.3830	-0.05	0.3555
$E\{IBamp\}$	0.14	**0.0448**	0.02	0.3830	-0.14	0.0680
$nBeats$	-0.05	0.3679	-0.08	0.2757	0.05	0.2291
$E\{h\}$	-0.02	0.5000	-0.08	0.2757	0.08	0.1325
$max\{h\}$	-0.08	0.2498	-0.14	0.1163	0.14	0.0307
$min\{h\}$	-0.02	0.5000	-0.11	0.1856	-0.14	0.0680
D^h	-0.02	0.5000	-0.05	0.3830	0.14	**0.0307**
t^h_{min}	-0.08	0.2498	-0.17	0.0676	0.08	0.1325
t^h_{min}	-0.20	**0.0205**	0.02	0.3830	0.05	0.2291
D^h_t	-0.05	0.3679	0.05	0.2757	-0.08	0.2291
δ^b_1	-0.02	0.5000	-0.14	0.1163	-0.02	0.5000
δ^b_2	-0.02	0.5000	-0.11	0.1856	-0.02	0.5000
$\delta^b_{\|1\|}$	0.11	0.0877	0.02	0.3830	-0.14	0.0680
$\delta^b_{\|2\|}$	0.11	0.0877	0.02	0.3830	-0.14	0.0680
$hann\{b\}$	0.11	0.0877	0.05	0.2757	-0.08	0.2291
$E\{b\}$	0.02	0.3679	-0.08	0.2757	0.02	0.3555
ULF	0.05	0.2498	-0.02	0.5000	0.05	0.2291
VLF	0.02	0.3679	-0.02	0.5000	-0.08	0.2291
LF	0.05	0.2498	-0.02	0.5000	-0.02	0.5000
HF	0.08	0.1553	-0.11	0.1856	0.17	0.0121
$\sigma\{RR\}$	-0.02	0.5000	-0.05	0.3830	0.02	0.3555
pRR_{50}	0.08	0.1553	0.11	0.1163	0.05	0.2291
$summation$	0.02	0.3679	0.05	0.2757	0.08	0.1325

SC Features								
	Relaxation		Frustration		Excitement		Boredom	
Feature	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*
$E\{s\}$	0.05	0.2557	-0.02	0.5000	0.09	0.0925	-0.12	0.1002
$\sigma\{s\}$	0.08	0.1620	-0.11	0.1553	-0.06	0.2858	-0.06	0.2923
s_{in}	-0.05	0.3714	0.08	0.1553	0.09	0.0925	0.12	**0.0494**
s_{last}	-0.05	0.3714	0.14	**0.0448**	0.12	**0.0436**	-0.03	0.4278
$max\{s\}$	0.05	0.2557	0.02	0.3679	0.06	0.1725	-0.09	0.1808
$min\{s\}$	0.08	0.162	0.02	0.3679	0.03	0.2858	-0.06	0.2923
D^s	0.17	**0.0235**	0.02	0.3679	-0.03	0.4253	-0.03	0.4278
t^s_{min}	-0.11	0.162	-0.08	0.2498	0.09	0.0925	0.06	0.1808
t^s_{min}	-0.05	0.3714	-0.05	0.3679	0.09	0.0925	-0.06	0.2923
D^s_t	-0.08	0.2557	0.08	0.1553	0.06	0.1725	-0.06	0.2923
δ^s_1	0.08	0.162	0.05	0.2498	-0.06	0.2858	-0.06	0.2923
δ^s_2	0.08	0.162	0.05	0.2498	-0.06	0.2858	-0.06	0.2923
$\delta^s_{\|1\|}$	-0.11	0.162	0.11	0.0877	0.15	**0.0178**	-0.18	**0.0214**
$\delta^s_{\|2\|}$	-0.17	**0.0494**	0.14	**0.0448**	0.18	**0.0063**	-0.18	**0.0214**

SC Features						
	Fun		Challenge		Predictability	
Feature	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*
$E\{s\}$	0.11	0.0877	-0.02	0.5000	0.05	0.2291
$\sigma\{s\}$	-0.02	0.5000	-0.08	0.2757	0.02	0.3555
s_{in}	-0.05	0.3679	-0.08	0.2757	-0.05	0.3555
s_{last}	0.02	0.3679	0.08	0.1856	-0.08	0.2291
$max\{s\}$	0.05	0.2498	0.02	0.3830	-0.02	0.5000
$min\{s\}$	-0.08	0.2498	0.05	0.2757	0.20	**0.0041**
D^s	0.17	**0.0205**	-0.02	0.5000	0.02	0.3555
t^s_{min}	-0.14	0.0877	-0.08	0.2757	0.02	0.3555
t^s_{min}	-0.02	0.5000	-0.02	0.5000	-0.05	0.3555
D^s_t	-0.02	0.5000	0.02	0.3830	0.05	0.2291
δ^s_1	-0.02	0.5000	0.11	0.1163	-0.02	0.5000
δ^s_2	-0.02	0.5000	0.11	0.1163	-0.02	0.5000
$\delta^s_{\|1\|}$	0.26	**0.0009**	0.08	0.1856	-0.08	0.2291
$\delta^s_{\|2\|}$	0.23	**0.0030**	0.08	0.1856	-0.05	0.3555

Wii Features								
	Relaxation		Frustration		Excitement		Boredom	
Feature	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*
$min\{a_x\}$	-0.56	**0.0000**	-0.53	**0.0000**	-0.39	**0.0000**	-0.45	**0.0000**
$max\{a_x\}$	-0.35	**0.0001**	-0.2	**0.0205**	-0.27	**0.0005**	-0.18	**0.0214**
$E\{a_x\}$	0.02	0.3714	-0.02	0.5000	-0.06	0.2858	0.00	0.4278
$min\{a_y\}$	0.05	0.2557	0.02	0.3679	0.00	0.4253	-0.09	0.1808
$max\{a_y\}$	-0.41	**0.0000**	-0.38	**0.0000**	-0.33	**0.0000**	-0.36	**0.0000**
$E\{a_y\}$	0.02	0.3714	0.08	0.1553	0.03	0.2858	0.00	0.4278
$min\{a_z\}$	-0.38	**0.0000**	-0.47	**0.0000**	-0.33	**0.0000**	-0.21	**0.0081**
$max\{a_z\}$	-0.41	**0.0000**	-0.26	**0.0030**	-0.18	**0.0178**	-0.3	**0.0002**
$E\{a_z\}$	-0.05	0.3714	-0.02	0.5000	0.06	0.1725	-0.03	0.4278
$min\{P\}$	-0.17	**0.0494**	0.02	0.3679	0.00	0.4253	-0.09	0.1808
$max\{P\}$	-0.17	**0.0494**	-0.02	0.5000	0.06	0.1725	-0.06	0.2923
$E\{P\}$	-0.08	0.2557	0.11	0.0877	0.09	0.0925	-0.06	0.2923
t^{idle}	-0.11	0.1620	0.02	0.3679	0.15	**0.0178**	0.09	0.1002
$E\{P^f\}$	-0.02	0.5000	-0.08	0.2498	0.00	0.4253	0.06	0.1808
$E\{P^{pf}\}$	-0.02	0.5000	-0.05	0.3679	0.00	0.4253	0.03	0.2923
$H\{spell\}$	-0.23	0.0100	0.05	0.2498	0.09	0.0925	-0.06	0.2923
$nSpell$	-0.2	**0.0235**	-0.26	**0.0030**	-0.06	0.2858	-0.15	**0.0494**
$E\{P^g\}$	-0.08	0.2557	0.02	0.3679	-0.09	0.1725	-0.09	0.1808
t^{lb}	-0.2	**0.0235**	0.08	0.1553	0.00	0.4253	-0.12	0.1002
$E\{t^{lb}\}$	-0.08	0.2557	0.05	0.2498	-0.09	0.1725	-0.12	0.1002
P^{lb}	-0.14	0.0939	0.02	0.3679	-0.03	0.4253	-0.15	**0.0494**
$nDefence$	-0.29	**0.0013**	-0.11	0.1553	-0.12	0.0925	-0.18	**0.0214**
$E\{t^{reac}\}$	-0.29	**0.0013**	0.11	0.0877	0.12	**0.0436**	-0.09	0.1808
$E\{t^{pfIdle}\}$	-0.05	0.3714	-0.05	0.3679	-0.03	0.4253	0.06	0.1808

Wii Features						
	Fun		Challenge		Predictability	
Feature	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*	$c(\vec{z})$	*p-value*
$min\{a_x\}$	-0.50	**0.0000**	-0.65	**0.0000**	-0.44	**0.0000**
$max\{a_x\}$	-0.32	**0.0003**	-0.38	**0.0001**	-0.23	**0.0041**
$E\{a_x\}$	-0.11	0.1553	-0.14	0.1163	0.02	0.3555
$min\{a_y\}$	0.11	0.0877	0.08	0.1856	0.02	0.3555
$max\{a_y\}$	-0.35	**0.0001**	-0.50	**0.0000**	-0.32	**0.0001**
$E\{a_y\}$	0.05	0.2498	0.14	0.0676	0.05	0.2291
$min\{a_z\}$	-0.50	**0.0000**	-0.62	**0.0000**	-0.41	**0.0000**
$max\{a_z\}$	-0.38	**0.0000**	-0.35	**0.0004**	-0.20	**0.0121**
$E\{a_z\}$	0.11	0.0877	-0.02	0.5000	-0.11	0.1325
$min\{P\}$	-0.08	0.2498	-0.11	0.1856	-0.20	**0.0121**
$max\{P\}$	0.02	0.3679	-0.11	0.1856	0.02	0.3555
$E\{P\}$	0.11	0.0877	0.26	**0.0033**	-0.05	0.3555
t^{idle}	-0.05	0.3679	-0.17	0.0676	-0.05	0.3555
$E\{P^f\}$	0.05	0.2498	0.05	0.2757	-0.08	0.2291
$E\{P^{pf}\}$	0.05	0.2498	0.14	0.0676	-0.11	0.1325
$H\{spell\}$	-0.05	0.3679	0.02	0.3830	-0.05	0.3555
$nSpell$	-0.08	0.2498	-0.14	0.1163	-0.26	0.0012
$E\{P^g\}$	0.14	0.0448	0.23	**0.0080**	-0.14	0.0680
t^{lb}	-0.05	0.3679	0.17	**0.0362**	-0.17	**0.0307**
$E\{t^{lb}\}$	-0.08	0.2498	0.08	0.1856	-0.11	0.1325
P^{lb}	-0.02	0.5000	0.11	0.1163	-0.11	0.1325
$nDefence$	-0.17	0.0448	-0.14	0.1163	-0.20	**0.0121**
$E\{t^{reac}\}$	-0.14	0.0877	0.14	0.0676	-0.20	**0.0121**
$E\{t^{pfIdle}\}$	-0.05	0.3679	-0.05	0.3830	-0.11	0.1325