How does difficulty influence a player's view of Artificial Intelligence?

TB246112

***Abstract*—Artificial intelligence (AI) has become an essential part of many video games, forming the opponent the player will spend most of their time engaged with. Game AI has evolved massively from its inception, with many techniques to construct an artificial intelligence being available. Developers are looking to push all aspects games forward. This dissertation investigates how players perceive AI behaviour with regards to the of AI. Analysing if players can successfully pick out the hardest and easiest AIs.**

A short first person stealth game was developed where player navigates through a level, sneaking past AI agents each with their own set of stats. The results from the study show that players were unable to reliably identify the correct agent.

I. Introduction

Artificial intelligence (AI) has become a fundamental component of video games, non-player characters (NPCs) are one of primary applications of AI. Almost every video game has some form of AI, whether an ally or an enemy that poses a challenge.

In the video game world, there is an expectation that the newest releases will push the boundaries in all aspects: graphics, story, gameplay, sound. As technology continues to advance, AI will play an even larger role in video games. By analysing both existing video games and academic literature this dissertation, will look at what game developers can do to create a well crafted, engaging player experience which is rewarding as possible. While simultaneously investing as little resources as possible.

The study in this dissertation looks at player psychology in a small isolated level, with an AI focus. It hypothesizes that players cannot tell the difference between variations of AI that are identical in programming, while having difficulty levels. Instead players will attribute the differences to being more 'intelligent'. This paper challenges the traditional view of 'intelligence' of an AI. With the comparisons made between different titles for the most intelligent AI. This paper will evaluate the techniques used by the most 'intelligent' enemies, analysing what makes them so highly regarded.

The paper is structured as follows: Section II covers existing works and topics, Section III discusses the research question, and the hypotheses for the study. Section IV details the computing artefact created to answer the hypotheses, Section V talks about the research methods, Section VI shows participants data, Section VII goes into detail about the data analysis and the limitations of the research, Section VIII outlines the strict ethical process this research has gone under and finally Section X is the conclusion to the study. References can be found just below as well as the appendices containing additional figures supporting the research.

II. Literature Review

This literature review details the history of game AI with the techniques that have been used and will look at games which broke new ground with AIs that defined the industry. By analysing these games, developers will be able to create AIs that maximise player satisfaction.

A. The history of game AI

The use of AI in a game context dates back to the 1950s where the first computing machines were used to simulate human opponents such as the match game Nim [1]. Chess and checkers became the next focal point of game AI development, using the Manchester Ferranti in 1951 by Dietrich Prinz. Since then, board game AI grew stronger and stronger eventually beating the then world champion Garry Kasparov in 1997, Deep Blue.

Halo 2 (2004) was the first mainstream to use behaviour trees [2]. It was revolutionary at the time, enemy characters would use their sense to perceive the world and then from that information, turns into action. Behaviour trees remain one of the resounding choices for creating an AI in video games.

B. Behaviour Trees

Behaviour trees are one of the most widespread AI techniques [3], allowing for customisable hierarchy of nodes [4]. Described by Martins as: 'behavioural flowcharts based on conditions on the world around them' [5]. BTs are made up nodes which control what behaviours are executed, each node will return either a SUCCESS, FAILURE or RUNNING state depending on the action node code. Alien Isolation is seen as one of most advanced AI systems created for a video game [6], featuring a MacroAI and a MicroAI. The MacroAI is a game director similar to that employed in Left for Dead [7]. The MicroAI is the alien AI itself, an NPC that reacts to the player using set of senses, as well as commands given by the AI director. OpenCage developed by Filer [8] allows the view of the behaviour tree, the microAI contains over 100 nodes showing just how advanced it is.

C. Game Directors

AI game directors manage the player experience by monitoring the players current situation. This feature was first developed for first person zombie shooter Left for Dead (L4D) [7]. As Booth puts it in his GDC talk *'The AI Director algorithmically drives overall pacing'*. The director does this by each survivor (the player) having an emotional intensity value, increasing when attacked by the zombies. This value

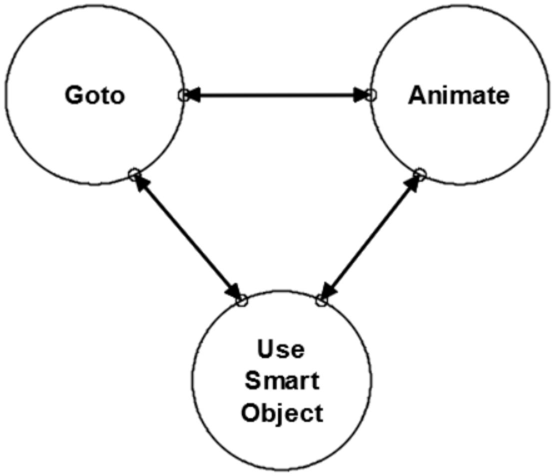


Fig. 1. Finite State Machine from F.E.A.R

decays over time toward zero only when the survivor safe from immediate danger. A finite state machine uses this intensity value to control the enemy amounts and to direct them to specific players to ensure all players get an engaging experience. The 4 states are:

1) **Build Up:** Create a full threat population, made up of standard zombies and special infected until emotional intensity crosses the peak threshold −> 'sustain peak'.
2) **Sustain Peak:** Maintain the population threat level for a short few seconds after peaking, then −> 'peak fade'.
3) **Peak Fade:** Switch to a minimal/low threat population, waiting until the intensity decays to peak threshold. Then −> 'relax'
4) **Relax:** Keep the minimal threat population for a short duration, or until the player are close to the end of the level. Then −> 'build up'

Game directors are one of the most significant developments in game AI, substantially improving the replayability due to the AI being able to modify and adjust itself based upon the player's ability.

D. Goal-Oriented Action Planning

GOAP is a popular AI structure originally implemented for video game F.E.A.R [9], GOAP executes a sequence of actions in a hierarchical manner to satisfy a goal. F.E.A.R uses Finite State Machine shown in figure 1. The system is very simple, for instance an AI taking cover is: move to a position, play crouch animation. The behaviour complexity comes from the planning, determining how the AI switches between states and the parameters which are set. F.E.A.R separates this logic from the FSM. Planning has several benefits: goals and actions can be separated, behaviours can easily be layered and enemies can dynamically react to the situation they find themselves in.

E. Machine Learning

Artificial intelligence in video games is an expansive topic that has been studied for decades. Recently published papers explore have taken a machine learning focus, largely due to advances in processing power. Machine learning has many applications in the real world making it a vital area of importance. The debate on what constitutes as 'intelligence' still a prevalent topic in the computer science community, from its inception in Turing's Computing Machinery and Intelligence paper [10]. Searle presents the Chinese Room Argument [11]. Stating a computer cannot be intelligent since it is just interpreting instructions it receives without any understanding. Hawkins proposes memory-prediction framework [12], believing that the brain has a common 'algorithm' which could be simulated by AI, using Bayesian networks.

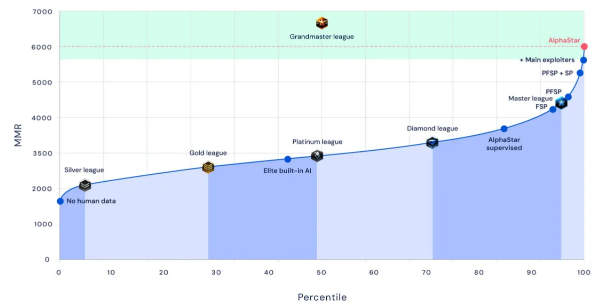


Fig. 2. AlphaStar MMR rating in StarCraft 2

Many in the scientific community argue that video game AI does not constitute as 'real' intelligence [13]. Instead when a video game AI is called 'intelligent' it is only a specific context that it has been programmed or trained. Researchers at Google Deepmind created AlphaStar, a machine learning artificial intelligence, based on reinforcement learning [14]. Intended to compete at the top level in StarCraft 2. This AI interpreted vast quantities of data, watching thousands of games played by the best players and could beat 99.8% of all players, as shown in figure 2. AlphaStar plays at a level almost unachievable by humans, AlphaStar was created to push and test the bounds of artificial intelligence, using StarCraft as a launch pad. However, whilst interesting as a case study, no player would want to play against this opponent.

III. Research Question & Hypothesis

The research question motivating this paper is: *'How does difficulty influence player perception of artificial intelligence'*. To answer this question, the computing artefact will investigate how enemy agents values can be adjusted to invoke a response from the player. By adjusting these values and therefore the difficulty, players experience can be improved.

The hypotheses for this paper can be seen in table I. The main hypothesis in this paper is players will not be able to tell the differences between the AI models.

There are two additional hypotheses, being an extension of the main hypothesis. Player will rate enemies which perform better i.e: detect the player more times, more favourably. This would be because they think the AI is more effective since it managed to catch them and pose a threat. The last

TABLE I
HYPOTHESIS TABLE

	Hypothesis	Data Source
1	Players will be unable to tell the difference between the AI models	Questionnaire
2	Player will rate enemies which perform better more favourably in the survey	Questionnaire
3	Players favourite AI will be the one they performed the best against	Questionnaire & In game data

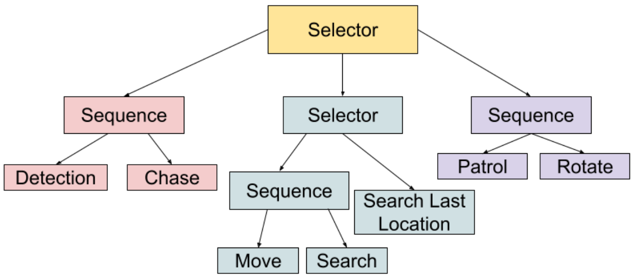


Fig. 3. Behaviour tree for enemy AI

hypothesis is that players favourite AI will be the AI they did the best against. As the player performed well against the AI, outplaying the AI managing to succeed. Players will view them in a more positive light.

IV. The Research Artefact

The research artefact is a short stealth game experience, with patrolling AI agents. Participants were tasked with completing 4 short levels, each with a different AI variant, represented by the colours red, blue, yellow and pink. These AIs have identical behaviours, operating with the same behaviour tree. The behaviour tree can be seen in figure 3. It is designed to be a generic system of a stealth game. It has 3 branches, chase, search and patrol.

1) **Chase:** Detection node creates a cone of vision (figure 13 shows the different cone sizes), covering a small area, if the player enters the cone then a detection timer will begin to tick up (there is a yellow bar to show detection, see figure 6). If the timer reaches the 'detected' level, then the 'Chase' node run causing the agent to pursue the player.
2) **Search:** Responsible for hunting for the player. The agent approaches the last known player position, upon reaching it will rotate similar to patrol state. This repeats until the player is found, or if the agent has failed to find the player after a set amount of attempts.
3) **Patrol:** Inside is the AIs idle state. When given an array of points the agent will move to each point one after the other, until repeating. After successfully reaching a point the agent will wait a small amount of time. While waiting the 'Rotate' node runs, causing the agent to rotate from side to side.

The computing artefact was developed in the Unity game engine version: 2021.3.4f1, using the C# programming language in Visual Studio 2022. Unity was chosen due to being

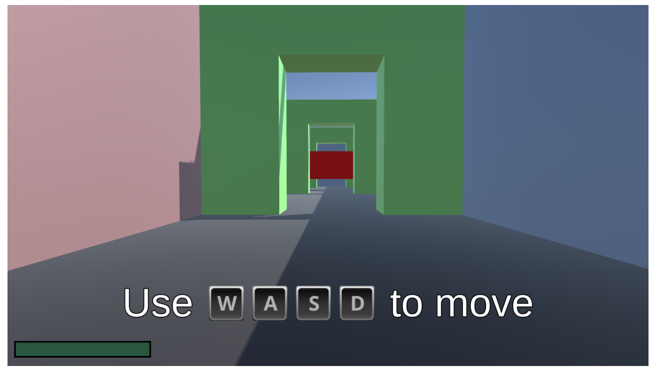


Fig. 4. Artefact tutorial, with control prompt

well-supported and allow for the quick building of basic applications, it is also the programming language that this author is most experienced with. The StreamWriter class from System.IO namespace was used to write participants data to a CSV file. The artefact uses no pre-made assets except an Input icon package [15] to assist with the tutorial prompts see figure 4. The computing artefact was stored on the Falmouth Universities GitHub Enterprise server [1].

A. The Game experience

The game demo consists of a tutorial, followed by 4 short levels, taking roughly 2 - 3 minutes to complete, ending with a questionnaire. The demo is designed to be as minimalist, mainly using the probuilder package, with simple materials composing the walls as seen in 4. This approach was chosen to strip away outside influence for participants, the only focus is upon getting through the level. The tutorial shows the user the controls and demonstrates how the AI enemies work. After completing the tutorial, one of the AI models is randomly selected, the level is the same for the different AIs. A bird eye view of the pink AI level is shown in figure 5, the player is allowed to take as long as possible to navigate through the level roughly following the red arrow. There is a maximum of 3 attempts per level, if the player exceeds this then the next level will load. This continues until all 4 levels have been played, then they will be prompted with the questionnaire (see figure 8). A video showing this is available at [2]

B. AI Variants

All the enemy behaviour values housed in the ScriptableObject **EnemyData**. Shown in figure 7, there are 4 main areas that

[2]Artefact Demo Video https://www.youtube.com/watch?v=4f8x1sx2xSQ

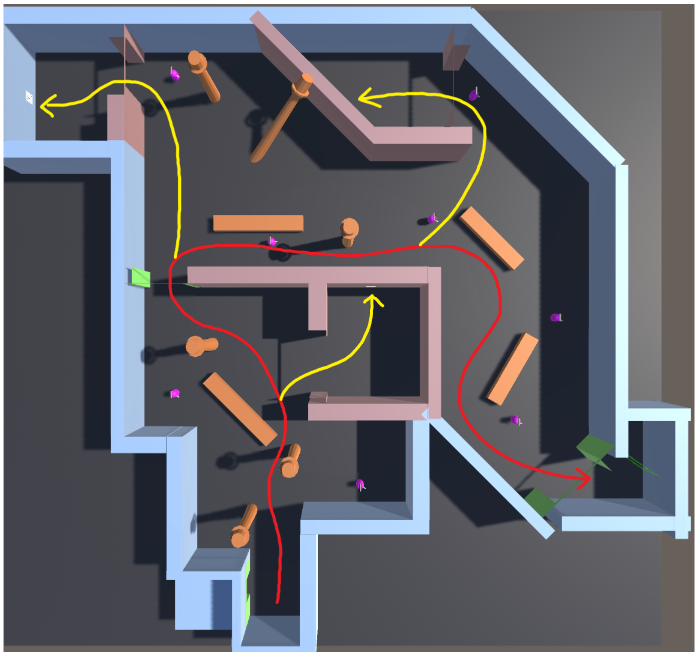


Fig. 5. Main artefact level, player path is annotated in red, with optional collectables shown by yellow arrows

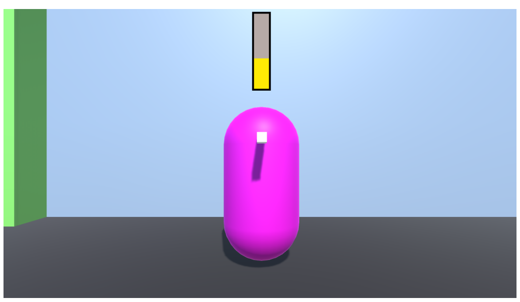


Fig. 6. A pink enemy agent, with a half full detection bar

Fig. 7. Stat values for all enemies

can be adjusted these being: Cone, search, patrol and speed. The cone variables influence the field of view (FOV), allowing for the angle, distance and detection time to be modified. Search effects how the agent reacts to almost detecting the player, or alternatively losing the player's position. The search radius is how far from the players last position the agent will search, the search attempts are how many times it will move to a spot around the last known position before giving up. Patrol simply controls how long the agent waits at its designated patrol points before moving to the next point. Speed controls the agent's movement in its standard (patrol) state and chase speed is used when the AI is pursuing the player.

For the experiment three distinct templates were created. The blue AI had the smallest detection cones and most forgiving detection time. The pink AI had the widest detection cones and the quickest detection time. The red AI values fell between these two extremes. Finally, there is also the yellow AI which is identical to the red AI. However, the detection bar UI normally seen above the AI is removed.

The blue enemy being the weakest, the pink the strongest and the red is in between. Figure 13 shows a size comparison between the different variants. There is also a fourth variant with the same values as red, but does not have any of the detection bar UI. After completing the levels, they will answer a questionnaire about their experience and their feelings towards the AI.

C. Data Collection

The computing artefact handles player stat data collection in the class 'StatisticsTracker'. This class uses Action events [] to update variables stored in the 'PlayerStats' scriptable object (see figure 14 in appendices). These action events are static instances that are invoked in the script. For example, the 'Detection Node' class handles the FOV & detection of the agent. When a player gets spotted the OnPlayerDetect event is triggered. Due to being an event, any number of scripts can be subscribed to it without a direct reference, compartmentalizing

[2]GitHub Link https://github.falmouth.ac.uk/Games-Academy-Student-Work-22-23

Fig. 8. Screenshot of the questionnaire at the end of the artefact

the code. This is used in 'StatisticsTracker' to then increase times detected. This approach is taken for all 'PlayerStats' variables except for play time and distance travelled. Play time is tracked from when the scene is loaded until the next scene transition, distance travelled is calculated at the same time

Each level has its own 'PlayerStats' class which is written to during play time. Then inside the 'StatsToCSV' class, a file path for writing to is specified, the level order is taken from the LevelLoader class. all of the 'PlayerStats' scriptable objects are stored inside an array. Once the final level has been completed, the SaveToCSV function will run. This function has a for loop iterating through each index of the playerStats array writing to a corresponding string array. Upon completing the for loop, a final **data line** is constructed combining all of the strings into one, each separated by commas. The 'PlayerStats' scriptable objects are now reset back to zero, as the data has been extracted into a single string. This data line string is written to the designated file path by the 'StreamWriter' System.IO class. 'FileStream' is also used to ensure the file appends and prevent multiple instances from accessing the file (preventing missing data).

V. Research Methodology

The research artefact was played by 22 participants (12 of which were online), the data collected was split into quantitative and qualitative data. The quantitative data came from statistics that were gathered in real time for each individual level. Whereas, the qualitative data was made up of participants responses. The data compiled was processed in R[3].

A. Validation & Artefact Development

The research artefact was extensively tested to ensure all features were working as intended. Throughout the computing artefacts development white box testing was used to test that elements had been implemented correctly. Table II show a unit test table. Following the white box testing method, unit tests were conducted using this table. With participants, the data obtained from their tests was not included in the data set.

To start with a maze level was constructed. Then simple character controller was quickly implemented, controlled by Unity's new input system. The behaviour tree framework (sequence, selector, tree etc) was based off an online tutorial [16]. After each feature was added it was placed into the maze scene and tested, only once there were no bugs was the next feature worked upon.

One of the main parts which got refactored was the 'StatsToCSV' and the 'FeedbackCollector' script. Originally the FeedbackCollector script was made to write to a Google form, using web request to write the values into the form from Unity. This approach was chosen since google forms responses can be downloaded as CSV files. During development there were major challenges with getting the 'StreamWriter' class to append data, this would allow circumnavigation of the problem. This worked and StatsToCSV was removed in favour of the Google method. However, google forms is not GDPR complaint by default. A solution was found to CSV append issue as such, 'FeedbackCollector' was changed with 'FileStream' & 'StreamWriter' being used to allow the data to append. Solving the problems of data writing over the same line.

B. Measuring Perception of AI

There are a variety of different methods to measure a players view towards an AI. Biometric data gives an insight into a player physiological response, examples being: heart rate, eye movements and skin conductance. These measurements can show fear, pressure, intimidation or anger. This is advantageous compared to a standard survey it is the body's automatic reaction to stimulus [17], which eliminates a large degree of bias.

For this study a survey was chosen, due to time constraints and lack of resources. The questionnaire is structured into 2 main sections, **checkboxes** & **likert scale**. Due to the need to compare the 4 different AIs, checkboxes were used to clearly obtain a participants view. The likert scales used in this study were based off the Game experience questionnaire (GEQ) [18] using a scale of 0 to 10, for how they felt. 0 meaning 'not at all', 3 = 'slightly', 5 = 'moderately', 7 = 'fairly' and 10 being 'extremely' A standard likert scale falls between 1 - 5 or 1 - 7 []. The wider scale was chosen as allows users to be more precise allow them to rate closer to how they actually feel.

C. Data Gathered

As stated earlier, the computing artefact collects both quantitative and qualitative data. This is required to get an understanding of the players overall experience.

1) **Player Statistics:** As shown in section IV-C the artefact measures player performance on several key variables (seen in figure 14). The data collection is encapsulated away from the participant. The player receives no information on how well they are doing, the only variables which is shown is the 'Collectables Found' counter at the top right of the screen. This prevents players from observing how the data is recorded from in game events, reducing the chance of exploitive behaviour. This is only

[3]R programming language https://www.r-project.org/about.html

a precaution since there is no reason for the participants to act maliciously.
The variables: detections, time in cone, time taken, distance travelled, times caught and collectables were chosen as the statistics to collect. As the artefact is a stealth game, the main interaction with the AI involving sneaking, these variables can show how player successful the player is as well as their approach.

2) **Questionnaire:** The questionnaire is inside Unity itself, as shown in figure 17. The questionnaire is a mix between toggle boxes, optional text boxes and likert scales In total there are 4 toggle boxes, 5 optional text boxes and 16 likert sliders. Since there are 4 AI variants, the same 4 statements are presented for each AI. The toggle groups is presented first, since they are discrete values, participants will have to consider their opinion carefully, meaning that when they fill out the likert scales they will have already thought about it. AI variant names being displayed in their colour, help locate them. In addition to the 4 AI colours, 'None' was also added. This was to give participants who thought the AIs were all similar an option. The text boxes were added to allow participants to explain their reasoning.

D. Research Philosophy

This research has taken an interpretivism approach [19], when trying to investigate the research question it is an inherently subjective topic. Players view towards AI is dependent upon what games they have played and their personal experience. However, when taking an interpretivist approach it is subjective in nature as the research. This can be mitigated by use of quantitative data to back up what is deduced from the qualitative data, participant comments can be paired with the in-game data to obtain more accurate results. As such, a pragmatism philosophy [20] provides an objective medium for pursuing this scientific yet social research.

VI. Data Analysis

The data present in this analysis was completed by 22 participants, predominantly from the Falmouth Games Academy. R studio was used to analyse the data gathered and to produce the figures 9 to 12. Pie charts have been used to present the qualitative data from the questionnaire. Whilst scatter plots compare different player statistics with the likert scales for individual AIs.

1) **Hypothesis I:** *"Players will be unable to tell the difference between the AI models"*. Pie charts in figure 9 show little consensus in identifying specific attributes of the AI. In terms of overall difficulty, blue is the easiest, followed by red, yellow then pink is the most challenging. Figure 9 (top left) shows that players can successfully pick out the toughest AI model.

2) **Hypothesis II:** *"Player will rate enemies which perform better higher in the survey"*. Figure 12 shows 4 scatter graphs, each plotting detections against one of the likert scales from the questionnaire. All graphs show a weak positive correlation, having the following r values: intelligence 0.352, pressured 0.210, skilful 0.407 and engaged 0.293. A stronger relationship is seen by running Spearman's R using the same variables as figure 12. These p values were: intelligence 0.00064, pressured 0.050600, skilful 0.00006 and engaged 0.00761. This shows that enemy detection rates is definitely how highly players rate.

3) **Hypothesis III:** *"Players favourite AI will be the one they performed the best against"*. Figure 9 shows what players rated as their favourite AI opponent. Using Spearman's R

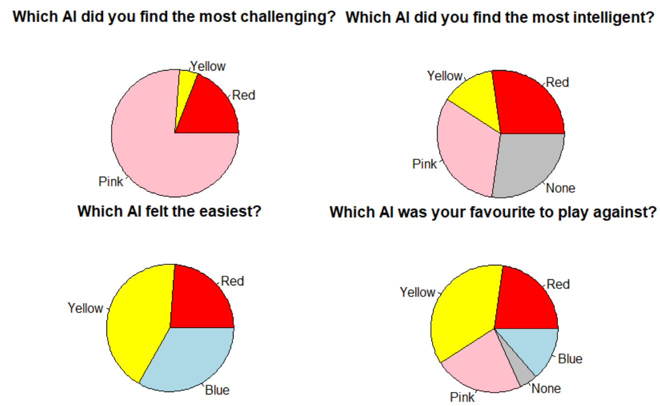


Fig. 9. Pie charts showing proportion of participants who rated different AIs

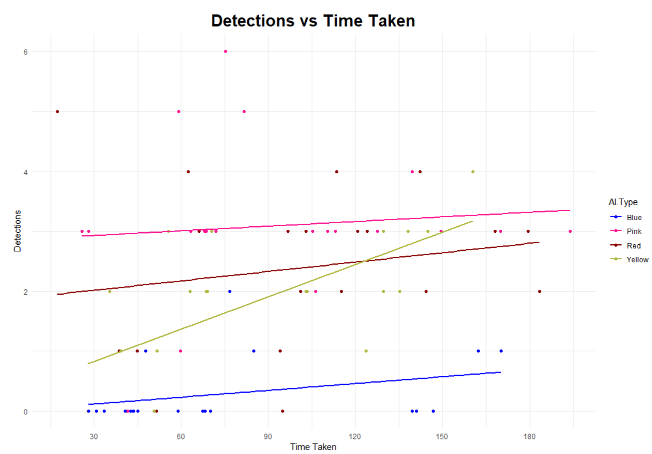


Fig. 10. Scatter graph showing player detections against level completion time

VII. Discussion

Analysis of the data shows that hypothesis I and II have some merit. With II being particular strong due to the low p values which are below 0.05. However, despite this the r value between detection and the different GEQ questions falls between 0.2 and 0.4. This would suggest that the enemy detection rates have a positive relationship with positive feedback. Though the limitations in the study hold back showing anything truly meaningful.

The pie charts show (figure 9) that players can clearly identify the strongest AI variant (pink). However, the other

Fig. 11. Time Taken against Distance Travelled

categories are more unclear. In particular, the *"Which AI felt the easiest?"* question is heavily divided. Only one third of participants correctly identified blue as the easiest AI, with the vast majority putting either yellow or red. As shown in figures 7 and 13, the red & yellow enemies are superior in every single way. This could be due to the way the artefact is presented to the player. For instance, in the tutorial level the red AI is used. Players may think (subconsciously or otherwise) that the weakest AI would be placed in the tutorial, as a way to ease the player in.

Interestingly figure 9 shows that the players were split on their favourite AI. The largest choices being yellow, red and pink. Yellow being most chosen is due to missing its detection bar, making it unlike any other AIs. Players had to adapt and change their approach, using their judgment to work out when they were in range. Figure 10 shows player detections plotted against time, the blue, red and pink AIs all have the same regression line. Whereas yellow has much stronger correlation, showing that players were rewarded for taking their time and had a much greater chance to succeed undetected. Hypothesis III was unable to be proven or disproven, as there was no data collected that could portray 'performance' in a balanced fashion.

A. Verbal Feedback

Participants were given the opportunity to fill in text boxes allowing them to explain their reasoning in the questionnaire for the multiple choice tick box questions and if they had any other comments at the end. The majority of participants filled in most of these input fields, which granted some interested insights into players perception and thoughts. One participant said *'All enemies felt uninteresting except for yellow as they just felt like variations on an uninteresting concept for stealth. It was either very forgiving or very unforgiving yellow felt like a good balance while still having some issues. None felt particularly challenging as a reactive enemy.* This comment shows they completely saw through the AI facade, as they thought all the AI opponents were just the same code but with different values, blue and pink being to the extremes.

B. Issues & Limitations

Levels are randomly chosen and can appear in any order (after the tutorial). Therefore, participants experiences vary greatly as there are 16 different level order combinations. From the 22 participants data, using R to split *'LevelOrder'* results in 15 different level orders. This diminishes the effectiveness of the data since the majority of participants have had a unique experience. Depending on the level order, you could be eased into with a combination of Blue −> Red −> Yellow −> Pink. Alternatively, participants may have that in reverse starting with pink which would be a large difficulty increase. Furthermore, the first level played is significantly more challenging as the player does not know the layout of the level. After completing the first level once, the player knows enemy locations, patrol paths and general route for success. As a result, only the player first level will produce unbiased accurate statistics.

The structure of the questionnaire shown in figure 17 may introduce some biased results from its layout. A common pitfall of the likert scale is that participants rating will avoid the extremes, typically rating in the centre, saving the low or high ratings for when they might need them [21]. As such, the AIs which are being rated first are likely to have more moderate responses.

A large limitation for this study is the obtained sample size. Only a sample of 22 people was obtained for the study. G* power [22] was used to get an estimated sample size for the study, having an effect size of 0.6 ρ at 0.05, results in. Consequently, the study's accuracy is weak, and the null hypothesis cannot be rejected. In addition, the majority of the sample was from Games Academy students. This has likely skewed my data as the students are experienced with game development and have encountered many AIs. Therefore, it is plausible students from the Games Academy have performed above average compared to the overall population. In the future random sampling should be used [23], in a wider demographic.

VIII. Ethical Considerations

The experiment conducted to the collect the data found in this paper is classified as medium risk research, since human participants are involved. Participants tested different AI models by playing through a short game experience, where data upon how they performed was collected. After completing the game, participants filled out a questionnaire.

The Declaration of Helsenski [24], Falmouth University ethical guidelines [25] and standards of the British Computer Society(BCS) [26] were strictly followed in this study. An information sheet and consent form were presented to participants (shown in figs. 15 and 16) as soon as the unity executable is run, only allowing the user to proceed once all tick boxes on the consent form are ticked. Additionally, users have to scroll down in order to see the 'continue' & 'proceed' buttons, meaning that there is a higher chance of participants engaging with the information materials. An exit button is always visible on the screen during the consent stage, allowing participants to opt out at any time.

No personal data was collected from participants. However, The General Data Protection Regulation (GDPR) [27], was

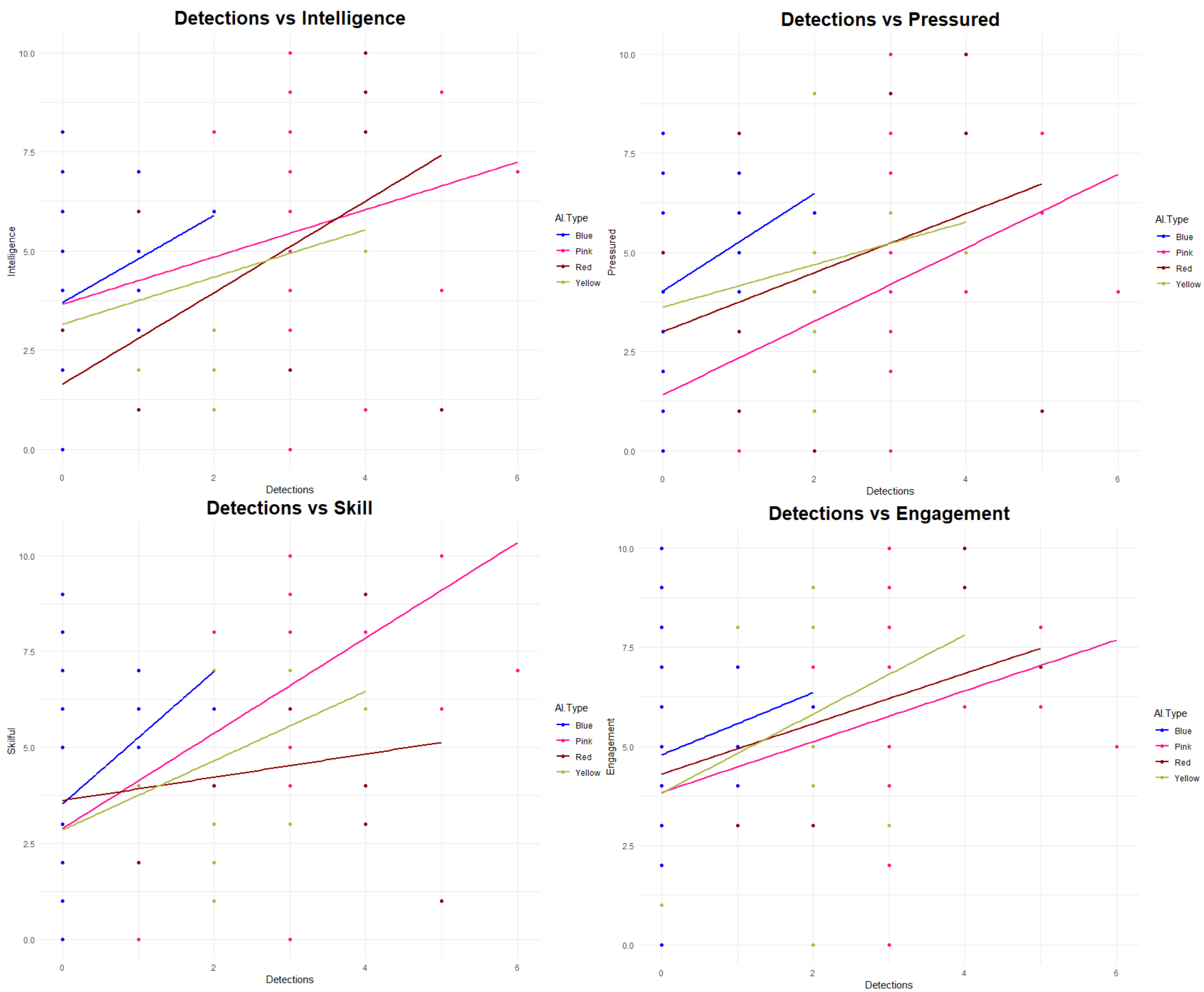


Fig. 12. Four scatter graphs comparing Detection to likert scale results

still followed. Users were given a unique personal ID in the information sheet, allowing their data to be removed at any time. The data collected was stored in OneDrive which is a safe and secure data storage location.

IX. Further Work

The computing artefact present in this paper, whilst it has been sufficient for investigating the research question and answering to hypotheses laid out. More informative data points could be created, in order to prevent 'Garbage in, garbage out' [], allowing for more in-depth analysis. As stated in VII-B the artefact was constructed in a way which created levels of bias in testing. Creating unique levels for each individual AI variant keep the results isolated and the experience more fresh. Alternatively, instead of participants playing each AI variant, a participant would play one AI.

Expanding upon the AI models presented in this research would allow for a greater insight into player perception into AI. Since the AI models all use the same behaviour tree, tests could be conducted between more advanced algorithms which act differently. Then see players can reliably identify which is the 'smartest' AI.

X. Conclusion

In conclusion, this paper gives an insight into player perception, player psychology and game AI. In order to find techniques game developers can use to make more engaging AIs, this dissertation proposes that players cannot identify the weakest or strongest AI enemies. Four AI variants were created each using the same behaviour tree, but unique stat values. These AI variants were placed into a first-person stealth game.

22 participants took part, collecting data using in game quantitative gathering and a questionnaire taking qualitative opinions. The results from the collected data partially support

the hypotheses particularly I & II. Showing that participants were unreliable in determining the different AIs ability levels. However, the limitations in this study reduce the strength of these findings, therefore the data cannot refute any of the null hypotheses. Despite the lack of data from the study, important points can still be drawn, showing the relationship between difficulty AI and player attitudes. Of which have many applications in video games, helping developers to understand what a player will enjoy.

Acknowledgments

Many thanks to Falmouth University, Dr Michael Scott, Victoria Rees and Mr Matt Watkins for their feedback and assistance in writing this dissertation. To those who played through the computing artefact, thank you for your data!

References

[1] E. F. Grant and R. Lardner, "It," *The New Yorker*, July 1952.
[2] *GDC 2005 Proceeding: Handling Complexity in the Halo 2 AI*, Mar. 2005.
[3] G. N. Yannakakis, "Game ai revisited," 2014.
[4] Y. Sekhavat, "Behavior Trees for Computer Games," *International Journal on Artificial Intelligence Tools*, vol. 26, Jan. 2017.
[5] C. Martens, E. Butler, and J. C. Osborn, "A Resourceful Reframing of Behavior Trees," tech. rep., Mar. 2018. arXiv:1803.09099 [cs] type: article.
[6] J. Švelch, "Should the Monster Play Fair?: Reception of Artificial Intelligence in Alien: Isolation," *Game Studies*, vol. 20, June 2020.
[7] M. Booth, "The ai systems of left 4 dead," Valve, Games Developer Conference, 2009. https://www.gdcvault.com/play/1422/From-COUNTER-STRIKE-to-LEFT PDF can be found here https://steamcdn-a.akamaihd.net/apps/valve/2009/ai_systems_of_l4d_mike_booth.pdf.
[8] M. Filer, "Open cage: Modding toolkit," *OpenCAGE - Alien: Isolation Mod Tools*, 2017.
[9] M. Productions, "F.e.a.r: First encounter assault recon," Sierra Entertainment, Inc, 2005.
[10] A. M. TURING, "I.—COMPUTING MACHINERY AND INTELLIGENCE," *Mind*, vol. LIX, pp. 433–460, Oct. 1950.
[11] J. R. Searle, "Minds, brains, and programs," *Behavioral and Brain Sciences*, vol. 3, pp. 417–424, Sept. 1980.
[12] J. Hawkins and S. Blakeslee, *On Intelligence: How a New Understanding of the Brain Will Lead to the Creation of Truly Intelligent Machines*. Macmillan, Apr. 2007. Google-Books-ID: Qg2dmntfxmQC.
[13] F. Chollet, "On the Measure of Intelligence," tech. rep., Nov. 2019. arXiv:1911.01547 [cs] type: article.
[14] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, May 1996.
[15] T. Froihofer, "Input Icons for TMPro | GUI Tools | Unity Asset Store," 2019. https://assetstore.unity.com/packages/tools/gui/input-icons-for-tmpro-213736.
[16] N. Mathur, "Building your own Basic Behavior tree in Unity [Tutorial]," *Packt Hub*, Oct. 2018.
[17] J. L. Andreassi, *Psychophysiology: Human Behavior & Physiological Response*. Psychology Press, July 2013. Google-Books-ID: 4Vc-nAAAAQBAJ.
[18] W. IJsselsteijn, Y. de Kort, and K. Poels, *The Game Experience Questionnaire*. Eindhoven: Technische Universiteit Eindhoven, 2013.
[19] M. Saunders, P. Lewis, and A. Thornhill, *Research Methods for Business Students*. Pearson Education, 2009.
[20] D. L. Morgan, "Pragmatism as a Paradigm for Social Research," *Qualitative Inquiry*, vol. 20, pp. 1045–1053, Oct. 2014.
[21] S. Y. Y. Chyung, K. Roberts, I. Swanson, and A. Hankinson, "Evidence-Based Survey Design: The Use of a Midpoint on the Likert Scale," *Performance Improvement*, vol. 56, no. 10, pp. 15–23, 2017.
[22] U. Düsseldorf:, "G*Power statistical power analyses," *General-Psychology-and-Work-Psychology*, 2007.
[23] A. Shorten and C. Moorley, "Selecting the sample," *Evidence-Based Nursing*, vol. 17, pp. 32–33, Apr. 2014.
[24] W. M. Association, "Declaration of helsinki," *Ethical Principles for Medical Research Involving Human Subjects*, 1967. https://www.wma.net/policies-post/wma-declaration-of-helsinki-ethical-principles-for-medical-research-involving-human-
[25] F. University, *Research Integrity and Ethics Handbook*, Dec. 2019. https://www.falmouth.ac.uk/sites/default/files/media/downloads/handbook_for_research_integrity_and_ethics.pdf.
[26] B. C. Society, *BCS-Code-Of-Conduct*, 2022. https://www.bcs.org/media/2211/bcs-code-of-conduct.pdf.
[27] EuropeanUnion, "General Data Protection Regulation (GDPR) – Official Legal Text," 2016.

Appendix A
Reflective Addendum

This research project has been one of the most challenging things I have done. Academic research is a serious undertaking that requires the upmost care and consideration of every aspect. Throughout the duration of this dissertation I have learnt many new skills and improving my programming, writing and analytical ability. Below are some of the difficulties I faced and what I learnt:

1) **Work Ethic:** I have found it difficult to spend time working on this module in comparison to the group team project. It's where excel working with other people to create something I'm passionate about. Therefore, I have always found it more gratifying to work on the team projects. Team members give immediate praise and approval. Whereas the dissertation is a slow burner, an intimidating project that I can sideline since I'm afraid of not knowing what to do.
2) **Lack of planning:** Due to problems with my work ethic above, the amount of time invested into this module is significantly lower than what it should have been. Resulting in hastily crafted plans, my research question was almost a spontaneous decision, time was dedicated to it but nothing was ever formed in advance. This led to further problems as I would be too ambitious and over scope, causing me to have to downscale my proposal.
3) **Quality Assurance:** I did not pay as much attention as I should have to key lectures about different topics, quality assurance was one of these topics. Naturally throughout the development of my artefact I tested, fixes bugs and refactored. However, I was unaware of the need to properly record it.
4) **Data Collection:** I am reserved and quiet person, who struggles talking to strangers. Combined with my late artefact deployment rate, data collection only began 2 weeks before the deadline. As a result I only got 22 responses, the majority were from friends online.

Appendix B
R Code

```r

library(ggplot2)
library(reshape2)
library(ggpubr)
library(questionr)

# Import the data set into R
# Quantitative Data
userStats = read.csv('DissertationData.csv')
```

Not-so-SMART- objective 1.A: Have a balanced split of work towards modules

Key Component	**Objective**
Specific - What is the specific task?	Create a work schedule to ensure time is well-balanced and follow it.
Measurable - What are the standards or parameters?	I will use a time tracker app to see if I'm sticking to the schedule.
Achievable - Is the task feasible?	Yes, A timetable schedule combined with a time tracker is clearly defined.
Realistic - Are sufficient resources available?	Schedule creators/planners are easily accessible on the internet.
Time-Bound - What are the start and end dates?	Week commencing the 17th of April until Games Academy expo (May 23rd).

SMART Objective 1.A: From the 17th of April to May 23rd. I will create a schedule, to organise my work and free time. I will follow it with assistance of a time tracker, reviewing my progress every 2 weeks

Not-so-SMART- objective 2.A: Analyse the marking rubric ahead of time and construct a timeline for tasks

Key Component	**Objective**
Specific - What is the specific task?	Construct a waterfall plan for the overall project, by looking at the rubric. With deadlines for specific tasks.
Measurable - What are the standards or parameters?	Following the deadlines set in the waterfall plan, adjusting if necessary.
Achievable - Is the task feasible?	Yes, the plan initially might take a lot of time but is well worth it.
Realistic - Are sufficient resources available?	Yes, planning out step by step should ensure a well constructed plan.
Time-Bound - What are the start and end dates?	Will start immediately and go until the end of this year, reviewing after each deadline and overall at the end of the year

SMART Objective 2.A: At the start of a new project, take the time to construct a plan with deadlines to keep on track

Not-so-SMART- objective 3.A:

Key Component	**Objective**
Specific - What is the specific task?	Take notes during key information briefings.
Measurable - What are the standards or parameters?	Ensure notes are taken for every big learning opportunity or key information.
Achievable - Is the task feasible?	Only requires writing a brief set of notes.
Realistic - Are sufficient resources available?	Pen and paper is only thing needed.
Time-Bound - What are the start and end dates?	From the 17th of April to May 23rd, reviewing notes every week.

SMART Objective 3.A: From the 17th of April to May 23rd, during all meetings I will write a set of notes on key facts

Not-so-SMART- objective 4.A: Improve confidence and ability to talk to strangers

Key Component	**Objective**
Specific - What is the specific task?	Have the confidence to ask someone to particpate in a study.
Measurable - What are the standards or parameters?	Measure the amount of new people I have interacted with and approached myself.
Achievable - Is the task feasible?	Yes, it is just talking to people.
Realistic - Are sufficient resources available?	There are many programs and videos online which assist with building confidence.
Time-Bound - What are the start and end dates?	Start of April to the end of the year. Reviewing amount of new people talking to each month.

SMART Objective 4.A: Starting April, reviewing each month make an effect to start conversations and interact with people I do not know.

```r
testData = read.csv('TestEntry.csv') # Stacked version, where Detection data is put on top of each other and separated with AI Colour

# Qualitative Data
survey = read.csv('SurveyData.csv')
combinedSurvey = read.csv('SurveyCombined.csv')

# Split the quantitative data by AI variant using column indexes
redData <- userStats[3:9]
yellowData <- userStats[10:16]
pinkData <- userStats[17:23]
blueData <- userStats[24:30]

# Set up data frames for ease of use

# Detections data frame
detections <- data.frame(Red = c(redData[2]), Yellow = c(yellowData[2]), Pink = c(pinkData[2]), Blue = c(blueData[2]))
colnames(detections)[1] = "Red"
colnames(detections)[2] = "Yellow"
colnames(detections)[3] = "Pink"
colnames(detections)[4] = "Blue"

# Time in cone data frame
cone <- data.frame(Red = c(redData[3]), Yellow = c(yellowData[3]), Pink = c(pinkData[3]), Blue = c(blueData[3]))
colnames(cone)[1] = "Red"
colnames(cone)[2] = "Yellow"
colnames(cone)[3] = "Pink"
colnames(cone)[4] = "Blue"

# Total level completion time
completionTime <- data.frame(Red = c(redData[4]), Yellow = c(yellowData[4]), Pink = c(pinkData[4]), Blue = c(blueData[4]))
colnames(completionTime)[1] = "Red"
colnames(completionTime)[2] = "Yellow"
colnames(completionTime)[3] = "Pink"
colnames(completionTime)[4] = "Blue"

# Distance covered
distance <- data.frame(Red = c(redData[5]), Yellow = c(yellowData[5]), Pink = c(pinkData[5]), Blue = c(blueData[5]))
colnames(completionTime)[1] = "Red"
colnames(completionTime)[2] = "Yellow"
colnames(completionTime)[3] = "Pink"
colnames(completionTime)[4] = "Blue"


# Collectables collected
distance <- data.frame(Red = c(redData[6]), Yellow = c(yellowData[6]), Pink = c(pinkData[6]), Blue = c(blueData[6]))
colnames(completionTime)[1] = "Red"
colnames(completionTime)[2] = "Yellow"
colnames(completionTime)[3] = "Pink"
colnames(completionTime)[4] = "Blue"

### Scatter Plots ###

# Detection vs Distance Traveled

cor(testData$Detections, testData$Distance.Travelled)
```

```
detectVdis <- ggplot(testData, aes(Detections, Distance.Travelled, colour = AI.Type)) +
    geom_point() +
    scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
    geom_smooth(method=’lm’, se = FALSE) +
    theme_minimal() +
    labs(x=’Detections’, y=’Distance Travelled’, title=’Detections vs Distance Travelled’) +
    theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

# Detection vs Time Taken

cor(testData$Detections, testData$Time.Taken)

detectVdis <- ggplot(testData, aes(Time.Taken, Detections, colour = AI.Type)) +
geom_point() +
scale_x_continuous(breaks = c(0, 30, 60, 90, 120, 150, 180)) +
geom_smooth(method=’lm’, se = FALSE) +
scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
theme_minimal() +
labs(x=’Time Taken’, y=’Detections’, title=’Detections vs Time Taken’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

# Detections against Engagement

cor(testData$Detections, combinedSurvey$Engaged)

detectVdis <- ggplot(combinedSurvey, aes(testData$Detections, Engaged, colour = AI.Type)) +
geom_point() +
geom_smooth(method=’lm’, se = FALSE) +
theme_minimal() +
scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
labs(x=’Detections’, y=’Engagement’, title=’Detections vs Engagement’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

# Detections against Intelligence

cor(testData$Detections, combinedSurvey$Intelligent)

detectVdis <- ggplot(combinedSurvey, aes(testData$Detections, Intelligent, colour = AI.Type)) +
geom_point() +
geom_smooth(method=’lm’, se = FALSE) +
theme_minimal() +
scale_color_manual(values = c(”blue”,’deeppink’,”darkred”, ”#B1B943”)) +
labs(x=’Detections’, y=’Intelligence’, title=’Detections vs Intelligence’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

# Detections against Skill

cor(testData$Detections, combinedSurvey$Skilful)

detectVdis <- ggplot(combinedSurvey, aes(testData$Detections, Skilful, colour = AI.Type)) +
geom_point() +
geom_smooth(method=’lm’, se = FALSE) +
theme_minimal() +
scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
labs(x=’Detections’, y=’Skilful’, title=’Detections vs Skill’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

# Detections against Pressured

cor(testData$Detections, combinedSurvey$Pressured)

detectVdis <- ggplot(combinedSurvey, aes(testData$Detections, Pressured, colour = AI.Type)) +
geom_point() +
geom_smooth(method=’lm’, se = FALSE) +
scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
theme_minimal() +
labs(x=’Detections’, y=’Pressured’, title=’Detections vs Pressured’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

### Time in cone

cor(testData$Cone.Detection.Time, combinedSurvey$Pressured)

detectVdis <- ggplot(combinedSurvey, aes(testData$Cone.Detection.Time, Pressured, colour = AI.Type)) +
geom_point() +
geom_smooth(method=’lm’, se = FALSE) +
scale_color_manual(values = c(’blue’,’deeppink’,”darkred”, ”#B1B943”)) +
theme_minimal() +
labs(x=’Time in detection cone’, y=’Pressured’, title=’Time in cone vs Pressured’) +
theme(plot.title = element_text(hjust=0.5, size=20, face=’bold’))
print(detectVdis)

### Bar Charts ###

p<-ggplot(data=testData, aes(x=AI.Type, y=Time.Taken)) +
geom_bar(stat=”identity”, fill = testData$AI.Type)
print(p)

p<-ggplot(data=testData, aes(x=AI.Type, y=Time.Taken)) +
geom_bar(stat=”identity”, fill = testData$AI.Type)
print(p)

completionTime = ggplot(testData, aes(x = AI.Type, y = Time.Taken), fill = ”AI.Type”) +
geom_boxplot()
print(completionTime)

### Spearman’s R Correlation Test ###

# Extremely low p value
skilful <- cor.test(combinedSurvey$Skilful, testData$Detections, method = ”spearman”, exact = FALSE)

# Very low p value
intelligence <- cor.test(combinedSurvey$Intelligent, testData$Detections, method = ”spearman”, exact = FALSE)

# Moderate p value
pressure <- cor.test(combinedSurvey$Pressured, testData$Detections, method = ”spearman”, exact = FALSE)

# Low p value
engaged <- cor.test(combinedSurvey$Engaged, testData$Detections, method = ”spearman”, exact = FALSE)

### Pie Charts
df <- data.frame(Challenging =c(survey$Which.AI.felt.the.most.challenging),
                 Intelligent = c(survey$Which.AI.felt.the.most.intelligent),
                 Easiest = c(survey$Which.colour.AI.felt.the.easiest),
                 Favourite = c(survey$Which.AI.was.your.favourite.to.play.against.))

# Easiest AI
easiestData <- split(df, f = df$Easiest)

slices <- c(nrow(easiestData$‘Red AI‘), nrow(easiestData$‘Yellow AI‘),nrow(easiestData$‘Blue AI‘))
```

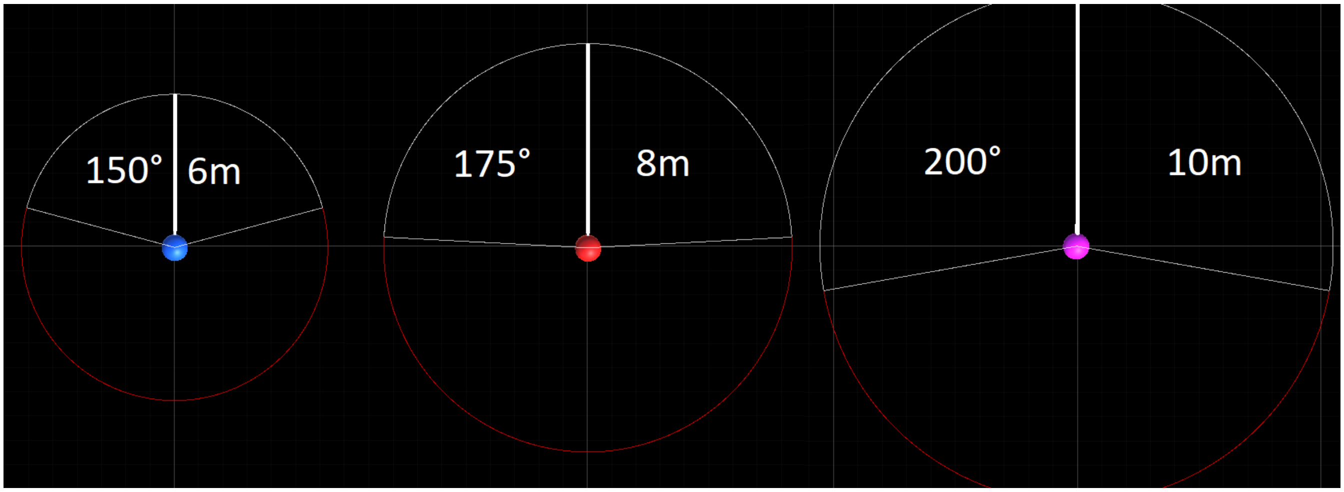


Fig. 13. FOV sizes shown, from blue to pink (yellow is not shown as it is the same as red)

```
lbls <- c(”Red”, ”Yellow”, ”Blue”)
pie(slices, labels = lbls, main=”Which AI felt the easiest?”,
     col=c(”Red”, ”Yellow”, ”Light Blue”))

# Challenge
splitData <- split(df, f = df$Challenging)

slices <- c(nrow(splitData$‘Red AI‘), nrow(splitData$‘Yellow AI‘),
       nrow(splitData$‘Pink AI‘))
lbls <- c(”Red”, ”Yellow”, ”Pink”)
pie(slices, labels = lbls, main=”Which AI did you find the most
       challenging?”,
     col=c(”Red”, ”Yellow”, ”Pink”))

#Most intelligent
intelligentData <- split(df, f = df$Intelligent)

slices <- c(nrow(intelligentData$‘Red AI‘), nrow(intelligentData$‘
       Yellow AI‘),nrow(intelligentData$‘Pink AI‘), nrow(intelligentData
       $None))
lbls <- c(”Red”, ”Yellow”, ”Pink”, ”None”)
pie(slices, labels = lbls, main=”Which AI did you find the most
       intelligent?”,
     col=c(”Red”, ”Yellow”, ”Pink”, ”Grey”))

#Favourite AI
favouriteData <- split(df, f = df$Favourite)

slices <- c(nrow(favouriteData$‘Red‘), nrow(favouriteData$‘Yellow‘),
       nrow(favouriteData$‘Pink‘), nrow(favouriteData$None), nrow(
       favouriteData$‘Blue‘))
lbls <- c(”Red”, ”Yellow”, ”Pink”, ”None”, ”Blue”)
pie(slices, labels = lbls, main=”Which AI was your favourite to play
       against?”,
     col=c(”Red”, ”Yellow”, ”Pink”, ”Grey”, ”Light Blue”))
```

.

Appendix C
Repository

Link to GitHub repository is https://github.falmouth.ac.uk/Games-Academy-Student-Work-22-23/TB246112-COMP320-Dissertation-Prototype

Test Case	Description	Test Step	Expected Result	Status
Consent	Player should only be able to proceed if all tick boxes are on 'Yes'	Check Toggle state is set to 'Yes'	Player is allowed through when 'Yes' has been ticked	Pass Or Fail
Movement	Player can move with WASD and can move camera	Check movement inputs are being picked up	Player is able to move around and use camera	Pass Or Fail
Functionality	UI buttons are functional and work	Have user go through and click if buttons respond	Buttons work and player will be able to complete game	Pass Or Fail
Loading	Level transitions work when you finish a level	Trigger the LoadLevel script	Selected level is loaded	Pass Or Fail
DataSaving	Data is written to the assigned CSV	Run the SaveToCSV function	Data appears in the CSV file	Pass Or Fail

TABLE II
UNIT TEST TABLE

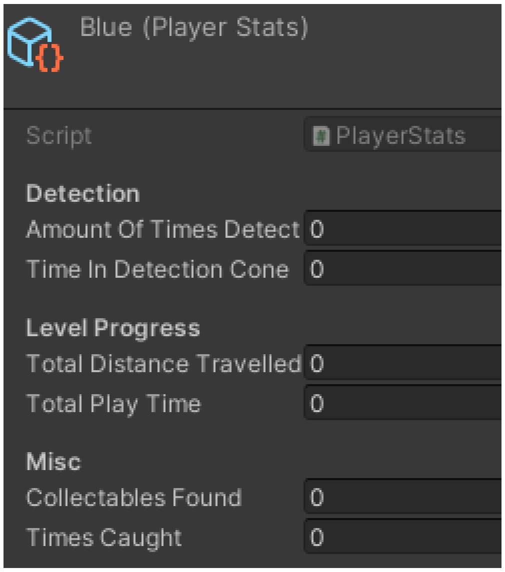


Fig. 14. Variables stored inside PlayerStats

FALMOUTH
UNIVERSITY

Research Information

Author: Thomas Billett
tb246112@falmouth.ac.uk
Supervisor: Matt Watkins
matt.watkins@falmouth.ac.uk

Introduction

In this research I am investigating artificial intelligence in video games, and I need participants to give their feedback on the systems that I have implemented.

The data gathered will be analysed to help game developers understand what influences player thoughts.

Please read through the following information carefully.

Data Collected

Game statistics: Information on how you perform in the game, such as: how many times did the player get detected. How long did it take for them to complete the level.

Questionnaire: After playing through the game, you will be asked some questions about your feelings and experience.

No personal data will be taken in this research. Though if you want to remove your data from the research at any time. Please contact the email above, along with your ID number. Which is **18fcd446**.

Study Information

The test is a 4 short game levels, where you need to sneak past enemies to reach safety. There are optional collectables. You will be presented with the following.

Consent: You will be required to fill out the consent form on the next page to start the study.

Tutorial: A short tutorial will show you the controls and mechanics of the game.

Level Gameplay: You will be tasked to complete 4 levels, navigating from start to end.

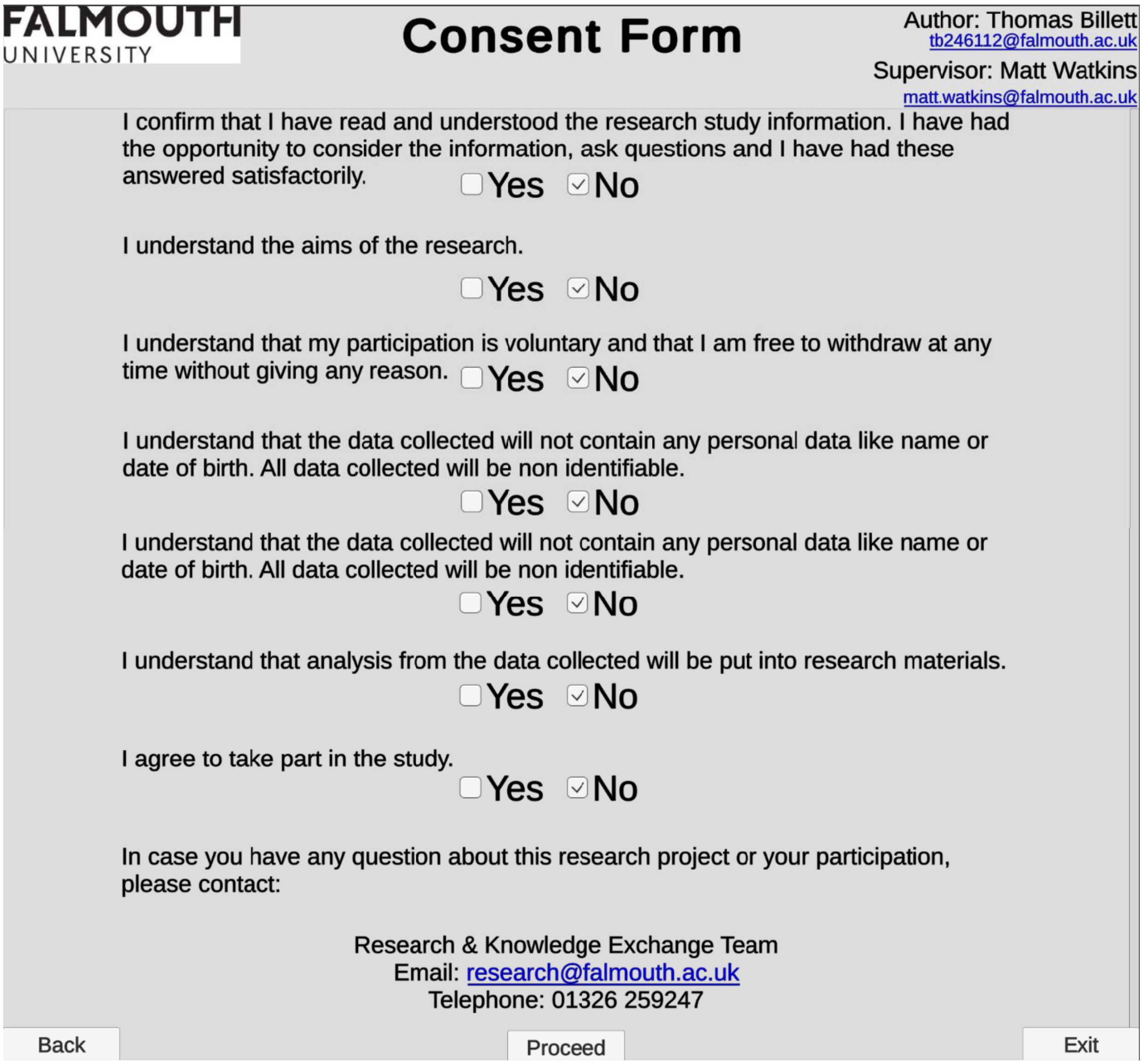
Questionnaire: You will be asked about your thoughts about the artificial intelligence.

The test should take around 15 minutes, each level is about 2 - 3 minutes. With a 5 minute questionnaire at the end. Taking breaks between levels is advised.

Thank you for participating in this research. If you have any further questions, contact me at my email at the top right of the screen.

Continue Exit

Fig. 15. Information sheet shown to participants



FALMOUTH UNIVERSITY

Consent Form

Author: Thomas Billett
tb246112@falmouth.ac.uk
Supervisor: Matt Watkins
matt.watkins@falmouth.ac.uk

I confirm that I have read and understood the research study information. I have had the opportunity to consider the information, ask questions and I have had these answered satisfactorily. ☐ Yes ☑ No

I understand the aims of the research.

☐ Yes ☑ No

I understand that my participation is voluntary and that I am free to withdraw at any time without giving any reason. ☐ Yes ☑ No

I understand that the data collected will not contain any personal data like name or date of birth. All data collected will be non identifiable.

☐ Yes ☑ No

I understand that the data collected will not contain any personal data like name or date of birth. All data collected will be non identifiable.

☐ Yes ☑ No

I understand that analysis from the data collected will be put into research materials.

☐ Yes ☑ No

I agree to take part in the study.

☐ Yes ☑ No

In case you have any question about this research project or your participation, please contact:

Research & Knowledge Exchange Team
Email: research@falmouth.ac.uk
Telephone: 01326 259247

Back Proceed Exit

Fig. 16. Conform form shown to participants

Fig. 17. Questionnaire at the end of the game (missing pink & blue slider section)