Towards Augmented Cognition in Games
- Psychophysiological signals and game events recognized through artificial neural network

Medialogy M.Sc
Master Thesis

By Andreas Wulff-Jensen
Aalborg University Copenhagen

Spring 2016

Aalborg University Copenhagen

Aalborg University Copenhagen
Frederikskaj 12,
2450 København S, Denmark

Semester Coordinator:

Secretary:

Semester:
MED 10

Title:
Towards Augmented Cognition in Games - Psychophysiological signals and game events recognized through Artificial Neural Network

Project Period:
February 2016 – May 2016

Semester Theme:
Master Thesis

Supervisor(s):
Luis Emilio Bruni

Project group no.:
20113367

Members:
Andreas Wulff-Jensen

____

Abstract:. In this thesis a few steps towards augmented cognition in games has been taken. By conducting an experiment where 31 subjects played Super Mario while Their Electroencephalogram(EEG) and Eye-tracking where measured.
Through analysis of the game 22 different events were tracked. 3 second epochs of the psychophysiological signals were divided between the different experienced game events, through which one dataset per subject was created those were inspected for abnormalities, which led to an exclusion of 18 out of 31 subjects and 18 out of 22 game events. Through the use of Independent Component Analysis significant features in the EEG and Eye-tracking data were found and given to the Artificial Neural Network (ANN), which was devised through the Encog framework. The results from the ANN, measured as accuracy and analyzed through ANOVA test and Kruskal–Wallis ANOVA test, were promising. Showing significant differences between the trained models based on EEG channels in respect to all four game events (means = 57.94% - 67.30%). The eye-tracking data significant differences between its measurements in relation to two of the events were found as well (means = 57.78% -69.48%). Through the grand average of the accuracies in respect to the different events significant differences both in relation to the accumulated EEG accuracies (means = 59.27% - 65.97%) and the accumulated Eye-tracking accuracies were found (means = 59-31% - 66.84%).
Albeit these results are promising they are not as prominent as other electroencephalogram recognition studies. This could be grounded in the very nature of the data, which can be harder to recognize and the structure of the ANN, which could be optimized further.
Conclusively this thesis points towards that it is possible to recognize psychophysiological patterns related to game events through an ANN.

Copies: 1
Pages: 102
Finished: 26/5 2016

Towards Augmented Cognition in Games
– Psychophysiological signals and game events recognized through artificial neural network

Table of Content

1 Introduction

Games can be viewed as a complex stimulus eliciting cognitive demanding tasks, visual appealing content and multilayered audio signals. All of which the gamer perceives in a top-down fashion to encourage embodied reactions towards the next course of actions. The receiving process and the immediate unconscious physiological reaction of the body are rather multifaceted and can tell us a lot about the state of the person and the stimulus he is receiving. However, taking two complex streams of signals and relating them directly to each other comes with a huge amount of uncertainty, as psychophysiological signals have the property of being mapped many to one. This essentially means that the measured signal can be elicited by a multitude of different kinds of stimuli.

Luckily, games can provide us with a controllable environment in which every tiny bit can be monitored, so by eliciting the same kinds of game events multiple times there might be a certain chance that the observed psychophysiological reactions will be the same, thus indicating a possible correlation between the specific pattern of reactions and the game events.

When such patterns are known, assumptions about the cognitive, psychological and affective mechanisms can be devised based on previous phenomena which had the equivalent psychophysiological features. These correlations can help expand the usefulness of games' investigations to other domains besides pure entertainment.

Referring back to the psychophysiological signals, great amount of data may occur, thus investigating it manually for patterns and relations might be troublesome and time consuming. Therefore, inquiries in deep machine learning algorithms, such as artificial neural networks, need to be done, as they are quite powerful predictors as well. Typically, they have been used within object recognition, where the algorithms have been trained by a big dataset of images to recognize different objects. Images in themselves consist of big arrays of data, and these algorithms have proven to have a quite high accuracy even if the object is somewhat obscured. This can be related to the psychophysiological signal if it is a little different than the perfect template typically elicited at certain game events. Thereby, they can be difficult to observe.

With these different elements in mind the investigation will proceed by exploring the following initial problem statement.

Can game events be predicted based on their corresponding psychophysiological features through the use of a deep machine learning algorithm?

2 investigation

Through this investigation there will be an elaborate description of the game including its features. Their temporal and spatial nature in the game. It will be important to determine how well they can be recognized and how often they will happen. It can easily be a huge list of features if all games are applied. Therefore, it will be delimited down to those found in a Super Mario game, because the source code is easy available [1] and it has been used as a test bed for multiple other experiments for instance within affective games [2]–[5] and AI research [6], [7]

When the different game features have been discussed the report will continue by looking at the psychophysiological signals and features. How they can be extracted and relations to stimuli found in previous studies. Before we reach this discussion it is worth mentioning that there will only be a focus on Electroencephalogram (EEG) and Eye-tracking as those instruments are available in the Augmented Cognition Lab, and we have successfully synchronized online together in a previous experiment protocol [8], [9].

At the end of this investigation the deep machine learning algorithms will be explained and examined with respect to studies recognizing psychophysiological patterns. In conjunction with the previous discussed topics, the algorithm and the features will be chosen to produce and fulfill the final problem statement.

2.1 The Game – Super Mario

In this game you play as the little game character called Mario. Your goal is to reach the end of the game level, which is marked by two poles and a white elevated marker (see Figure 2). On the way towards the goal Mario has to go through different jumping puzzles, avoid or kill enemies, collect coins and collectable power up items. Mario can move back and forth, duck, sprint, jump and shoot. However, he can only shoot if he is in his third and highest form (see Figure 1). Mario can be in three different forms. In the first form he is small, and is killed by a single touch of an enemy. However, he is capable of jumping on them, thus killing them instead. He can get to the second form if he picks up a mushroom (see Figure 1 under collectables). Now he is capable of smashing boxes and when he is hit by an enemy he will degenerate to the first form. By entering the third form, which he does by picking up a flower, he will gain the ability to shoot. In case he is hit by an enemy, he will degenerate to the second form.

Figure 1 shows different enemies, collectables and forms of Mario. The first collectable is the mushroom, the second collectable is the flower and the third is the coin. The forms of Mario are in order of strength, where if you play with the last one you are able to shoot.

Figure 2 shows Mario in his first form jumping into the finish line

2.2 Game features

Game features are essentially the building blocks of a game. A feature can be spatial, discrete, temporal, audible and visual. An interplay between these features can be called an event, if one feature is left out the event is not correctly constructed and will not have the same effect. Moreover, each feature can elicit its own effect as well. If a combination of different features is present at the same time, one of the features might be dominant and thus its effect will become the most evident one.

In this section the different kinds of features will be inspected and elaborated on in relation to EEG and other psychophysiological domains. Afterwards, the Super Mario game, will be analyzed in relation to those features.

2.2.1 Discrete and continuous visual

A discrete visual feature is a features, which temporarily only occurs momentarily. The feature is recognizable for the player, which means that its appearance should not be shorter than the time it takes an average human to process. According to [10] if a complex imagery of nature or an animal appears in front of the subjects for 20ms, the observable differences in the EEG event related potential (ERP) is certain approximately 150ms after the onset of the stimulus. When the subjects consciously detect that the image depicts an animal they had to release a button. This process took between 382ms – 567ms. A related work [11] about visual attention, simpler stimuli being rectangular white boxes were shown. They appeared for 50ms and the ERP reaction around 90-140ms after the onset. When the white boxes were cued a higher EEG voltage amplitude were measured compared to none cued boxes.
based on this evidence, a discrete visual stimulus is a very short event lasting around 20-50ms.
A continuous visual feature appears on the screen until it is erased as a consequence of some sort of action. This feature can very well be a combination of fast successive discrete visual features, which seems to be producing a coherent lasting stimulus. This can either be a sequence of snapshots (frames) that induce motion or a stable image due to the repeated firing of the cells in the retina [12]. Moreover, to produce the illusion of a continuous visual stimulus the firing rate of the cells in the retina is unknown. However, the important point is the difference between this feature and the discrete feature, which is whether or not the stimulus lasts on the screen.

2.2.2 Simple and complex visual

A simple visual feature has a basic shape like a circle or a square, consist of one or few colors and do not have any textures or high frequency details [13], [14]. Opposite, a complex visual feature can consist of odd shapes, a great variety of colors and high detailed texture, like for instance a face or a peacock [13], [14]. In Eng, Chen, Jiang and Walle's [13] research they studied how fast humans can search through simple visual stimuli versus complex visual stimuli. They found it was quicker to search through a series of simple stimuli. This suggest that recognizing and acting upon a simple stimulus might indeed be faster than a complex stimulus.

2.2.3 Discrete and complex audio

A discrete audio feature is somewhat similar to a discrete visual feature. It appears briefly without any repetitions. This feature has its lower limit, which is the temporal perceivable limit of the human auditory perceptual system. In [15] this limit was documented to be 400Hz, which means that the auditory system could detect changes in modulation of the sound smaller than every 2.5ms. Furthermore, the temporal distance between discrete sounds can be as small as 50 ms [16], [17]. This implies a discrete audio feature can very easily be continued by another audio feature right after the stimulus has ended and still be recognizable as an individual feature. In relation to the ERP a short discrete audio feature may elicit an

auditory N1 (The N1 component will be visited in section 2.3.1.1) , which occurs 90- 140 ms after onset of an audio stimulus [18].
On the other hand, an audio feature is continuous, when it keeps on playing or repeating itself until the source is erased. A continuous audio feature can also contain breaks as long as the consecutive signal can be perceived as belonging to the previous heard signal.

2.2.4 Simple and complex audio

A simple audio feature is defined in the frequency space, if only a few frequencies composes the audio or the used frequency spectrum is narrow, the feature is seen as simple. For instance, in [19] they decompose nature sound into simple frequency features in order to classify them. If such features stand-alone, they are considered as a simple audio feature.
A complex audio feature uses a broad frequency spectrum, and may very easily be composed by many simple audio features.

2.2.5 Game features in relation to Super Mario Game

The visual part of the game consists of a combination of all the mentioned features, but with primary inclusion of continuous complex visual features.
In Figure 3 different visual features are depicted. Super Mario consist of 14 within the Continuous Complex Visuals category. Their common attributes are their detailed textures and multiple colored color schemes. Some of them also have a non-basic shape like e.g.: the turtle. Within this category there are basic shapes as well, even though this attribute belongs to the simple visuals the objects' texture gives them complexity. All of these features are present on the screen until they are erased by an action.
Those features within the Continuous Simple Visuals has a simple color scheme with only few colors, a low detailed texture and their shapes are basic squares or circles. One noteworthy object is the coin. It is both within the Continuous and the Discrete categories. The coin has the property of either being spawned into the game world until it is picked up or be elicit briefly as a reaction upon Mario hitting a box.
There is only one Discrete Complex Visual in the game, which is a short animation of Mario and a girl enclosed in a circle. This feature appears on the screen for a few seconds when you accomplish a level. It has a rather detailed texture, multiple colors and a complex inner shape, but a basic circle as an outer shape.

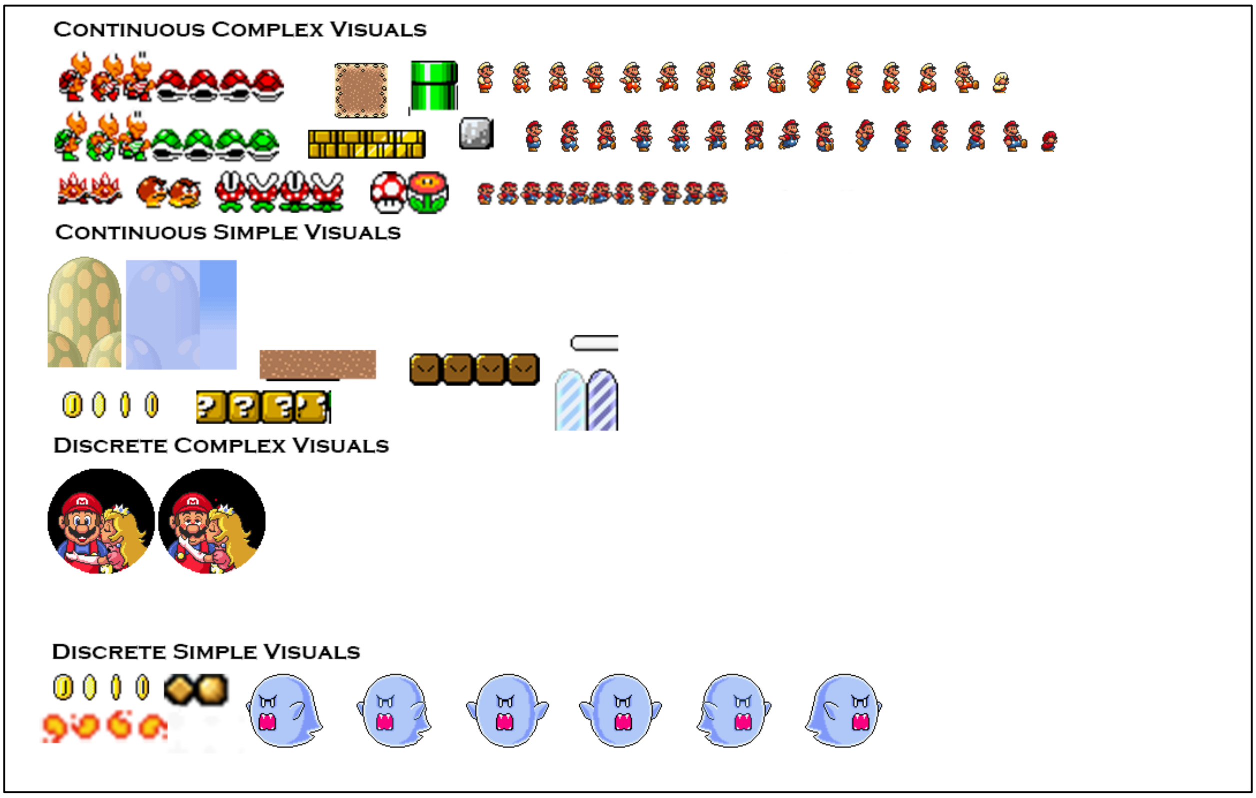

Figure 3 shows an overview of the different kinds of visual features appearing in the game Super Mario

The last group of visuals is the Discrete Simple Visuals. These appear briefly, spanning from a few hundred milliseconds to a few seconds. Their shapes are simple circles or squares and they have low detailed textures.

The audio in Super Mario is a little differently distributed than the visuals (see Figure 4). It is more discrete than continuous, but nearly equally distributed between complex and simple.
The continuous complex audio is the score running in the background, it is playing throughout the whole game as the only sound that is constantly present. The score is complex as it covers a broad frequency spectrum and contains different melody pieces.
The Discrete Complex Audio features contain different small sound samples elicited at different interactions such as when you accomplish a level or pick up a power up item. All of the sound samples cover a broad frequency spectrum, and consist either of a melody or sort of noise for example the sound when a box is being broken by Mario. Moreover, there is a melodic sound which is placed between the discrete and the continuous audio area, it is the sound elicited when Mario dies. This sound starts with three small discrete beeps, which only lasts 100 milliseconds. They are continued by a discrete complex section which last 2 seconds and 750 milliseconds. This part is a melody and covers a broader frequency spectrum than the first part (The frequencies and temporal information is accessed through Adobe Audition [20]).
The rest of the Simple Discrete audio features are short samples below 400 milliseconds and only consist

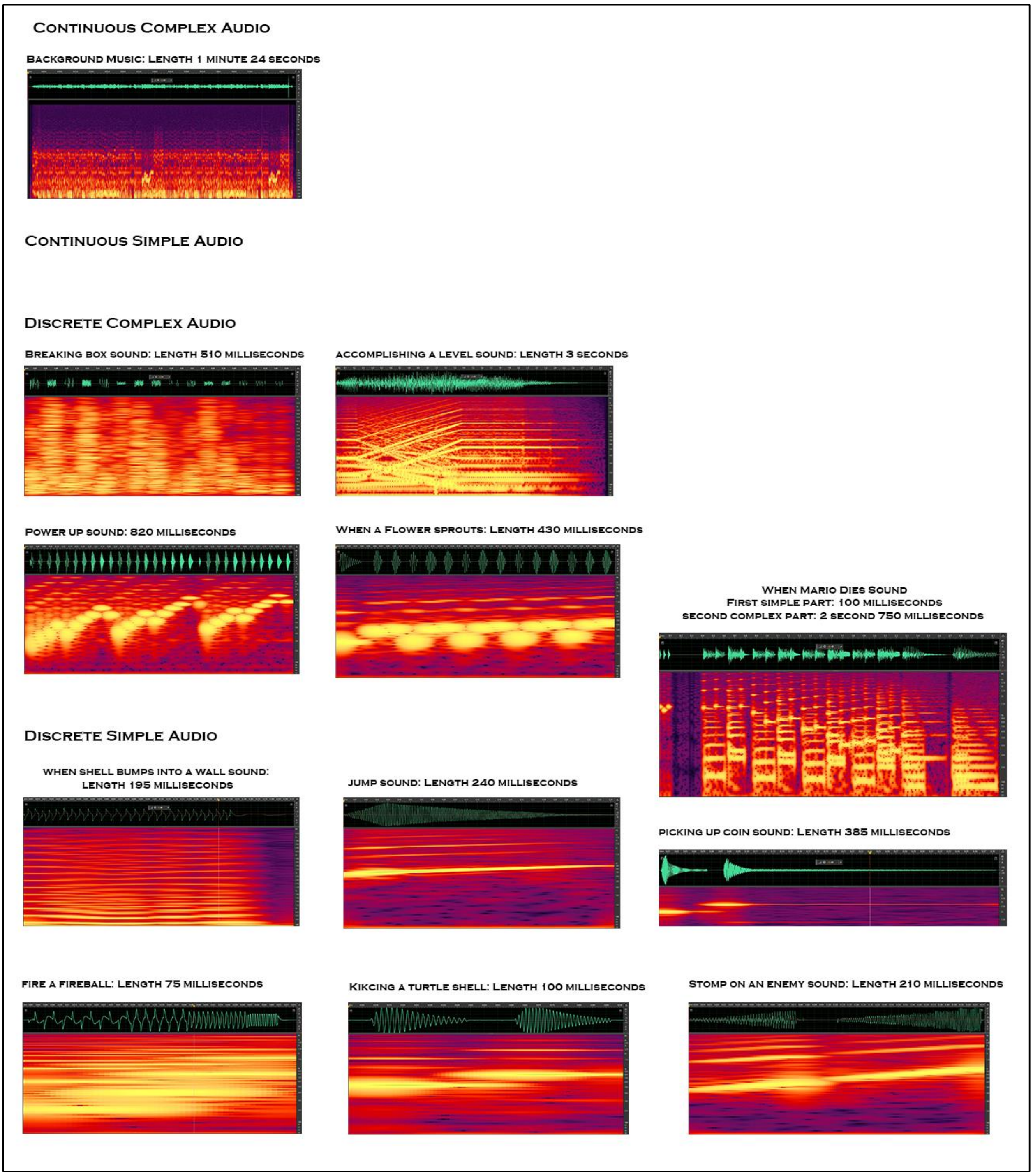

Figure 4 shows screenshots from Adobe Audition Cs6 in which the different sounds were analyzed. Through the bright yellow areas, the main frequencies can be seen. The bright green part shows the amplitude of the sound.

of one or two burst of sounds. These have a narrow frequency spectrum and as all the other discrete sounds they are elicited at different points of interaction.

The different features shown in Figure 3 and Figure 4 does not occur in isolation in the game. Their appearance is typically a mix, and will give the player a holistic input. It is therefore important to note that the few previously mentioned EEG characteristics might not be present or might be altered when the user

is exposed to a combination of the different stimuli at once. Furthermore, the EEG signal elicited through the game might reflect the cognitive response to the stimuli rather than the pure sensation of them. Figure 5 showcases the combination of different features and actions. The static ones being the mountain background, the seven boxes, the writings, the red arrow in the top middle and the brown platform layout. All of these are within the continuous simple visual category. Since these are static they might only call a little amount of visual attention to them and thus low amount of cognitive processing [21].
The features which might call for some visual attention are the ones with a more dynamic (moving) nature. These are: The character Mario, the two enemies, the growing flower, the rotating coins, and the rotating question mark in the right side of the screen. Out of those items Mario and the flower might be the ones the subject most consciously will work on, as the flower is just about to appear based on the action of Mario bumping into the box [21]. At this very moment the subject might as well think about the next array of decisions to make and in which order [22]. For example, he could interested in killing the enemies as well. The flower will upgrade Mario to be able to shoot, thus a thought about moving towards the flower could also be part of the subject´s strategy. Lastly the subject could think about picking up the coins.

With these available actions the subject has to rank them in order of importance to reach different internal sub-goals, which have been ranked in order to reach the final goal (completing the level), and execute them in that order [22]. Since this is an ongoing process the subject is constantly reflecting and rearranging upon what he should do, thus he is cognitively active. This is hypothesizing and may or may not be reflected in the EEG.

In the following section literature within EEG and cognition field will be reviewed in order to clarify and to better foresee how the EEG signal caused by a different combination of features will look like.

Figure 5 shows a snapshot from the Super Mario game. Mario has just hit a box containing a flower, while an enemy is approaching and another is distancing from Mario

2.3 EEG features

In the literature EEG is explored as being capable of reflecting cognitive and perceptual processes. These processes have been related to different features found in the EEG signal. Such features can either be positive or negative charged voltage responses found milliseconds after the elicited stimuli (Event Related Potentials (ERPs[1])) ranging from an amplitude of -10 – 10 microvolt. Or a dominant frequency band (quantitative or continuous EEG[2]).

2.3.1 ERP

The first features which will be examined are the different responses which have been recorded in relation to particular time-locked stimuli. These are found through averaging all the EEG snippets related to the stimuli. These snippets are called epochs and they are typically 1100ms long 100ms before stimuli to 1000ms after [23], [24].

2.3.1.1 N1

The N1 event related potential is a negative voltage amplitude found 150 to 200 ms after an onset of a visual stimulus [25].It is mostly pronounced for attended[26] , discriminated [27], cued visual stimulus [26], or when the processing of attended location is heightened [28]. Furthermore the amplitude of N1 has been observed to be decreasing overtime as a result of attention fatigue, from an average of -3 or-4 microvolts to an average of -1 or-2 [29] thus related to goal-directed attention (top-down)[30] as opposed to stimulus-driven attention, which were discussed to be more prominent at the end of a three hour long experiment conducted by Boksem, Maijman and Lorist [29].
Similar to the visual stimuli, auditory stimuli elicit an N1 component as well. It has a higher voltage amplitude when attended and is also prominent in an oddball paradigm[3], where the mismatch negativity[4] is around the N1 mark showing an amplitude of -5 microvolts [31]. Moreover, the auditory N1 has a lower latency than the visual N1 and can be found between 50-150 ms.
A subcomponent, defined by a specific temporal area within the N1 is the N170. This negative potential can be found from 130 – 170 ms with a peak at 170 [32]. It has been associated with early face and object recognition. It has been demonstrated that human and animal faces have the same amplitude elicited (around -4 microvolts) [33]. Objects, such as cars, elicit a lower amplitude (around -3 microvolts) compared to words and faces (-6 microvolts) [32]. Words have proven to have a higher amplitude compared to symbols and pseudo words in the left occipital-temporal area compared to the right occipital-temporal area where their amplitudes were equal (In the left occipital-temporal area from -1 - -2 in the right approximately -1.5 microvolts) [34].

2.3.1.2 P1

Is a positive response occurring approximately 50 ms after the onset of an auditory stimuli and 100 ms after the onset of a visual stimuli [28]. The amplitude of the P1 (1-4 microvolts) have both been associated with the amount of attention that has been given to the stimuli and to when the reaction speed towards the stimuli has been emphasized [27], [35]. Thus also mere selective attention [36].

---

[1] ERP is a negative or positive voltage impulse found after the onset of a stimulus. It is correlated to it if the voltage impulse is still clear after 20 or more EEG data snippets called epochs have been averaged together.
[2] Continuous EEG are the different frequency bands which can be measured. These can prevail for longer time and can be independent from concrete stimulus.
[3] The oddball paradigm is a test procedure where a sequence of audio or visual stimuli is infrequently interrupted by a deviant stimulus.
[4] Mismatch negativity is a negative voltage component in the ERP elicited by odd or deviant stimuli.

The auditory P1 on the other hand is typically associated with sensory gating, which has been proven by playing paired audio stimuli with a short pause between the stimuli [37].

2.3.1.3 P2

The P2 Potential is a positive potential found 150-275ms after the onset of the stimulus. It has been seen in conjunction with the N1, which is typically referred to as the N1 P2 complex [28]. The P2 has a relation to selective attention tasks [38], short term memory task [39] and object categorization tasks [40], under which valence and arousal distinction of the presented pre-subject evaluated stimulus is found [38], [41]. In Carretie et al. study [41], about valence and P2, the stimuli with negative valence was observed to elicited a higher amplitude P2 than both stimuli with positive valence and neutral stimuli. Pernet et al. [40] studied the relation between the delay of the P2 and the difficulty of categorizing the stimuli. Noticing that reaction time for categorizing letters and geometric shapes were far shorter than structured textures and Asiatic characters. Ranging from 670ms to 1300ms. They connected those observations with the latency of the P2 for the categories. In relation to the negative valence P2 and the object categories, it has been found if a non-compressed image is followed by a compressed image a P2 is elicited[42]. This effect is not seen if the non-compressed image is followed by another non-compressed image. In a later study by Mustafa and Magnor [43], they use this fact to access the pleasantness of a picture in order to alter it based on the subject [43]. It is also worth noting that in Lindemann and Magnor's paper [42]. The latency of the negative potential elicited by the images can be interpreted as changing from being an N1 to an N2 if a non-compressed image is followed by a compressed image.

2.3.1.4 N2

This negative potential is elicited 200 – 350 ms after the onset of the stimulus. It has been found in go/no go experiments[5]. The amplitude of the N2 at the no go condition decreases as the probability of its occurrence rises. The N2 has as well been associated with novel, unusual and deviant stimulus in combination with oddball experiments, thus being a mismatch negativity. In sequence matching task experiments where the stimulus has been cued or anticipated beforehand, a non-cued or non- matching complex unusual stimuli elicits a N2 [28], [44]. Lastly an N2 has been found in relation to cognitive control paradigms, which is an immediate control of an action, such as changing the expectancy of an upcoming stimuli. An example of such is the stop signal paradigm, where a visual selection task can be continued by a stop stimulus [44].

2.3.1.5 P3/P300

The P3 also known as the P300 is a potential elicited 300 – 800ms after the onset of the stimulus [45]. It is sensitive to attention, as it is only elicited when then the stimulus is attended [28], [45], [46]. The frequency of the attended stimuli has an impact as well, the more frequent the lower amplitude does the P3 has. However, it is only true if the frequency is within 6-8 seconds [46]. The latency of the P3 has been associated with the difficultness of distinguishing the target stimuli from the standard stimuli, as a higher difficulty has a bigger latency than lower difficulty [45]. Expectancy, suspense and surprise has as well been associated with the P3 [47], [48]. In Belitski et al. [47] a combination of the expected audio-visual stimulus showed a higher P3 amplitude compared to visual and audio stimulus alone. 1 microvolt versus 1.5 microvolt . In the paper by Bruni et al. [48]. it was demonstrated that suspense governed by clues in an interactive narrative led to an expected surprise when the right stimulus was found thereby producing a P3. Moreover, it is further discussed that the P3 is essentially provoked by a combination of different cognitive

[5] For a go/no go paradigm the subject has to press a button when a target stimulus is present (go) and avoid pressing it if the stimulus is non-target (no go)

processes such as the complexity of both the task, the stimulus and the importance of the stimulus [45]. These are essential elements, which deals with how recognizable the stimulus is, how complex and detailed it is and how much attention and cognitive energy the subject funnels towards the stimulus.

2.3.1.6 N400

This potential occurs between 350 – 550 ms after the onset of the stimulus. This negative response is found associated with incongruence in relation to linguistics, syntaxes, photo sequences, and sound sequences [28], [46]. This essentially means, when a string of stimuli is continued by an odd or strange stimulus, which does not fit into the context the N400 is seen. The voltage amplitude has been reported to be larger for word and language than photos and sounds (-10 microvolt vs -5 microvolt) [46]. Furthermore, it has been reported to be associated with familiarity of the stimulus in an old – new memory test where new stimulus elicited a N400 [28].

2.3.1.7 P600

The P600 is a positive potential occurring 400- 600 ms after the onset of a stimulus. It has been associated to memory and syntax comprehension. In memory it has occurred when an old known stimulus has happened [28]. In relation to syntax comprehension abnormalities like "the cats won't EATING" showed a p600 [49]. Furthermore, the p600 has been associated with discourse update of an event based on new information [50]. An example on this could be "The boy was in the garden yesterday, I just found his ball" By mentioning the ball, the boy was suddenly playing in the garden, thus the discourse of the event has been changed.

2.3.2 Frequency bands

Apart from analyzing the signal based on voltage responses after the onset of a stimulus the power and presence of different oscillating frequencies does have an implication towards the exposed stimuli as well. These can both be evoked or induced. Evoked frequency oscillations are synchronized (phased locked) to the target stimulus and clear in averaged epochs, while induced frequency oscillations are not synchronized, but around the same temporal mark and related to the elicited stimulus. It is therefore cancelled when averaged [51].

2.3.2.1 Alpha

The alpha frequency band is between 8-12Hz [51] It is clearly observed when a subject closes its eyes[52]. It has furthermore been seen in relation to relaxation[51], [53], [54]. However, the alpha activity has been seen as an evoked oscillation 250 – 300 ms after visual or auditory stimulus [51], [53]. A relationship between working- memory and short term memory and alpha oscillation has also been demonstrated by observing synchronized alpha waves prior to a temporal point where an assumed stimulus should have been elicited [53]. Lastly the alpha oscillations have been associated with attention and visual awareness [51].

2.3.2.2 Beta

The Beta frequency band is between 12-30 Hz. Beta oscillations have been associated with mental motor imaging studies, where the subject has to image the movement of limbs. However, in relation to real movement of the limb a beta reaction is firstly monitored one second after the action [51]. These waves have been observed to be related to endogenous top-down processing, where the subject has to maintain a cognitive or perceptual state, whereas on the other hand a decrease in beta band has been observed when a novel or unforeseen exogenous stimuli interrupts this state [55]

2.3.2.3 Gamma

These waves are either denoted to be 30-70Hz[54], [56], 30 – 80 Hz[51], 30-100Hz [52]. These different ranges just imply that the broadness of the gamma waves is not very well defined. It might indeed postulate that the instruments for capturing EEG have been more sophisticated thus being able to capture higher frequencies without noise. Moreover, gamma waves have been observed to both have a relation to stimuli and to cognitive functions. In relation to stimuli, phased locked[6] gamma responses have been observed 100 ms and 300 ms after the onset of an auditory or visual stimuli [54], [56]. Furthermore, the cognitive correlations to gamma have been observed to be attention [51], [54], [56], perceptual switching between ambiguous and reversible visual input[7] [54], [56], perceiving meaningful objects [51] and lastly towards emotional valence [57], [58]

2.3.2.4 Delta

The delta wave has a frequency range from 0-4 Hz[51] and it has been associated with decision making, signal detection and reaction to deviant stimuli as a part of negative mismatch event related component [54], [56].

2.3.2.5 Theta

These oscillations are between 4-8 Hz. They have together with Delta been associated with the major components of P300, N400 and P600, working memory functions, and together with Gamma they have been associated with memory performance [51]. Furthermore, prolongation of the theta waves has been linked to selective attention. A last remark, a coordinated theta response called "orienting" has been used to indicate alertness, arousal or readiness, thus related to oscillations during, searching, exploration and motor behavior [54], [56].

2.3.3 Relating ERPs and frequency bands to events in the Super Mario Game

The different stimuli found in Super Mario and the cause of different brain potentials and frequency bands have been examined. Based on the known correlations presented in the previous pages, this section will make connection between game events, and the brain's reactions, both in terms of event related components and the dominant frequency band. Moreover, it shall be mentioned that at this stage these are mere hypotheses, and might be confirmed or dismissed in a later study (see section 9.1).

At the beginning of the game different events can

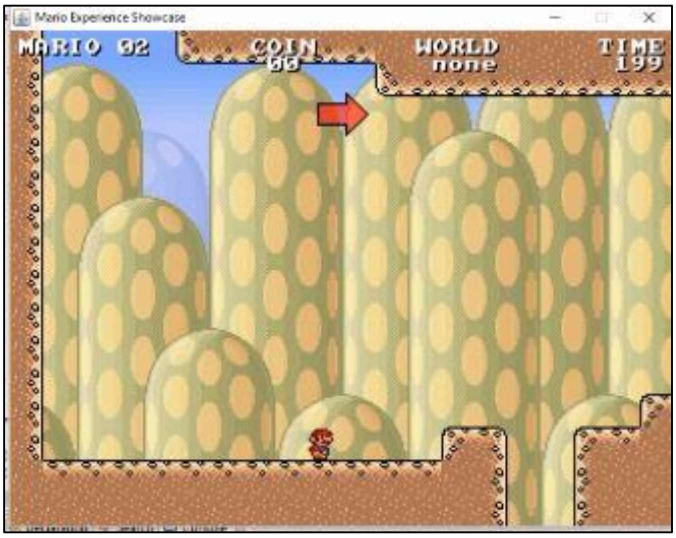

Figure 6 shows the start of a Mario level, where Mario is small and the user can start progressing.

---

[6] Phased locked oscillations are found when epochs are averaged together, and the oscillation pattern still prevails.

[7] Such as the Necker cube or vertical flashing light. In both of them the perceived orientation or direction can be altered by the observer, even though the object in itself does not change.

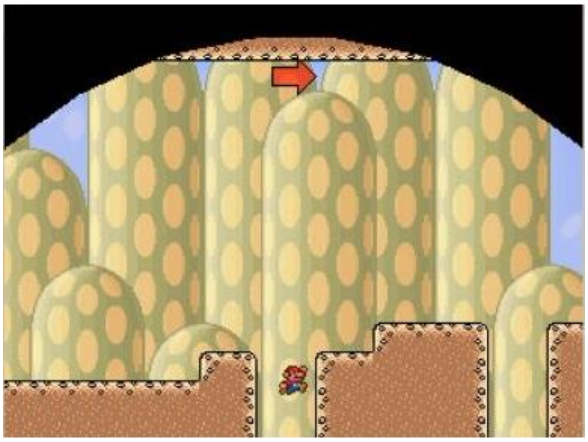

Figure 7 shows the moment, where the user has just failed to jump over the gap. At this point the animation for Mario has died is about to end. The automated jump animation is ending and the circular transition is in the middle of its process

happen. In Figure 6 Mario will soon jump over the gap. The jump in Itself is not very interesting. It is rather its context that is relevant. In this case the jump may be a fail or success jump. The subject has probably already made up his mind towards succeeding the jump [22]. Therefore, succeeding might produce one unique brain potential compared to failing the jump. Succeeding might lead to the expected outcome, thus a p300 might be observed. While failing the jump on the other hand might produce P2 due to its emotional low valence compared to succeeding a jump. A third event that can occur at this moment, is Mario jumping up of the gap again. This is an unusual action, thus an N2 or an N400 might be present. A P1 might also be present when such event happens, as it is a sudden call for attention. Furthermore, it will enable interaction with the system again, thereby changing the subject's internal game schema from "time to start over", to "Mario is still alive let us progress".

Assuming the player has lost due to the gap, automated animations are elicited. Those might cause different ERPs. Firstly, Mario autonomously jumps out of the gap just to fall down again; secondly the screen fades through a circular transition (see Figure 7) followed by returning to the game through the same transition. The jump is a fast discrete event, to which the subject has no control of. If attention is paid towards it a P1 and N1 might be present for the pure attention to the stimulus. Furthermore, whether or not the subject expect this to happen the N2 or P3 might happen. In relation to the frequency bands Gamma could be present as a result of the attention and the relation it has to valence.

The circular transition stimulus might activate the N2 or P3, whether it is expected or not. Furthermore, since it blackens out the screen alpha waves might be present as a result of closing the visual stimulus. The return of the stimulus might therefore lower the alpha waves while at the same time enhancing the theta waves as a result of being alert and attentive to the game again.

Figure 9 shows Mario who is about to kill an enemy with a fireball

The next interesting events that can happen in the game can be seen in Figure 8**.** It illustrates Mario who has just killed an enemy by stomping on it. This could shortly elicit some brain potentials or certain signal activations, such as gamma waves as a result of the positive impact of killing an enemy. Furthermore, the reaction of the enemy is expected, thus a P3 might occur as well. Apart from stomping there are two other ways of killing an enemy; by a fireball (see Figure 9) and by a provoking a shell to hit it . These interactions might produce different theta patterns, as they all require different amount of pre action calculations. This could as well lead to difference in the elicited arousal.

Furthermore, in Figure 8 other future events are depicted. Such as dying to an enemy, devolve by colliding with an enemy (degenerating to a lower form level) and picking up a coin. Dying to an enemy is an unfortunate event. When that happens all dynamics stop briefly, before it is continued by the automatic jumping discussed before when Mario failed to jump over the gap. A result based on the low valence of dying a P2 might be observed. Through, the inhibition of all dynamics, interaction and user agency a N2 could be present as well.

However, instead of dying Mario could be fortunate to be in an evolved form when he collides with the enemy. In that case Mario will flicker shortly, while agency is taken away from the player. When the agency is back Mario is in a devolved form. This event is possible not a part of the subject's game schema, thus not expected. Therefore, it might create a N2 as something that was not cued and a P2 for the low valence it could imply.

In Figure 8 Picking up a coin was also a possible future action this not show any significant brain potentials, as they have minor impact on the game schemas. But, if the disappearance of the coin is attended and it is the only thing the subject is focused on a P3 might appear due to its expectedness. However, it is rare in the game, that picking up of coin is given a significant amount of attention.

The next couple of events there will be discussed can be found in Figure 10**.** This figure shows what happens when Mario collides with a box. This action could cause a mushroom, a flower, a brief view of a coin, or an enemy to pop out of the box. Furthermore, if the box does not portray a question mark it is more likely to explode (see Figure 10 for all the reactions)**.** All of these have some common traits and some distinctive properties. They all appear as a result of the interaction and thus calling for immediate attention for the subject to rate its significance, thereby recognizing it. That may provoke the N1 P2 complex, as the stimulus needs to be processed and categorized. The similarities however stop here, as the valence and the likeliness of the different events could provoke the elicited signal in different directions. The expectancy of the flower, mushroom, coin and the destroyed box is way more established than the enemy, which may elicit a P3 for the former and a N2 for the latter. Furthermore, the N2 could as well be elicited based on the go / no go paradigm as the appearance of the enemy changes the subjects mind from approaching the appearing stimulus to distancing himself from it. Moreover, the P2 for the latter could be of a higher amplitude as the enemy has a negative impact on the subject compared to the rest.

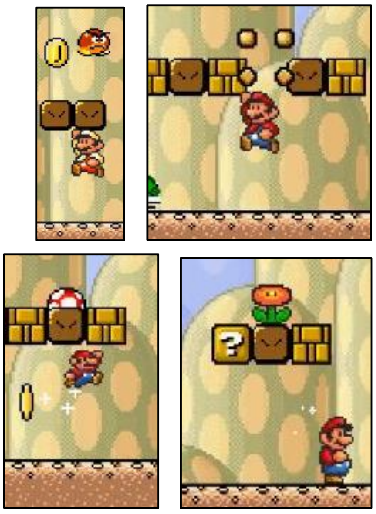

Figure 10 shows five different objects a box can contain, which is elicited when Mario collides with it. The upper left shows a coin popping out of one and an enemy popping out of another. The frame in the upper right corner shows how a box looks like when it breaks. The lower left depicts a raising mushroom and the lower right shows a flower had appeared.

The flower, coin, breaking box and mushroom event might have differences as well. They are assumed to be subtler, but by looking at the semantics of the game and the effect these components may have on Mario and thus the subject, different brain potentials might occur. The P2 and the Delta waves might have different signatures across the four different objects, as the decision process towards which kind of action should be taken afterwards is different as well as the recognition of the objects. The decisions that could be implied post stimulus is after the mushroom or the flower have popped up. This action might change one of the player's sub goals to "picking up object" before continuing towards the finish line. On the other hand, the decisions needed when the box breaks or a coin pops up do not change afterwards as the stimuli disappears right after their occurrence. Gamma and Theta could as well have different signatures as the valence and arousal of the appearance of the different items could be very different. The arousal and valence of the mushroom and the flower could potentially be a lot higher than that of the coin and the breaking box, because the two former items helps and supports the player to win the game.

When the player picks up either the mushroom or the flower Mario will evolve if he is not already in the evolved form. The reaction is the same as when he is devolving. Mario flickers shortly, while inhibiting the user's agency continued by the return of the agency. The subject expects it to happen and the event is possibly attended, thus a P3 might occur for both of them when evolving. A P1 might also occur as a result of the attention. The difference between them might be in the Gamma waves as the valence of picking up a flower might be higher than picking up a mushroom as it enables the player to kill enemies from a distance thus completing the level more easily. Furthermore, these are not the only ways Mario can evolve. Occasionally when Mario collides with an enemy he will evolve instead of devolve. This is an unusual change in the semantics for subjects not used to this deviant behavior, thus a N2, N400 and P600 might occur, but only as long as the subject does not accept this event, which can very easily be perceived as a glitch, or a genuine part of the game.

The last events Mario contains is when the level is lost or won. When winning the player collides with the finish line found in Figure 2**.** When the game is lost it is due to the subject losing all three of "Mario's lives". Both of these events might be expected thus eliciting a P3. Moreover, the gamma would be different in them as losing is more unpleasant than wining and thus a P2 might be available when losing a level.

All the discussed events and their hypothetical brain potential signatures are summarized in Table 1 below

Events:	ERPs and Frequency bands
Succeeding a Jump	P300
Failing a jump in a gap	P2
Bounce out of gap	P1, N2, N400
Automated jump animation when dying	P1, N1 (N2 or P3), Gamma
Automated fade to black	(N2 or P3) Alpha
Automated fade back to game	Theta, Low Alpha
Dying by an enemy	N2, P2
Devolve by an enemy	N2, P2
Evolve by an enemy	N2, N400, P600
Evolve by a mushroom	P1, P3
Evolve by a flower	P1, P3
Killing an enemy by a stomp	P3, Gamma, Theta
Killing an enemy by a fireball	P3, Gamma, Theta
Killing an enemy by a shell	P3, Gamma, Theta
Picking up a coin	P3 but rarely
Breaking an empty box, as a result of a collision	N1, P2, P3, Delta, Gamma, Theta
Hitting a box containing a coin	N1, P2, P3, Delta, Gamma, Theta
Hitting a box containing a mushroom	N1, P2, P3, Delta, Gamma, Theta
Hitting a box containing a flower	N1, P2, P3, Delta, Gamma, Theta
Hitting a box containing an enemy	N1, P2, N2, Delta, Gamma, Theta
Winning a Level	P3, Gamma
Losing a Level	P2, Gamma

Table 1 the Hypothetical correlation between game events and EEG signal signatures

2.4 Eye-tracking features

In the introduction EEG and Eye-tracking were introduced as the measureable psychophysiological signals. EEG has been explored, in this section Eye-tracking will be examined. Eye-tracking data is comprised of two streams of information the gaze information and pupillometry. In this section these two streams will be investigated and correlated to the game events found in Super Mario.

2.4.1 Gaze information

Gaze information comes from the direction and position of the subject's vision. In relation to a computer screen it is translated to an x and an y position. When the subject briefly focus on an item it is called a fixation. This focus is approximately adjusted three – five times a second, thus making them last 200-300ms [59]. Those adjustments are mostly not noticeable, and they are primarily influenced by the onset of a stimulus, task and sub goal relevant stimuli. It has been observed if an embodied reaction is elicited upon such stimulus the gaze has fixated at it 0.5-1 seconds prior to the embodied interaction, if the context is fast paces, like tennis, the eye-interaction delay can be as small as 100 ms. This task relevant focus has both been proven in real life and within virtual reality. Another context this phenomenon can happen is, if a perceived threatening object is approaching you then there is a tendency that you will fixate on it whether it is animated or not. Thereby it can be assumed that enemies in Super Mario might indeed get some fixations as they walk towards Mario. They can both be perceived as threatening and the subject should act upon it. Furthermore, It should also be noted, since the subjects will focus on a computer screen through the experiment ,their gaze fixation is likely to be around the center of the screen to get as wide view as possible [60, pp. 3–32].
Given the gaze is governed by task and sub goal relevance, it would be expected that the game events in Super Mario which will require a conscious reaction would be fixated at before interaction. Such interaction could be hitting a box or an enemy, both of which requires the user to immediate look at the

target and determine when an embodied interaction is required. Moreover, in a game where stimuli come continuously and the game schema is constantly updated in the subject's fixation pattern might as well reflect that. Thus, detecting newly arrived stimulus, interpret it and put it in the game schema as a task on the working memory list. Furthermore, when a stimulus is needed to be acted upon a new wave of fixations might occur in favor of the stimulus which calls for interaction. If these assumptions are true, then the attention from the subject will be directed towards the task relevance stimulus, thus attention related ERPs could have a chance of emerging.

2.4.2 Pupillometry

Pupillometry is the contraction and dilation of the pupil, it have been connected to cognitive load [61]–[64], emotional arousal. [61], [63], [65] and the amount of brightness (the pupillary light reflex) [61]–[63]. The latter is greatly important in relation to this project because the screen emits light and thus making the pupil more contracted to compensate for the brightness. If this measurement should be taken into the final results, the application's brightness needs to be constant to not provoke any unnecessary contraction. This factor is in fact very crucial as the pupil diameter can change from 3 – 9 mm between light and dark environments [61]. Such high fluctuations can very easily overrule and mask cognitive or affective pupil dilation, which rarely exceeds 0.5 mm [61].

in respect to the cognitive load experiments, pupil dilation has been observed increasing when the subjects need to remember different amount of numbers from 3-7 digits (average pupil diameter ranged from 3.7 – 4.2 mm). The highest dilation was recorded when the subject needed to remember 9 digits. When the subject had to tell them loud the pupil progressively contracted. The same pattern has been observed in relation to multiplication tasks. More difficult tasks produced more increased dilation than easier tasks, The difference in pupil dilation was pronounced 1000 ms post multiplication task (the relative change in pupil diameter ranged from 0.3mm for easy tasks to 0.5mm for difficult tasks) [62]. An increased dilation has also been seen in relation to incongruent color and word pairs compared to congruent (relative average pupil diameter change of 0.16 mm compared to 0.08mm). The onset of the dilation difference occurred between 1000ms – 2000ms after the onset of the stimulus. Interestingly, attentional shift has also proven to influence pupillometry, the dilation changed from the baseline starting from 244ms pre attention shift to 1552ms post attention shift with a peak dilation at 602ms [61].

In respect to Super Mario pupil dilation might not give many unique patterns. Moreover, it might be more increased during scenarios where many sub goals have to be taken into consideration, thus a higher cognitive load compared to when there are not many elements in the scene. The attention shift provoked dilation could be available as well. However, it could be hypothesized that it would only be available when it is shifted between different stimulus categories in the game for instance from a collectable to an enemy. This makes it less likely to occur as a result of an interaction during the game, as the attention is already set on the interact able object. Moreover, as visited in the previous subsection the gaze is likely to move at task relevant objects 0.5-1 second prior to the interaction, thus an attentional shift pupil dilation effect might occur at the time of interaction.

2.5 Categorizing events in Super Mario

The components Super Mario is composed of and the interactions which may elicit different EEG and Gaze tracking patterns have been visited. An overview of the different type of events and how they relate to each other has thereby been established. This examination has emitted hypothesized theories of which events that may elicit significant psychophysiological patterns compared to others.
These types of events will also aid the decision of which event's data that will be used to train the Artificial Neural Network, which will be visited in the next section. Below is a table which have categorized the different types of events by their common semantics.

Categories:	**Types of event:**
Mario changing form by collision	Devolve by an enemy
	Evolve by an enemy
	Evolve by a mushroom
	Evolve by a flower
Box collision events	Breaking an empty box
	hitting a box containing a coin
	hitting a box containing a mushroom
	hitting a box containing a flower
	hitting a box containing an enemy
Killing enemy events	Killing an enemy by a stomp
	Killing an enemy by a fireball
	Killing an enemy by a shell
Ending level events:	Dying by an enemy
	Winning a Level
	Losing a Level
	Failing a jump in a gap
Automated ending level events:	Automated jump animation when dying
	Automated transition to black
	Automated transition back to game
Avoiding dying in a gap events	Succeeding a Jump
	Bounce out of gap
Object collision without effects	Picking up a coin

Table 2 shows the examined types of events and their categories, which are based on common semantics

In Table 2 seven different categories have been devised based on the semantically uniformity of the events. In the "Mario changing form by collision" category does all the types of events belong which makes Mario change form also denoted as evolve or devolve. This can be understood as upgrade or downgrade of Mario as well.

The "Box collision events" category contains all the types of events which are causally related to Mario hitting a box by jumping into it.

"Killing enemy events" This category are all the methods of which the subject can cause harm to an enemy.

In the category "Ending level events" are all the events which either resets a level or makes the program progress to next level, but they all stops the game for a few hundred milliseconds before continuing.

"Automated ending level events" are events which are automated when the subject fails a level by either falling into a gap or dying to an enemy. The subject does not have any influence on these events, thus they can just be observed like a short movie.

In the "Avoiding dying in a gap events" are the two ways which can safe Mario from falling into a gap, either by jumping over it or by luck through activation of "bouncing out of gap" event.

The last category only has picking up a coin in it. The category is not called by the name of the event is because, the same effect can happen when Mario collides with a flower or a mushroom as well. However, only if is he already in that form, which the object would evolve him to.

2.6 Artificial Neural Network

The multiple different psychophysiological reactions related to different game events have been discussed. The fact that there are both noteworthy differences and hypothesized similarities in the signals produced by these events represents quite an interesting technical recognition and detection problem. Before the problem is formulated it is important to know how deep machine learning algorithms work and how useable they are in regards to this very problem. In particular, in this section the theory and practices behind artificial neural network (ANN) will be discussed. The author is well aware that there are other deep machine learnings algorithms, which could solve the problem as well. However, to maximize the understandability of the mechanisms, the know-how and report coherency ANN will be the only one explored (A performance discussion between difference algorithms can be found in section 8.2).

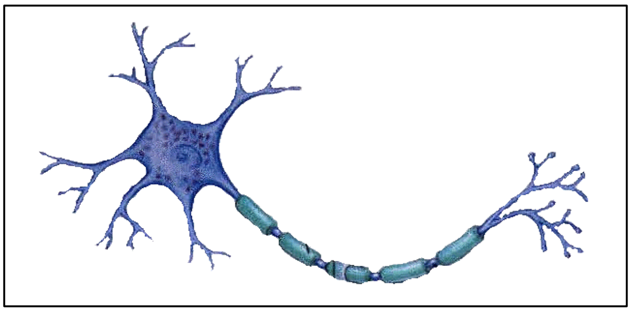

Figure 11 shows a Neuron. Its axon to the right, the big pointy cell in the middle and the synapses to the left.

At its core the ANN is a bunch of mathematical equations used to find the relationship between the input value and the output value in a particular problem. There are essentially only a few unique equations in an ANN. However, these can be multiplied and stacked together to form unique models, which can be trained to detect the connection between abstract input values and concrete output values or categories [66]–[68]. The structure of these equations can be thought of as the neuron network in the brain. Where every single neuron is connected to multiple others. The neuron receives signals through its synapses, computes the signals in its cell, then it fires a resulting signal through its axon [67] (see Figure 11 for an illustration).

2.6.1 Studies utilizing ANN

This analogy has led to the idea of creating an ANN. It has successfully been used in for instance computer vision [67], chemical engineering [69], forecasting of future electrical load and temperature [70], ecology [71]the relationship between user experience and game metrics or psychophysiological signals [2], [3], [72]–[78], and EEG and different cognitive or perceptual phenomena [79]–[87]. These are just a few of these areas where ANN has been used, but the list of different fields and areas of application is much longer.

From this immense amount of literature the articles revolving around EEG will be given a closer look in order to get an idea about the possibilities and limitations of using the ANN for EEG pattern detection. Some of the papers look at simple EEG experiments, where the attended stimuli produced P300 [81], [86], [87], They were governed by expectancy or counting. Other looked at the continuous EEG from different frequency bands [83]–[85]. In the ERP experiments where they should count they did it when the right sound frequency was heard [81] or when a preselected image appears [86]. In regards to the expectancy experiment, the subject where introduce to match audio and visuals, where conflation was the expected condition [87]**.** These experiment were conducted on either disable and able body subjects[86] , head injured subjects [87] or both healthy subjects and subjects suffering from Multiple Sclerosis [81]. Thus giving them different classification problems. One thing those problems have in common is however, that it is either target or no target ERP (counted tone or image or not counted tone or image, healthy subject p300 or not healthy p300). The classification problems in themselves are rather straight forward, which is reflected in their results . The auditory p300 from healthy or not healthy subjects gave 75% accuracy for the not healthy subject and 87 % for the healthy ones [81]. The accuracy for detecting the P300 at the matching task gave three different accuracies as the data was manipulated in three different ways: average of all EEG channels to one stream of data, training and averaging six different algorithm based on six different electrodes and eight trials, and the last was given based on the same algorithms as the previous, but only trained on four trials. The one trained on all electrodes gave 85% accuracy, the first set of individual algorithms trained on eight trials resulted in 79% accuracy and the second set trained on four trials gave 69% accuracy. All trials were based on head injured' p300 [87].
The last of the papers working with p300 recognition gained their results based on different types of neural networks. The first being a neural network which employs a back propagation training algorithm [88], the second an adaptive neural network without autoregressive feature selection method and the third being an adaptive neural network with autoregressive feature selection. The backpropagation driven network gave 93% average accuracy. The adaptive neural network gave 96.5% accuracy and the adaptive neural network with autoregressive feature selection gave 97.3% accuracy in respect to enabled and disabled bodied subjects [86]. Apart from these papers a review paper has debated the accuracies of ANN trained on EEG data [82]. Through that review it concluded that there is a promising future for using ANN to recognize and classify EEG. In regards to evoke potentials in healthy subjects whether it was auditory, visual or somatosensory the results were not consistent, but encouraging. Equally relevant for the future of EEG classification and games. With the help of neural network the review paper looked at brain computer interaction (BCI) as well. In the papers achieving the highest accuracies the results were between 83-90% [82].
Related to the BCI studies, which looked for imagined motor control, is the last paper which will be covered in this review before continuing describing the ANN.
This paper [85] worked with working memory load recognition both within a task and cross different working memory task. They chopped their signals into the five different frequency bands Alpha, Beta, Gamma, Delta and Theta, those signals where cut into five second pieces. In the study there were three memory tasks with two difficulty levels each, one low and one high. Classifying these difficulty levels within each task gave accuracies between 86-89%. Moreover, when the ANN models were tested on data based on the other tasks it gave an average accuracy at 44.3% [85]. The same accuracy tendencies have been found within workload classification of real life challenges such as air traffic monitoring, in which Eye-tracking, heartrate and respiration have been used as well [83], [84]

2.6.2 The ANN – components, parameters and equations

The brief look into the literature has confirmed ANNs successful usage within EEG recognition and classification. Through this subsection will an elaborate description of the ANN be devised.

2.6.2.1 Components – Layers, Neurons and Connections

In the network there are several components with each of their distinctive properties. (a relationship between them can be found in Figure 12)

Neuron – a simple calculation unit which summarize its inputs. This sum is let through an *activation function*, which has been a sigmoid or a hyperbolic tangent (tanh) in the literature [67], [79], [86], [89]. Through this function an output between 0-1 is sent, if it was a sigmoid, or if was a tanh an output between -1-1 is sent. The output is sent through the connections towards the next layer of neurons.

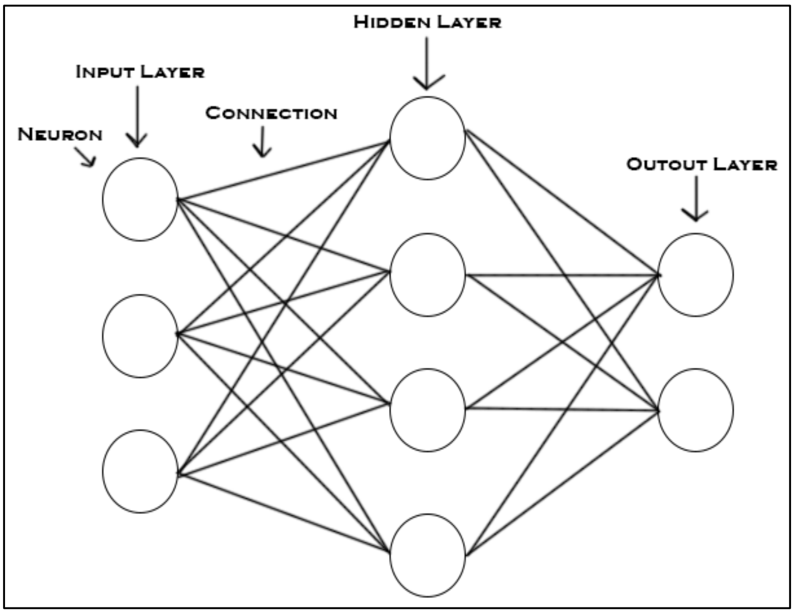

Figure 12 shows an example of an ANN

Connection – the role of the connection is to scale the number the neuron has just fired towards the next neuron. It is scaled through a parameter called *weight.* The weight can be interpret as the memory of the network, as it is the element which is changed through training, thus "learned" what it should be in order to create the ideal or close to ideal output.

Layer – this is a collection of neurons, there are three types of layers with their distinctive functions: *input layer, hidden layer* and *output layer*. The input layer takes the data the network has been fed and sends it through the network to the hidden layer. The hidden layer receives the data through the connections from the input layer. Runs them through the mentioned calculations. Afterwards sending them to either another hidden layer or the output layer. The hidden layer has been given its name because its data is not accessible by the user. At the output layer. The data hits the last activation function after that the data is compared to the ideal output data through *mean square error* [67], [90], thus finding the error rate.

2.6.2.2 Test Run – feedforward

The structure and their individual functions have been covered. The process of the input data going through the network to reach the output neuron is called feedforward, to make it clear an example of the math is provided below:

In Figure 13 an input, some random weights and two ideal output has been assigned.

First the input is multiplied by each weight. The four hidden neurons will then receive: 0.06. 0.09, 0.12, 0.18 respectively. In the neurons the value will be applied to the sigmoid activation function, which look like Figure 15. Through that the hidden neurons will output the following values to their connections: 0.5150, 0.5225, 0.5300, 0.5449. As before the output are multiplied by the weights and send to the next neuron. At the output neuron the sum of the connections is calculated and send through the activation function. For the upper neuron the result is 0,6550 and for the lower output neuron the result is 0,7628 (code example can be found in appendix D).

The output value has been found, but they are not corresponding to the wished output values, which have already been plot into the system. To give the system a value it can use to determine how far the output value is from the wished output value The mean square error function is used. A modified version can be found in Figure 14**,** however the usual function also incorporates a sum, as it is a sum of errors, but in the case of the trial neural network developed by the author the errors were treated as individual instances. in this example the values are 0.2145 and 0.0281.

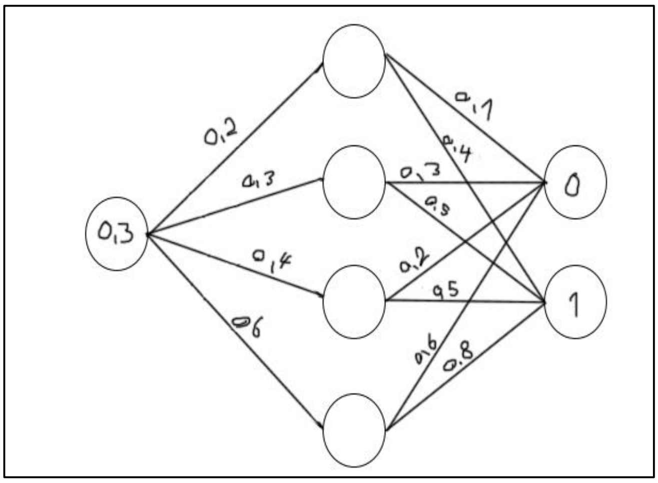

Figure 13 shows an example of a simple ANN with assigned values

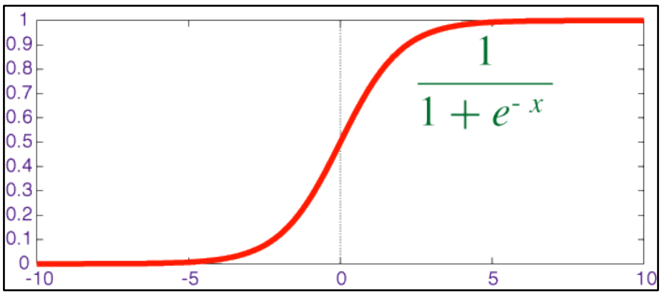

Figure 15 shows the sigmoid activation function

$$X = \frac{1}{2}(Y - \hat{Y})^2$$

Figure 14 square error function. Y = ideal output. Y hat = actual output

2.6.2.3 Backpropagation, resilient backpropagation and gradient descent

From the example there are clearly differences between the output from the feedforward mechanism and the wished output. A way to get around this is by using gradient descent and backpropagation. Gradient descent is an algorithm which is used to search for the local minimum, in this case it will search to find the lowest error rate possible. This strategy is utilized through Backpropagation, which is an algorithm which broadcast the error rate back through the system in respect to the weights, thus changing them.

In order to know how they should be changed the partial derivative of the functions utilized in the system is used and a learning rate is multiplied to the equation. The partial derivatives are highly important as those finds the slope of the function in respect to the different weights, thus guiding the gradient decent in a certain direction.

The weights which firstly will be updated are those connecting the hidden layer and the output layer (W2 see Figure 16). They are updated through the equation found in Figure 16**.** Through it each weight will be

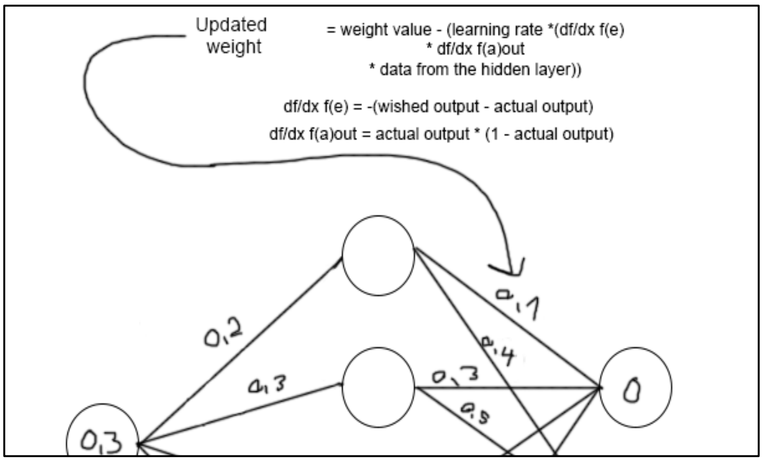

Figure 16 shows the first step of the backpropagation, where it changes the weights (W2) in respect to the broadcasted error rate. df/dx f(e)= partial derivative of the error function df/dx f(a)out partial derivative of the activation function in the output layer

adjusted relative to the influence of the weight and the learning rate. The learning rate is usually a low constant number like 0.001. With that in mind the weight pointed at in Figure 16 will be changed to: 0.1 –(0.001*((-(0-0.6555))*(0.6555*(1-0.6555))*0.515)) = 0.0999.
This value descent makes sense, as the wished value is o, and the last value the output neuron received was 0.515.
we assume that all the weights between the hidden layer and the output layer are updated the next layer of weights (W1 Figure 17) will be updated in the same manner. However, their equation needs to take more parameters into consideration, as the backpropagation mechanism through W2 influence those weights. This is expressed through delta3 found in Figure 17**.** This parameter takes the sum of the partial derivatives used in Figure 16 multiplied by the W2 weights relevant for the individual weights in W1 (fx the relevant weights for W1 = 0.2 are W2 = 0.1 and W2 = 0.4). All these expressions are usually expressed as matrixes; thus matrix multiplication is done. However, for this example each of the elements are inspected individually to make it clearer what is happening. Delta3 is further multiplied by delta2, which is the partial derivative of the activation function in the hidden layer. Lastly it is multiplied by the input from the input layer. These parameters are as well as the equation in Figure 16 multiplied by the learning rate. All together subtracted from the weight value in question. This is a rather simple network if there were more hidden layers, a delta3 component would be added for each one of them to take into consideration the summed weight and partial derivative function between the two hidden layers.
The equations in order to update the weight which is pointed at in the network (see Figure 17) is explained and the numeric equivalent will look as follows: updated weight = 0.2-(0.001*((((-(0-0.6555))*(0.6555*(1-0.6555))*0.1)+ (((-(1-0.7628))*(0.7628*(1-0.7268))*0.4))*(0.515*(1-0.515))*0.3))) = 0.2 -3.7450e-07. The update with that learning rate is very subtle and nearly invisible, but after many iterations it will have an effect. The backpropagation has been done, and the network will run through the feedforward once again to hopefully find lower error rates than before. Those will be back propagated once more. Thereby will the iteration continue to follow one of the following exemplified strategies: reaching an error rate threshold, looking for when

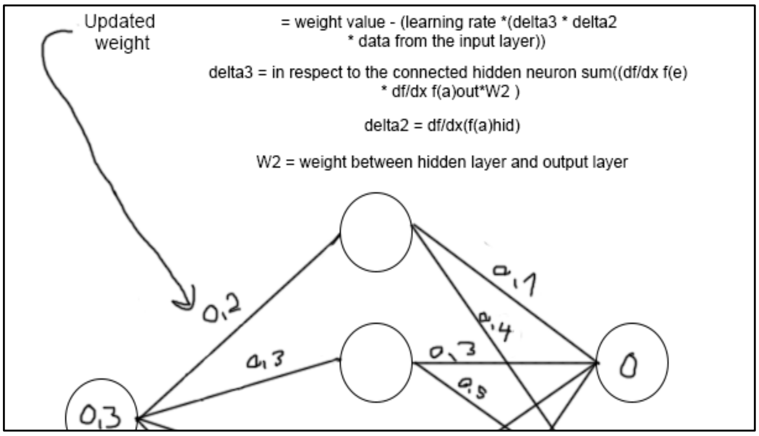

Figure 17 second step of the backpropagation, where the weights (W1) between the input layer and the hidden layer is changed in respect to how the broadcasted error rate have changed W2. df/dx(f(a))hid = partial derivative of the activation function in the hidden layer

the error rate starts to increase or listen to when a certain amount of iterations have been reached. (an example of the simple implementation can be found in Appendix D and through the link in Appendix I)
The backpropagation algorithm dates quite far back [88], and thus optimizations have been done afterwards one of which is the resilient backpropagation, which have been quite popular due to its efficient training [91]. It works locally at the individual weights. Every time the local minimum is missed, derived from sign change of the partial derivative of the error function in relation to the weight in question, a new term called update value is decreased by a constant factor. This value is determined by the difference of the weight update during learning iterations. If the local minimum is not missed the update value is slightly increased by the factor. Afterwards the weight is either added or subtracted by the update value dependent on the sign of the partial derivative. If the partial derivative is positive the update value is subtracted, if it is negative the update value is added to the weight. However, if by the previous steps the partial derivative changes sign, thus missing the local minimum, the previous weight update is reverted [91]

2.6.2.4 Important adjustable parameters

The ANN algorithm has been visited. Moreover, it have some parameters which can be adjusted and tailored to the input and output data. The amount of layers, neurons in each layer, the learning rate and the ending strategy.

Amount of layers: the amount layers can be all from two to infinity. Infinity is however a quiet bad idea and is strongly discouraged. In relation to the specific problem at hand the more layers can usually find and distinguish finer and more subtle patterns, which will give a lower error rate. Moreover, it comes at a cost, as the amount of computations and weights are increased thus more time is needed on training the model [92].

Neurons per layer: the input layer and output layer are depending on the amount of input data and output categories. Moreover, the amount of neurons in the hidden layers can be adjusted. Since there are a trillion amount of combinations it can be a tedious trial and error approach for finding the right amount of neurons. An adaptable approach has however been made, but a model running it can run forever [89]. Thus, guessing the amount of neurons based on the complexity of the data might be a good shot. Furthermore, the more neurons the more training time does it take to adjust all the weights [92].

Learning rate: this variable adjusts how aggressive the weight should learn. Moreover, if the rate is too high the local minimum might be skipped, and if it is too low the weights would not change remarkably, thus the error rate would stand still [92].

Ending strategies: in order to tell when the network should stop training itself there are different strategies that can be chosen from, with different advantages and disadvantages. Firstly, **stop when a certain error rate is reached.** This strategy is pretty straight forward as the error rate will denote when the training should stop. This seems like a fine strategy as it can be set very low and thus achieving a high prediction rate when it is reached, but reaching that level is never guarantied. The error rate could very well have converged way before the wished error rate, thereby never reached. The model could rick training itself for an eternity without getting any better.
secondly, **stop when a certain amount of iterations is reached.** This strategy is like brute forcing the algorithm to stop. It can be smart if there is only a certain amount of time to work with or if the strategy above has confirmed a convergence at that certain amount of iterations. Moreover, stopping it too early risks loss of optimization and providing worse results than if it was continued. Thirdly, **Looking for when the error rate starts to increase.** This strategy gives the network to stop as soon as it sees an increase

in the error rate. At first it might sound like a good idea to stop at that point. Naively, it can be thought that the model has reached its potential, but in case the model needs to adjust it weights to a worse state in order to be able to continue towards a better error rate, then that opportunity is missed [90], [92].

2.6.3 ANN Summary

Through this section the usability of ANN and the functions behind it has been described. It gives a good indication of it utilization in relation to EEG and ERPs. Moreover, the section has gone through a simple implementation of the network to identify the different equations behind it, how they interplay and update themselves in order to make it a machine learning algorithm.

2.7 Investigation Summary

Through this investigation the Super Mario game have been decomposed into different visual and auditory features which combined with a player's interaction converges into diverse categories and types of events. These events are good candidates to elicit unique psychophysiological patterns extracted from EEG and Eye-tracking. These can very well be a combination of different EEG ERPs (N1, P1, N2, P2, P3/P300, N400, P600), EEG frequency bands (alpha, beta, gamma, delta, theta), Gaze position and Pupillometry (pupil dilation / contraction). Moreover, since the events occurs in random order, time and space, thus an algorithm which can find non-linear subtle patterns needs to be developed. To this artificial neural network has been investigated and thereby different papers utilizing this algorithm. Their results and future prospects in relation to EEG recognizing research are promising and gives an idea about the success of implementing it in this project.

This leads to the Final Problem Statement, which this thesis intends to create an experiment upon based on the knowledge gathered through the investigation:

3 Final Problem Statement (FPS)

Is it possible to recognize psychophysiological patterns, measured in relation to specific videogame perceptual-events, through an Artificial Neural Network?

3.1 Requirements to answer the FPS

In order to answer the FPS the following list of requirements needs to be taken into consideration.

1. Create a version of Super Mario which
    a. Ensures that many of the type of events are happening at least 20 times, as ERP analysis and the ANN requires repetitions to be analyzed and trained properly.
    b. Create levels which are uniformly constructed without creating identical replications, this will make the ERPs more likely to be uniform thus analyzable, and it will give the ANN a higher chance of learning how the ERPs looks like.
    c. Implement triggers at every single type of events mentioned in Table 2
    d. Implement an UDP networking paradigm to send the triggers to the computer running the EEG recording (more details about this go to [8])
    e. Connects with the Eye-tracker thereby having it connected to the stream of triggers.
2. Setup an EEG helmet configuration to support signal from as broad a spatial coverage as possible
3. Setup Matlab Simulink patch from 9 semester [8] to work with the new EEG helmet configuration.
4. Implement Artificial Neural Network
    a. Utilizing Resilient backpropagation
    b. Can both work with EEG and Eye-tracking data without much reprogramming of the program
    c. Implement a statistical supported paradigm to express recognition rate
5. Construct a paradigm for extracting the data from Matlab to the ANN.
    a. In which feature selection and artifact cancellation should be a part, in order to train the ANN with the best quality data as possible.

4 Method

With the final problem and the final goal in place, this chapter will provide an overview of how the experiment was constructed and planned in order to be able answer the FPS. The requirements devised in the previous section will be taken into consideration to reach as valid results as possible.
This chapter will first visit the design of the experiment, then the procedure, which both includes pre, during and post experiment paradigms.
The post experiment paradigm will also concern data processing and analysis.

4.1 Experimental Design

The experiment was conducted in the Augmented Cognition lab in a "within group" style. It was only possible for the subject to be exposed to one condition, thereby excluding the possibility of doing a "between group" experimental.

In order to avoid as much bias in the signal as possible noises and user distraction were inhibited as much as possible at the location of the experiment, because of the EEG and eye-tracking signals. In case any sort of noise or distraction entered the sphere of the subject it was immediately detected by the equipment and mask out the proper signals.

To be able to recognize and compare signals to the natural state of the brain and eyes. A baseline was settled through exposure to a uniform grey screen with a fixation cross on it.

The Super Mario application was designed to expose the subject to 10 levels of approximate equal length, same diversity and intensity of events in order to ensure the comparability between events experienced in the beginning with those experienced towards the end of the game. This was especially essential for EEG and Eye-tracking as those signals are highly sensitive to content and context changes.

In order for the eye-tracker to capture the most accurate tracking as possible different factors was taken into consideration. The distance between the user and the eye-tracker, the comfortability of the user and the screen dimensions. The Eye-tracking that was used in this experiment was the EyeTribe eye-tracker [93] The distance between the EyeTribe and the user was approximately 60cm. At this distance the freedom of head movement within the detectable area of the tracker was the highest, though it was recommended to keep the head in the same position during the experiment. In order to avoid user fatigue, the comfortability of the user was needed to be assured. Lastly the screen could maximum be 24 inches [93].

The EEG was another story. To ensure possible important signals were not omitted the setup of 16 electrodes placed diversely around the scalp was used. The placement of the electrode followed the international 10-20 system [94] and was mounted at the following positions to ensure broad spatial coverage. F3, FZ, F4, T7, C3, CZ, C4, T8, P7, P3, PZ, P4, P8, O1, OZ, O2 (see Figure 18 to get a visual representation of the placements). In this setup a ground electrode was placed at FPZ and the reference electrode was placed at the right earlobe.
The signal was captured through a EEG capturing system developed by g.Tec. It consisted of a

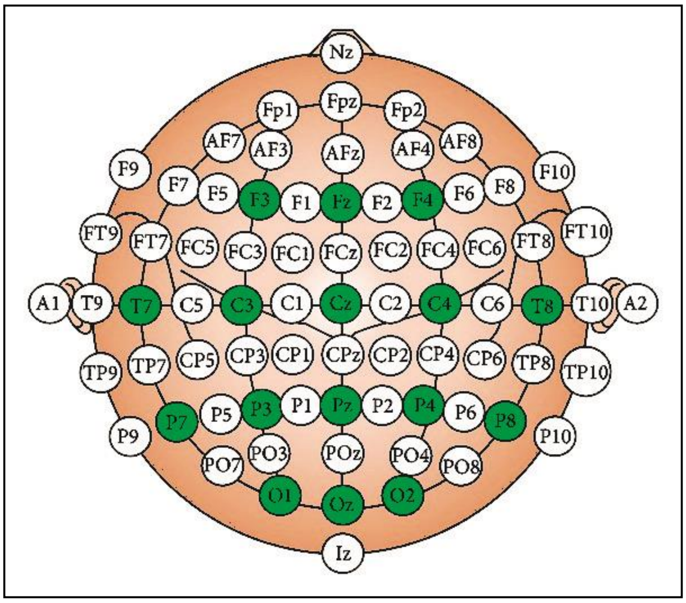

Figure 18 shows the 10-20 electrode system. The green areas are where the electrodes are placed in this

g.Gammabox, in which all 16 electrodes were inserted. This box was connected to the g.USBamp. which amplifies the signal and served as an analog to digital converter. It sampled the data at 256Hz and it was possible to setup different spatial filters for the recordings. For this experiment raw recordings were preferred, thus no spatial filters were applied [95].

4.1.1 Positioning the participant absolute to the screen and the eye-tracker

Mentioned in the last section the dimension of the screen should maximum be 24 inches for optimal eye-tracking signals. However, the screen that was available in the Augmented Cognition Lab was 27 inches. In relation to the recommended screen dimensions the subject should be placed so the distance between the subject to the monitor was 45-75cm [93].

Since the available monitor was bigger dimensionally than recommended it has some clear disadvantages. Firstly, if the monitor is placed at the same distance from the user as recommended the visual angle will be bigger, which can cause the eye-tracker to be less accurate [96]. The visual angle is the size of the visual representation of the observed object on the retina.
Secondly, the EyeTribe performs pretty badly at tracking the eyes when the subject's eyes reach the borders of a screen. This has been tested with a test setup consisting of a 20 inch monitor 60 cm from the user and the EyeTribe 45 cm from the user [97]. In this context it is worth mentioning that the accuracy of the tracker is dependent on the distance between the user and the eye-tracker as well.
In order to compare the visual angle of the theoretical working setup: 75cm distance between the user and a 24-inch screen with the two examined setups found in the papers which were; a 21-inch screen at a distance of 70 cm from the user and a 20-inch screen at a distance of 60 cm from the user. The equation for visual angle found in [98] will be used (for calculations look at appendix G). Through that equation the angles are 0.67928, 0.641357 and 0.70289 respectively. Those seems to be quite close together. However, [97] mentioned they had troubles with accurately tracking the eyes when reaching the borders, thus

indicating that It might be beneficial to have a smaller visual angle than the one they have used. In the lab the setup are as follows, 27-inch screen, 110 cm distance between the user and the monitor and 65 cm between the user and the eye tracker (see Figure 19). This is equal to a visual angle of 0.532504, which is definitely smaller than the others. In

Figure 19 shows the setup and the relationship between the screen, eye-tracker and the chair the subject is placed in

perspective of the recommended setup, then at a distance of 75 cm the screen size will be equal to 18.5 inch to occupy the same visual angle. To make that distance more concrete it is somewhat equal to you sitting in a relaxed back leaning position while still being able to interact with a laptop with that screen size.

4.2 Experimental Procedure

4.2.1 Recruiting participants

The participants was recruited through non-probabilistic convenient sampling [99], [100] by the help of social media and the internal communication systems on campus. The only requirements for the user was that they should not be diagnosed by any mental illnesses, as those alter the brain signals, thus making average predictions across participants harder [101]–[103].
When they were recruited they were asked to not take any psychoactive drugs, like alcohol or caffeine, at the day they have signed to go through the experiment. Such drugs do as well alter the signals gathered trough the EEG [104], [105]. The lowest amount of participants for this experiment was 30, as there can easily be many artifacts and noise in the EEG signal, thus a chance for exclusion of some of the subjects during post experiment analysis.

4.2.2 Execution of the experiment

When the participant arrived to the lab he was greeted and asked to fill out a consent form (an example of the form can be found at appendix A), which would simply tell the participant that their brain, gaze and gameplay would be recorded in a non-harmful way. The gameplay recordings were not taken into consideration during this project as the data were out of its scope, but it will be useful for later research (see future perspective section 9.2). After the participant has signed the consent form a short questionnaire was handed to them consisting of a few demographic elements such as gender, age and occupation. Furthermore, the form also asked if the participant has consumed any alcohol or caffeine and if he had been diagnosed by any mental illnesses. If he had consumed alcohol, caffeine or was diagnosed his dataset was excluded when across participant analysis was accessed, but the experiment was still proceeded. On top of that how often they play video games and if they have played Mario was asked, as it could have an impact on their dataset (demographic questionnaire can be found in Appendix C).
The subject was then asked to sit at the computer which ran the Mario game. The EEG helmet was placed and a conductive gel was applied. While the gel was applied the conductor asked the subject to notify when he/she felt the gel. This check confirmed that the right quantity of gel was applied. To test the validity of the signal a few steps was taken. First a validation mechanism inside the Simulink patch checked how big the standard deviation was, if it was smaller than or around 25 micro volts the signal was good. However, if the standard deviation was higher the contact between the scalp and the electrode was poor, which could be caused by a broken electrode or lack of gel. After this validation the subject was asked to blink with their eyes and clinch their teeth. By doing this the conductor could ensure that he was looking at EEG signals and not EMG or EOG, as those would be clearly visible when the subject performed the aforementioned tasks.

The helmet was successfully mounted and tested. The subject was then asked to sit as comfortable and still as possible while still being able to look and interact with the computer running the Mario game. The eye-tracker would be placed so it could sense the user's eye as described in the experimental design (section 4.1). A native eye-tracking calibration program was executed to ensure the eye-tracker was calibrated to the eyes of the user. The calibration program gave different scores on how well the tracker has been calibrated all from poor to perfect. It was preferred if the calibration procedure was repeated until the score was perfect to ensure the quality of the eye-track data.

With all the data acquisition devices calibrated the experiment was ready to run.
Firstly the Simulink patch, which received and recorded all the psychophysiological signals and the event triggers was started. Secondly, the gameplay recording program was turned on.

Thirdly, the EEG baseline calibration of the subject began which consisted of a grey picture with a fixation cross in the middle. It was shown to the participant for 1 minute. lastly, the Mario game was activated.

While the subject was playing the game the conductor observed the EEG signals and the eye-tracker's eye monitoring program for any ambiguous output. If he observed anything out of the ordinary like, if the environment elicited any auditory stimuli or the eye-track monitor notified, that it could not see the eyes, then those were noted, to be rendered out during the analysis phase.

After the subject has played the game for 10 levels, the subject was un equipped and thanked. The subject was asked to take some refreshments before he/she left the room.
If another participant was waiting the EEG electrodes were briefly cleaned with wet wipes otherwise the electrodes were cleaned with water to get all the excessive gel away, as dried gel can potentially damage the electrode.

4.2.3 Method of analysis

4.2.3.1 Data preparation

After the data was gathered from the experiment it was prepared for the artificial neural network. First the data was cut into different pieces called epochs which responded to the onset of the different game events. These epochs were 3 seconds long, 1 second before the onset and 2 seconds after the onset. This range was to ensure as little crucial information from either the EEG or the Eye-tracking is cut out (this amount of information were discussed here 2.3.1, 2.3.2, 2.4.1 and 2.4.2).

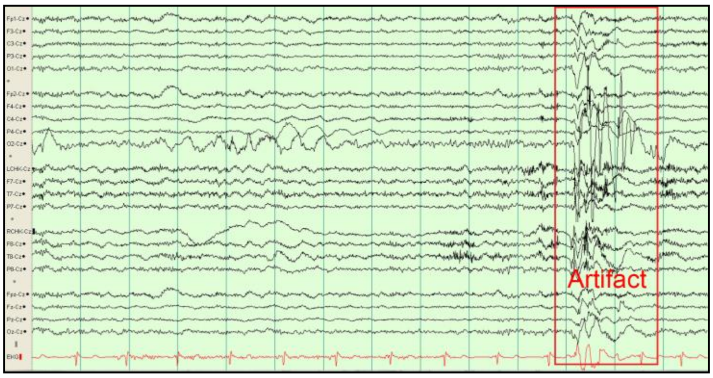

Figure 20 shows EEG with the artifact marked [133]

The EEG data was then filtered by different spatial filters. First a notch filter of 50Hz is applied to cut out the constant noise from the ground in the power supply [106].
Afterwards a band pass filter between 0.1 and 100 Hz was applied. This filter removed all higher frequency noises found in the EEG signal.
There could still be eye blinks and Electromyogram (EMG) artifacts in the EEG data. Through visual inspection those epochs were removed. Eye blinks and EMG artifacts were rather easy to detect as they have a higher amplitude than EEG an example can be found in Figure 20**.**
After the most obvious artifacts was found and removed many epochs were as well, thus any game metrics gathered through the gameplay noting the amount of events that have happened cannot be used anymore.
The next step was to look at the amount of events that were left at each subject. Each event per subject which was recorded 20 or more times was noted down. Those events which were noted at least 10 times was taken into consideration for further analysis. Referring back to the categories in the investigation session (see section 0) it was expected that at least one type of event in each category would be

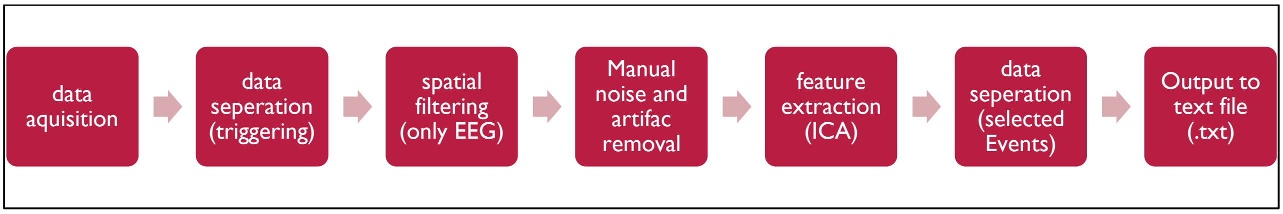

Figure 21 shows the data preparations procedure from data acquisition to the output .txt file

represented at least 20 times. Moreover, if there were not enough events in a category it was excluded from the dataset. In worst case scenario where only one event has met the criteria the ANN would be given that particular event and 20 random samples of others, so the FPS could be partially answered.
The events the ANN should be trained to recognize have been set. With the given datasets the signal quality could be improved further, thus increasing the recognition rate of the ANN. To improve the data, feature extraction models like Principal Component Analysis (PCA)[107] and Independent Component Analysis (ICA)[108]–[110] could be applied based on the chosen events. ICA has proven to be great for separating artifacts and noise from the signal[108] and to find hidden brain activity components[109]. It does it in the following way, firstly it creates a number of matrixes corresponding to the number of EEG channels, in this case 16. Secondly it assumes all the data is statistically independent, has zero-mean, non-Gaussian and is whitened out. These means that; the signals analyzed should not be able to change each other if modified. Should have a center point at 0.0 in a 2 dimensional coordinate system. It should not be normal distributed, and lastly, the data should be spread out in such way it covers -1-1 on both the y and x axis in a 2D coordinate system, which will simulate something like white noise. After these assumptions it starts maximizing the nongaussianity which ends up being a number between -1 and 1. If it reaches 1 or -1 it is super Gaussian or sub Gaussian, thus strongly independent. Moreover, if it is 0 it Gaussian, thus can be dependent[108]. In this thesis the ICA was used to project the data upon the 5 most independent variables normalized across channels. An initial test study worked with the different feature extraction models in order to figure out which of them gave the highest accuracy validates this decision (see appendix E),(accuracy is explained in Table 4). After the signal was feature extracted, the chosen event's epochs were separated from the rest of the dataset. Furthermore, to delimit the bias of variance within the datasets, it was assumed that the different participants had different amount of the chosen events. The amount of events was cut down to the least amount found throughout all the datasets to make them uniform. An example of this could be a subject having the following division of event repetitions (30,55,75,23) In this case all events would be reduced to 23. The datasets were then stored in a text file, which the ANN could read.
In regards to the Eye-tracking, then it followed nearly the same procedure as the EEG signals. However, spatial filters were not applied, as these signals were not influenced by the same kind of artifacts and noises.

In short the following (see Figure 21) was applied in order to prepare the data for event recognition:

4.2.3.2 ANN preparation

Through the procedure above the data was ready to train the ANN to recognize the different selected events. There were several parameters that was taken into consideration before the data was inserted into the network. How much of the data? How was the ANN model constructed? Which end strategy was employed?
The data length depended on the modality. If it was EEG data or Eye-tracking data. If it was EEG data it was

assumed that all evoked potentials and related frequency bands were elicited post the onset of the event (for a review of the components go to section 2.3), but there were not any significantly interesting signals after 1000 ms from the onset. With a sample frequency of 256Hz then it gave the model 256 data points to work with from the EEG. The eye-tracking on the other hand has proven to elicit interesting patterns both before and long time after the onset of an interactive event (for review on eye-tracking see 2.4). This implied the whole epoch was used thus 768 data points which was equal to the 3 second long epoch.

The model of the ANN was structured to elicit the highest recognition rate as possible. For this study an initial test on one subject's EEG signal from the electrode at F3 was used to see which of two models elicited higher recognition rate. The first model consisted of 256 input neurons for the data. Three hidden layers with 32, 16 and 8 Neurons respectively and an output layer of 4 neurons to represent the four different events it should learn to recognize. The activation function in the three hidden layers was a Hyperbolic Tangent and for the output layer a sigmoid was utilized, as subzero values were redundant in this step. The utilized learning algorithm was the resilient backpropagation algorithm. The second model had only one hidden layer with 16 neurons. The rest of the model was the same as the first one. Through the initial test the accuracy showed significant differences between the two models at most of the four events. The results were in favor of the first model (the results can be found in appendix F) This concluded that the model was set to be the complex one, as it seemed more effective in predicting the game events. Moreover, the amount of max iterations trained on was 4800. If more iterations were used the models could potentially begin to find subtler patterns, thus eliciting higher accuracies.
Through the initial test accuracies were sampled 10 times at every 100th epoch, which gave an idea of how many iterations the model should be trained with in order to elicit the best recognition rate possible. Through the test the amount of iterations were applied to be the ending strategy as the others visited in section 2.6.2.4 either implied more time than available

or uncertainty whether the network would meet the ending criteria e.g. low error rate. Through the test the amount of training iterations was set to be 700, as it resulted in highest averaged accuracy.

4.2.3.3 *Data modelling for recognition*

The ANN was devised and ready handle the data. In order to avoid overfitting, the data was modelled. First it was divided into a training dataset and a test dataset. The training dataset consisted of 70% of the data and the test dataset consisted of 30%. Furthermore, through the training K fold Cross validation was applied. It has previously been found as a good candidate to avoid overfitting and have been delivering promising results [111], [112], in this case K =10, thus dividing the training data into 10 partitions. Every time the algorithm gave an error rate and a validation rate[9] it was a results of the sum of 10 training sessions with nine random partitions and tested with the last 10th partition [107, pp. 180 – 181]. When the algorithm reached the goal of the end strategy the 30% test dataset was tested on the ANN model.

16 Channel of EEG per subject (F3, FZ, F4, T7, C3, CZ, C4, T8, P7, P3, PZ, P4, P8, O1, OZ, O2)	4 channel of eye-tracking per subject (Gaze position x, y[8] and pupillometry: right eye, left eye)

Table 3 shows the different channels of data from EEG and Eye-tracking

4.2.3.4 *Result analysis and hypothesis*

Through the test Accuracy and prediction (description see Table 4) were given. Moreover, the ANN ran the data from each channel through the system 10 times in order to compensate for different weight initiations (the different channels can be found in Table 3). This gave an average of 10 X Accuracies and predictions at every channel at every event. These averages helped answer the final problem statement thus used to find significant differences. The datasets which was used to find differences both within and across participants were with these three element groups: (Accuracy, Precision) X (EEG, Eye-Tracking) X (Event categories) from which different null hypotheses was created:

H0: There would be no significant differences in the accuracy between the models trained on the different EEG channels at different event types

Accuracy and Precision

Accuracy = True Positive + True Negative / (True Positive + False Positive + True Negative + False Negative) it is equal to the proportion of true in the test population [134]. It is useful to tell the overall recognition rate and give a broad picture of how well the model is at correctly recognizing the events.

Precision = True Positive / (True Positive + False Positive) it is equal to the amount of true recognized positive within all classified positives [134], [135]. The reason for including precision is because the quality of each event class in each dataset can vary dependent on the quality check during data acquisition a low Precision can occur with a high Accuracy if the channel's quality of a specific data is low relative to the other channels in the dataset.

Table 4 shows the way to calculate Accuracy and Precision.

---

[8] Gaze position x and Gaze position Y, means X coordinate or the Y coordinate on the screen, which the subject is gazing at.

[9] It is based on the same principle as the error rate, but it is only in relation to the validation partition of the 10 fold cross validation.

H1: There would be no significant differences in the precision between the models trained on different EEG channels at different event types

H2: There would be no significant differences in the accuracy between the models trained on different eye-tracking measurements at different event types

H3: There would be no significant differences in the precision between the models trained on different eye-tracking measurements at different event types

H4: There would be no significant differences between the different game event types based on EEG trained model's accumulated accuracies.

H5: There would be no significant differences between the different game event types based on EEG trained model's accumulated precisions.

H6: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated accuracies.

H7: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated precisions.

hypotheses H0 – H3 were analyzed through Kruskal–Wallis one-way ANOVA test[113], as it only concerns one independent variable and it was assumed that the data from the participants will not be parametric. While H4-H7 were answered through the use of One-way ANOVA test as the hypotheses only concerns one independent variable. It was assumed that the accumulated accuracies or precisions approximately followed a normal distribution as well. All tests showed significant differences if $P < 0.05$.

4.2.4 Method Summary

In this chapter the methodology behind the experiment and data processing have been covered. The procedure behind the experiment which denotes the importance of the subject moving as little as possible while playing the game as it concerns the psychophysiological signal quality. The protocol for analyzing the data has been visited as well which describes in detail the data acquisition, filtering of the data, feature and artifact extraction, selection of event and the amount of data per epoch. The ANN architecture has been defined to work the with following parameter, 3 hidden layers, with 32, 16 and 8 neurons in the layers respectively, Hyperbolic tangent activation function in the hidden layers and a sigmoid function in the output layer, for learning a resilient backpropagation algorithm was employed. Lastly this chapter concerned evaluation of the data, which was accessed through 8 Null hypotheses where one-way ANOVA tests and Kruskal-Wallis ANOVA test provided the answers and thus answer the FPS.

5 Design

In this chapter will the design of the Super Mario application be accessed. Since it is a pre created game it will mostly concern the design of the levels, which should support uniformity albeit not creating identical replicas.
This chapter will be divided into two sections: one concerning elements crucial for this study and a second section to support data gathering for a secondary study (which can be read more about that in section 9.3).

5.1 Level layout

The Super Mario game adopted for this thesis has been taken from the Mario AI Procedural content generation competition which ran from 2009-2012[7]. It had the goal to automatically produce level layouts, which supports subject's perceived fun of the game [5], [114]. With this procedural content generation feature different level layouts have been generated for this project. Moreover, those have nothing to do with optimization of the perceived fun of the levels, as it was not the focus for this project.
The different generated level can be found in Figure 22. There are certain consistencies between those which will be addressed. First all the levels start with an area with 10-11 gaps (Marked by green). This area does not contain anything else. Thus showing extreme case of homogeneity. The next area (Marked with red) consist of a mixture of enemies and boxes. The amount of boxes and enemies seems to only vary little across the levels, albeit the amount of boxes is between 32 -58 and the amount of enemies is between 22-33. In respect to the amount of required events the user has to experience in order to make the type of event considerable in the analysis (see section 4.2.3.3) then this amount is promising. When it comes to "colliding with box events (**CB**)" there are five types of events (see section 0), if we assume that the

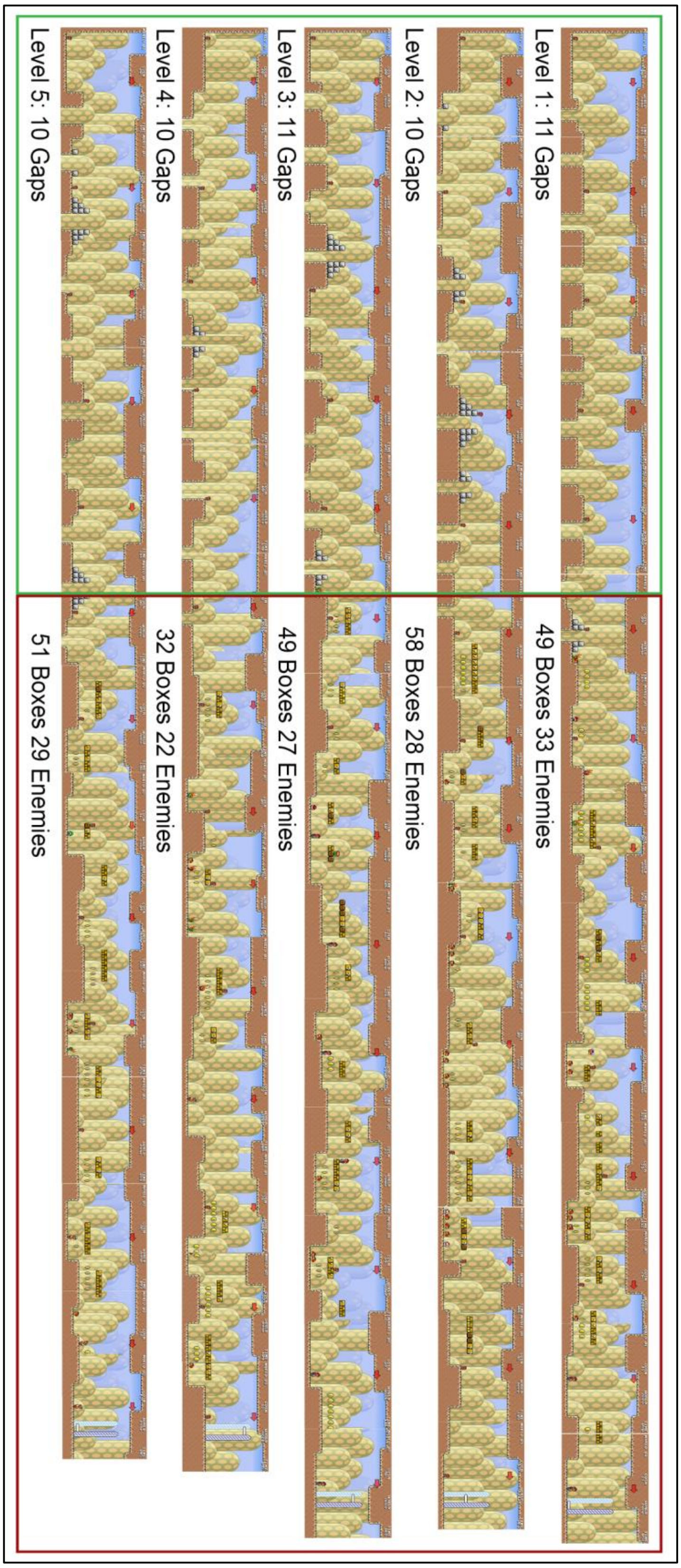

Figure 22 shows five different level layouts, which the experiment is based upon.

chance of getting either of the types of event at the boxes is equal then: sum of boxes / 5 = 47. Which indicate that there is a fair chance that the amount of wished events for this category can occur. Moreover, that will only happen if the subject interacts with all the boxes. In relation to the "killing enemies (**KE**)" category the situation is similar as: the sum of enemies / 3 types of killing = 46, but only if the subject kills them all with equal distribution between the 3 types of killing. Through these level designs these two categories might be the only one with a fair chance of being elicited enough times. Moreover, the coin is also apparent a lot of places in the level, but discussed in the investigation (see section 2.3.3) its psychophysiological foot print might not be unique enough to be recognizable. For the other categories "Mario changing form by collision events (**MFC**)", "Ending level events (**EL**)", "automated ending level events (**AE**)" and "avoid dying in a gap events (**ADG**)" it mostly depends on the performance of the subject. If the subject is performing badly several AE happens and ELs as well. Moreover, depending if the user fails to jump over the gaps or manages to get killed by enemies a lot. would influence the amount of MFC, ADG, KE and CB, because the subject will never reach many events due to bad performance. If the subjects perform good, the amount of ADG will be secured, as there is at least 10 of those event per level. A good performance might instead conflict with the amount of MFC as the subject will be able to change form a few times in order to accomplish the levels. When Mario is in his highest form the interest of colliding with the boxes is low, as those does not contain anything the subject is crucially in need of, thus not part of the game schema [22]. Due to the layout of the level the sufficient amount of elicited types of events is only certain for a few of the categories. Furthermore, if different player types [115] are taken into consideration the sufficient elicitation of the different types of categories becomes uncertain.

5.2 Design for secondary study

In the method chapter (see section 4.1) it was stated that the subject will experience 10 levels of Super Mario. This is achieved by duplicating the five created levels. In this way the player will be playing two identical levels after each other. This conflicts with the requirement that they should not be similar. However, there is a catch. Three of the events found in this Super Mario game are not normal within the realm of the traditional Super Mario game [116], those events are "Bouncing out of gap", "Enemy appearing by collision of a box" and "evolving if colliding with an enemy". They provide an opportunity to study the effect of unexpected game events on the user experience. Given that it would not influence the main purpose of this study these events have been incorporated in every second level, although in switching order.

5.2.1 Design of preference and user experience questionnaire

In order to access the user experience a 4-alternative forced choice (4-AFC[10]) questionnaire have been designed similar to those used in the following studies[2], [3], [72], [77]. The questionnaire is designed to interrupt the player after every second level that has either been won or lost. Challenge, surprising, novelty, frustration and engagement is accessed to learn about the user's preference and experience (see Figure 23 to get a view of the questionnaire). These elements are thought to be influenced by the added

[10] the 4-AFC is a questionnaire consist of four different answers, which the user is "forced" to choose from in respect to the different user experience categories. "The first level", "the second level", "both of them", and "neither of them".

unexpected events. Moreover, there is a chance that the user will not experience them enough to be able to distinguish the two levels from each other, thus making them completely identical.

5.3 Design Summary

Through the procedural content generator embedded in the Super Mario application different levels which supports the requirements for the study can be created. Moreover, this thesis supports a second study which design elements have been taken into consideration thus being able to include in the final design of the application.

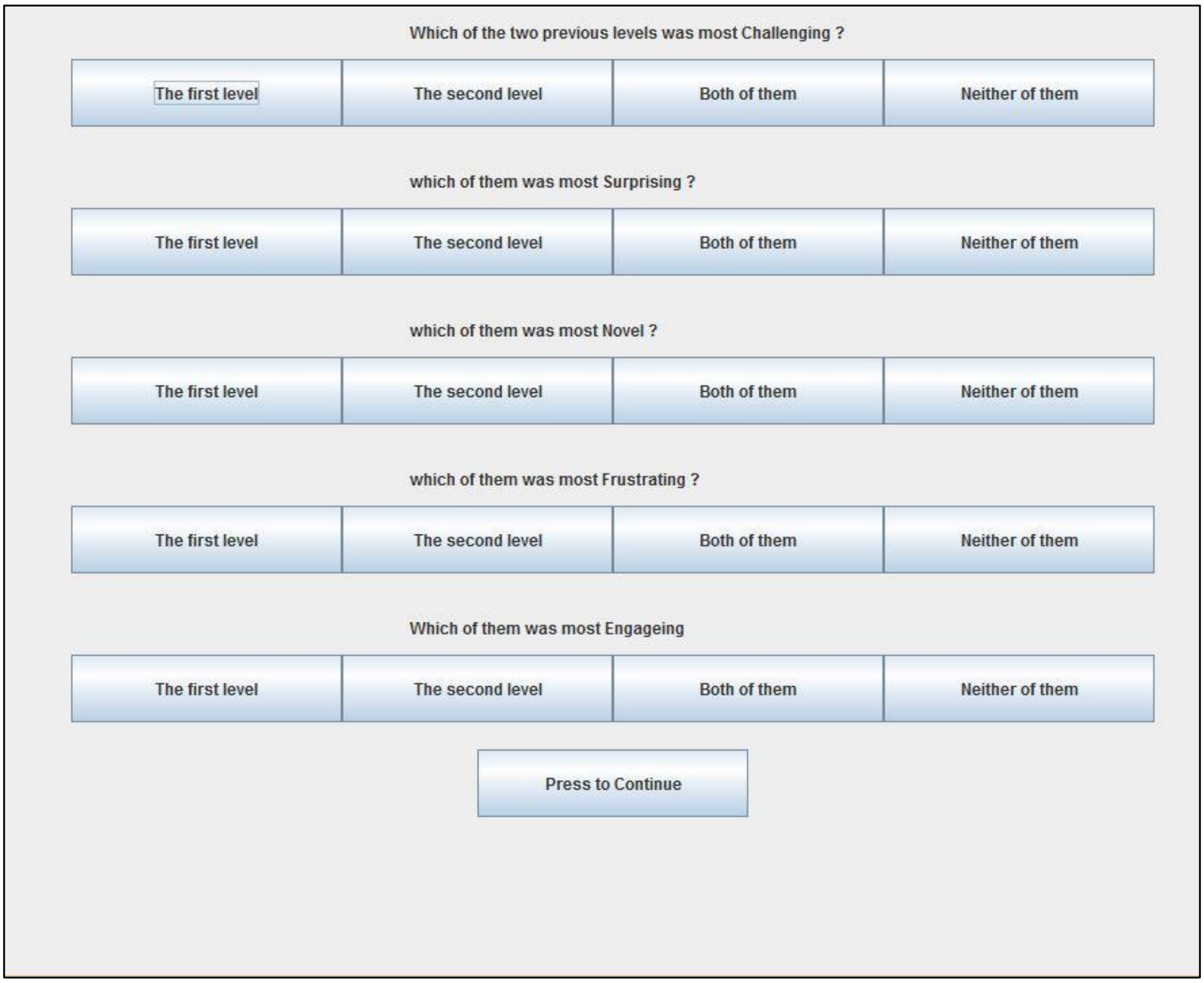

Figure 23 shows the questionnaire elicited after every second level

6 Implementation

This chapter will go through the data communication setup and the three different applications the communication protocol is made of, thus making the thesis possible. Firstly, the Super Mario application will be addressed, but only those elements which the author has implemented and the structure around those elements. Secondly, the communication and synchronization structure between the game and the EEG-recording. Lastly, the implemented Artificial Neural Network will be addressed.

6.1 Data communication overview

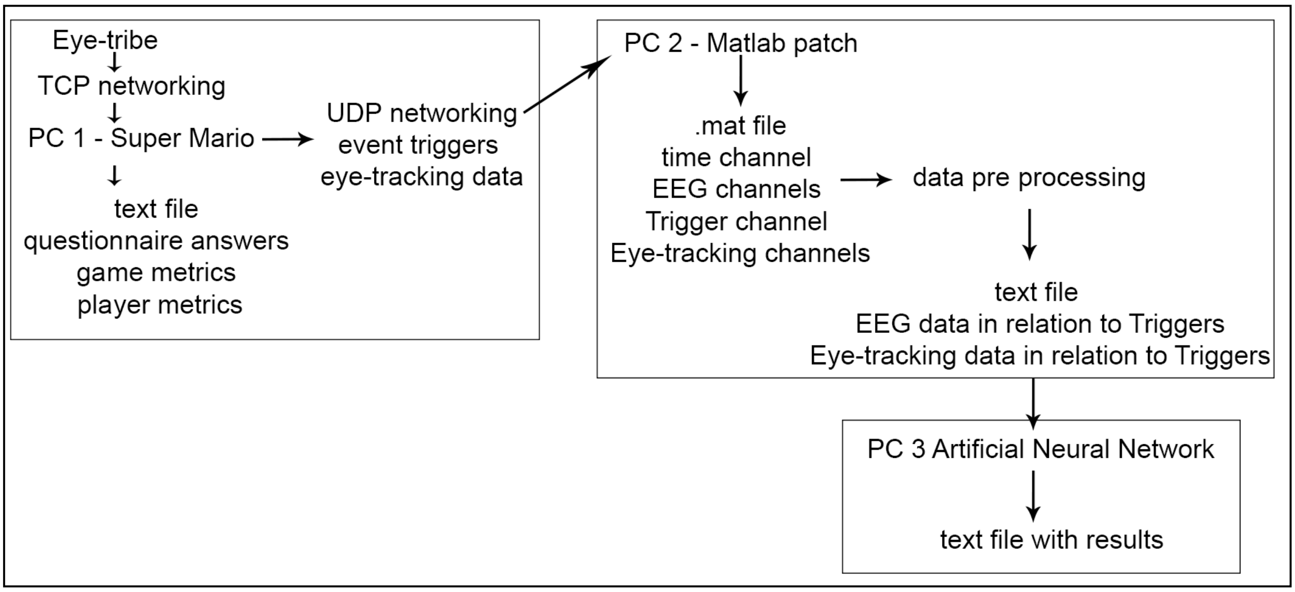

Figure 24 shows an overview of how the data is communicated between computers and programs through the use of different modalities.

To get a quick idea of how the data is communicated throughout this thesis the protocol in Figure 24 is followed. First the Eye-tracking and Game event triggers are gathered in PC1, which runs the Super Mario game. The Eye-tracking data and the game events is sent through UDP networking to a second PC (PC2), which runs the Matlab patch It is responsible for synchronizing and recording the EEG with the Eye-tracking and the event triggers. After the experiment a .mat file is created with all the data. It is preprocessed following the paradigm in method section 4.2.3.1. It extracts a text file which the artificial neural network can read. The network on the third PC works with the data, through training it provides the measureable results.

6.2 The Super Mario Application

The standard Super Mario application as it is when downloaded from the competition website gives a working prototype coded in Java which can be modified as you please [7]. The application can generate two sort of levels, one which is random and another which is dependent on the player metrics from the user. With the random level player metrics can be generated which influence the layout of the second type of level. As a standard the application lets you play one of either level then it quits the application. Since the game is in a working condition only a few things needs to be implemented in order to satisfy the requirements stated in previous sections (3.1, 4.1, and 5.2). Those are:

automated continuation to the next level – as the application quits after each level an automatic mechanism is created to shift levels.

unexpected events – Mario bouncing out of a gap, enemy appearing as a result of hitting a box and Mario evolve when colliding with an enemy.

Triggers – every time one of the investigated events happens a unique paired number(trigger) should be elicited.

Questionnaire – in which the user will answer the 4 – AFC questions

extraction of information gathered through the game and questionnaire – For the secondary study the player behavior through the game and the answers to the questionnaire needs to be exported to an external file, so the information can be accessed at any time.

Eye-tracker communication – in order to make them synchronized by the rest of the data streams

UDP networking – to send information about eye-tracking and events to the computer recording the EEG signal.

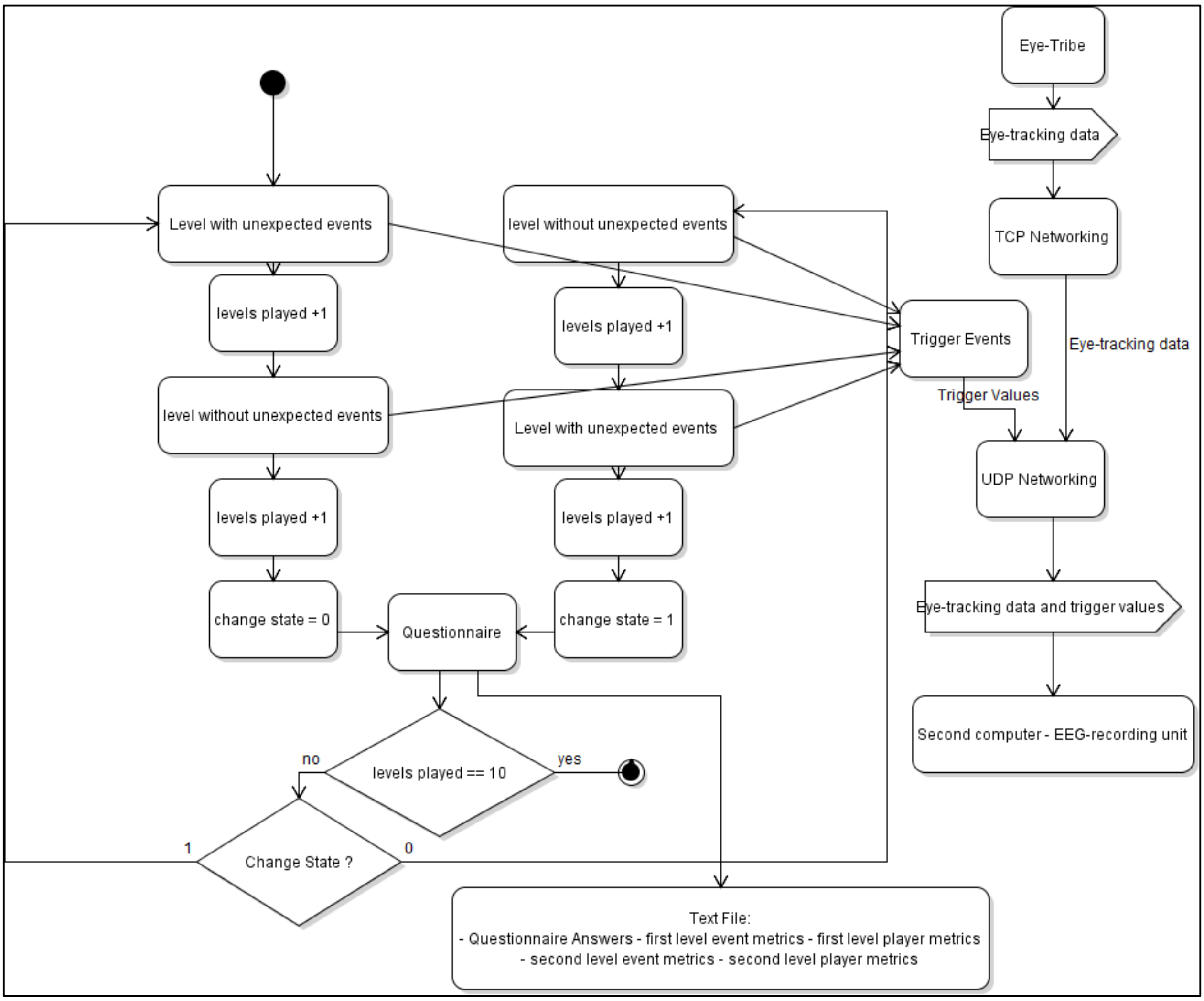

Figure 25 shows the implemented structure of the Super Mario Game, where the data communication is presented as well.

6.2.1 Automated continuation to next level

In the standard Mario game, the gameplay and the level is running on a thread [117]. This device keeps track of whether the game is running, having a break or is stopped. When the user has reached the ending point of a level either by dying or winning a corresponding screen is shown. It is present until the user quits the program. This action will stop and delete the thread running the program. In order for the program to continue to next level a new thread is started containing the new level. A problem with this is that the old thread from the old level is never stopped. If we imagine that a player plays this game for 30 levels the computer will be running 29 inactive threads which consumes memory, thus making the computer and the Mario application slow. Fortunately, the stop function of the old thread can be called at the wake of the new one thus only having one game thread running at a time.

6.2.1.1 Shifting between level conditions

In the design chapter a secondary study was encountered, which wanted the game to shift between two types of levels: one with unexpected events and one without. it is controlled by a simple shifting mechanism and can be seen in the diagram in Figure 25. This mechanism is implemented when the level is awoken. It is controlled by the amount of levels the subject has been through. It shifts condition after each level and at the same time it changes order for every second level the user has been through, thus avoiding order bias in thequestionnaire.

6.2.2 Unexpected events

These events are created as replicas of existing events, but elicited in an unfamiliar context. As mentioned these event are: bouncing out of gap (BoG), enemy appearing when hitting a box (EB) and evolving when colliding with enemies (ECE). They have been implemented in a jeopardy system where program decides by random whether it should appear or not. For BoG and EB there is a 2/5 chance that it would happen and for ECE there is 1/5 chance. These chances are chosen as they seem to result in making the unexpected events happen as often as they are noticeable as a part of the game, but without eliciting them as often as they seem dominant thus changing the player's perception of consequences of different interactions. The chance for ECE is lower than the others, due to the way collision is calculated. When Mario collides with an enemy their pixels overlap. This overlap can last several updates[11], and at every single update the jeopardy system activated. This makes it automatically it more likely that ECE will happen, as it is given more chances. Therefore, is its chance of occurrence lowered.

6.2.3 Triggers

These elements are simple but important elements of this experiment. They are numerical representations of every type of event occurring throughout the game, and they are elicited at the very moment the event occurs. This number is sent through UDP networking to the computer which records the EEG signal. Through these numbers the EEG and the Eye-tracking can be related to specific game events and to each other. Based on the table of type of events (see section 0) 22 uniquely numbered triggers has been implemented in the system in order to distinguish between the different types of events.

6.2.4 Questionnaire

The main changes internally to the Super Mario program has been explained. The next implemented elements are added to the program, thus having nothing natively to do with Super Mario.
The questionnaire is implemented to occur after every second level. An example is provided in the design chapter (see section 5.2.1). The main mechanics of this questionnaire is accessed through the use of JPanel, JLabels, JButtons, JToggleButtons & JButtonGroups[12]. The JPanel is the area on which the different elements of the questionnaire are represented. This makes the questionnaire acts like an automatic pop-up window when the player has played two levels. In this window several JLabel, JtoggleButtons and a JButton is manually placed to give the questionnaire its layout without following any structure rules, which the JPanel can inherit. However, none of them fitted the needs for this questionnaire (read more about JPanel structures here [118]). The JLabels in the questionnaire are represented by the written questions in the window. The JButtonToggles are all the answer options. Through the use of JButtonGroups, JButtonToggles are grouped together in respect to the different questions so they are mutually exclusive. The last element on the questionnaire is the JButton which is used for the "Continue" button at the bottom. This button is

[11] An update is every time the program has run all its accessible functions through. It is like one big while loop. While program is active start from the top and read all the instructions, when the bottom is reached start from the top again. If the program is stopped == not active, stop reading the instructions.
[12] The J components are visual building blocks natively available through the Java development kit. They are primarily used as Graphical User Interface elements (GUI)[132]

only clickable if all of the questions have been answered, thus ensuring that the user have clicked them. When the continue button is clicked the answers are recorded and saved in a text document.

6.2.5 Exporting game information and questionnaire answers

This text document contains not only the answers from the questionnaire, but also player and game metrics such as all the events encountered through the game and all the actions like running and jumping the user has done. The text file is created at the time the questionnaire is answered, thus including metrics from both levels that have been played prior to the questionnaire. The order of information in the document are as follows: numerical representation of questionnaire answers[13], encountered events and player metrics from the first played level and then metrics from the second level. The order of the level is noted in the title of the document. This is employed so it is easier to distinguish the information from each other. The level with the unexpected events are denoted as "level with surprise", the other level is denoted as "level with no surprise".

6.2.6 Eye-tracking integration

In order for the eye-tracking data to be easily synchronized with the system, integrating it in the game seems like an optimal solution. The Eye-tribe eye tracker which is used in this project can be communicated to with the help of an eye-tribe API[93] which can be integrated in Java developed systems. This API can be used to retrieve data from the Eye-tracker through its TCP networking paradigm. When the game is started it instantiate a connection to the eye-tribe through TCP connection, it receives all gaze positions and pupillometry data, packs them into array packages and sends them via UDP networking to the EEG-recording computer. In this way all the real time data is collected on one machine.

6.2.7 UDP Networking

The last important element implemented in the game system is the UDP networking component. It is responsible for sending all event triggers and eye-tracking data to the computer which records the EEG-signal. Essentially there are two part of the UDP networking component: the one responsible for the event triggers and the other responsible for the eye-tracking signal. The part responsible for the event triggers packs the numerical value in a byte package. This package is send to an assigned IP address through an assigned network port.

The second part of the UDP component is less straightforward. It is known that the receiving program (see reference [8]) reads a package of bytes through the Big-Endian[14] procedure. Moreover, if the game coded in Java is told to encode its package through the big-endian procedure it sends it like little-endian[15]. In order to ensure the right byte-order the byte package is manually accessed in the program in order to re-order and reverse the bytes so it emulates Big-endian instead.

[13] As the 4-AFC questionnaire consist of four different answers the following coding have been employed: "the first level" = 1, "the second level" = 2, "both of them" = 3, "neither of them" = 4.

[14] Big-endian is a way of storing bytes in the memory and reading them. Through this the received array of bytes are stored and read the same way. Thus the first memory address is the first byte, second memory address is the second byte, and so on.

[15] Little-endian is the opposite of the Big-endian. If an array of 8 bytes is encoded through this, the first byte will be stored in the last memory address of eight, the second in the sevenths, and so on. Some programs read the byte backwards thus sometimes it is necessary to encode it this way.

6.3 Matlab Patch for signal synchronization

When the Eye-tracking and the event triggers have been sent from the computer running the Super Mario game, a second machine responsible for synchronizing all signals receives them. This machines runs a program within Matlab called Simulink. In Simulink it is possible to create real-time applications for recording and synchronizing psychophysiological signals. The g.Tec system found in the Augmented Cognition Lab is connected to this application and has a time resolution of 256 Hz. Since it is the fastest update rate of the recorded signals the Simulink program is configured to record and receive everything at 256 Hz. Moreover, the eye-tribe signal is being send with a temporal resolution of 60Hz, which can cause some interesting issues. However, those have been discussed and dealt with in my 9th semester project in which this program has been described thoroughly as well [8]. In the Simulink program the EEG signal is also viewable in a filtered version. Moreover, it will be saved in raw format in this experiment, so it can be manipulated more versatilely. The synchronized data file this program creates contain the following parameter 1 stream of time data, 16 streams of EEG data, 1 stream of Trigger data, 4 streams of Eye-tracking data.

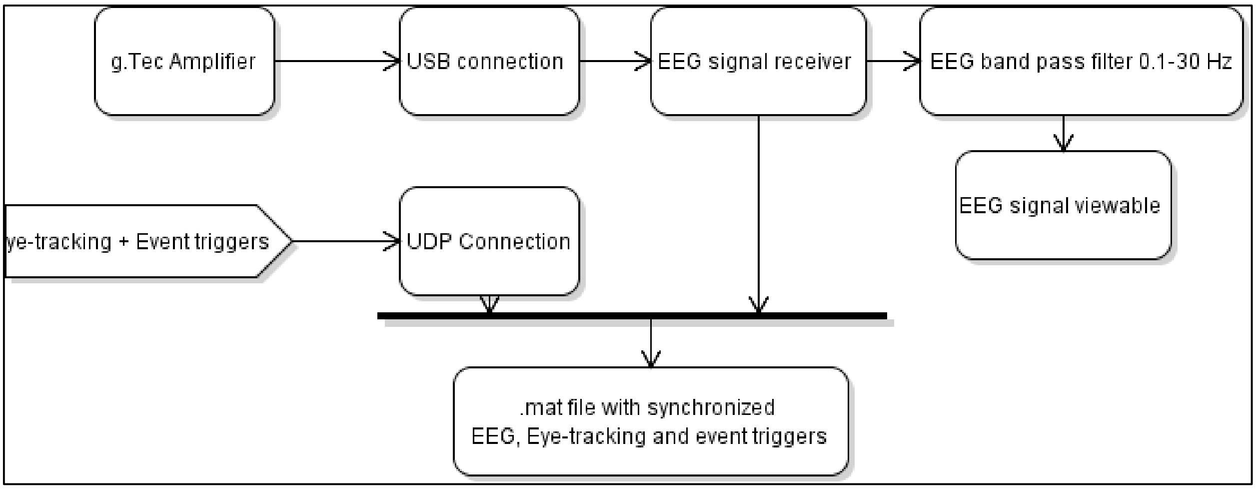

Figure 26 shows the implementation of the Simulink program responsible for synchronizing all data

6.4 Artificial Neural Network

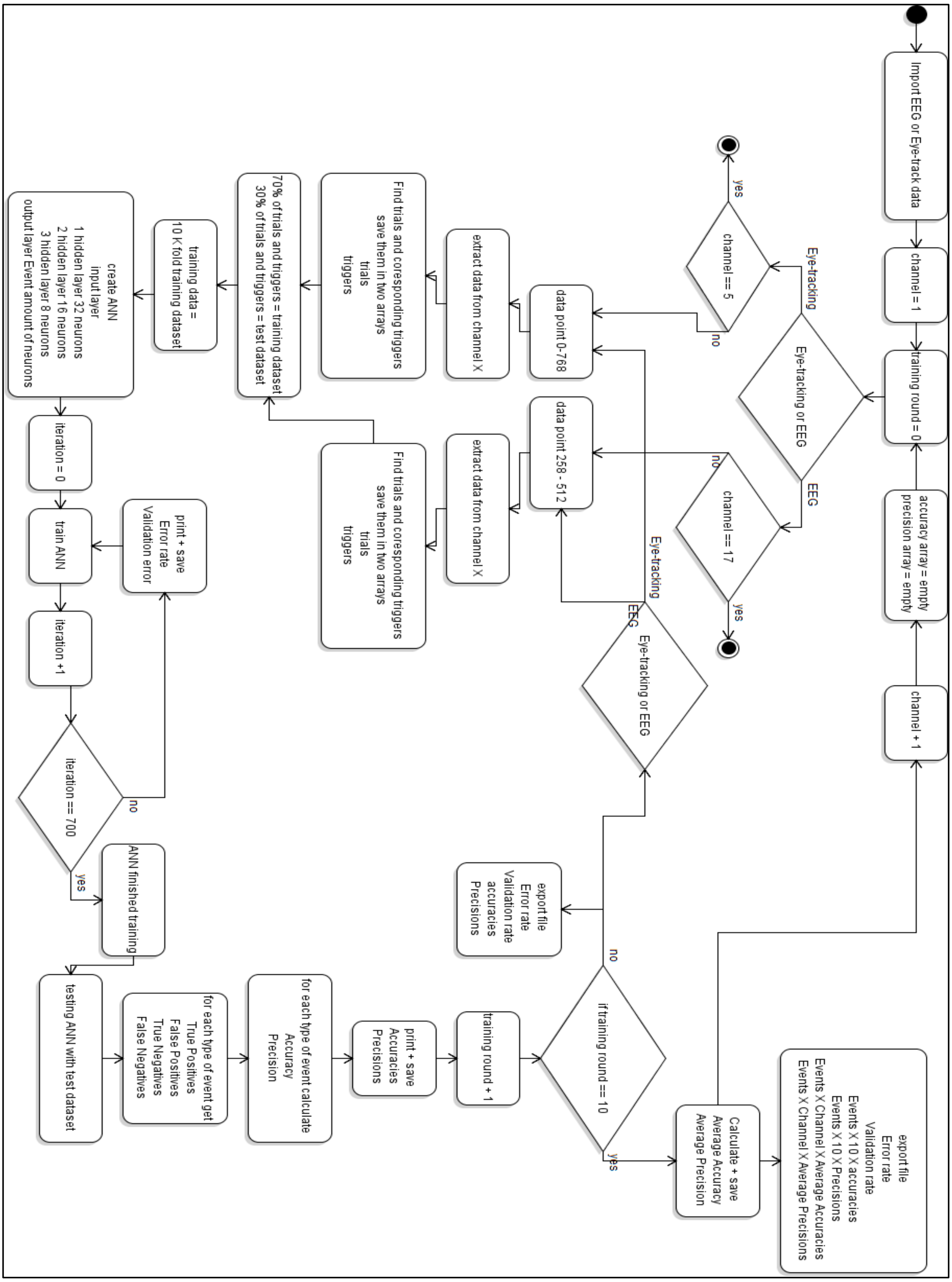

Figure 27 shows the implemented Artificial Neural Network. all way from data input, data organization, ANN training, result calculation to result output

Figure 27 shows the structure of the ANN program. From importing the data to the export of the final results. It is a flexible program, which with smaller adjustments can work with any input data. In the beginning it asks for a filename, the range of data it should take into consideration and the channel number. The channel number is a uniformly used both in the EEG and Eye-tracking text files from Matlab. The channel numbers either correspond to different electrode placements or the sort of eye-tracking data (puillometry, gaze position). When the channel number has been chosen the program finds every time the channel occurs. Stores the data needed, told by the range of data, in an expanding array List. When it is done, the data is normalized and divided into two two-dimensional arrays. 70% of it into a training dataset and 30% of it into a test dataset. The data is converted to arrays instead of lists, because the machine learning framework (EnCog [119]) works with 2D arrays.

6.4.1 EnCog Machine Learning Framework

This framework consists of several advanced machine learning algorithms like artificial neural network and generic algorithms [119]. The framework is open source for everybody to use and upgrade, which have resulted in a rather huge expansion since its beginning. The framework is constructed in multiple languages (e.g. Java, C#, C /C++, JavaScript) and designed to be easy recode able when an implementation is working. Implying, if an specific machine learning algorithm is implemented a new one can exchange place with it by changing few lines of code [119] In this thesis the C# version of EnCog was employed. It has served as a resource for creating the artificial neural network, training it though cross validation, creating error rate, creating validation rate, and testing the network. This framework is the very backbone of the application, and provides great flexibility in relation to exchanging activation function, end strategies and training algorithms.

In the code snippet in Figure 28 the network is created within it the activation function (ActivationTANH,

```
var eegNetwork = new BasicNetwork();
eegNetwork.AddLayer(new BasicLayer(null, true, fr.getTotDataPoints()));
eegNetwork.AddLayer(new BasicLayer(new ActivationTANH(), true, 32));
eegNetwork.AddLayer(new BasicLayer(new ActivationTANH(), true, 16));
eegNetwork.AddLayer(new BasicLayer(new ActivationTANH(), true, 8));
eegNetwork.AddLayer(new BasicLayer(new ActivationSigmoid(), false, 4));
eegNetwork.Structure.FinalizeStructure();
eegNetwork.Reset();
```

Figure 28 shows a code snipped used by EnCog to create the ANN

ActivationSigmoid) and the amount of neurons (32, 16, 8, 4) is defined as well. Those can be easily exchanged to a different amount of neurons and other activation functions like a bipolar activation, which either gives you a 1 or a -1 as an output.

```
IMLTrain trainCrossFoldEEGData = new ResilientPropagation(eegNetwork, crossFoldEEGDataset);
```

Figure 29 shows a short code snippet, where the training algorithm is applied

The training algorithm can as easily be changed as long as it is within the EnCog repository. In Figure 29 ResilientPropagation is coded, this can be changed to normal back propagation by replacing ResilientPropagation with BackPropagation or others without changing anything else.

```
} while (epochEye < amountOfIteration);

crossValidated.FinishTraining();
```

Figure 30 shows the ending strategy and the code snippet which denotes that the network has finished

The last piece of the algorithm shown in Figure 30 serves the purpose of ending the training of the algorithm. In this case a certain amount of iterations needs to be met before the ANN model can be tested. This can very easily be changed to be a certain error rate the training algorithm needs to achieve before quitting the training.

When the algorithm is ready to be tested the test dataset is applied. Since the last activation function is a sigmoid the outcome is between 0 and 1. For the purpose of calculating the statistics afterwards it can be rather arbitrary to compare this floating number with 0 and 1, which is the two only numbers given as a true result. These numbers can be directly translated to event or not event where 1 = event and 0 = not event. In order to compensate from this a simple threshold has been created. It sorts every number above 0.5 to be 1 and every number below 0.5 to be 0. Through the threshold True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) can be derived (see Table 5).

Ideal output :	1	1	0	0
Actual output :	1	0	1	0
	TP	FP	FN	TN

Table 5 shows the relationship between ideal output and actual output, and what their accumulated results is equal to.

After TP, FP, FN and TN has been found accuracy and precision will be calculated following the guidelines found in the method chapter (4.2.3.4). These are printed to the console and saved in a text file together with all the error rates and validation rates calculated through the training session.

The whole procedure covered this far will run uninterrupted ten times with the same data to try out different weight initiations in the network and thus giving different accuracies and precisions. At the tenth time the average of the accuracies and the precisions is calculated, stored and printed in a text file. The channel number is then added by one, and the whole paradigm will terminate again with the new data from that particular channel. If the data comes from EEG and the reached channel is 17 the program will quit itself, because it is redundant to look for a channel that does not exist. If the data comes from eye-tracking the channel number is five.

6.5 Implementation Summary

Through this implementation the different technical parts of the thesis have been described. The journey of the data starting from the modified Super Mario game, synchronized through Matlab and analyzed by the ANN application. The Super Mario game is modified so it continuously let the subject play through different uniform levels with or without unexpected event. Through the game session the play is interrupted every second level with a questionnaire created by J-components in Java. The Matlab program synchronize the EEG data with Eye-tracking and Event triggers through a temporal resolution of 256Hz. The data is exported into .mat files, which can be accessed and modified. The ANN received the data after it has been preprocessed (see preprocess method in section 4.2.3.1) The ANN divides the data into channels and trials and is trained to predict them through 10 training sessions per data channel. From the ANN average Accuracies and Precision is calculated and exported.

7 Result

In this chapter an overview of the demographic data will be given. The amount of events all participants' datasets contains, after they were cleared for EEG artifacts are examined as well. With those events the different hypotheses devised in the method chapter will be answered and ultimately answering the final problem statement.

7.1 Demographic Questionnaire results

The total amount of participants acquired for this experiment were 31. 7 of them were females and 24 of them were males. Their average age = 26.16(SD = 6.9). Most of them were students or interns as they accounted for 27 of the participants. 18 of these were graduate students and 9 of them were undergraduate students.

Apart from their background all the participants were asked about their general gaming behavior and whether or not they have played Super Mario before. 29 of 31 have played super Mario before. The majority of the participants play video games on a weekly or daily basis, those accounted for 21 of them. 5 of 31 play video games on a monthly basis and 5 play video games rarer or never.

These results suggest that the participant is mostly a homogenous group of videogame playing students in their twenties.

7.2 The events – which the ANN shall predict

In appendix H a table with all the events is presented. It shows how many of the different events have occurred per subject after all the events with artifacts in the EEG signal has been sorted out. Through the requirements for selecting events found in the method chapter (see section 4.2.3.1). The following type of events can be chosen for recognition: Jump, pick up a coin, breaking an empty box, hitting a box containing a coin, hitting a box containing a flower, killing an enemy with a stomp and killing an enemy with a fireball. At first "jump" and "pick up a coin" have been discussed to possibly not eliciting significant psychophysiological patterns (see section 2.3.3), thus they will be excluded from the dataset to the ANN. Within the other events it is important to find a combination of event, which can prevail the maximum amount of categories of events and event types per category while still maintaining as big amount of participants as possible.

Event combinations	Subjects
BB, CC, CF, KS, KF	11
BB, CC, KS, KF	13
CC, CF, KS, KF	13
BB, CF, KS, KF	12
BB, CC, CF KF	12
BB, CC, CF KS	15

Table 6 shows the amount of subjects that has elicited minimum 20 triggers per event in each combination of events.
Breaking an empty box (BB), Colliding with a box containing a coin (CC), Colliding with a box containing a flower (CF), Killing an enemy with a stomp (KS) and Killing an enemy with a fireball (KF)

The combination of events which showed most participant while still having the greatest amount of participants is "BB, CC, CF KS" (see Table 6 description for definitions). However, it suffers from having one event in the killing category and three in the box collision category (see section 0 for categories). The combinations which has equal amount of event types in each category is "BB, CC, KS, KF" and "CC, CF, KS, KF" They both have 13 participants to work with. However, through a discussion in section 2.3.3 BB is chosen as its discrete nature could produce clearer ERPs than CF. Thereby will the following combination of events will be fed into the ANN BB, CC, KS, KF. Through Appendix H it can further be seen that the least amount of repeated events within either of the types is 20. This gives the following datasets (see Table 7) per participant according to the dataset requirements found in the method chapter (see section 4.2.3.3).

	BB	CC	KS	KF
Training dataset	14	14	14	14
Test dataset	6	6	6	6

Table 7 shows the distribution of events between training datasets and test datasets.

7.3 Answer the Hypotheses

7.3.1 H0: There would be no significant differences in the accuracy between the models trained on the different EEG channels at different event types

Since there are four events, the answer to this hypothesis will be based on four Kruskal–Wallis one-way ANOVA tests (K-WANOVA). One for the "Breaking an empty box event", one for "colliding with a box containing a coin event", one for "killing an enemy with a stomp" and lastly one for "killing an enemy with a fireball". In case there are differences between pairs of them, the answer to the hypothesis will be based on K-WANOVA s on pairs of models based on electrodes as well.

For the "Breaking an empty box event" the result was Chi-sq(15,192) = 17.2 P = 0.307 thus not showing any significant differences between the models trained on different electrodes' data.

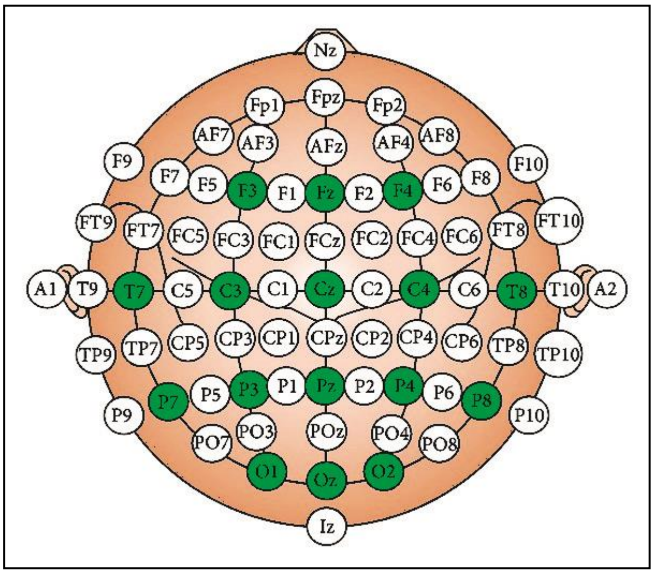

Figure 31 shows where all the used electrodes are placed spatially on the scalp

Moreover, there are pairs within these models trained on electrodes' data that showed significant differences, if they exclusively were analyzed and compared. (the used electrode placements can be found in Figure 31)
Ch3 F4 (Mean 64.64% (SD = 1.7%)) was significantly higher than Ch11 Pz (Mean 61.02% (SD = 3.0%))
Chi-sq (1,24) = 8.16 P = 0.0043.
Ch3 F4 was significantly higher than Ch5 C3 (Mean 62.30% (SD = 2.5%))
Chi-sq (1,24) = 6,09 P = 0.0136.
Ch11 Pz was significantly lower than Ch14 O1(Mean = 64.45% (SD = 2.57%))
Chi-sq (1,24) = 6.59 P = 0.0102
Ch5 C3 (Mean = 62.30% (SD = 2.5%))was significantly lower than Ch14 O1
Chi-sq (1,24) = 3.91 P = 0.0479
One pair of electrode which did not show a conclusive significance, but if more data was collected could have showed.
Ch9 P7 (Mean = 63.55% (SD = 2.75%)) and Ch11 Pz
Chi-sq (1,24) = 3.71 P = 0.0541

For the "Colliding with a box containing a coin" event the result was Chi-sq (15,192) = 13.86 P = 0.5363 Thereby it did not show any significance between the models trained on different electrodes' data

With this event one pair of electrodes' data showed significant difference
Ch5 C3 (Mean = 61.41%(SD = 2.82%)) was significantly higher than Ch9 P7(Mean = 57.91% (SD = 3.81%)
Chi-sq (1,24) = 4.12 P = 0.0423

For the "Killing an enemy with a stomp" event the result was Chi-sq (15,192) = 14.15 P = 0.5141 This didnot show any significance between the models trained on the electrodes' data.
Models based on the electrodes' data were inspected pair wise and the significant different pairs were

Ch2 Fz (Mean = 61.89% (SD = 2.78%)) was significantly higher than Ch4 T7(Mean = 57.69% (SD =4.71%))
Chi-sq (1,24) = 4.65 P = 0.031
Ch2 Fz was significantly higher than Ch10 P3 (Mean = 57.94% (SD = 3.93%))
Chi-sq (1,24) = 6.6 P = 0.0102

For the last event "Killing an enemy with a fireball" the result was Chi-sq (15,192) = 15.57 P = 0.4108 Thereby, it did not show any significant difference between the models trained on the electrodes' data.

Between the electrodes' data for this event significant differences can be observed between
Ch7 C4 (Mean = 67.30% (SD = 2.17%)) was significantly higher than Ch14 O1 (Mean = 64.87% (SD = 3.00%))
Chi-sq (1,24) = 4.14 P = 0.0418
Ch7 C4 was significantly higher than Ch16 O2 (Mean = 64.23% (SD = 4.09%))
Chi-sq(1,24) = 3.85 P = 0.0497
Closely if more observations were available:
Ch1 F2 (Mean = 65.35%(SD = 2.47%))and Ch7 C4
Chi-sq (1,24) = 3.55 P = 0.0595

Through these K-WANOVA tests differences between selected electrode pairs have been found thus rejecting the null hypothesis. Moreover, based only on K-WANOVA including all electrodes then the null hypotheses cannot be rejected. Supportive box plots can be found in Appendix J.1.

7.3.2 H1: There would be no significant differences in the precision between the models trained on different EEG channels at different event types

The same way as with the last hypothesis this one will be based on four K-WANOVA test over the four different events and on K-WANOVAs between pairs of electrodes.
For the "breaking an empty box" event the result is Chi-sq (15,192) = 16.4 P = 0.3561 thereby, it did not show any significant differences between the models trained on the electrodes' data in relation to the precision.

The pairs of electrodes' data which showed significant differences were

Ch1 F3 (Mean = 25.28% (SD = 7.24%)) was significantly higher than Ch14 O1 (Mean = 19.63% (SD = 4.75%))
Chi-sq (1,24) = 4.53 P = 0.0333
Ch3 F4 (Mean = 26.54% (SD = 8.04%)) was significantly higher than Ch10 P3 (Mean = 19.82% (SD = 6.87%))
Chi-sq 1,24) = 4.31 P = 0.0378
Ch3 F4 was significantly higher than Ch14 O1 Chi-sq (1,24) = 4.13 P = 0.0378
Ch6 Cz (Mean = 24.34%(SD = 4.63%))was significantly higher than Ch14 O1
Chi-sq (1,24) = 5.44 P = 0.0196

For "Colliding with a box containing a coin event" the result was Chi-sq (14,192) = 31.22 P = 0.585 Thereby it did not show any significant difference between models trained on the electrodes' data in relation to precision.

Between pairs of electrodes' data, the following significant differences could be observed.
Ch1 F3 (Mean = 26.06% (SD = 5.74%)) was significantly higher than Ch8 T8 (Mean = 20.77% (SD = 5.05%))

Chi-sq (1,24) = 4.98 P = 0.0257.
Ch7 C4 (Mean = 26.55% (SD = 6.50%)) was significantly higher than Ch8 T8
Chi-sq (1,24) = 7.25 P = 0.0071
Ch8 T8 was significantly lower than Ch15 Oz (Mean = 26.09% (SD = 5.53%))
Chi-sq (1,24) =4.98 P = 0.0257
Ch8 T8 was significantly lower than Ch16 O2 (Mean = 26.07% (SD = 8.29))
Chi-sq(1,24) = 4.53 P = 0.0333

For the event "Killing an enemy with a stomp" the result was: Chi-sq (15,192) =16.32 P = 0.3609 thereby, it did not show any significant difference between the models trained on the electrodes' data in relation to precision.
Through accessing different pairs of electrodes' data the following significant differences were found
Ch4 T7 (Mean = 22.94% (SD = 5.52%)) was significantly lower than Ch15 Oz (Mean = 29.12% (SD = 5.13%))
Chi-sq (1,24) = 6.19 P = 0.0129
Ch7 C4 (Mean = 23.42% (SD = 7.25%))was significantly lower than Ch15 Oz Chi-sq (1,24) = 5.44 P = 0.0196

For the "Killing an enemy with a fireball" the result was Chi-sq (15,192) = 10.82 P = 0.7652 thereby, it did not show any significant differences between the models trained on the electrodes' data in relation to precision.

Between different pairs of electrodes' data there were no significant differences. Their mean differed between 19.85% - 25.50% where the highest mean was found at Ch13 P8 (SD = 8.16) the lowest mean was found at Ch11 Pz (SD = 8.85%)

For this hypothesis difference between electrode pairs were significant in three of the four events and there were no significant differences between any of the K-WANOVA tests based on the models trained on all electrodes' data. This means that the null hypothesis is rejected in three out of four events for specific electrode pairs. (for supportive box plots see Appendix J.2.)

7.3.3 H2: There would be no significant differences in the accuracy between the models trained on different eye-tracking measurements at different event types

To answer this hypothesis, the K-WANOVA test have been tested on the accuracy from the four different measurements' data in relation to the four different events. Thereby creating four different K-WANOVA tests. Pairs of eye-tracking measurements' data will be tested through K-WANOVA as well to exploit were the significant differences are.

For the first event "Breaking an empty box" the result was Chi-sq (3,48) = 2.94 P= 0.4007 Thus not showing any significant differences between the models trained on the different eye-tracking data in relation to accuracy.

differences between pairs of eye-tracking measurements' data were not significant. Although, their means were between 57.78% – 60.99% where the highest was found at the left pupil (SD = 5.27%) and the lowest was found at Gaze position X (SD = 4.38%).

For the next event "Colliding with a box containing a coin" the result was Chi-sq (3,48) = 1.66 P = 0.6459 Thus indicating that there was no significant difference between the models trained on the eye-tracking

measurements' data in relation to accuracy
K-WANOVA tests on pairs of the models accuracies did not show any significance differences either. Furthermore, their means differed between 58.62% - 60.51% where the highest was found at the Right Pupil (SD = 4.15%) and the lowest was found at Gaze position X (SD = 3.40%)

For the next event "Killing an enemy with a stomp" the result was Chi-sq (3,48) = 8.76 P = 0.0327 shows that there was a significant difference between the models trained on the eye-measurements based on accuracy.
The significant different were caused by these pairs of models trained on these data.
Right pupil (Mean = 62.62% (SD = 4.54%)) was significantly higher than Left pupil (Mean = 58.26% (SD = 4.82%))
Chi-sq (1,24) = 4.01 P= 0.0452
Right pupil was significantly higher than Gaze position Y (Mean 56.69% (SD = 5.55%))
Chi-sq (1,24) = 6.59 P = 0.0103

For the last event "Killing with a fireball" the result was Chi-sq (3,48) = 13.59 P = 0.0141 which proves that there is a significant difference between the models trained on different Eye-tracking measurements' data In relation to accuracy
The significant difference was between.
Right Pupil (Mean = 69.48% (SD = 2.43%)) was significantly higher than Left Pupil (Mean = 64.67% (SD = 2.92%))
F(1,24) =11 P = 0.0009
Through these K-WANOVA tests the hypothesis can be rejected in two of the four events. both in relation to the K-WANOVA tests based on all model's trained on the measurements' data and between individual pairs of models. (see supportive box plots in Appendix J.3.)

7.3.4 H3: There would be no significant differences in the precision between the models trained on different eye-tracking measurements at different event types

To answer this hypothesis the same procedure will be followed as towards answering the last hypothesis

For the first event "Breaking an empty box" the result was Chi-sq (3,48) = 2.24. P = 0.5235 thus no significant difference between the models trained on the eye-tracking measurements' data in regards to precision.
Neither of the paired K-WANOVA tests showed any significant differences. Although, there means were between 22.35% - 25.38% where the highest were at the Left Pupil (SD = 5.67%) and the lowest were achieved at Gaze position X (SD = 4.88%)

For the next event "Colliding with a box containing a coin" the result was Chi-sq (3,48) = 6.93 P = 0.0743 Thus not showing significant difference between the models trained on the different Eye-tracking measurements' data in regards to precision
However, the pair which showed significant difference was:
Left Pupil (Mean = 28.57% (SD = 7.28%)) which was significantly higher than Gaze position X (Mean = 22.30% (SD = 6.18%))
Chi-sq(1,24) = 5.69 P = 0.0171

For the next event "Killing an enemy with a stomp" the result was Chi-sq (3,48) = 4.84 P = 0.1838 This does not show any significant difference between the models trained on the eye-tracking measurements' data in regards to precision.
By inspecting the pairwise K-WANOVA tests there were no Significant differences to be found. Moreover, the means were between 22.90% - 28.33% with the highest prediction coming from Right Pupil (SD = 7.03%) and the lowest from Gaze Position Y (SD = 6.66%)

For the last event "killing an enemy with a fireball" the result was Chi-sq (3,48) = 1.15 P = 0.7662 thereby, it did not any significant difference between the models trained on eye-tracking measurements' data in regards to precision.
Through inspection of the pairwise K-WANOVA tests no significant differences were found. Furthermore, there means differed between 21.66% - 26.66% where the highest were given by Gaze Position Y (SD = 12.30%) and the lowest by Left Pupil (SD = 8.45%).

The K-WANOVA tests performed to answer this hypothesis could only reject it at one out of four events. (Supportive box plots can be found in Appendix J.4)

7.3.5 H4: There would be no significant differences between the different game event types based on EEG trained model's accumulated accuracies

To answer this hypothesis accumulated accuracies from the model's trained on the different EEG channels. They are collected both for each event category and event type. Thereby not only considering the models eliciting the highest accuracies.

The result for the different types of events is F(3,828) = 134.47 P = 5.74278e-71 Thus showing that there was a significant difference in the accuracy between the different events based on all EEG data.

The paired ANOVAs shows the following pairs were responsible for the observed significant difference.
"Breaking an empty box" (Mean = 62.86% (SD = 3.52%)) showed significantly higher accuracies than "Colliding with a box containing a coin" (Mean = 59.27% (SD = 4.44%))
F(1,414) = 83.04 P = 3.49612e-18
"Breaking an empty box" showed significantly higher accuracies than "Killing an enemy with a stomp" (Mean = 59.55% (SD = 4.17%))
F(1,414) = 74.74 P = 1.22197e-16
"Breaking an empty box" showed significantly lower accuracies than "Killing an enemy with a fireball" (Mean = 65.97% (SD = 3.43%))
F(1,414) = 81.64 P = 6.51198e-18
"Colliding with a box containing a coin" showed significantly lower accuracies than "Killing an enemy with a fireball"
F(1,414) = 294.71 P = 2.8429e-50
"Killing an enemy with a stomp" showed significantly lower accuracies than "Killing an enemy with a fireball"
F(1,414) = 291.48 P = 7.34092e-50

Through a combination of the events into their respective categories the result was F(1,830) = 27.1 P = 2.43323e-7. Colliding with box events (Mean = 61.06% (SD = 4.39%)). Killing an enemy events (Mean = 62.76% (SD = 4.98%))

The ANOVA test on both categories and types of event showed significant difference thus rejecting the null hypothesis. (A supportive box plot can be found in Appendix J.5)

7.3.6 H5: There would be no significant differences between the different game event types based on EEG trained model's accumulated precisions.

To answer this hypothesis, the same measurements will be taken into consideration as the last hypothesis.

The result for different types of events was F(3,828) = 6.24 P = 0.0003 thus indicating that there was a significant difference in precision between the events based on EEG data trained models.

Between the events the following significant differences were present.
"Breaking an empty box" (Mean = 22.79% (SD = 7.26%)) had significantly lowerprecisions than "Colliding with a box containing a coin" (Mean = 24.44% (SD = 6.44%))
F(1,414) = 5.94 P = 0.0152
"Breaking an empty box" had significantly lower precisions than "Killing an enemy by a stomp" (Mean = 25.09% (SD = 5.89%))
F(1,414) = 12.51 P = 0.0005
"Colliding with a box containing a coin" had significantly higher precisions than "Killing an enemy by a fireball" (Mean = 22.68% (SD = 7.91%))
F(1,414) =6.14 P = 0.0136
"Killing an enemy by a stomp" had significantly higher precisions than "Killing an enemy by a fireball"
F(1,414) = 12.38 = 0.0005

By combining the events into their categories the result was F(1,830) = 0.31 P = 0.5783 Thereby it did not show any significant differences between the categories of event in relation to their precision based on EEG data. Their means were 23.61% (SD = 6.91%) and 23.88% (SD = 7.07%) respectively to "Colliding with box" events and "Killing an enemy" events.
This means that the null hypotheses cannot be rejected in respect to event categories, but it can in relation to types of events. (A supportive box plot can be found in appendix J.6)

7.3.7 H6: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated accuracies.

Like the previous hypothesis. This will be answered is by accumulation accuracies from models trained on eye-tracking data into the different events types and categories without taking into consideration which measurements showed the highest accuracies.

The result for the different event based on the accumulated Eye-tracking based accuracies was
F(3,204) = 28.81 P = 1.42233e-15

inspection of ANOVAs performed between pairs of the events shows the following significant differences.
"Breaking an empty box" (Mean = 59.31% (SD = 5.29%)) showed significantly lower accuracies than "Killing an enemy with a fireball"
(Mean = 66.84% (SD = 4.37%))
F(1,102) = 61.17 P = 4.97367e-12
"Colliding with a box containing a coin" (Mean = 59.52% (SD = 4.64%)) showed significantly lower accuracies than "Killing an enemy with a fireball"

$F(1,102) = 67.1$ $P = 7.85356e\text{-}13$
"Killing an enemy with a stomp" (Mean = 59.59% (SD = 5.23%)) showed significantly lower accuracies than "Killing an enemy with a fireball"
$F(1,102) = 57.67$ $P = 1.53226e\text{-}11$

Through accessing the categories of event the ANOVA test shows the following
$F(1,206) = 24.23$ $P = 1.75281e\text{-}06$
Where the "Colliding with box" events and "killing an enemy" events had the following means and Standard deviations respectively (Mean = 59.42% (SD = 4.98%)) (Mean = 63.21% (SD = 6.60%))

With these ANOVA tests the null hypotheses can be rejected. (A supportive box plot can be found in appendix J.7)

7.3.8 H7: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated precisions

This hypothesis will be approach the same way as the previous an accumulation of the precision gathered through the different eye-tracking measurements in relation to the different types of events.
The result between all events was $F(3,204) = 0.91$ $P = 0.4377$ thereby indicating that there is no significant different between all the event's accumulated precisions based on Eye-tracking.

ANOVA based on pairs of the event did not show any significant differences either. The means were 23.50% (SD = 6.27), 25.30% (SD = 7.03%), 25.86% (SD = 6.38%), and 24.60% (SD = 10.20%) respectively to the four events "Breaking an empty box", "Colliding with a box containing a coin", "Killing an enemy with a stomp", and "Killing an enemy with a fireball".
By combining the events into their categories the result is $F(1,206) = 0.6$ $P = 0.4391$ Which neither showed any significant difference. The mean for "collision with box" events were 24.40% (SD = 6.77%) and for the "killing enemy events" it was 25.23% (SD = 8.53%) in respect to the precisions.
This concludes that the null hypothesis cannot be rejected. (A supportive box plot can be found in Appendix J.8)

7.4 Summary of means and hypotheses conclusions

Through the last pages, different findings have been presented. In this subsection they are summarized to make them easier to compare. First a table with all the highest accuracies and predictions per event (Table 8). Then a table with the grand average of accuracy and prediction per event (Table 9). Afterwards a table with the grand averages per category of events both in relation to accuracy and precision (Table 10). Lastly, the conclusions for the hypotheses is shown (Table 11).

Event	EEG highest mean Accuracy	Eye-tracking highest mean Accuracy	EEG highest mean Precision	Eye-tracking highest means Precision
Breaking Empty Box	F4 = 64.64% (SD = 1.79)	Left Pupil = 60.99% (SD = 5.27%)	F4 = 26.54% (SD = 8.04%)	*Left Pupil = 25.38% (SD = 5.67%)*
Colliding with a box containing a coin	C3 = 61.41%(SD = 2.82%)	Right Pupil = 60.51% (SD = 4.15%)	C4 = 26.55% (SD = 6.50%)	*Left Pupil = 28.57% (SD = 7.28%)*
Killing an Enemy with a stomp	Fz = 61.89% (SD = 2.78%)	Right Pupil = 62.62% (SD = 4.54%)	*Oz = 29.12% (SD = 5.13%)*	*Right Pupil = 28.33% (SD = 7.03%)*
Killing an Enemy with a fireball	C4 = 67.30% (SD = 2.17%)	Right Pupil = 69.48% (SD = 2.43%)	*P8 = 25.50% (SD = 8.16)*	*Gaze Position Y = 26.66% (SD = 12.30%)*

Table 8 shows the highest averaged accuracies and predictions based on data from various EEG electrodes and eye-tracking measurements

Event	Averaged accuracy from EEG	Average Accuracy from Eye-tracking	Average Prediction from EEG	Average Prediction from Eye-tracking
Breaking Empty Box	62.86% (SD = 3.52%)	59.31% (SD = 5.29%)	22.79% (SD = 7.26%)	23.50% (SD = 6.27%)
Colliding with a box containing a coin	59.27% (SD =4.44%)	59.52% (SD = 4.64%)	24.44% (SD = 6.44%)	25.30% (SD = 7.03%)
Killing an Enemy with a stomp	59.55% (SD = 4.17%)	59.59% (SD = 5.23%)	25.09% (SD = 5.89%)	25.86% (SD = 6.38%)
Killing an Enemy with a fireball	65.97% (SD = 3.43%)	66.84% (SD = 4.37%)	22.68% (SD = 7.91%)	24.60% (SD = 10.20%)

Table 9 shows the grand averaged accuracies and precisions accumulated from all EEG electrodes and from all Eye-tracking measurements in relation to the different types of events.

Event Categories	Averaged accuracy from EEG	Average Accuracy from Eye-tracking	Average Prediction from EEG	Average Prediction from Eye-tracking
Colliding with box category	61.06% (SD = 4.39%)	59.42% (SD = 4.98%)	23.61% (SD = 6.91%)	24.40% (SD = 6.72%)
Killing an enemy category	62.76% (SD = 4.98%)	63.21% (SD = 6.03%)	23.88% (SD = 7.07%)	25.23% (SD = 8.53%)

Table 10 shoes the grand averaged accuracies and precisions accumulated from all EEG electrodes and Eye-tracking measurements in relation to the different event categories

Hypotheses	Conclusion
H0: There would be no significant differences in the accuracy between the models trained on the different EEG channels at different event types	**Rejected in four out of four events**
H1: There would be no significant differences in the precision between the models trained on different EEG channels at different event types	**Rejected in three out of four events** not in relation to "killed enemy with a fireball" event
H2: There would be no significant differences in the accuracy between the models trained on different eye-tracking measurements at different event types	**Rejected in two of the four events** not in relation to the collision with box category events
H3: There would be no significant differences in the precision between the models trained on different eye-tracking measurements at different event types	**Rejected in one out of four events** only in relation to the colliding with a box containing a coin event
H4: There would be no significant differences between the different game event types based on EEG trained model's accumulated accuracies.	**Rejected both in relation to event categories and event types**
H5: There would be no significant differences between the different game event types based on EEG trained model's accumulated precisions.	**Rejected only in relation to event types, but not in relation to event categories**
H6: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated accuracies.	**Rejected both in relation to event categories and event types**
H7: There would be no significant differences between the different game event types based on Eye-tracking trained model's accumulated precisions.	**not rejected at either event types or categories.**

Table 11 shows all the hypotheses and their conclusions whether they are rejected or not.

7.5 Result discussion

This discussion will elaborate on the results from two different perspectives. Firstly, they will be discussed in relation to them self and compared to each other. Secondly, will the results be compared to other studies training ANN with EEG data, thus giving an indication towards how prominent the findings are.

Through the hypotheses it is evident that the precision is more similar across data streams compared to accuracy as the fewest rejections were done in relation to precision. On top of that, across EEG electrodes also showed more diverse results than across eye-tracking measurements. It could be hypothesized that choosing eye-tracking measurement does not matter as much as choosing an electrode in an EEG system in relation to recognizing different game events. Moreover, an argument against that is clearly seen through where the highest average accuracies and precisions have been found. The majority of them have been observed through pupillometry compared to gaze position. That is rather surprising. In the Investigation (see section 2.4) the opposite relationship would have been expected due to the visited literature. It mostly spoke about correlations between cognitive load and pupillopmetry, and correlations between attention and gaze position (see section 2.4). Moreover, it was argued that stimulus was not so well attended at the moment of interaction, as it has already been stored in the subject's game schema, thus the attention might be visiting other stimuli that will be interacted with afterwards.
In relation to the highest averaged accuracies and precisions from the EEG. A quite interesting pattern is shown as three of them are from the frontal lobe and three others are from the central cortex. Those

areas are for instance about motor control [120] and temporal planning of goal-directed sequence of actions [121]. Seeing that the user has just executed an action at which a finger needed to move, the recognized activation in the central cortex could make sense. In relation to the temporal planning of events, combined with the theory about game schema [22] correlations could be possible. The player develops a new set of sub goals toward the final goal as a reaction upon the event. Thereby, can this theory about making a game schema during the game be a possible theory to why a higher accuracy and precision is found in the frontal lobe.
Accumulating all accuracies and predictions towards the different type of event and event categories projected some rather interesting results. The accuracies from both the EEG and Eye-tracking shows clearly that the accuracy of the "Killing enemy with a fireball" event is higher than the rest. This could give an idea about its ERPs being slightly more alike each other than the rest of the events. This is further defended by the fact that all training and test datasets contained the exact same amount of events, thus given them all an equal chance of being recognized across participants.
In regards to the accumulated precisions the event "killing an enemy with a stomp" proves to have the highest average. Although, its accuracies are not the highest. This shows that the none event ERPs towards that event are the smallest tendency to be misclassified as positives. Thus, indicating that the algorithm potentially could learn how exactly that event looks like both in relation to EEG and Eye-tracking measurements.

The results have been discussed in relation to itself and what it could mean. Towards other EEG trained ANN studies the performance of the algorithm will be compared. From this thesis the highest averaged accuracies will be used to compare, as all the papers mentions their benchmark tests through the use of accuracies.

Study:	This thesis	[81]	[87]	[86]	[85]
Accuracy:	61.41% - 67.30%	75% - 87%	69% - 85%	93%-97.3%	86%-89%

Table 12 shows the highest accuracies gathered from the studies where an ANN have been used to recognize EEG

In Table 12 There is a clear difference between the accuracies found through this thesis and the accuracies found in some of the literature. There can easily be many reasons why the observed accuracy is not matching the literature. First, the data. The data used in the literature are either coming from controlled p300 evoking stimuli or narrow working memory tests. Thus very isolated replicable nearly context less stimuli. Where this study works with ERPs from a rather complex context where more stimuli are represented at the same time. Thus, influencing the signal of the ERP and obscuring the evoke potential related to the very event. Second, the structure and the ending strategy of the ANN. Even though initial tests were done to maximize the accuracy there are certain faults in those which can influence their results negatively for example the training data. The data they were trained with were not the same as the final results were based on. Their initial training data were not homogenously divided between the events in either the training dataset or the test dataset. Thereby, could the amount of iterations before stopping the training very easily be another in order to reach the highest accuracy. In relation to the chosen structure for the network, further optimization of the topology could be done. In the initial test only two different topologies were tested towards the direction of the topology. Although, they could be representable of simple and complex topologies, they do not take into account the thousands other layer constructions that could be created. Moreover, a test of every combination will end in a combinatory explosion and thus take forever to find the most optimal configuration if no structure limit is set.
The construction of the ANN model might not be the most optimal, and thus not creating as prominent

results as found in the literature. But, the data is unified, artifacts are reduced to a minimum and the data is projected through the strongest components found through the ICA. Thereby, reducing possible bias in the data as much as possible, thus deeming the results gathered through this thesis to be evidential and reproducible.

8 Discussion

In this discussion different parts of the thesis will be covered. It will cast a critical light upon the different aspects of thesis from the Investigation to the Results. Through the different discussions, the validity of the thesis and the usefulness of the results will be covered. After the critical review of the thesis future research perspectives will be given towards both creating more replicable and solid results and investigate further into the dataset obtained through the experiment. Furthermore, a future real-time feedback scenario incorporating the ANN will be briefly devised as a concept.

8.1 Would other games apply

Since the beginning it was settled to work with the Super Mario game as it was convenient, and easy to access the source code. Moreover, through the analyzed results and the chosen categories, could other games work for this experiment and possibly give the same or similar results? possibly yes, the only thing it requires for those games is they should have the same semantics as Super Mario and provide the player with the same goals and sub goals thus a similar internal game schema. Such games which could work are most if not all sidescroller or platform games that incorporates running from the beginning of a level to the end, avoiding obstacles, killing enemies and collecting items, which either gives you point or makes you stronger. Furthermore, the elements need to be repeated throughout the game, without giving the sense of repetitiveness. Such game could very easily be Crash Bandicoot [122], Tarzan [123], or Rayman [124]. All of these are platform games with several repeating elements such as: the same collectables, containers with surprises, enemies, obstacles, few ways of killing the enemies and a level based structure with the goal of reaching the end of each level without dying. Surely the visual aspects of these games differs from Super Mario, but if the goals and the semantics are the same does the aesthetics matters in relation to the internal processes of the player? And what can potentially be captured through EEG and Pupillometry? However, as the semantics used in Super Mario are clearly found in context of different other games it can be theorized that similar results could be captured with other test beds.

8.2 Was the ANN the best choice for the machine learning algorithm?

The ANN with resilient backpropagation has been available since 1993[91]. Within computer science, researchers thrive to figure out new and novel ways to make algorithm smarter and quicker for instance the alphaGo [125] where ANN, Monte Carlo tree-search and reinforcement learning are combined to play Go better than even the human master in Go. If it was possible to incorporate such advancements into the ANN, developed through this thesis then there would certainly have been a higher recognition rate. Moreover, a more realistic advancement or replacement of the implemented ANN could be one of the following EEG trained algorithms: Convolutional Neural Network [126]–[128], Support Vector Machine [129] or Fisher's Discriminant analysis [130]. Through their results, which are both based on ERP and continuous EEG they achieved the following accuracies:

Study	This thesis	[128]	[126]	[127]	[129]	[130]
Accuracy	61.41% - 67.30%	49.95% - 53.47% difference flickering frequencies of a box	67.39% - 79.19% P300 speller	55.83% - 95.51% different network constructions to recognize Steady state visual evoke potential	49.38% - 90.35% different amount of emotions and channels	Mean between 60-70% Max 82.1%-97.4% for different emotion dimensions of emotion recognition.

Table 13 shows the highest accuracies form different studies utlizing a machine learning algorithm or a classifier to recognize EEG signals.

Through this table it can be found that the accuracies in this thesis can compete with the lower averages and mean accuracies in other studies using different classifiers. Moreover, a classifier solution which seems promising for adoption is a version of convolutional neural network found in [127]. The advancement Cecotti [127] has done within optimizing the network to work with EEG ERPs seems promising due to the high accuracies. Thus a change to such algorithm could potentially give higher accuracies, albeit the different natures of the ERPs.

8.3 Methodology problem – conducting the test as a game experiment vs an EEG experiment

When the experiment was conducted the subjects were asked to play Mario as they would usually do. For user experience game studies that would be the most optimal as the subject falls into their own gamer mindsets and follow the goals of the game in order to succeed. Moreover, for an EEG study, where the controllability is the key. This method does not apply. The amount of events each subject would be going through is variable and the exact context is not predictable. A way to make the study more controlled could be through asking the subject to interact with every possible event present in the level. This would make the amount of interactions more unified and the dataset more like each other from the beginning before artifact reduction. Even though this instruction could have helped the controllability of the study it could as well wreck a havoc in the EEG signals. The goal of the gamer in the game would be changed by this instruction. Instead of playing Mario as it should, with the main goal of reaching the end without dying, the main goal would instantly be changed to interact with every single object. This will change the internal game schema [22] of the user and all interactions would not have the same effect on the user as with the usual goal. The effect would be decreased to yet another interaction and the EEG signal might very well be influenced by that. Through such instruction the speculations about how the signal could end up looking like at each interaction will never be properly confirmed or rejected (see section 2.3.3) as those are based on the subject plays the game as it is supposed to.
Thereby, the following can be extracted. Even though the experiment does not have the same controllability as other EEG experiments it follows the free space of a user experience game experiment, which in the end provides the dataset with the potential most reliable ERPs.

8.4 Methodology problem – creating different ANN models for different events vs creating one model for them all.

For recognizing the different events found in the game a single ANN was developed to deal with all events at once. As seen in the results it proves to be quite flexible at learning how the different events looked like to different extends. Moreover, in the following papers they have dealt with different amount of variables for recognition and observed different accuracies, depending on the amount of tasks the network were

trained to recognize [84], [129] both of them shows higher accuracies when they are only trained to recognize two different tasks compared to more. [129] drops from 90.35% with a 32 channel configuration when recognizing two different emotions to 69.53 % when it has to recognize eight different emotions. In [84] their accuracy dropped from approximately 96% – 99%, when using a 17 EEG channel setup to recognize two different conditions in relation to cognitive load, to 71%-91% when trying to recognize seven different degrees of cognitive load. Thereby it is quite evident that reducing the amount of variable the ANN should learn to recognize the more accurately will it learn how they look. In relation to this thesis a future work could be to investigate how different events in pairs can push the accuracy upwards and thus find the maximum accuracy for the different events.
Moreover, as the network is modelled right now gives a better idea of how it will perform in a real-time context, as there might rarely be only two cases that should recognized, unless it is binary decision making problem. Such problem could be BCI where the subject has to turn left or right by imagining left or right hand movement. In that case a network dealing with two kinds of recognized groups of signals is optimal.

8.5 Methodology problem – Would a unique network for the eye-tracking enhance their accuracies

In the initial test (see Appendix F) optimization of the accuracy was only in respect to the EEG data. Since the Eye-tracking data is different than the EEG data it is very likely that another network than the current one would produce even higher accuracies for the eye-tracking data. It is worth investigating as for instance in these papers [3], [114] the researchers had different features to which they tried to predict Frustration, Engagement, fun and Challenge. They used different topologies of the ANN with different kinds of features in order to optimize the accuracy. Their strategy for during so is pretty straight forward. Frist they define a threshold which have been a network with three hidden layers with 10 neurons in each. Their networks then start evolve from one hidden layer with two neurons up till the threshold. On the way a max accuracy has been achieved, thus estimating the optimal topology for the neural network. This strategy could very easily be acquired in order to figure out the optimal topology for a network recognizing eye-tracking signals and another unique topology for the EEG.

8.6 Can the results tell us something about the ERPs?

Considering the nature of the ANN it tries its best to determine what an event looks like based on the training set. Through the weights, which can be seen as simple scalars for amplifying or nullifying different part of the epoch. The model starts to learn how the event looks like and finds subtle re-occurring components in the epochs. This could imply, if a high accuracy is achieved it is very possible that there is some part of the epoch which contains re appearing potentials. Moreover, if it is possible to achieve accuracies which are like the studies recognizing P300 then it is highly likely that there are significant potentials in the EEG epochs. Furthermore, looking at the maximum achieved accuracies in relation to the "killing an enemy with a fireball" event single accuracies from different subjects shows 69.5% - 76.5%. This suggest that there are some recognizable and similar between the potentials found in the epochs. Moreover, it does not tell anything about where the potential is and how it looks like. (In Appendix I is a link to a folder, which contains an excel sheet with all the acquired accuracies and precisions.

8.7 Are the chosen events for recognition those with the most recognizable ERPs – theoretical discussion

Referring back to the investigation chapter where it was discussed which kinds of ERPs that potentially could be available at different events (see section 2.3.3) it can be discussed that dealing with other events could produce more recognizable ERPs. For instance, evolving and devolving. When one of either is happens the time stops shortly and full attention is called to Mario through a flashing animation. As

discussed evolving could produce P1 and P3, while devolving could produce N2 and P2. This action can change the player's internal game schema as well. One of their sub goals could just have been completed, if evolving happened, or a new sub goal could be added if devolving happened.
Those events that has been recognized, which potentially could have elicited less recognizable ERPs, are "braking an empty box" and "colliding with a box to get a coin". Those are less significant for the progress of the game and thus to the internal system of the player. Even though they are described as potential elicitors of N1, P2, P3, Delta, Gamma and Theta waves, they could easily be obscured by other more important processes. This could possibly lead to considering whether or not the classified epochs solely come from the events. Thereby, not a combinations of different events at once, where the ones calling for most attention is the key promoter of brain potentials. An argument against this is the observed accuracies, as the top average is 64.64% and 61.41% for the different box colliding events. Moreover, an event of the same category which has been excluded from the recognition process despite it were elicited enough times is "colliding with a box containing a flower". The importance of this to the player's game schema and sub goals compared to breaking an empty box is significantly higher, thus a potential candidate for eliciting the discussed components.
Through this brief discussion it can be concluded that it might not be strongest events chosen in relation to eliciting the most recognizable EEG patterns and other events could theoretically elicit clearer ERPs. But the undeniable fact that they are recognized to some extend gives hope for those events which have been deemed to elicit weak brain potentials.

9 Future perspective

In the result discussion and the discussion different changes have been elaborated in order to discover more prominent results towards the final problem statement. Moreover, those are short sighted improvements, which can help the results from this thesis to reach publication potentials. Moreover, in this section long sighted topics in relation to this thesis will be proposed. They will both cover where the development of the ANN can go and looking at the data from different other perspectives.

9.1 Analysis of the epochs for ERPs in relation to the recognized events.

There has been a lot hypothesizing in relation to how the ERPs for the different events could look like (see section 2.3.3). This makes it increasingly interesting to look at the data from the individual events to see whether or not the conjecturing will stand. If so, it will open up for EEG ERP research in more complex scenarios than the controlled simple scenarios seen this far. Moreover, if the observed epochs do not contain any significant components. Then it could suggest that many competing stimuli will cancel each other's ERPs out, or even make them less noticeable. Furthermore, through observing high accuracies from the ANN it could be suggested that significant components are present in the ERP. This will especially hold true if the accuracies match those in the papers, which tries to recognize P300 components [81], [86], [87], [126]. In those they have first established that there are significant components present before they try to train the algorithm to detect them.
On the other hand, the procedure the data from this thesis will approach are more like these studies [83], [84], [129] They have trained their algorithms before they have investigated how the EEG looked like in its pure form. Albeit, they have used different feature selection models to determine which features should be fed into the system in order to maximize the accuracy compared to this thesis. Furthermore, these studies do not look at ERPs, but continuous EEG, thus having other analysis focusses when it comes to the look of the data.
In relation to the data from this project. If the following hypothesis is true "there are clearer brain

potentials in the data from the events with highest accuracies compared to those with low accuracies" Then the ANN could be used as a ERP significant quality detection algorithm. Thus could be used to scale down the visual search area for exploratory ERP research.

9.2 The potential use of the Game footage

In the method chapter (see section 4.2.2) it was mentioned that the subjects' gameplay was recorded. For future prospects they could be reviewed in respect to the events fed into the ANN for recognition. The point of it is to determine if there are uniformity between the events occurring right before the recognized events and the events occurring right after the recognized event. By figuring out how similar the pre events interactions are and how similar the post event interactions are. The processes that have influenced the EEG Epochs and the Eye-tracking behavior will be clearer. This will help to determine more exact what the signals are representing, as they are influenced by previous interactions and idea of what is happening just after. Through this analysis the timing of pre and post event interaction should be noted as well to be able to hypothesize which part of the signal are caused by pre event interaction and which are preparing for the post event interactions. In respect to usual EEG studies inter stimulus pauses are implemented to control the measured ERPs[27], [45], [46], [50]. Thereby, through analysis of the gameplay footage a potential framework for game EEG ERP studies can be derived as to ensuring that the Epoch the researcher is looking at is primarily caused by the event in question and not a variation of other events.

Another area the gameplay footage can help is to see where the user had their gaze attention. Moreover, with the Eye-tracking and the video stream in two separate files an application to fuse the two stream has to be devised. In other more advanced eye-tracking systems like the Tobii eye-tracker a software package including this feature is available [131]. Moreover, for the cheaper alternative like the Eye-tribe which have been used in this experiment such feature is not integrated. Creating such program would not only give this research further validation of the eye-tracking and raise speculations towards its impact on the EEG, but will give other students interested in viewing the Eye-tracking data upon a video sequence for post analysis a great opportunity to see the conjunction without having to focus on the integration, if it is not their target problem.

9.3 The data for the secondary study

Through the design (see section 5.2) it was decided to incorporate elements which can be used to determine the consequence of incorporating unexpected events into the game which elicits the opposite effect compared to the usual event. To analyze the effect a 4-AFC questionnaire were devised. The results from this questionnaire will consist of 5 answers per participant, thus 31x5 = 155 answered questionnaires. Through this it will be determined if there is an answer tendency toward either of the game modes in respect to the different questions. Furthermore, the amount of unexpected events will be fed into an ANN as a feature together with other features inspired from these studies [2], [3], [75] to see if the ANN can be trained with those to predict the questionnaire answers. Thereby, eliciting the possibiblity ofproposing new correlations.

9.4 Test the ANN on clear p300 components

To consider this ANN as a research tool for future EEG projects it is important to observe how it performs when recognizing clear p300 components either elicited through the P300 speller[47] or a previous surprise – suspense experiment done in the augmented cognition lab [48]. If the algorithm manages to meet the benchmarks of the other P300 detection studies, it will be worth to work further with it.

Forecasted the algorithm should work as expected in relation to the P300 or even better, then there is an

opportunity to try to integrate it in an interactive feedback loop with games and media.
This will open up research on affective games and media based on the overall model devised in a previous project [8] (see Figure 32)

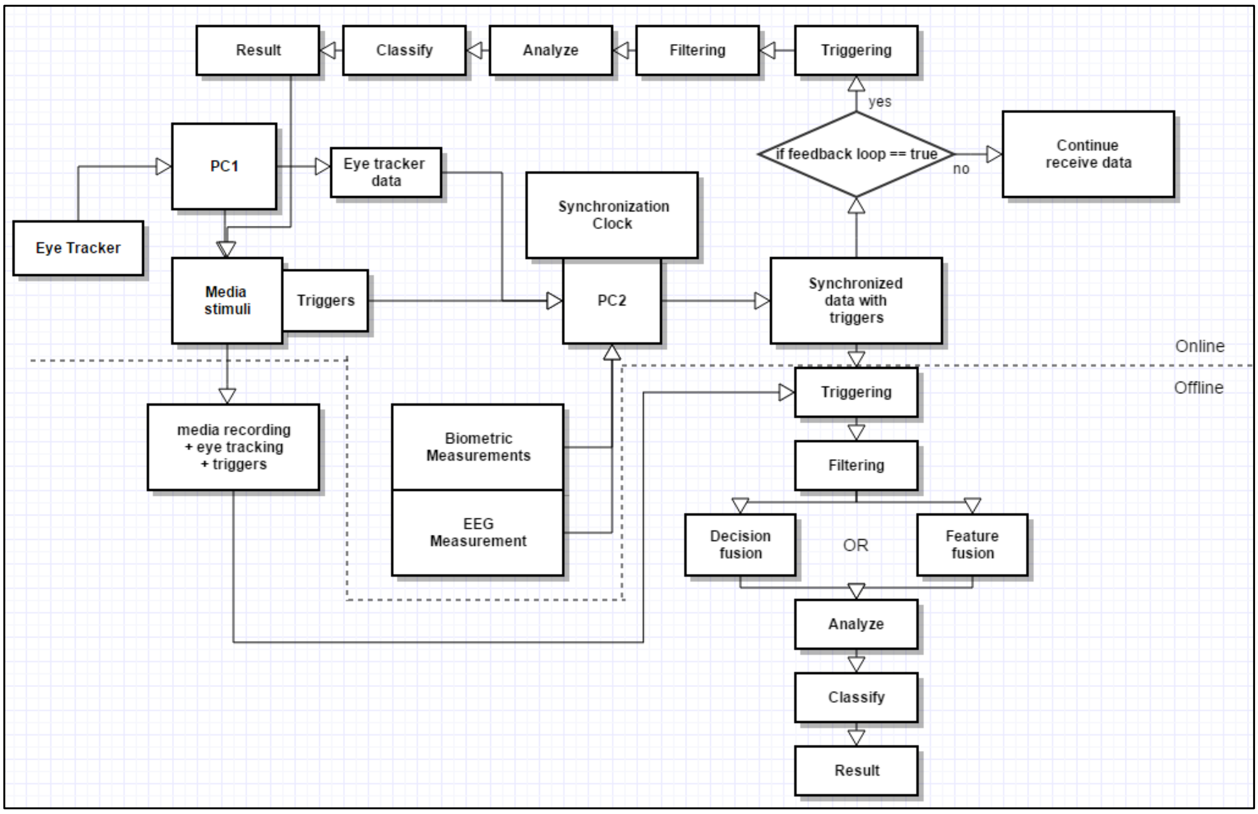

Figure 32 shows the Multimodal Sensor Triangulation Protocol (MMSTP) which has been devised on my 9th semester.

An addition to the model in Figure 32, which have been covered in this thesis, is integration of feature extraction before classification as it helps promote the accuracy through removal of unnecessary features. In relation to this real time integration of the ANN many questions have to be answered in order to adjust many parameters to make the system run as robust as possible and with as little delay as possible. Such questions are how much training data should it be fed with per subject? Should the ANN weights from different sessions be saved and reused in the next session? Is it feasible to make the program connect with the live data stream found through the Matlab patch? Shall it also be developed to work with different frequency bands, and thus use the power of those to predict the event or a higher order cognitive reactions? And many more questions are possibly worth to ask in order to move towards augmented cognition in games.

10 Conclusion

In this report, the game Super Mario have been investigated in order to analyze its perceivable component, thus finding the building blocks for any interactive and reactive event occurring in the game. A dissecting analytical approach have been utilized to diverge components from Electroencephalogram and eye-tracking data streams. A hypothetical convergence of the psychophysiological blocks with the game blocks have been conducted in order to visualize different categories and types of game events which could potentially have identifiable and recognizable by their psychophysiological patterns. The last puzzle piece that has been examined is the Artificial Neural Network. How it has been used to recognize different psychophysiological data features in relation to a diverse amount of stimuli, and how its core mechanisms work. Through a junction of the three elements the Final Problem Statement have been devised:

Is it possible to recognize psychophysiological patterns, measured in relation to specific videogame perceptual-events, through an Artificial Neural Network?

To answer this problem a Super Mario game have been modified to fit the experiment, by running 10 levels continuously. In real time while playing the game event triggers and eye-tracking have been communicated via UDP to a signal recording computer. From this computer EEG, eye-tracking and Event triggers have been saved.
in an offline manner the EEG signal was filtered through a bandpass filter form 0.1 – 100 Hz and a notch filter at 50 Hz. Both the EEG and Eye-tracking signal were divided into 3 second epochs in relation to the event triggers. 1 second before stimulus and 2 second after stimulus. The signals were then individually send through an independent component analysis to sort out artifacts and to enhance significant components. All of the eye-tracking data which counts 768 data points were fed to the ANN and 256 data points from the EEG representing the first second after stimuli were given to the algorithm.
The structure of the algorithm was as follows; input layer 768 or 256 neurons. Three hidden layers with respectively 32, 16 and 8 neurons. Lastly an output layer with 4 neurons one for each classifiable game event (Breaking an empty box (BE), Colliding with box containing a coin (CC), Killing and enemy with a stomp (KS) and Killing an enemy with a fireball (KF)). Within each layer, apart from the output layer were hyperbolic tangent functions implemented. In the output layer was a sigmoid function implemented. The algorithm was trained on 56 events, 14 of each through resilient backpropagation and 10-fold cross validation. After the algorithm have been through 700 iterations it was tested on 24 events 6 from each group.
Through 13 usable out of 31 subjects the results show significant difference between models trained on EEG channels in relation to accuracy in respect to the different events (BE (61.02% - 64.64%) CC (57.91% - 61.21%) KS (57.69% - 61.89%) KF (64.87% - 67.30%)). Between models trained on eye-tracking measurements the same could be found in two of the four events (BE (57.78% - 60.99%) CC (58.62% - 60.51%) KS (56.69% - 62.62%) KF (64.67% - 69.48%)). Accumulated accuracies between the events given by models trained on EEG signals showed significant difference as well (59.27% - 65.97%). This pattern was repeated as well through accumulated Eye-tracking accuracies (59.31% - 66.84%). These results are rather weak compared to other similar studies, but with further fine tuning of the parameters in the ANN stronger accuracies can be found. **To answer the final problem statement then the report can conclude: Yes,** it is possible to recognize psychophysiological patterns, measured in relation to specific videogame perceptual-events, through an Artificial Neural Network.

11 Reference

[1] http://www.marioai.org/LevelGeneration/source-code. [Accessed: 14-Mar-2016].

[2] -based corpus for analysing the *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6975 LNCS, no. PART 2, pp. 536-547, 2011.

[3] modeling user experience in g *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1519-1531, 2013.

[4] N. Shaker, M. Shaker, and M. Abou *Proceedings, Elev. Aaai Conf. Artif. Intell. Interact. Digit. Entertain.*, pp. 191 197, 2015.

[5] - pp. 147 161, 2011.

[6] *2010 IEEE World Congr. Comput. Intell. WCCI 2010 - 2010 IEEE Congr. Evol. Comput. CEC 2010*, 2010.

[7] - *AI Mag.*, vol. 34, no. 3, pp. 89-92, 2013.

[8] A. Wulff- synchronization paradigm for user sensing-

[9] S. Baceviciute, L. E. Bruni, P. Burelli, and A. Wulff- . 1 12, 2015.

[10] *Nature*, vol. 381, no. 6582, pp. 520-2, 1996.

[11] 447, 1998.

[12] *Nat. Neurosci.*, vol. 6, no. 2, pp. 119-126, 2003.

[13] *Psychon. Bull. Rev.*, vol. 12, no. 6, pp. 1127-33, 2005.

[14] *J. Exp. Psychol. Hum. Learn. Mem.*, vol. 6, no. 2, pp. 174 215, 1980.

[15] a Kohlrausch and - 44, 1999.

[16] *Nature*, vol. 408, no. December, 2000.

[17] of auditory temporal- *J. Neurosci.*, vol. 17, no. 10, pp. 3956-3963, 1997.

[18] -visual integration during multimodal object recognition in *J Cogn Neurosci*, vol. 11, no. 5, pp. 473-490,

1999.

[19] S. Chu, S. Narayanan, and C. cy *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 17, no. 6, pp. 1142-1158, 2009.

[20]

[21] *Digit. Mediat. Hum. Commun.*, pp. 1 32, 2004.

[22] *Int. J. Comput. Games Technol.*, vol. 2008, pp. 1-7, 2008.

[23] nt- *Cogn. Neurosci. Neuropsychol.*, vol. 00, no. 00, pp. 1-4, 2006.

[24] valence effects *Int. J. Psychophysiol.*, vol. 72, no. 3, pp. 299-306, Jun. 2009.

[25] A. S. Makeig, M. Westerfield, T. Jung, S. Enghoff, J. Townsend, E. Courchesne, and T. J. Sejnowski, 295, no. 5555, pp. 690-694, 2002.

[26] P. Sauseng, W. Klimesch, W. Stadler, M. Schabus, M. Doppelmayr, S. Hanslmayr, W. R. Gruber, and *Eur. J. Neurosci.*, vol. 22, no. 11, pp. 2917-2926, 2005.

[27] *Psychophysiology*, vol. 37, no. 2, pp. 190-203, 2000.

[28] *Dev. Neuropsychol.*, vol. 27, no. 2, pp. 183-215, Apr. 2005.

[29] *Cogn. Brain Res.*, vol. 25, no. 1, pp. 107-116, 2005.

[30] -Directed and Stimulus-Driven Attention in the *Nat. Rev. Neurosci.*, vol. 3, no. 3, pp. 215-229, 2002.

[31] onse to sound: a *Psychophysiology*, vol. 24, no. 4, pp. 375-425, 1987.

[32] face, word, and object *Neuroimage*, vol. 20, no. 3, pp. 1609-1624, 2003.

[33] G. a Rousselet, M.-J-M. Macé, and M. Fabre- *J. Vis.*, vol. 4, no. 1, pp. 13-21, 2004.

[34] *Behav. brain Funct.*, vol. 1, p. 13, 2005.

[35] V. P. Clark and S. A. Hil *J. Cogn. Neurosci.*, vol. 8, no. 5, pp. 387-402, Sep. 1996.

[36] *Proceedings of the 31st Annual Conference of the Cognitive Science Society*, 2009, no. January, pp. 383 388.

[37] and catecholamine *Psychiatry Res.*, vol. 44, no. 1, pp. 21 32, 1992.

[38] L. Carretié, M. Martín studied through event- *J. Cogn. Neurosci.*, vol. 13, no. 8, pp. 1109 1128, 2001.

[39] -related qualitative differences in auditory cortical responses during short- *Clin. Neurophysiol.*, vol. 111, no. 12, pp. 2234 44, 2000.

[40] C. Pernet, S. Bas *Cogn. Brain Res.*, vol. 17, no. 2, pp. 327 338, 2003.

[41] studied through event- *Int. J. Psychophysiol.*, vol. 41, no. 1, pp. 75 85, 2001.

[42] *2011 18th IEEE Int. Conf. Image Process.*, pp. 3109 3112, 2011.

[43] *Comput. Graph. Appl. IEEE*, vol. 34, no. 6, pp. 46 56, 2014.

[44] on the N2 component *Psychophysiology*, vol. 45, no. 1, pp. 152 170, 2008.

[45] - *J. Clin. Neurophysiol.*, vol. 9, no. 4, pp. 456 479, 1992.

[46] C. C. Duncan, R. J. Barry, J. F. Connolly, C. Fischer, P. T. Michie, R. Näätänen, J. Polich, I. Reinvang, -related potentials in clinical research: guidelines for eliciting, recording, *Clin. Neurophysiol.*, vol. 120, no. 11, pp. 1883 908, Nov. 2009.

[47] - *J. Neural Eng.*, vol. 8, no. 2, p. 025022, Apr. 2011.

[48] - 175, 2014.

[49] - *Trends Cogn. Sci.*, vol. 1, no. 6, pp. 203 209, 1997.

[50] P. Burkhardt *Cogn. Neuropsychol.*, vol. 18, no. 17, pp. 51 54, 2007.

[51] *Event-related potentials: A methods handbook*, 2005, p. 229.

[52] O. Jensen, E. Spaak, and J. M. Zumer, *Human brain oscillations: From physiological mechanisms to analysis and cognition*, vol. 9783642330. 2014.

[53] E. Basar, M. Schurmann, C. Basar n brain functioning: an

Int. J. Psychophysiol., vol. 26, pp. 529, 1997.

[54]
248, 2001.

[55] -band oscillations- *Curr. Opin. Neurobiol.*, vol. 20, no. 2, pp. 156165, 2010.

[56] -
event- *Neurosci. Lett.*, vol. 259, no. 3, pp. 165168, 1999.

[57] M. Stikic, R. R. Johnson, -based classification of positive and negative *Brain-Computer Interfaces*, vol. 1, no. 2, pp. 99112, 2014.

[58] M. Soleymani, S. Asghari-
sig *IEEE Conf. Multimed. Expo*, vol. 231287, no. 231287, pp. 18, 2013.

[59] *ACM Trans. Appl. Percept.*, vol. 4, no. 2, p. 11es, 2007.

[60] R. G. Reilly
Springer, vol. 1, M. Horsley, M. Eliot, B. A. Knight, and R. Reilly, Eds. Cham: Springer International Publishing, 2014, pp. 13155.

[61] B. Laeng, S. Siro *Perspect. Psychol. Sci.*, vol. 7, no. 1, pp. 1827, 2012.

[62] -evoked pupillary response with a remote eye
Proc. 2008 Symp. Eye Track. Res. Appl. ACM, vol. 1, no. 212, pp. 6972, 2008.

[63]
Int. J. Psychophysiol., vol. 52, no. 1, pp. 16, 2004.

[64] O. Palinko, A. L. Kun, A. Shyro
Symp. Eye-Tracking Res. Appl., pp. 141144, 2010.

[65] e
2011 5th Int. Conf. Pervasive Comput. Technol. Healthc. Work., pp. 7884, 2011.

[66]
Proc. Natl. Acad. Sci. U. S. A., vol. 79, no. 8, pp. 25542558, 1982.

[67] *Communications*, vol. 29, pp. 3144, 1996.

[68] D. E. Rumelhart and J. L. MacClelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Foundations. 1, Volume 1*, vol. 1, no. June. 1987.

[69]
Comput. Chem. Eng., vol. 12, no. 910, pp. 881 890, 1988.

[70] D. C. Park, M. a.-El
IEEE Trans. Power Syst., vol. 6, no. 2, pp. 442449, 1991.

[71] M. Gevrey, I. Dimopoulos, and S. L

Ecol. Modell., vol. 160, no. 3, pp. 249264, 2003.

[72] *User Model. User-adapt. Interact.*, pp. 155 183, 2015.

[73] *Int. Conf. Found. Digit. Games*, pp. 107 114, 2011.

[74] P. Burelli, G. Triantafyllidis -invasive player experience estimation from body *IEEE Conf. Comput. Intell. Games*, no. December, pp. 17, 2014.

[75] N. Shaker, M. Shaker, and M. Abou . 191 197, 2012.

[76] *Affect. Comput. Intell. Interact.*, pp. 267 276, 2011.

[77] : Combining Discrete Events and *Proc. 16th Int. Conf. Multimodal Interact.*, pp. 34 41, 2014.

[78] *User Model. User-Adapted Interact.*, vol. 18, no. 12, pp. 207 243, 2008.

[79] selection and classification of electroencephalogram signals: An artificial neural network and genetic algorithm- *Artif. Intell. Med.*, vol. 55, no. 2, pp. 117126, 2012.

[80] Principal Component Analysis with Neural Ne *2013 IEEE Int. Conf. Emerg. Trends Comput. Commun. Nanotechnol.*, no. Iceccn, pp. 227231, 2013.

[81] - 264, 1993.

[82] C. Robert, J. *Clin. Neurophysiol.*, vol. 113, no. 5, pp. 694701, 2002.

[83] -time assessment of mental workload using psychophysiological *Hum. Factors*, vol. 45, no. 4, pp. 635643, 2004.

[84] *Hum. Factors*, vol. 45, no. 3, pp. 381 389, 2003.

[85] metrics for within- and cross *Neuroimage*, vol. 59, no. 1, pp. 4856, 2012.

[86] A. Turni -P300 signals using adaptive neural *Int. J. Innov. Comput. Inf. Control*, vol. 8, no. 9, pp. 64296443, 2012.

[87] -Network Approa *Brain Cogn.*, vol. 27, no. 3, pp. 311330, 1995.

[88] R. Hecht *Proc. Int. Jt. Conf. Neural*

Networks, vol. 1, pp. 593605, 1989.

[89] Y. Hirose, K. Yamashita, and S. Hijiya, -propagation algorithm which varies the number of *Neural Networks*, vol. 4, no. 1, pp. 6166, 1991.

[90] --- a versatile artificial neural network *Comput. Phys. Commun.*, vol. 81, no. 12, pp. 185 220, 1994.

[91] *IEEE Int. Conf. Neural Networks - Conf. Proc.*, vol. 1993Janua, pp. 586591, 1993.

[92] I. A. Bas *J. Microbiol. Methods*, vol. 43, no. 1, pp. 331, 2000.

[93] https://theeyetribe.com/products/. [Accessed: 15-Nov-2015].

[94] - *Electroencephalogr. Clin. Neurophysiol.*, vol. 10, no. 2, pp. 371375, 1958.

[95] n, Processing and Analysis Products 2013 96, 2013.

[96] -It- *Proc. CESCG 2011 15th Cent. Eur. Semin. Comput. Graph.*, pp. 1 7, 2011.

[97] ccuracy and precision of fixation locations recorded with the low-cost Eye Tribe tracker in different experimental set 20, 2015.

[98] *Cogn. Psychol.*, vol. 10, no. 3, pp. 356 389, 1978.

[99] J. Lazar, J. H. Feng, and H. Hochheiser, *Research Methods in Human-Computer Interaction*. 2010.

[100] -Computer Interaction (3rd *Lect. Notes Comput. Sci. Incl. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinforma.*, vol. 5623 LNCS, p. 229,230, 2011.

[101] *Clin. Neurophysiol.*, vol. 116, no. 12, pp. 27192733, 2005.

[102] S. Koehler, P. Lauer, T. Schreppel, C. Jacob, M. Heine, A. Boreatti-Hümmer, A. J. Fallgatter, and M. J. *J. Neural Transm.*, vol. 116, no. 1, pp. 97104, 2009.

[103] T. A. Stroganova, G. Nygren, M. M. Tsetlin, I. N. Posikera, C. Gillberg, M. Elam, and E. V. Orekhova, *Clin. Neurophysiol.*, vol. 118, no. 8, pp. 1842 1854, 2007.

[104] *Neuropsychobiology*, vol. 45, no. 3, pp. 161166, 2002.

[105] *Alcohol*, vol. 37, no. 3, pp. 129133, 2006.

[106] -enabled affective human-n *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8513 LNCS, pp. 536547,

2014.

[107] er estimation and state *John Willey Sons, 2004. 434 p*, vol. 399, no. 6735, p. 423, 2004.

[108] *Neural Networks*, vol. 13, no. 45, pp. 411 430, 2000.

[109] S. Makeig, A. J. Bell., T. *Adv. Neural Inf. Process. Syst.*, vol. 8, pp. 145151, 1996.

[110]

[111] *2014 9th Iberian Conference on Information Systems and Technologies (CISTI)*, 2014, pp. 16.

[112] *Proc. 29th Annu. Meet. Cogn. Sci. Soc.*, vol. 1, pp. 905910, 2007.

[113] - *Biological Handbook of Statistics*. pp. 165 172, 2009.

[114] *Computer (Long. Beach. Calif).*, no. Hudlicka 2008, pp. 6368, 2009.

[115] *Mud*, p. 1, 1996.

[116] *Nintendo of America, Inc.*, 1988. [Online]. Available: https://web.archive.org/web/20141114102808/http://www.allgame.com/game.php?id=1322. [Accessed: 14-Sep-2014].

[117] https://docs.oracle.com/javase/7/docs/api/java/lang/Thread.html. [Accessed: 25-May-2016].

[118] *Oracle*, 2015. [Online]. Available: https://docs.oracle.com/javase/tutorial/uiswing/layout/layoutlist.html. [Accessed: 22-May-2016].

[119] *J. Mach. Learn. Res.*, vol. 16, pp. 12431247, 2015.

[120] W. *J. Educ. Psychol.*, vol. 82, no. 4, pp. 881884, 1990.

[121] *J. Neurocytol.*, vol. 31, no. 35, pp. 373 385, 2003.

[122] N. Do - bandicoot-3-warped/. [Accessed: 18-May-2016].

[123] - https://www.youtube.com/watch?v=GaI2U3Gvpl. [Accessed: 18-May-2016].

[124] [Accessed: 18-May-2016].

[125] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I.

Nature, vol. 529, no. 7587, pp. 484–489, 2016.

[126] brain- *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 433–445, 2011.

[127] -frequency convolutional neural network for the offline classification of steady *Pattern Recognit. Lett.*, vol. 32, no. 8, pp. 1145–1153, 2011.

[128] ral Network with embedded Fourier Transform for *2008 19th Int. Conf. Pattern Recognit.*, pp. 1 4, 2008.

[129] -enabled affective human- 1, pp. 536 547, 2014.

[130] - 17, 2006.

[131] http://www.tobiipro.com/product-listing/tobii-pro-studio/. [Accessed: 19-Nov-2015].

[132] *Oracle*, 2015. [Online]. Available: https://docs.oracle.com/javase/tutorial/uiswing/components/jcomponent.html.

[133] ://eegatlas-online.com/index.php/en/artifacts/movement-tongue. [Accessed: 25-May-2016].

[134] -Score and ROC: A family of *Adv. Artif. Intell.*, vol. 4304, pp. 1015–1021, 2006.

[135] - *Proc. 23rd Int. Conf. Mach. Learn. --* , pp. 233 240, 2006.

Appendix

A. Consent Form

Participant No.______

In this experiment you will be play 10 levels of a game.
During the experiment your eyes, electro brain activity and in game behavior will be monitored.
None of the measurements will cause you any harm.
The electro brain activity will be captured through the electroencephalogram helmet. Which needs to be placed on your head. To enhance the connectivity between the scalp and the electrodes a water based gel will be applied.
All the data will be anonymously stored and will only be used to train and test an artificial neural network classifier in respect to detecting different game events.

The procedure of the experiment will go as follows.

1. You will have to fill out a simple demographic questionnaire
2. The EEG helmet will be placed on you
3. The gel will be applied to every electrode
4. You will be asked to blink and clinch your teeth to test the connectivity
5. You will be asked to find a comfortable sit position, which you will remain in approximately 10-15 minutes
6. The eye-tracker will be calibrated
7. A screen recording program will be turned on to capture in game footage
8. To make a baseline for the EEG measurement a grey screen with a white cross in the middle will be shown to you for one minute.
9. The game application will be started and last for 10-15 minutes.
10. After the game you will get free from the helmet and you will get some refreshments.

You can at any point during the experiment stop and quit the experiment.

Please sign the contract if you agree to have your eyes, brain and game behavior monitored.

Name　　　　　　　　　　　　　　　　　　　　　　Date

______________________　　　　　　　　　　　　______________________

Signature

____

B. Demographic questionnaire

Participant No.______

Gender:
____

Age:
____

Occupation:
____

Level of education you currently possess:
____

Have you played Mario before? Yes___ No___

How often do you play video games? Daily___ Weekly___ Monthly____ Rarer___ Never___

Have you consumed Alcohol within the last 4 hours? Yes___ No___

Have you consumed any caffeine within the last 4 hours? Yes___ No___

C. Experimental procedure protocol

Before The Experiment

1. name the .mat file in the Simulink patch: MarioSubEEG01. MarioSubEEG02. MarioSubEEG03… etc. dependent on the number of participants we have reached.
2. Make sure the active folder in matlab is AndreasThesis, so the .mat file will be saved there
3. The iMac computer needs to be in resolution 1280 X 760
4. The volume on the iMac should be set to 0%
5. The baseline application, Mario game, EyeTribe Software and the recording software (for either gameplay or external behavior) should be easy accessible on the iMac's desktop.

During The Experiment

1. ask the participant to fill out the consent form
2. ask the participant to fill out the demographic questionnaire
3. Equip the user with thy EEG helmet
4. Apply gel into helmet
5. Run the matlab patch to check the conductivity
    a. Ask participant to blink and clinch their teeth
6. Stop the matlab patch
7. Make the user sit comfortably in front of the iMac while he can reach and interact with the keyboard.
8. Adjust the eye-tracker so it is 65 cm from the participant
9. calibrate the eye-tracker
10. ask the participant not to move as far as it is possible
11. run the matlab patch
12. turn on the recording software (for either gameplay or external behavior)
13. run the baseline application
14. after the baseline application is done. Run the Mario game
15. when the game is done stop the matlab patch
16. stop the recording software
17. un equip the subject and give him/her refreshments and a thanks

After The Experiment

1. Save the recording from the software as: MarioSubRec01, MarioSubRec02, MarioSubRec03… etc.
2. If there will come another participant clean the helmet with wet papers. Else, wash the cap.

D. Simple Artificial Neural Network application

In order to understand the basics and the core algorithms of ANN a simple program emulating such network was developed. It consisted of Neurons, Connections, activation functions, error rate calculations and backpropagation algorithm.

The core of the application is done within the Network class, which can be seen in Figure 33**.** In this class the network is assembled, all the neurons and output neurons are created. The Input neurons are given values and the output has been given ideal values, which the system will try to match by learning the relationship between the input values and those. All the connections between the different layers are created and random weights are assign to them. The structure is now ready to be trained. In this application the training will take 1000 iterations. Firstly, the feedforward mechanisms are done and the activation functions are activated. Secondly, the cost function within the output neurons will be activated, and an output error rate is given. With this error rate the backpropagation can start, first from the output layer to the hidden layer. With the partial derivatives and the values given by the first part of the backpropagation the second part is started between the hidden layer and the input layer. When it is done the program checks whether or not it has been running for 1000 iterations, if not I'll start the feedforward mechanism again. If it has it will quit the application (an activity diagram can be found at Figure 34).

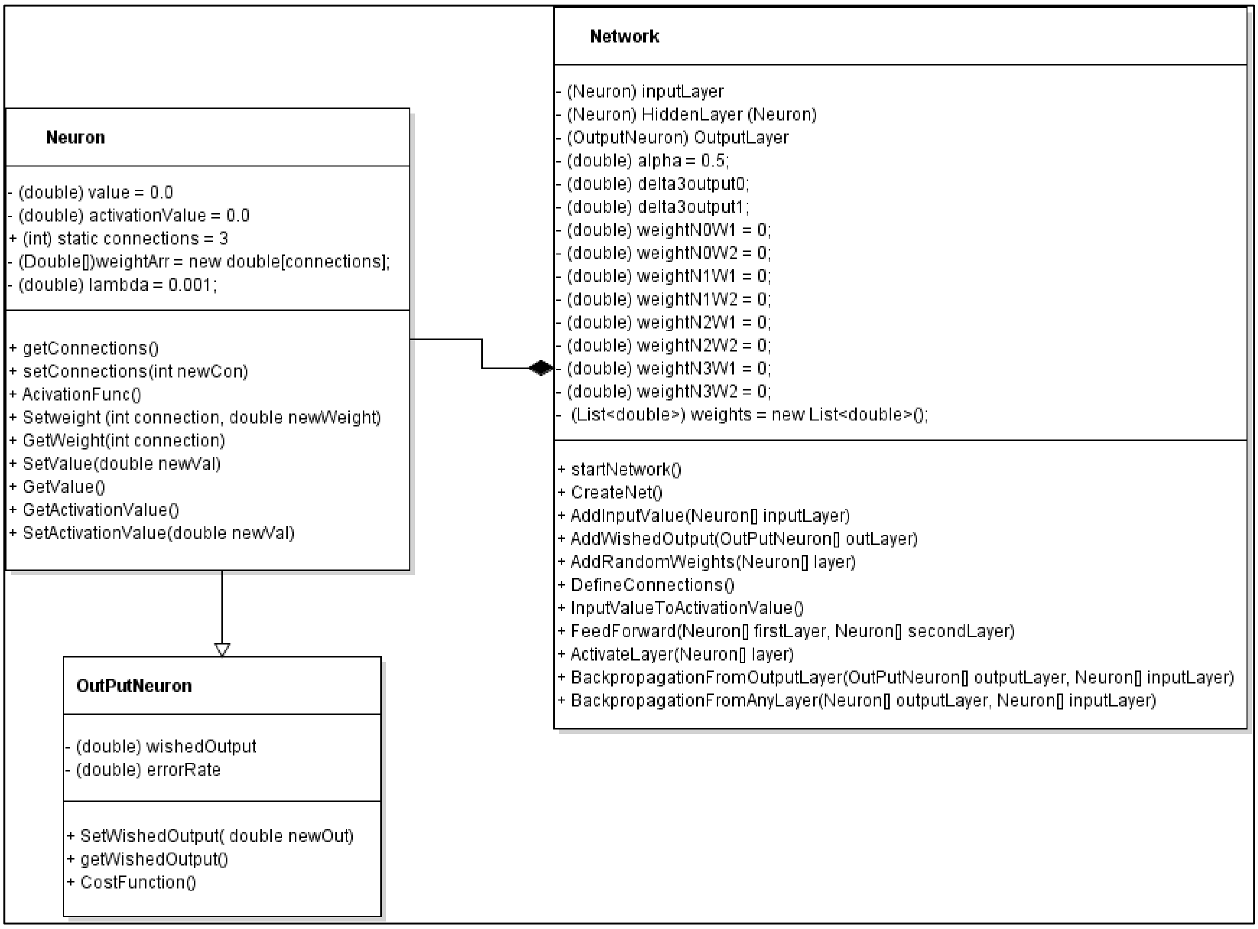

Figure 33 shows the class diagram for the simple Artificial Neural Netowrk

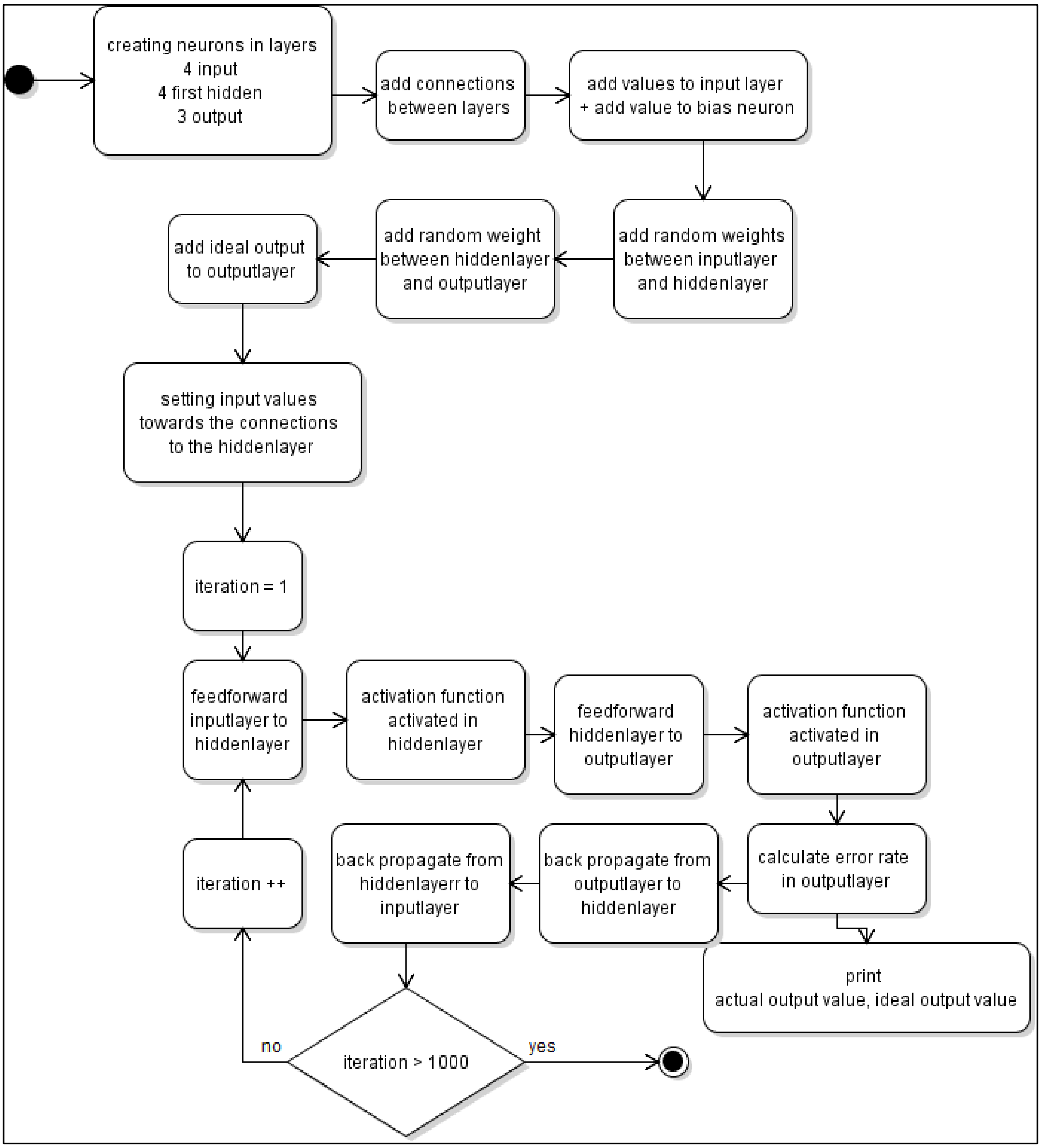

Figure 34 shows the activity diagram of the simple ANN. how it works and in which oreder the different calculatios are done.

E. Initial test 1. Decide the feature extraction model

The aim of this initial test is to see whether or not feature extraction of different kinds will enhance the accuracy. Five different conditions will be compared 1: no feature extraction, 2: all ICA features, 3: all PCA features, 4: 5 best ICA components, 5: 5 best PCA components.

The events they were tested on were four different box collision events: when a flower popped up, when an enemy popped up, when a coin popped up and when a mushroom popped up.

The average accuracies were gathered by one participant. One average for each of the EEG electrodes.

With the average accuracies ANOVA test were ran. The results are as follows.

Box collision with a flower: $F(4,75)= 1.58$ $P = 0{,}1882$

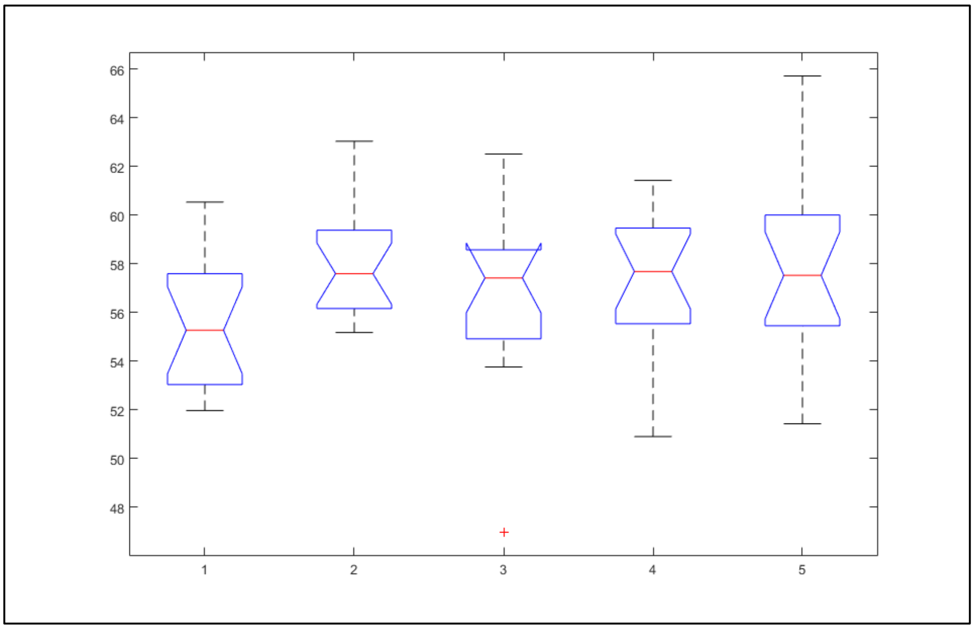

Figure 35 accuracies based on five different ways of feature selections. 1 = none, 2 = ICA, 3 = PCA, 4 = 5 strongest componenets of ICA, 5 =5 strongest components of PCA. All is in relation to the event "colliding with a box containing a flower" and they are based on data from the electrode placed at position F3.

The box collision with an enemy: F(4,75) = 2.25 P = 0.0711

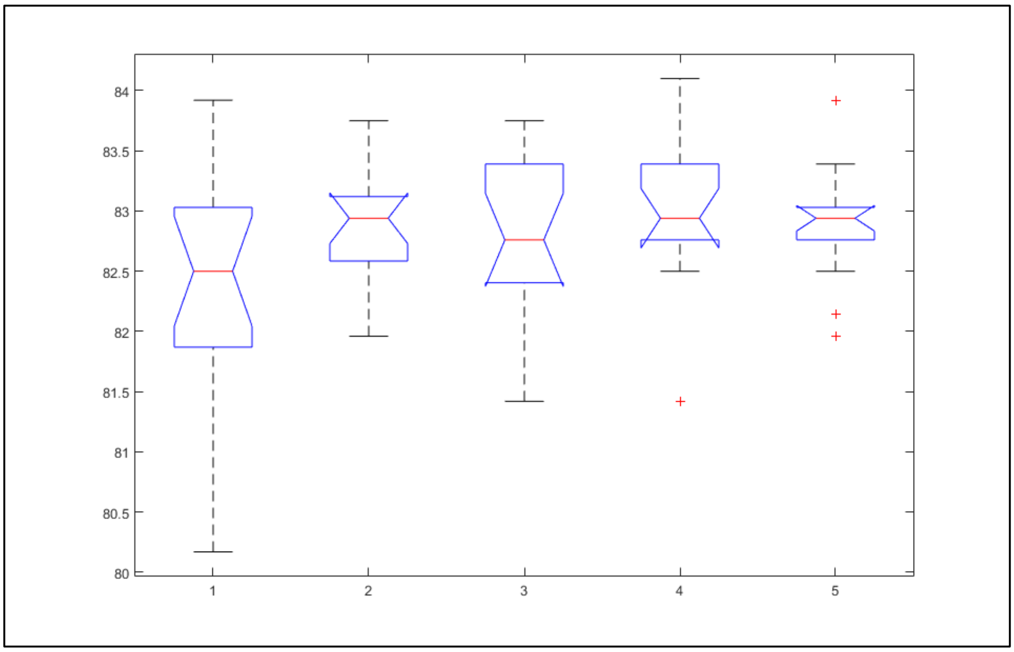

Figure 36 accuracies based on five different ways of feature selections. 1 = none, 2 = ICA, 3 = PCA, 4 = 5 strongest componenets of ICA, 5 =5 strongest components of PCA. All is in relation to the event "colliding with a box containing an enemy" and they are base

The box collision with a coin: F(4,74) = .87 P = 0.4886

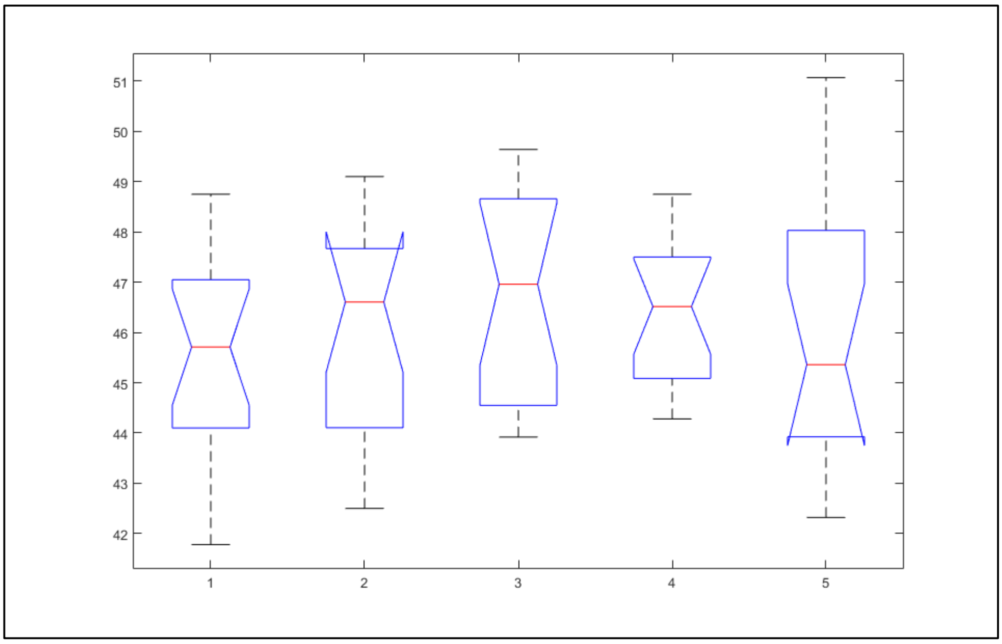

Figure 37 accuracies based on five different ways of feature selections. 1 = none, 2 = ICA, 3 = PCA, 4 = 5 strongest componenets of ICA, 5 =5 strongest components of PCA. All is in relation to the event "colliding with a box containing a coin" and they are base

The box Collision with a mushroom: F(4,74) = 0.87 P=0.4873

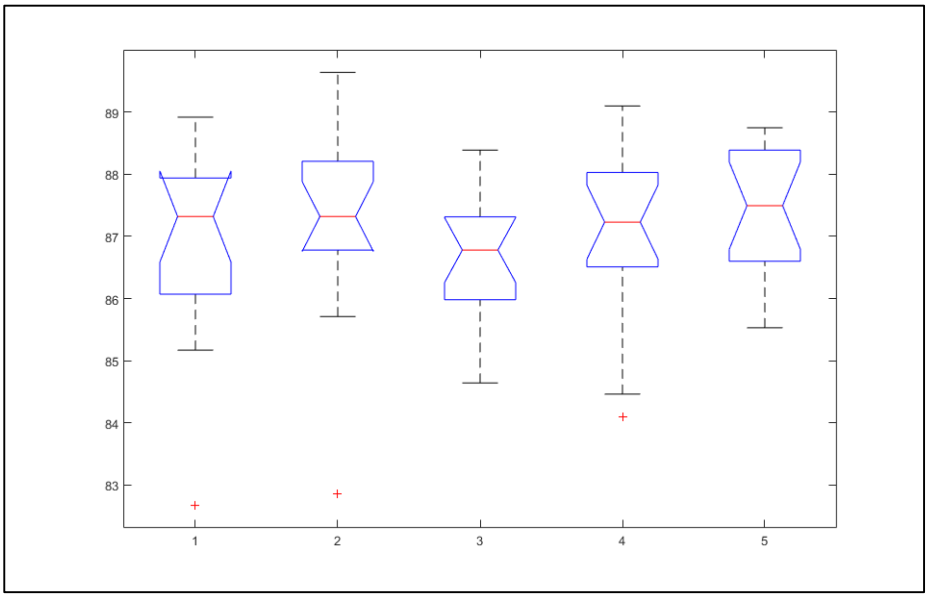

Figure 38 accuracies based on five different ways of feature selections. 1 = none, 2 = ICA, 3 = PCA, 4 = 5 strongest componenets of ICA, 5 =5 strongest components of PCA. All is in relation to the event "colliding with a box containing a mushroom" and they are base

Though the ANOVA tests no significant differences could be found between the feature selection models. Moreover, the difference between none and the feature extraction models in the Flower and Enemy event is noticeable. Furthermore, looking at the maximum values elicited per channel per event gave the following distributions.

None	PCA	ICA	ICA5	PCA5
8	15	15	17	14

Table 14 shows how many times the highest accuracy have occured with the different feature selection algorithms.

This tells us, that even though the difference are not big, the feature selection model with most highest values are ICA with 5 components. It is not with much, but it will be seen as preliminarily the best decision to go with.

F. Initial test 2. Decide ANN topology and amount of iterations

This test has the purpose of finding the best topology of the ANN and the most optimal amount of iterations in order to get the highest accuracies. The best topology was between two different structures. One simple structure containing 256 input neurons, 16 hidden neurons and 4 output neurons. Another complex structure: 256 input neurons, 32 in first hidden layer, 16 in second hidden layer, 8 in the third hidden layer, and lastly 4 in the output layer.

The structures were trained on one subject\s EEG data from the F3 electrode over the course of 2 days during that they followed the following paradigm. For every 100th iteration the accuracy was measured. The same number of 100th was reached over 10 training sessions in order to account for different weight initiations. Thus, the accuracies shown in the following figures are averages of 10 accuracies.

The amount of iterations reached by both of the models after they have ran 2 days were 4800 iterations. Thereby, giving 48 different averaged accuracies per structure.

Lets first look at the error rate. How that evolved during the 4800 iterations:

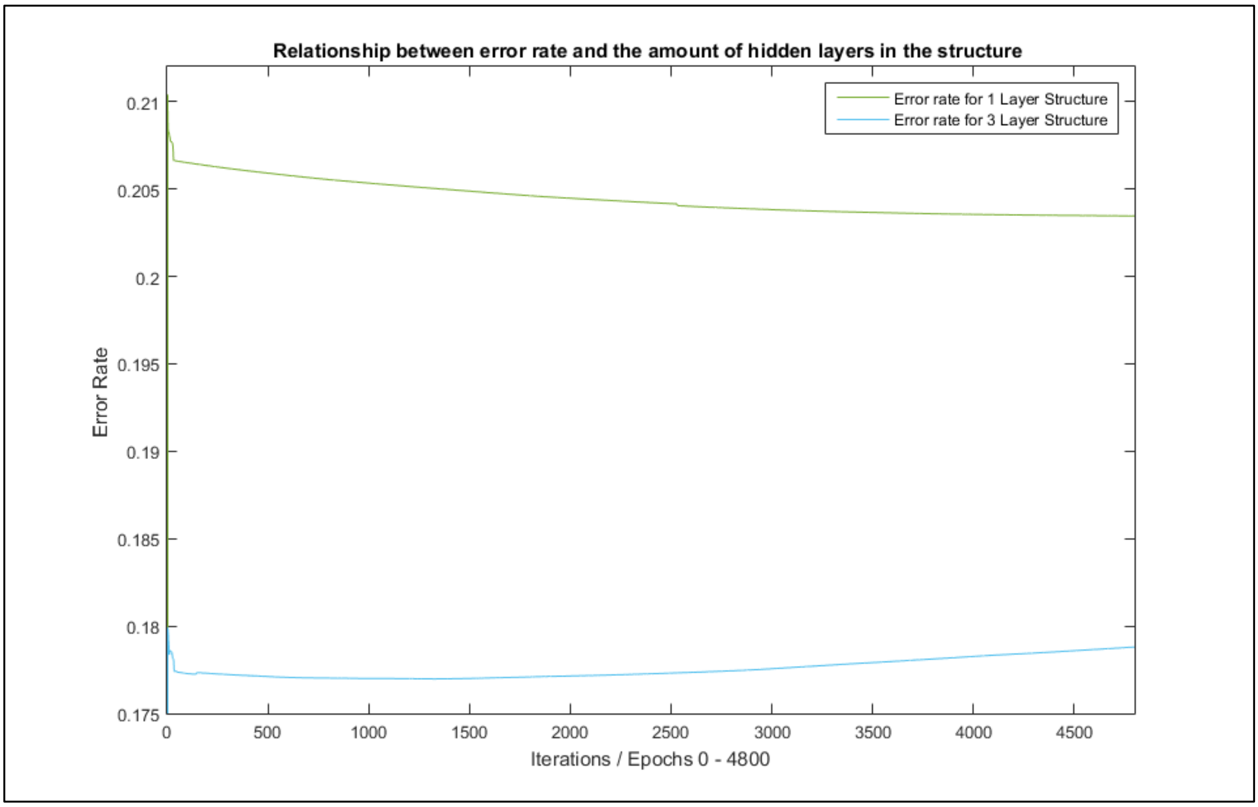

Figure 39 shows the error rate from the two models and how it evovls over the course of iterations

By Figure 39 it is clear that the complex model has lower error rate than the simple model. Moreover, the complex model reaches its minimum early at around 1300 iterations, where the simple model has not reached its, but it is converging in the end, so its minimum might be reached around 5000 iterations.

The error rates are in favor of the complex model. The same occurs when we look at the validation error. In Figure 40 it shows that the validation error is falling for both of the structures. Moreover, it seems to be steeper for the complex model and lower. Compared to the error rates the validation error are not converging, thus not showing the same tendencies.

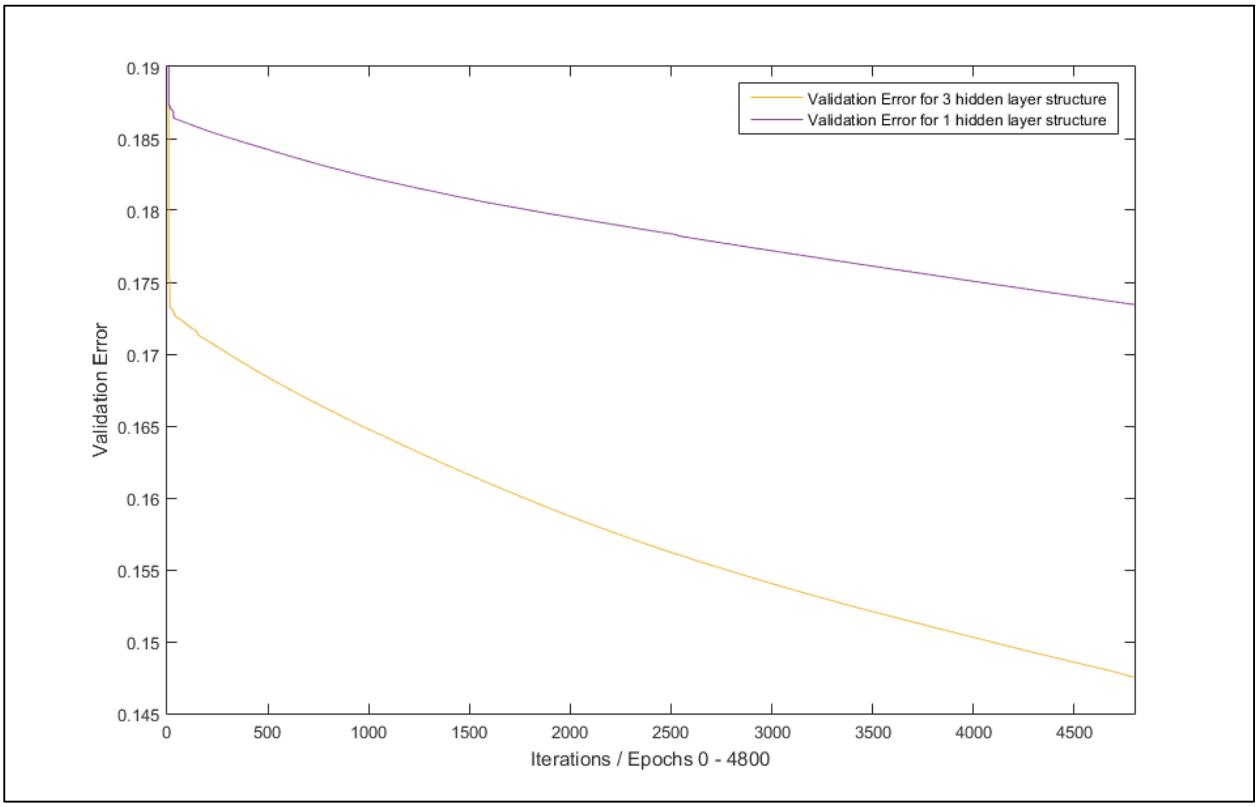

Figure 40 shows validation error from the two models and how it evovles over the course of iterations

As it was mentioned there were 4 output neurons. This gives four streams of accuracies, which are related to the following four events. breaking an empty box, hitting a box containing a coin, killing and enemy by stomp and killing an enemy by fireball.

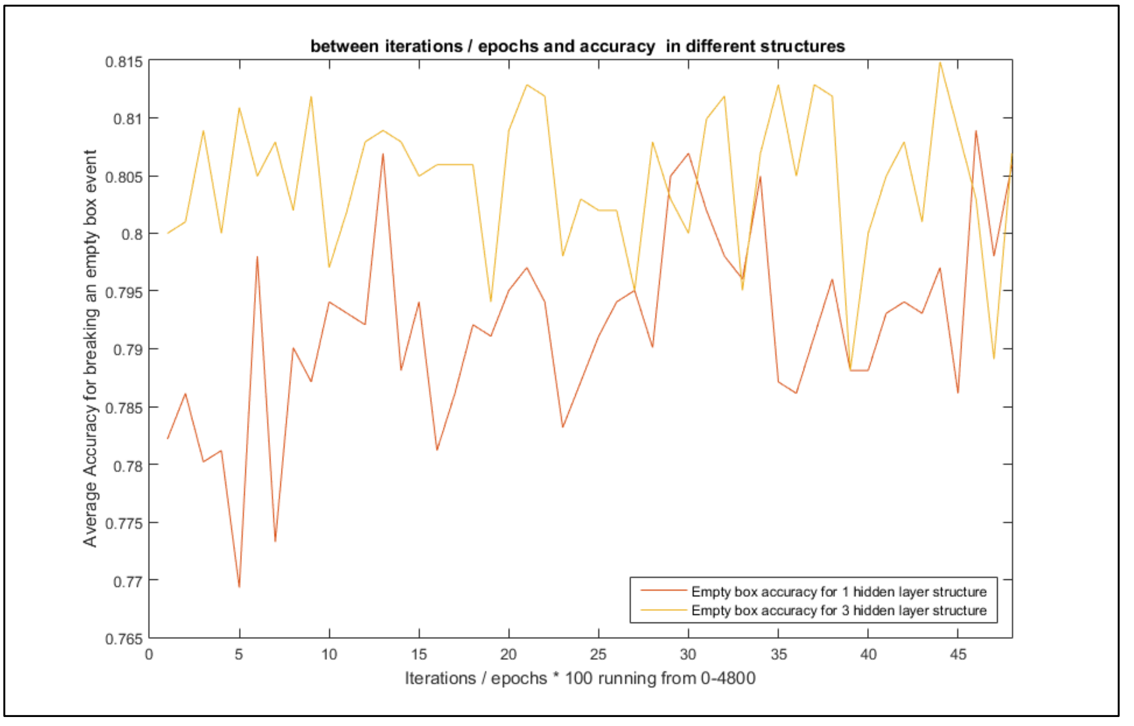

Figure 41 shows how the average accuracy evolves over the course of iterations in relation to the breaking an empty box event.

by the breaking box event it can be seen that the complex model before the 2500 iteration reaches higher accuracies than the simple model. However, after 2500 iterations they seems to be more alike. The simple model could be reaching higher accuracies than the complex model if it trained more.

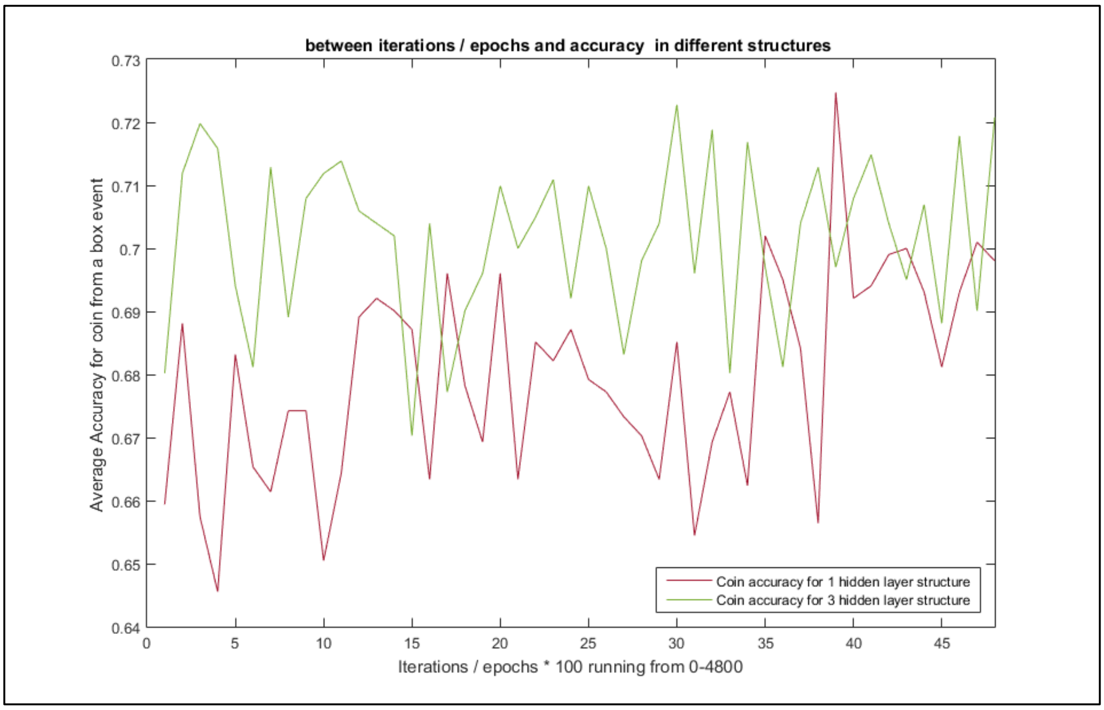

Figure 42 shows how the average accuracy evolves over the course of iterations in relation the colliding with a box containing a coin event.

The accuracies given by "hitting a box containing a coin" event shows the same tendencies as the "breaking a box" event. With the complex model generally shows higher accuracies and the simpler model seems to getting higher accuracies with more iterations.

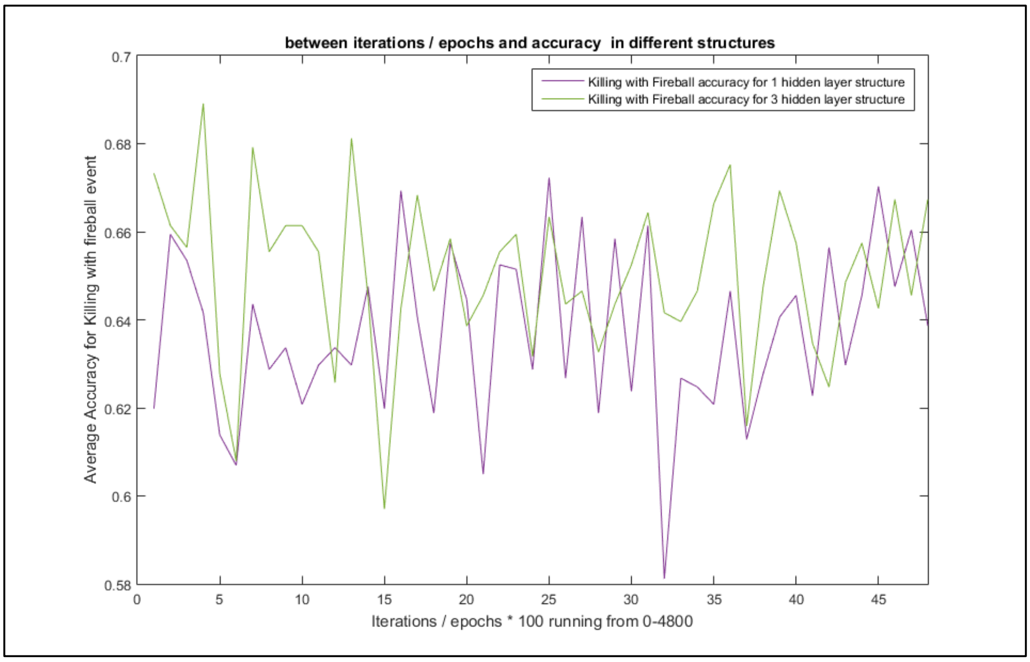

Figure 43 shows how the average accuracy evolves over the course of iterations in relation to the killing an enemy with a stomp event.

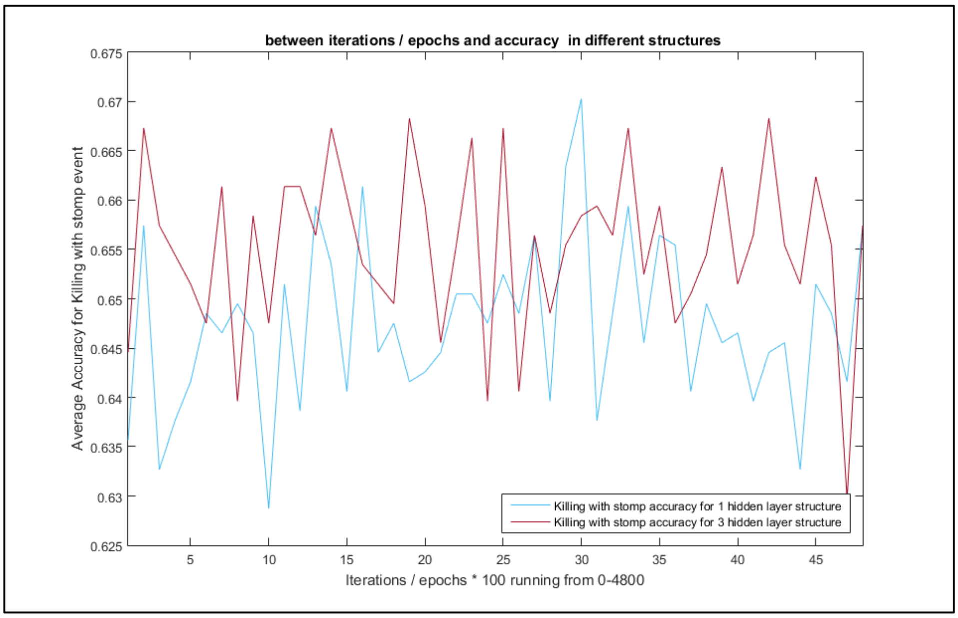

Figure 44 shows how the average accuracy evolves over the course of iterations in relation to the killing an enemy with a fireball event.

For the two other events types the accuracies seems to be more mixed together. However, the complex structure seems dominantly to have the highest accuracies. This dominance is even more certain if all the accuracies from all the events are averaged together, thus producing Figure 45.

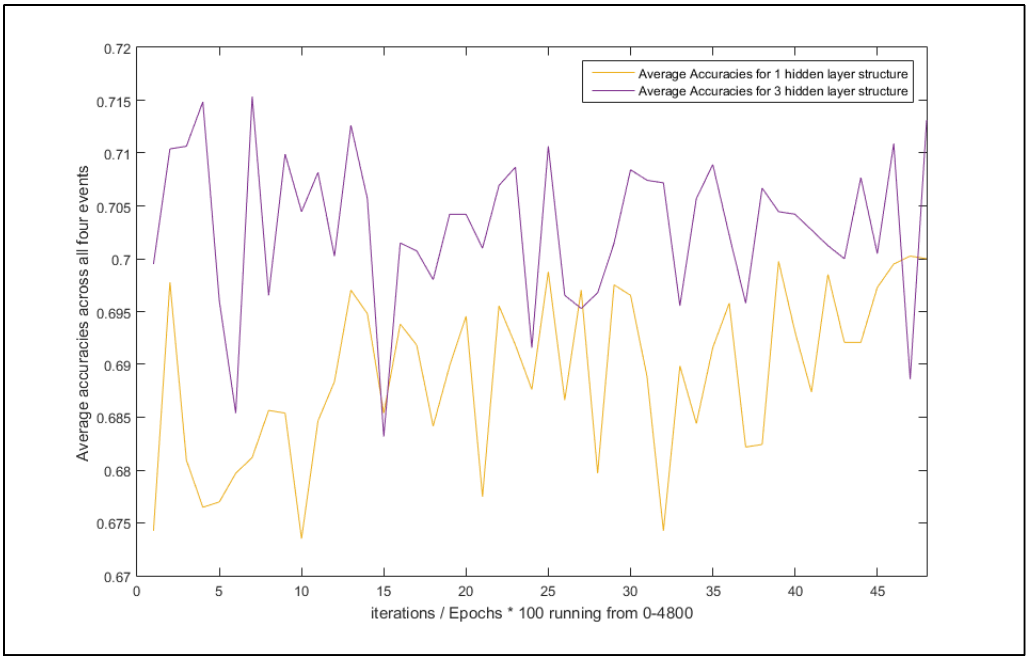

Figure 45 shows the grand average accuracies for the two models based on all the events.

Though Figure 45 it is stigmatized that the complex model is more favorable than the simple model. The last question needs to be answered is at which amount of iteration does the model perform best? Zooming in on Figure 45 gives the Figure 46 This shows the accumulated highest accuracy occurred at 700 iterations.

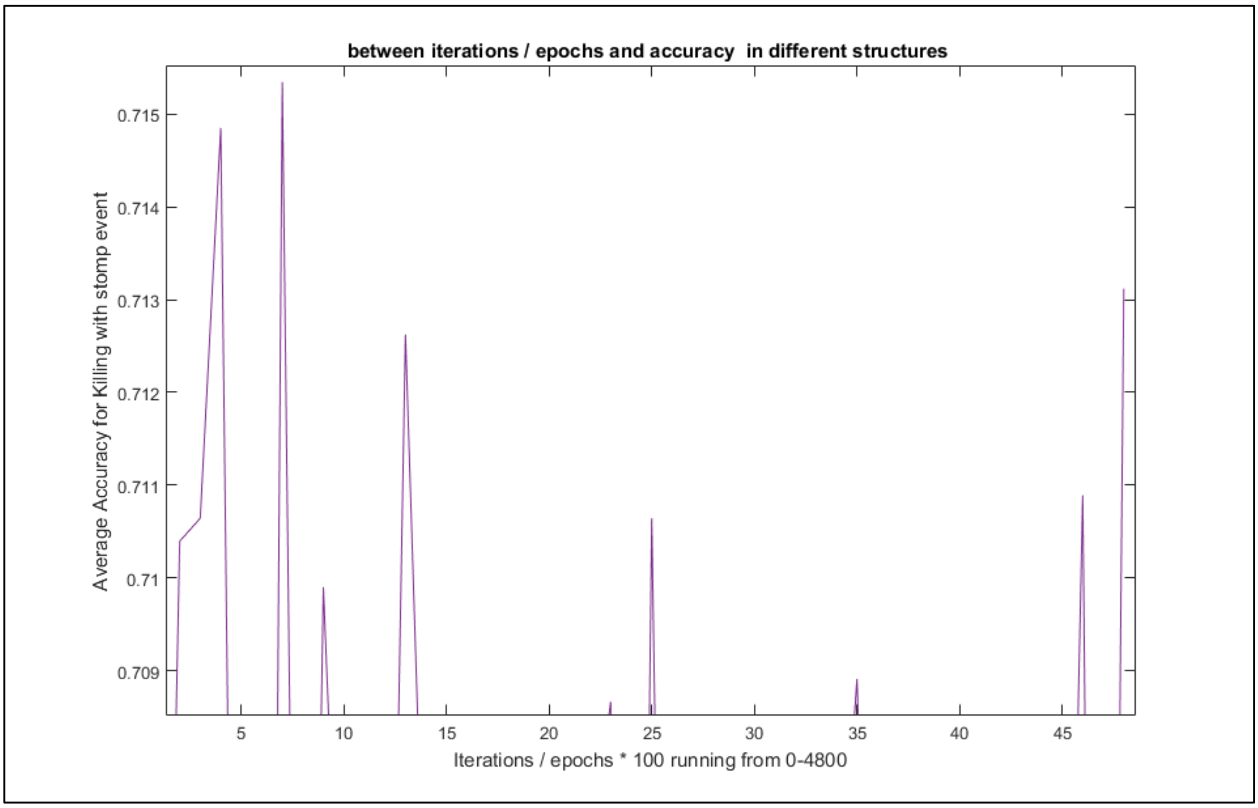

Figure 46 shows where the accumulated accuracies are highest for the complex model.

G. Calculation of Visual Angles

Kosslyn 1978 visual angle

correct cal is $\theta = 2\arctan\left(\frac{\left(\frac{e}{2}\right)}{d}\right)$

e = longest axis

d = distance

our setup	Kowalik 2011	Dorms 2015	eyetribe page Theory optimal
$d = 110$	$d = 70$	$d = 60$	$d = 75$
$e = 60$ 27 inch	$e = 46{,}5$ 21 inch	$e = 44$ 20 inch	$e = 53$ 24 inch
$\theta = 0{,}532504$	$\theta = 0{,}641357$	$\theta = 0{,}70289$	$\theta = 0{,}67928$

How close is the distance really to the others so it can be compared?

$$0{,}641357 = 2\arctan\left(\frac{\left(\frac{60}{2}\right)}{d}\right)$$

$$0{,}3206785 = \arctan\left(\frac{30}{d}\right)$$

$$\tan(0{,}3206785) = \frac{30}{d}$$

$$\frac{30}{\tan(0{,}3206785} = d = 90{,}32\text{ cm}$$

$$0{,}70289 = 2\arctan\left(\frac{\left(\frac{60}{2}\right)}{d}\right)$$

$$0{,}351445 = \arctan\left(\frac{30}{d}\right)$$

$$\frac{30}{\tan(0{,}351445)} = d = 87{,}81\text{ cm}$$

$$0{,}67928 = 2\arctan\left(\frac{\left(\frac{60}{2}\right)}{d}\right)$$

$$\frac{30}{\tan(0{,}33964)} = d = 84{,}90\text{ cm}$$

with 75cm distance and the θ from our setup how big is the screen relatively

$$0{,}532504 = 2\arctan\left(\frac{\left(\frac{e}{2}\right)}{75}\right)$$

$$\tan(0{,}266252) = \frac{\left(\frac{e}{2}\right)}{75}$$

$$75 \cdot (\tan(0{,}266252)) = \frac{e}{2}$$

$$2 \cdot 75 \cdot (\tan(0{,}266252)) = e = 40{,}9 = 18{,}5\text{ inc}$$

H. Amount of events different subjects have experienced

Subjects/Events	Jump	Jump failed in gap	auto jump at end	fade to new scene	death by falling into gap	death by enemy	devolve	evolve by enemy	bounce out of pit	box with an enemy	pick up a coin	pick up a mushroom	pick up a flower	evovle by item	breaking box	box with a coin	box with a mushroom	box with a flower	killing by stomp	killing by fireball	killing by shell	winning level	losing level
sub 01	55	5	5	5	3				3														2
sub 02	330	2	2	3	11	6	1		6	2	206	6	16	8	42	40	6	19	13	67	4	2	7
sub 03	404	1	1	1	4	7	6	3	1	23	378	7	11	12	13	62	8	43	35	105	16	6	2
sub 04	322	1	1	1	3	3	4	1	1	11	323	10	14	16	20	42	11	31	27	103	10	6	
sub 05	303	1		1	3	3	10	2	1	7	167	10	9	16	10	14	10	13	41	33	5	5	3
sub 06	451	2	2	2	5	10	12	3	5	2	307	11	11	19	34	30	13	21	51	52	24	6	3
sub 07	398	1	1	1	7	7	16	4	4	23	472	12	13	21	59	93	21	72	91	105	17	7	1
sub 08	290	1	1	1	4	4	10		2	8	192	7	9	15	13	32	10	11	38	7	9	4	1
sub 09	472	1	2	1	4	7	12	3	6	17	480	13	62	21	40	94	18	76	58	127	16	3	4
sub 10	338	4	4	4	2	4	11	1	1	7	275	8	12	17	20	34	10	25	35	58	10	6	2
sub 11	335	1	1	1	5	2	18	9	2	11	272	8	18	15	34	35	14	26	65		22	4	1
sub 12	302	3	2	2	1	1	14	6	2	22	391	13	11	19	35	77	14	53	50	121	22	5	1
sub 13	223	8	7	7	3	7	1	1	11	1	60					1	6		24		1		6
sub 14	306	2	2	2	1	4	14	9		26	351	5	11	15	25	38	6	26	36	138	4	5	1
sub 15	313		1		1	1	12	1	1	8	278	13	14	25	33	34	11	33	25	59	19	8	
sub 16	332				7	4	5	3	9	12	174	7	14	8	25	19	10	15	19	46	6	2	3
sub 17	300	2	3	3	6	4	14	4		2	178	11	3	13	41	48	9	30	52	15	3	1	3
sub 18	346				4	3	14	4	1	12	314	10	15	19	51	53	10	27	39	70	8	6	1
sub 19	126	2	2	2	2	1	3	3	6	23					2			1	2				2
sub 20	368	1	1	1	5	5	19	6	6	6	209	11	18	16	42	51	14	22	50	2	22	1	6
sub 21	286				2	4	14	6	2	8	332	9	7	16	22	31	10	13	69	52	16	7	1
sub 22	245	7	6	6	3	5	8		4	5	114	5	10	9	7	20	6	14	27	2	16	3	3
sub 23	324	1			2	3	6	2	1	21	350	7	10	14	72	60	8	40	53	73	10	5	
sub 24	114				5	2	4	2	4	1	33	1	3	3	4	2	1	1	9				2
sub 25	258	1	1	1	5	2	11	5	1	3	217	8	6	9	19	21	8	18	32	38	6	2	3
sub 26	202				2		6	2	2	5	222	7	14	14	29	42	6	21	22	55	7	5	1
sub 27	207	3	2	2	1	3	1		1	9	127	1	2	2	23	16	1	27	11	42	6		3
sub 28	367	2	2	2	1	3	9			17	403	9	21	23	85	81	12	71	42	110	5	7	1
sub 29	266	1	1		1		2			11	305	10	13	18	9	15	14	15	18	98	10	8	
sub 30	182		1			4	8	3		7	231	7	8	15	11	21	9	8	21	76	4	3	1
sub 31	261	7	8	8	3	5	8	3	15	7	160	6	7	11	18	29	6	7	24	37	11	2	5
																				20<			
																				30<			

I. Link to Implementations, presentation video, Accuracies, Precisions, raw data and unpublished reports.

https://www.dropbox.com/sh/gbhvdrclt4844zw/AABNRmzvWjfcI25R2MNMODjga?dl=0
This link leads to a Dropbox folder containing all results, implementations, raw data, this thesis, thesis presentation video, and unpublished work such as my 9 semester report, as it has been referenced throughout this thesis.

J. Box plots for results

J.1. H0

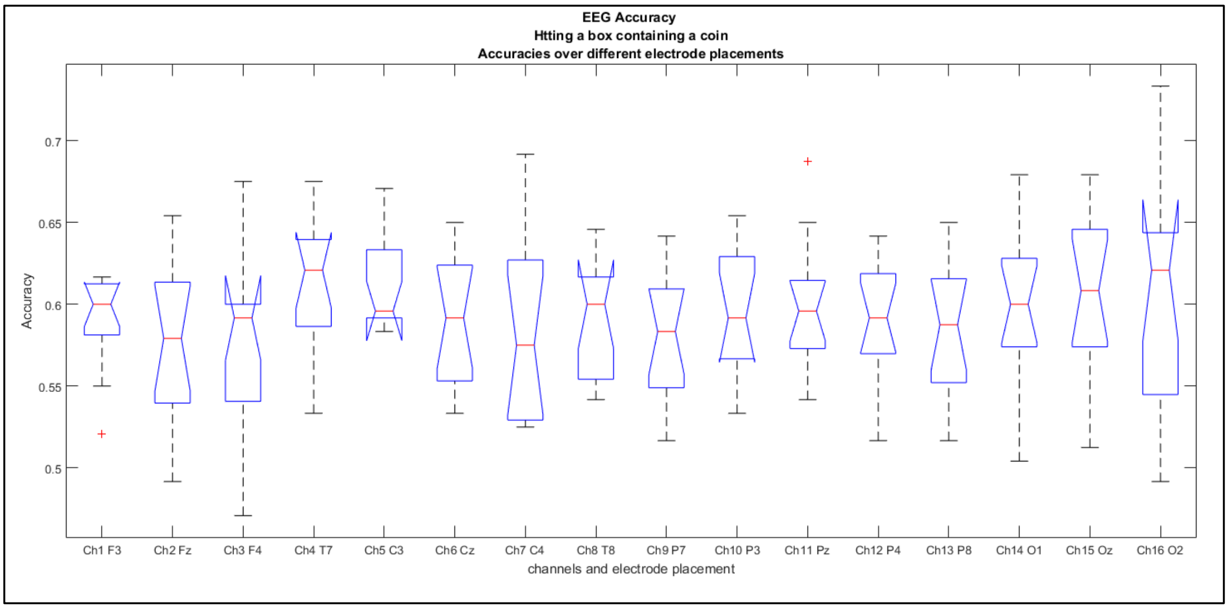

Figure 47 shows a box plot of the accuracies based on the different models trained on EEG electrodes' data in relation to the "hitting a box containing a coin" event

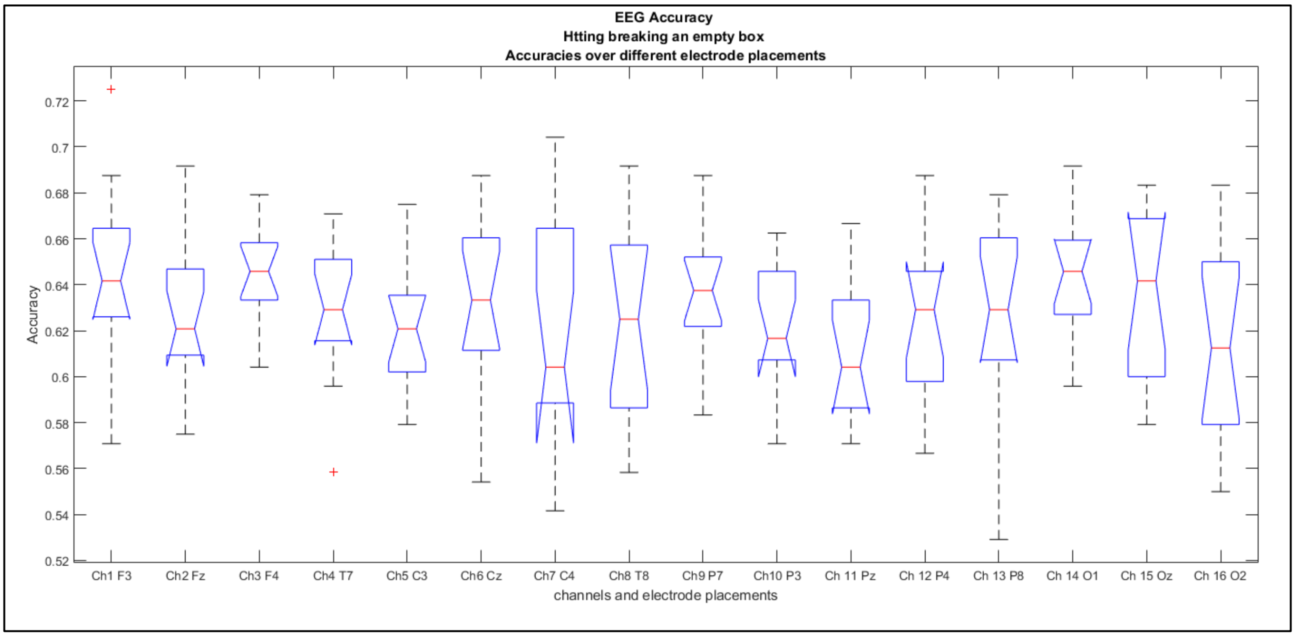

Figure 48 shows a box plot of the accuracies gathered from the different models based on the EEG electrodes in respect to the "breaking an empty box" event

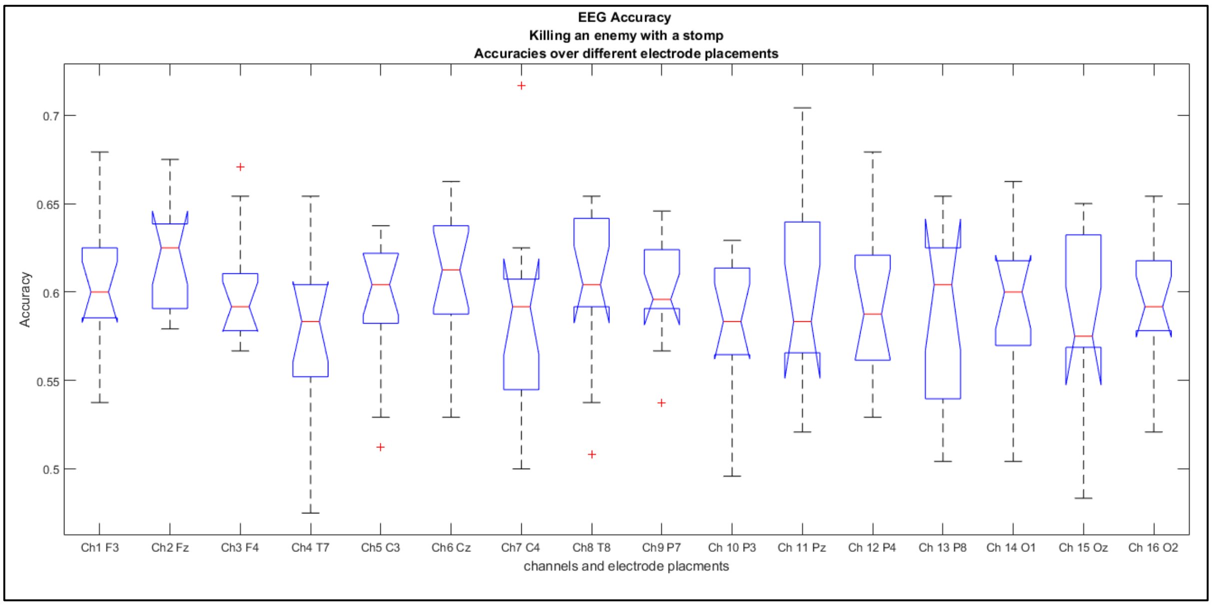

Figure 49 shows a box plot of the accuracies based on the different models trained on EEG electrodes' data in relation to the "Killing an enemy with a stomp " event

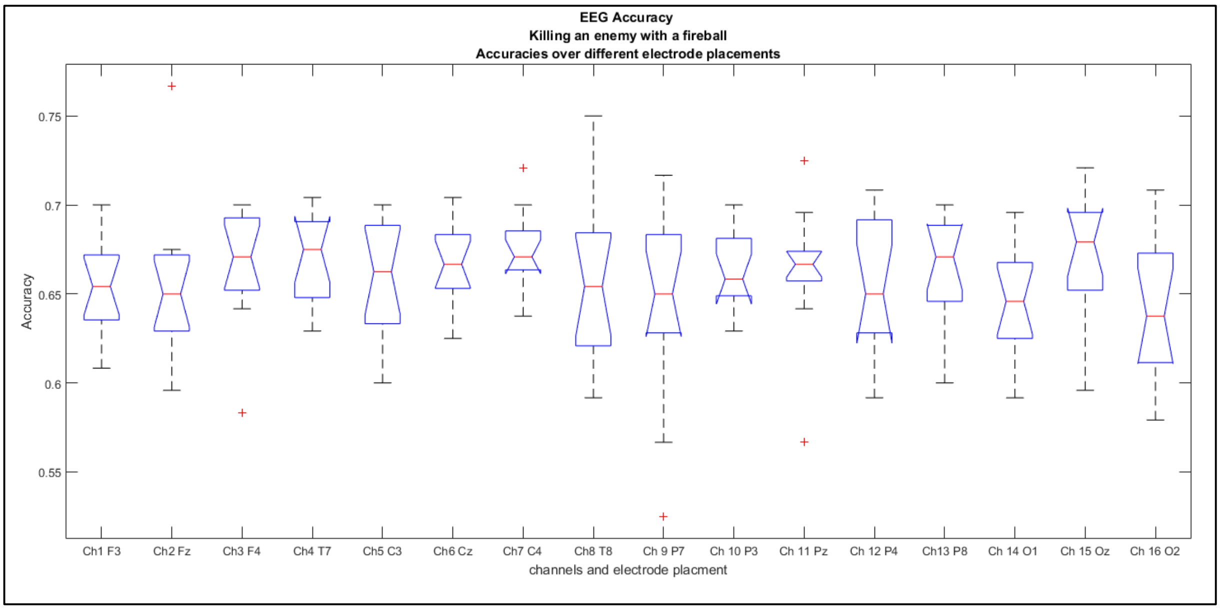

Figure 50 shows a box plot of the accuracies based on the different models trained on EEG electrodes' data in relation to the "Killing an enemy with a fireball" event

J.2 H1

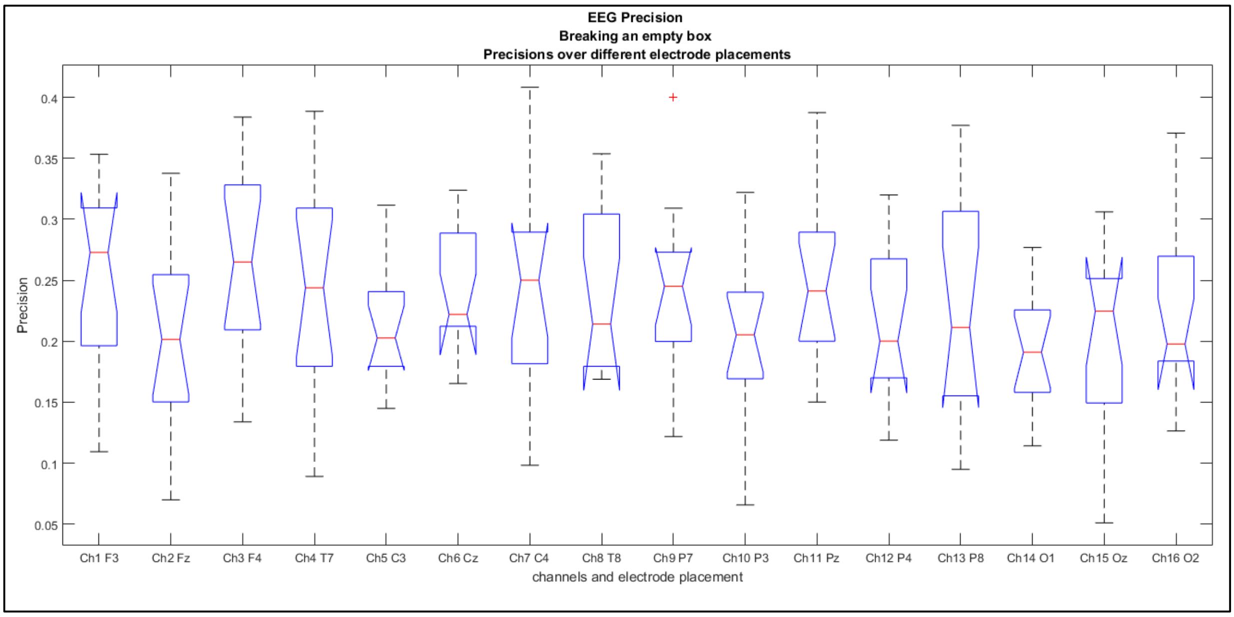

Figure 51 shows a box plot of the Precisions based on the different models trained on EEG electrodes' data in relation to the "Breaking an empty box" event

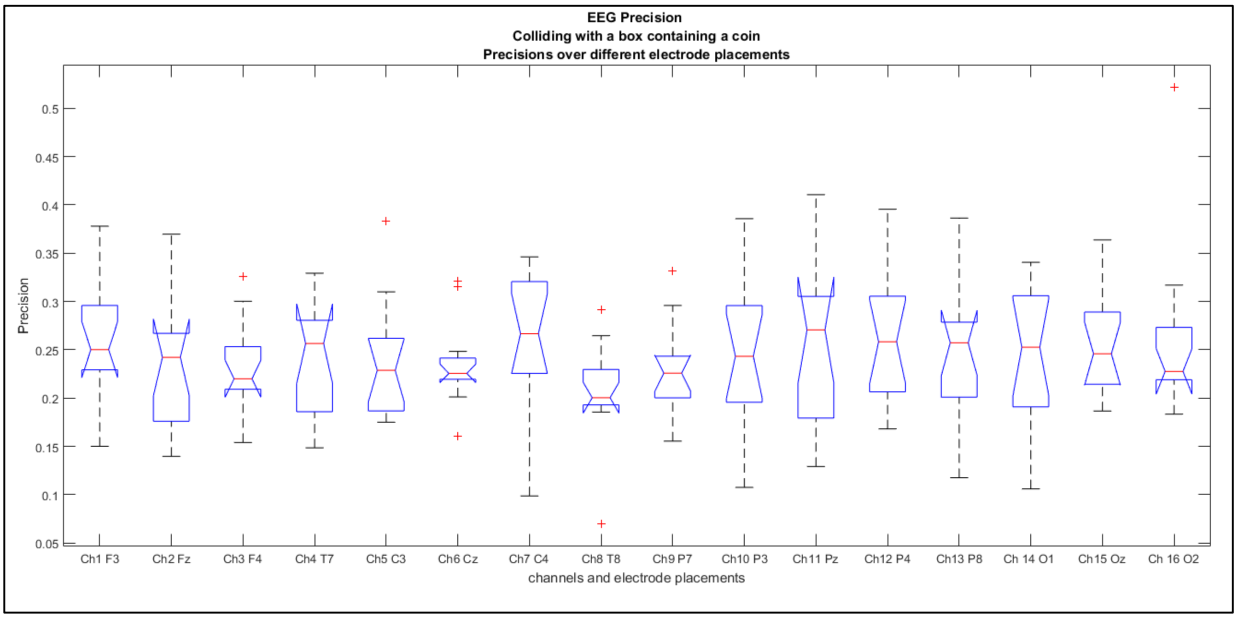

Figure 52 shows a box plot of the Precisions based on the different models trained on EEG electrodes' in relation to the "colliding with a box containing a coin" event

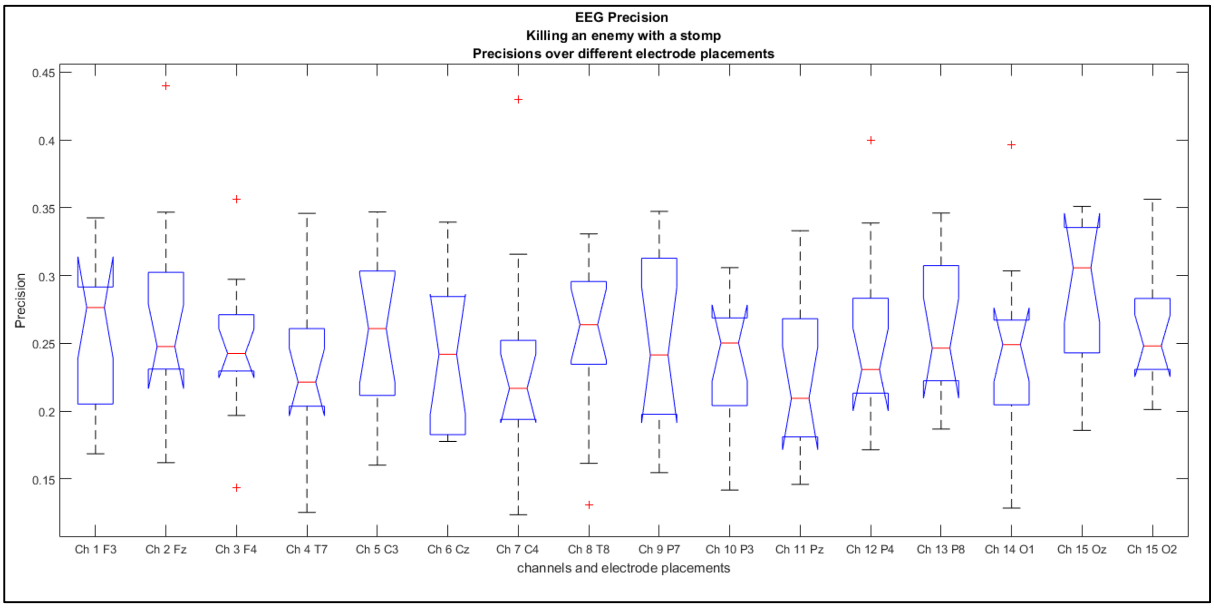

Figure 53 shows a box plot of the Precisions based on the different models trained on EEG electrodes' data in relation to the "Killing an enemy with a stomp " event

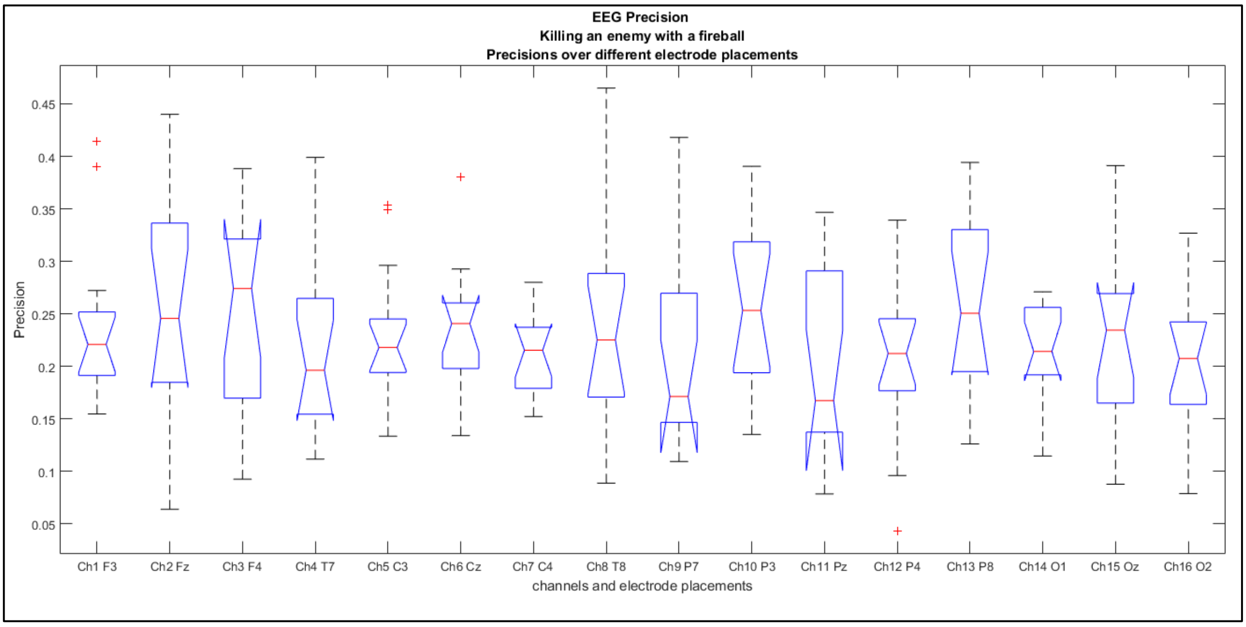

Figure 54 shows a box plot of the Precisions based on the different models trained on EEG electrodes' data in relation to the "Killing an enemy with a fireball " event

J.3 H2

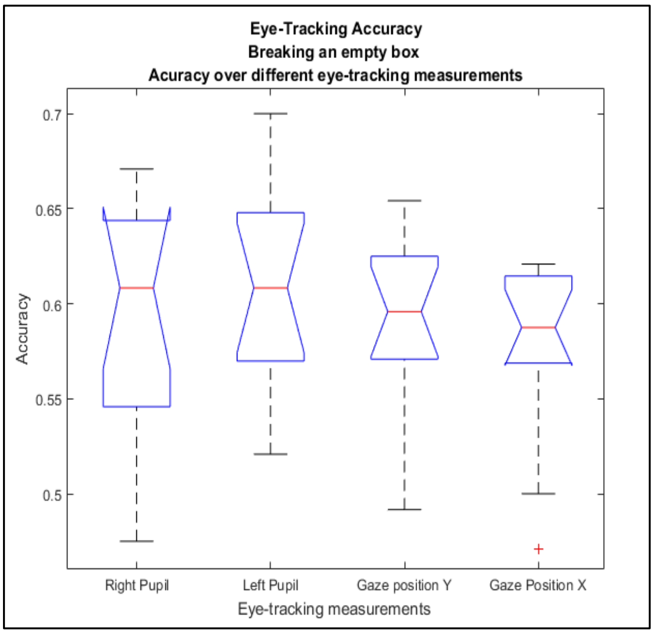

Figure 55 shows a box plot of the accuracy based on the different models trained on Eye-tracking measurements' data in relation to the "breaking an

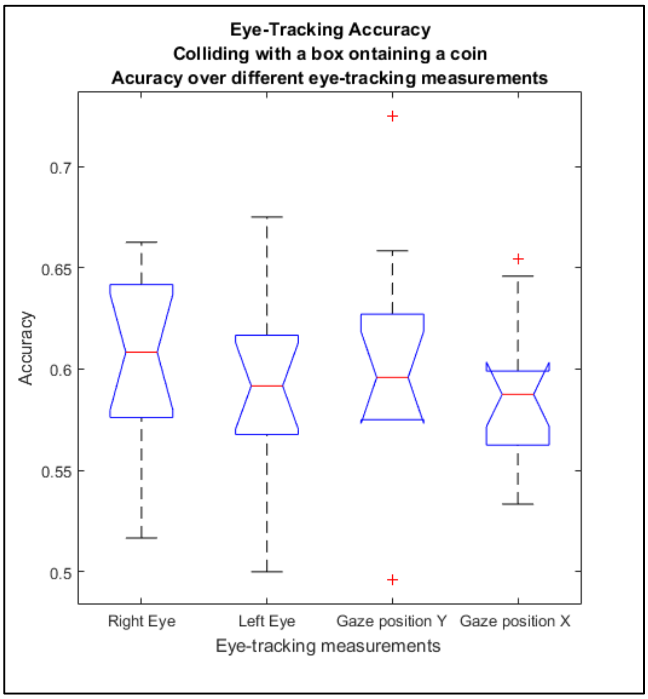

Figure 56 shows a box plot of the accuracy based on the different models based on Eye-tracking measurements' data in relation to the "colliding with

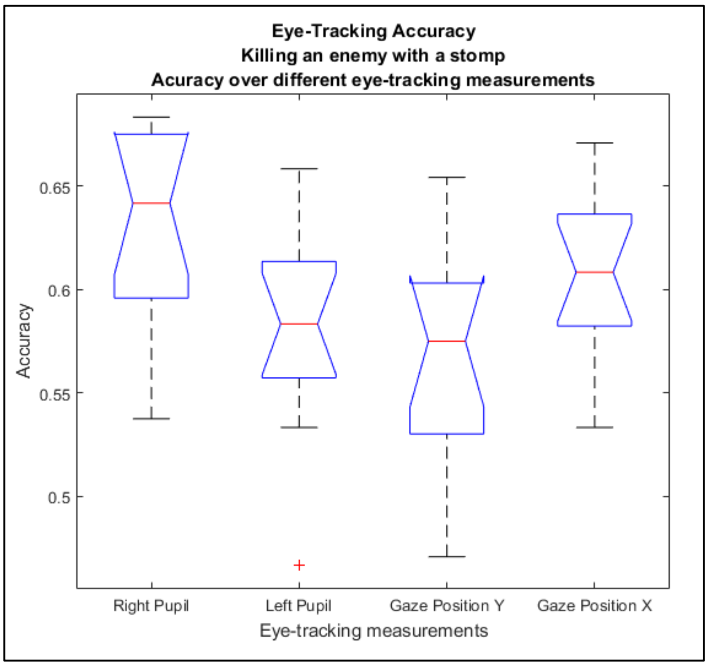

Figure 57 shows a box plot of the accuracy based on the different models trained on Eye-tracking measurements' data in relation to the "Killing an enemy with a stomp "

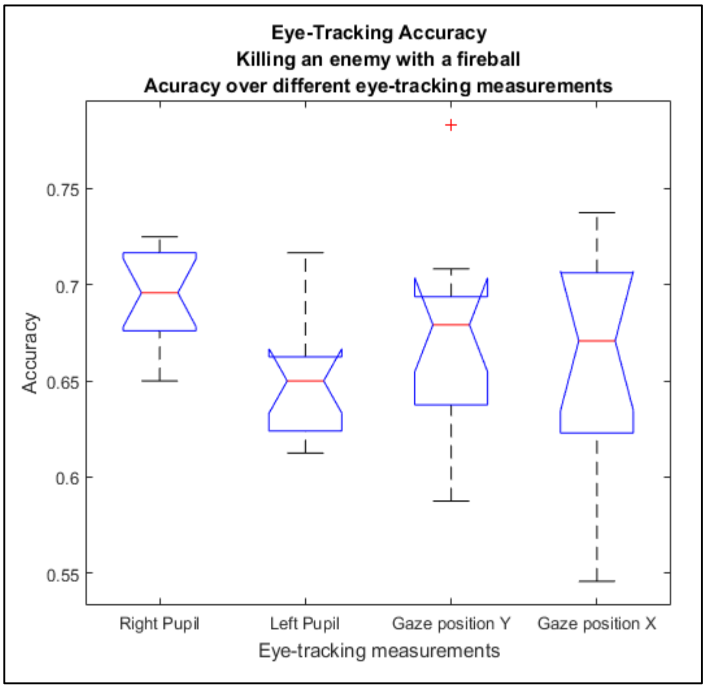

Figure 58 shows a box plot of the accuracy based on the different models trained on Eye-tracking measurements' data in relation to the "Killing an enemy with a fireball"

J.4 H3

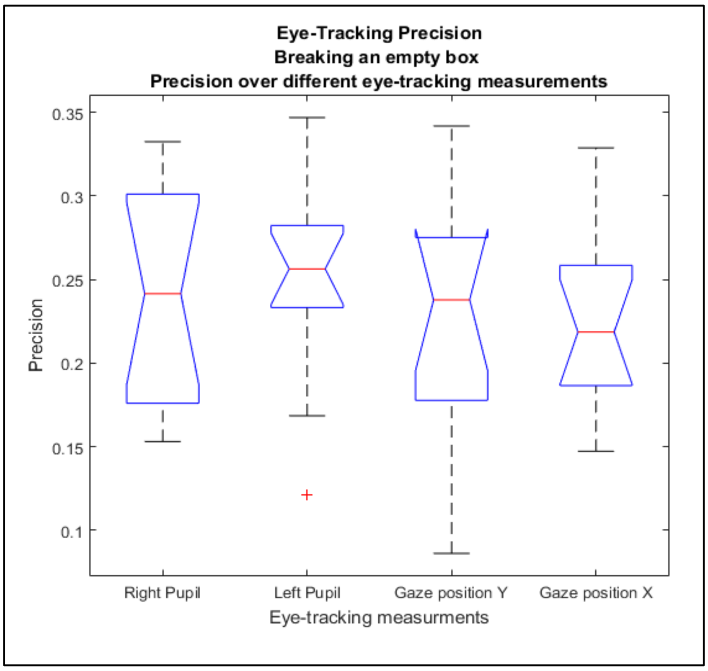

Figure 60 shows a box plot of the Precisions based on the different models trained on Eye-tracking measurements' data in relation to the "breaking an empty box " event

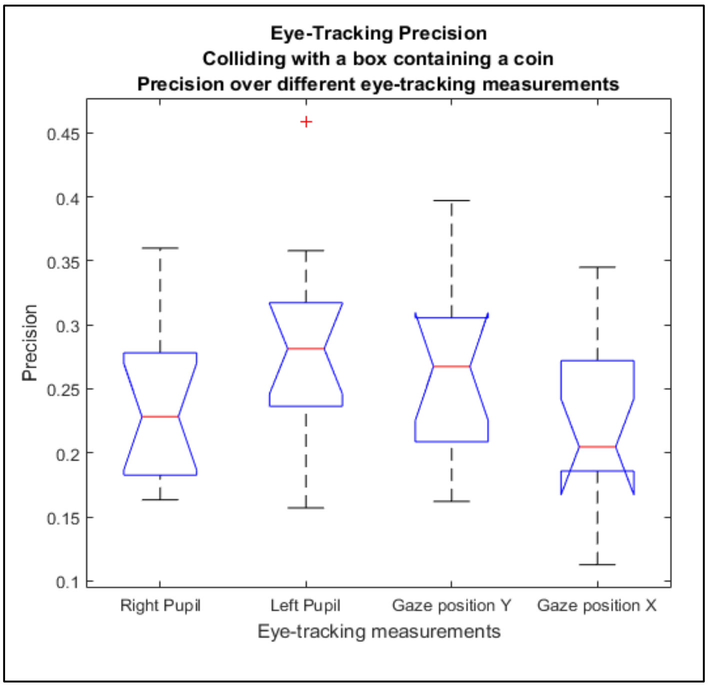

Figure 60 shows a box plot of the precisions based on the different models trained on Eye-tracking measurements' data in relation to the "Colliding with a box containing a

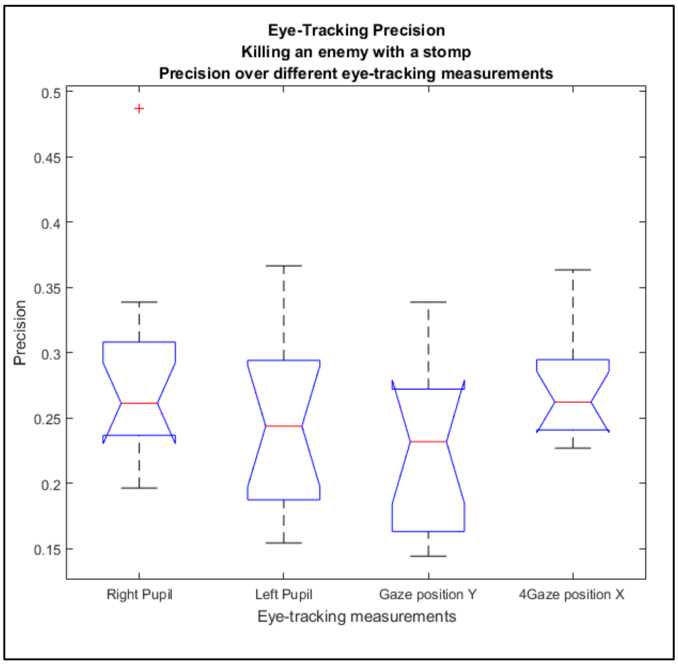

Figure 62 shows a box plot of the precisions based on the different models trained on Eye-tracking measurements' data in relation to the "Killing an enemy with a stomp " event

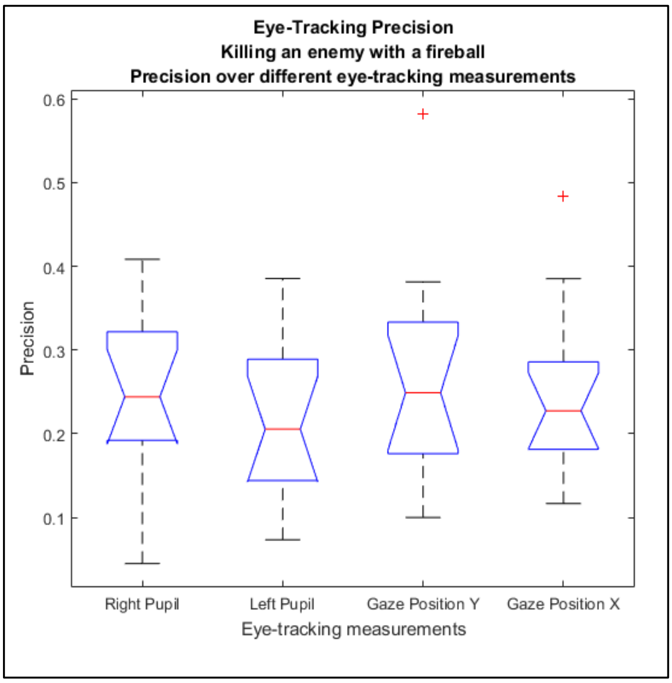

Figure 62 shows a box plot of the precisions based on the different models trained on Eye-tracking measurements' data in relation to the "Killing an

J.5 H4

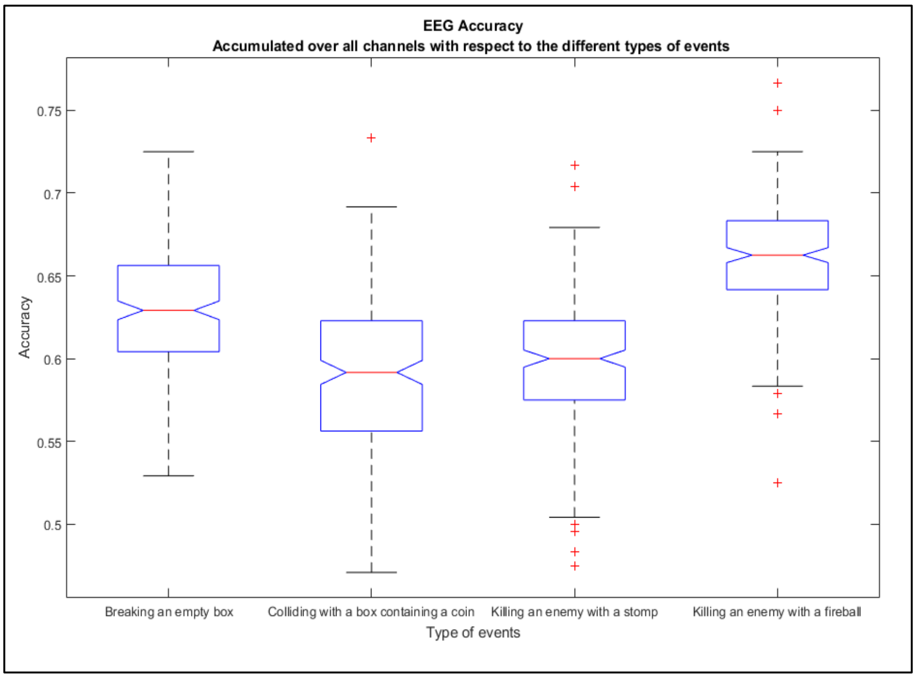

Figure 63 shows the accumulated accuracies based on all EEG channel trained models in relation to the different events

J.6 H5

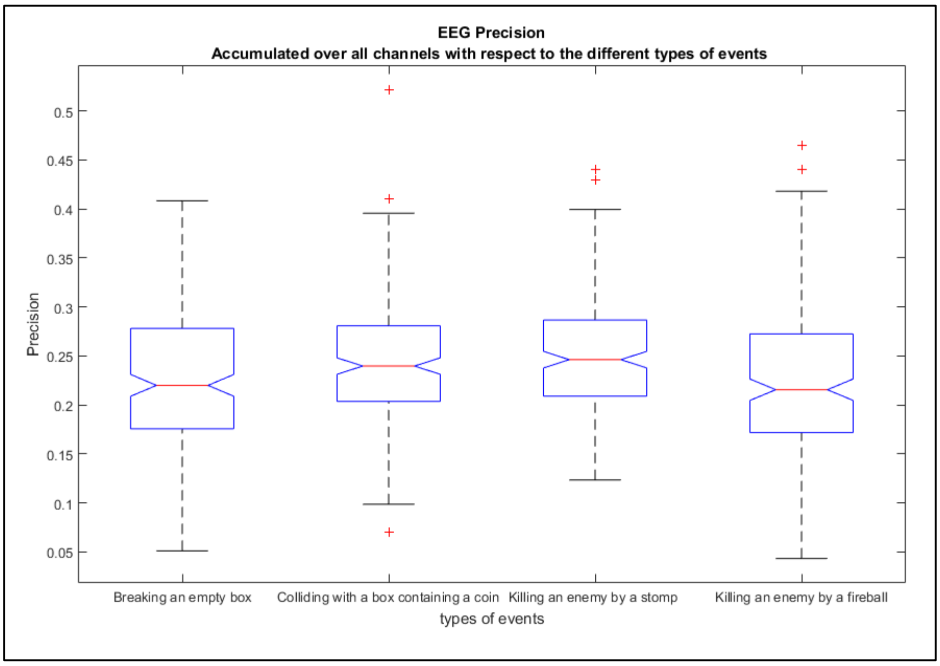

Figure 64 shows the accumulated precisions based on all EEG channel trained models in relation to the different events

J.7 H6

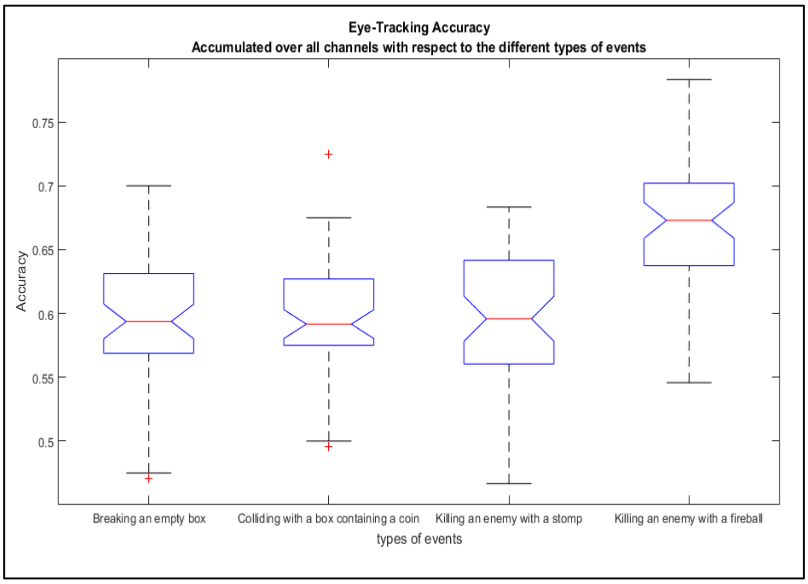

Figure 65 shows the accumulated accuracies based on all eye-tracking measurement trained models in relation to the different

J.8 H7

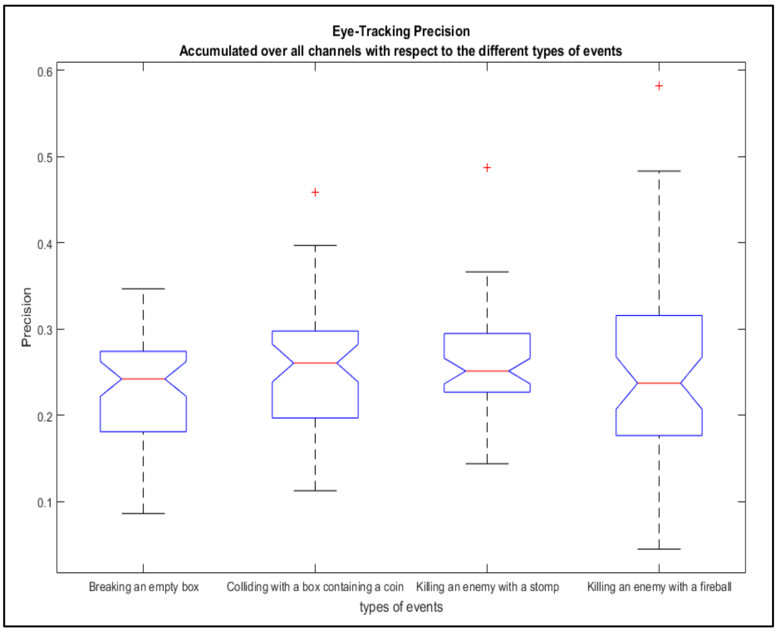

Figure 66 shows the accumulated precisions based on all eye-tracking measurement trained models in relation to the