Machine Learning Automation for Virtual Reality

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at George Mason University

By

Erdem Murat
Bachelor of Science
George Mason University, 2021

Director: Dr. Lap-Fai Yu, Professor
Department of Computer Science

Fall Semester 2022
George Mason University
Fairfax, VA

Table of Contents

Page

List of Tables v
List of Figures vi
Abstract vii
1 Introduction 1
1.1 Background 1
1.2 Development Issues with the State-of-the-Art 1
1.3 User Experiences with Today's VR Games 2
1.4 Proposal 3
2 System Overview 5
2.1 The Game 5
2.1.1 Game Difficulty Manipulation 5
2.2 Data Collection Methodology 7
2.3 Additional Data Collected 8
3 Machine Learning 9
3.1 Goal 9
3.2 Training and Testing Data 9
3.3 Best Algorithm and Prediction Error 10
4 Discussion 13
4.1 Why Use Machine Learning? 13
4.2 Prediction Error 13
4.2.1 Comparison of Rating Percentages and Prediction 14
4.3 Explanation for Randomness and Outliers 15
5 Limitations and Future Work 19
5.1 Main Limitation 19
5.2 Contribution 19
5.3 Expansion of Data 20
5.3.1 Fun and Stress Rating 20
5.3.2 E4 Wristband Medical Data 20
5.3.3 Positional, Perspective and other Gameplay Data 22

5.4 Optimization, Automation, and Generation of New Levels 23

6 Conclusion 26

List of Tables

Table	Page
2.1 Example Rank and Adjustment of Parameters	7
3.1 Example Training Data	10

List of Figures

Figure		Page
2.1	Annotated Birds Eye View of Game Scene	6
2.2	Game view from POV of player	6
3.1	Rating Percentage Prediction of ML Algorithms	11
3.2	Rating Percentage Prediction in-comparison to user results	12
4.1	Prediction Error Distribution Amongst Users	14
4.2	ML Prediction and Rating Percentages of 4 Users with prediction error above plot.	17
4.3	H1 and H2 User Rating Percentages	18
5.1	Example data (EDA) collected by E4 Wristband during gameplay	22
5.2	Intermediate Levels and Predictions in Comparison to Default Game Levels	24

Abstract

MACHINE LEARNING AUTOMATION FOR VIRTUAL REALITY

Erdem Murat

George Mason University, 2022

Thesis Director: Dr. Lap-Fai Yu

Virtual Reality (VR) game development techniques are relatively new in relation to conventional 2-dimensional (2D) content. Although there has been significant research conducted in this new field, more work is still needed as there are still some prevalent issues. A significant issue reported by some users is that the perceived difficulty of a game can vary drastically between users. This is because the nature of VR gives more autonomy to users and lets them play games differently than the developer might've intended. To address this, I have proposed a system that tracks user difficulty perception on the manipulation of various game parameters that affect difficulty. The collected user data is used to train a machine learning regressor to predict the perceived difficulty of different game levels. The initial findings show a 53% prediction error. However, further analysis has shown that the predictions are realistic and adequate. Anomalies in prediction are explainable and prediction error can be reduced to 26% through the removal of some outliers. Limitations of this work, like the limited dataset size, are also addressed for future work to improve accuracy and performance. This thesis was primarily written with future work in mind, as the addressed problem is complex and requires further examination for a final and applicable model. The final model proposed uses MCMC optimization and is aimed at automating optimization of game parameters to tailor experiences to intended difficulty and/or emotions. Thus, the main contribution of this paper

is its address of an insufficiently covered issue by producing a key approach and proposing detailed suggestions for future research.

Chapter 1: Introduction

1.1 Background

The recent emergence of VR and the Metaverse [1] in the consumer field has brought a new range of content development considerations with it [2]. This is because the traditional consumption of content has never been as immersive. Thus, the developer community that transitioned to this field adapted their earlier visual simulations to VR through game development research [3]. This research was influenced by traditional game development techniques and focused on improving usability and user experience. Researchers were able to present a systematic approach to VR game development [4–6]. As such, VR games are developed with things like user comfort in mind, as movement mechanics were needed to be carefully designed to decrease motion sickness and increase immersive interactions with regards to the typical user play-space [7, 8]. Another consideration that came out of this research was the ability to adapt the previous user interfaces to VR. These interfaces were designed with the goal of replacing keyboards, mouses, and gamepads with VR controllers that work more naturally for the virtual environment. Gameplay and difficulty were also addressed and adjusted to VR mechanics.

1.2 Development Issues with the State-of-the-Art

While there was a variety of useful outcomes from VR game design research, one of the issues about the current state-of-the-art is that some previous techniques that have been adapted to VR might not be well suited. One reason for this is that VR game mechanics are vastly different and this affects difficulty. Developers of non-VR video games are able to manipulate gameplay in many subtle ways that simply are not an option for VR. For example, in non-VR games, developers can move the camera angle to make the player focus on a part of the game that is required to proceed to the next level. This forces the player to interact with game elements in a way that the developer intends to dictate the gameplay. However, this is not an option in VR as the entire view and interaction system is in the control of the user. Thus, it is not a viable option to force the player to look or move in certain directions as such unnatural and forced movement is likely to cause motion sickness which will ruin gameplay [8]. Furthermore, the inability to dictate gameplay often leads to players missing critical parts of the objective, making the completion of tasks/levels much harder.

1.3 User Experiences with Today's VR Games

Many game studios are entering the VR market to expand their most famous titles to the new medium that is VR [9, 10]. Some modders[1] are also making a living out of modding PC games to support VR [11]. This demand for VR content is driving the VR market [12] as consumers show attention to their favorite games gaining VR support.

Unfortunately, the user experiences of these games can sometimes be mixed as the problems discussed in section 1.2 are prevalent. For example, a user on an online blog [13] noted that they found such issues in the VR game Resident Evil 4:

> "I was playing RE4 for the first time in VR and was enjoying the heck out of it. [I] was playing it on normal difficulty but somewhere around chapter 3 it just became too difficult. ... Now on [the] easy difficulty the very first time I enter the village I get dynamites thrown at me plus the chainsaw guy shows up who did not even show up just a few chapters later on normal difficulty. And it was supposed to be easy."

Following this post, a user commented that the issue is likely because the player missed picking up an item that would make gameplay easier. As discussed previously, VR games provide more autonomy to the player and less dictation for the developer. This can result in users being unable to pick up on clues about how the developer wants the game to flow. In this situation, the user tried to power through this problem that they as an end-user were clueless about. Not giving the player options and taking this one-dimensional development practice is an issue that is likely to make

[1]A modder is someone who modifies games to add new features to them. In this case, modders add VR functionalities to games.

the user eventually never visit the game. A solution to this would be if the developer had a system implemented that would account for such missing parameters and automate difficulty. Similar issues for other users persisted as they commented [14–16] on other games:

> "I have been playing Stride VR and I've come across an issue ... In the Endless mode, when I play on harmless or easy, it changes difficulty instantly to expert after about 200 points. Which is extremely sudden [at] the very beginning of the run."

> "This is a problem a lot of VR games have, even the major releases like Bonelab. Too many VR games feel too easy. The only two exceptions I've found so far are Thrill Of The Fight and Until You Fall."

> "The game was built around the special movement system featured in the flat screen version, but I'm finding the increased movement and weapon speed and ability to do multiple things at once [are] really smashing any difficulty the game had."

As can be seen from the reviews on various games, some users experience problems with difficulty in VR. The issue can sometimes be that the game is too easy, or that it is too hard, and that the game is unable to automatically adjust itself to the correct setting.

1.4 Proposal

This paper proposes a solution to the mentioned problem by creating a system that can understand the game parameters that control difficulty, the relationship between those parameters, and how their change impacts difficulty perception. The presented system works by using qualitative user data to train a machine-learning algorithm that can be used to predict the game's difficulty as perceived by users. As the problem is complex, the presented work is mostly a proof-of-concept followed by a meticulous proposal and suggestions to guide future research that can build off of the presented methodology. While there was similar research done using game parameters in level generation [17], my proposed work does so in VR and with reference to qualitative user data which differs from the previously used level generation and completion data. My proposed future work does also incorporate a similar understanding of the level generation and completion but differentiates itself by integrating a greater dimension of game parameters, gameplay data, and user data (i.e., medical data).

Chapter 2: System Overview

2.1 The Game

The VR game I developed for this research is called "Why did the chicken cross the road?". The goal of the game is to cross the street without getting hit by cars on the road. The player plays in the form-factor of a chicken, which makes environment objects like the cars and road seem larger, making it more challenging to cross the road. Cars in this game only travel in one direction (left to right). Each car in a given lane has the same speed and the speed of each lane gets slightly faster going from start to finish. While players can physically walk/move to play this game, the primary control for player movements was done using the Quest 2 controllers, given the considerable distance needed to travel in the game. The game has 5 car types; two sedans (small sized cars), one minivan (medium sized car), one bus (large car), and one truck (large car). While I did not incorporate functionality to control the frequency of big/small cars, I left car spawning to be completely random to keep a factor of randomness in the game.

2.1.1 Game Difficulty Manipulation

The difficulty of the game is determined by 5 parameters that are manipulated on each level to make the game easier or harder.

1. Number of Lanes - The number of lanes the player has to pass to complete the level.
2. Speed of Cars - The uniform speed of the cars.
3. Player Tardiness - The tardiness/slowness of the player's movement.
4. Size of Cars - An increase in car size results in the same car taking up more space in its lane.
5. Car Frequency / Number of Cars - Frequency of cars in every lane.

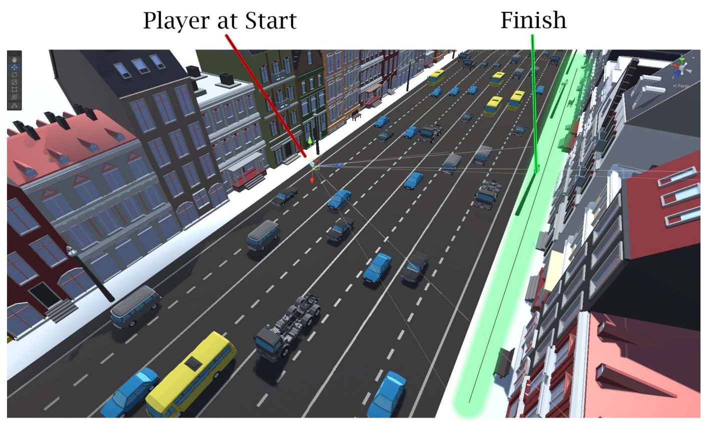

Figure 2.1: Annotated Birds Eye View of Game Scene

Figure 2.2: Game view from POV of player

The game has a base level where all parameters are set to be at their easiest. In addition to that, the 5 difficulty parameters are incrementally increased individually three times each to create $3 \times 5 = 15$ additional levels. So, the game has a total of 16 levels. The increased version of each level is referred to as a ‘rank’. To make referral to the ranks of each parameter simpler, they will be referred to as <parameter name><rank number>. Table 2.1 is an example of how the rank system works. For example, ’number of lanes 1’ would be a reference to the first rank of the number of lanes, where the number of lanes is increased by 2, for a total of 4 (all other parameters remain unchanged). ’Number of lanes 2’ would be a reference to the second rank of the number of lanes, where the number is even higher (8). A similar trend in difficulty increase is applied for all parameters individually in their own ranks. This was done to simplify the referral of ranks, from easiest to hardest.

Table 2.1: Example Rank and Adjustment of Parameters

	Base	Rank 1	Rank 2	Rank 3
No. of Lanes	4	6	8	10
Car Speed	20	24	28	32
Car Size	3	3.2	3.4	3.6

2.2 Data Collection Methodology

For data collection, I conducted 10 user-studies [1] to collect qualitative data from users playing my game. In each experiment session, users first play a test-round for approximately 5-10 minutes to get adjusted to the gameplay and difficulty, to eliminate outliers from the learning curve. Once the user completes the test levels and seems adjusted to the game, the game is restarted and the data collection begins. The game is designed to cycle through the 16 levels in a randomized order to cancel bias about difficulty and order. At the end of each level, users are prompted to rate the given level on a scale of 0-10 based on the following questions:

- How difficult was this level?
- How fun was this level?
- How scary was this level?

User responses and progress of level completion are locally saved. The user's level completion progress is also separately saved to ensure that they play each level once. Once the user has played all 16 levels, their progress recording for levels played is reset and they can play through the levels

[1] IRB Certificate for Group 1 Social & Behavioral Research has been obtained for user studies. Certificate available: https://www.citiprogram.org/verify/?k8fee8fbc-5df1-4343-9694-590dced0d37c-52588619

again if they wish. Although the game was designed to have only 16 levels played once, depending on the comfort level of the user in the game, users were allowed to continue playing and replay levels but weren't given the information that they have completed all levels. This replay of levels was done to increase the amount of data and decrease the prevalence of outliers. While the user answers ranked the game based on three questions, this thesis only focuses on their difficulty rating and saved additional data collected for future research.

2.3 Additional Data Collected

Although the data presented in this section wasn't used in this paper, it was collected to be used in future research. With this abundance of relative data, I hope to increase the dimensions at which VR user experience can be studied by expanding on the methodology presented in this paper. The possible future research that can be done using this data is reported in detail in the Limitations and Future Work chapter.

The additional data collected are as follows:

- User position, rotation, and view during gameplay (recorded every 4 frames).
- Position and type of cars (recorded every 4 frames).
- Time of start and end of level, and result (win/lose).
- E4 Medical Wristband Data during gameplay.

Chapter 3: Machine Learning

3.1 Goal

The goal of using machine learning is to try and find an algorithm that is well-suited to predict subjective qualitative data while also being trained on subjective qualitative data. The biggest challenge in this approach has been that subjective data can have a large variance between users. While it can be expected that the result will show an accurate comparison of difficulty between levels, subjective data is vulnerable to results with vast differences due to personal opinions. For example, level *number of lanes 3* may be regarded as 40% more difficult than the *number of lanes 2* level by most users, but some users might regard it as even harder with 80%. While this doesn't cause a misunderstanding of difficulty, it creates a larger prediction error. My goal is to reduce prediction errors as much as possible and present proof of the methodology so it can be used in future research with expanded parameters.

3.2 Training and Testing Data

The user studies were conducted with a total of 10 users which isn't considerably large. I used cross-validation to improve training and testing predictions. The x values of the training data consisted of rows of the numeric values of the 5 difficulty parameters of each played level. The y values were the percentage difference between the user rating to the given level and the base level (hereafter referred to as the *rating percentage*). This was done to reduce the effect of subjective data as users' overall difficulty perception or ranking scales are likely to differ. As the rankings were given on a scale of zero to ten, each rating was also increased by one to enable percentage calculation.

To set up the training data, some outliers were eliminated. Outliers were selected and eliminated with reference to it's adjacent rank. If a rating percentage for a rank-1 level is higher than the highest

Table 3.1: Example Training Data

Level	Parameters	Rating Percentage	Rating
Car Speed. 3	[4,**32**,0.1,3,85]	50%	8
No. of Lanes 1	[**6**,20,0.1,3,85]	33%	7
Car Speed 1	[4,**24**,0.1,3,85]	33%	7
Base	**[4,20,0.1,3,85]**	**0%**	**5**

rating percentage for a rank-2 level, the given rank-1 rating is regarded as an outlier and removed. If a rank-2 level rating is higher than the highest rank-3 rating percentage, or lower than the lowest rank-1 rating percentage, it is also removed. And if a rank-3 level rating is lower than the lowest rank-2 rating percentage, it is removed. Discussion on the reasoning for outlier removal can be found in section 5.1.

3.3 Best Algorithm and Prediction Error

To find the most accurate machine learning algorithm, I tested 6 different algorithms. The plots in figure 3.1 show the predictions generated by each algorithm with respect to the rank of each difficulty parameter.

To find the most accurate prediction, the prediction error of each algorithm was calculated. This was calculated by taking the difference between the user's rating percentage and the predicted rating percentage for the level. The prediction error associated with each algorithm was found by calculating the mean prediction error for each user. Following these tests, the random forest regression came out to be the algorithm with the lowest prediction error at 53%. Figure 3.2 shows the rating percentage prediction and the actual user ratings given by users. This shows that the predictions visually make sense, they increase through each rank, similarly to how user ratings increase. The plots also show that there is a large distribution amongst users which is to be expected with subjective data.

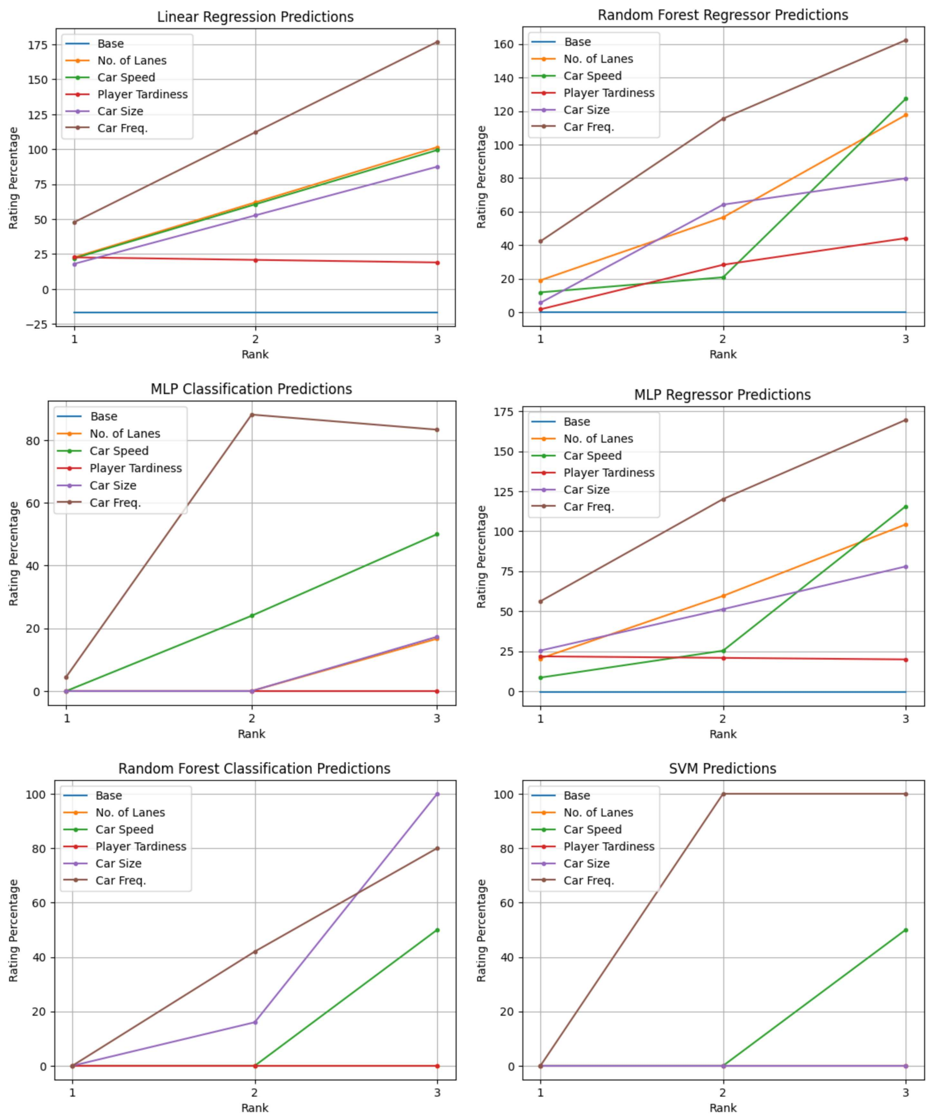

Figure 3.1: Rating Percentage Prediction of ML Algorithms
Clarification: The base level is presented for ranks 1, 2, and 3. While the base level does not have separate ranks, this was done for visualization purposes to show a comparison between all levels.

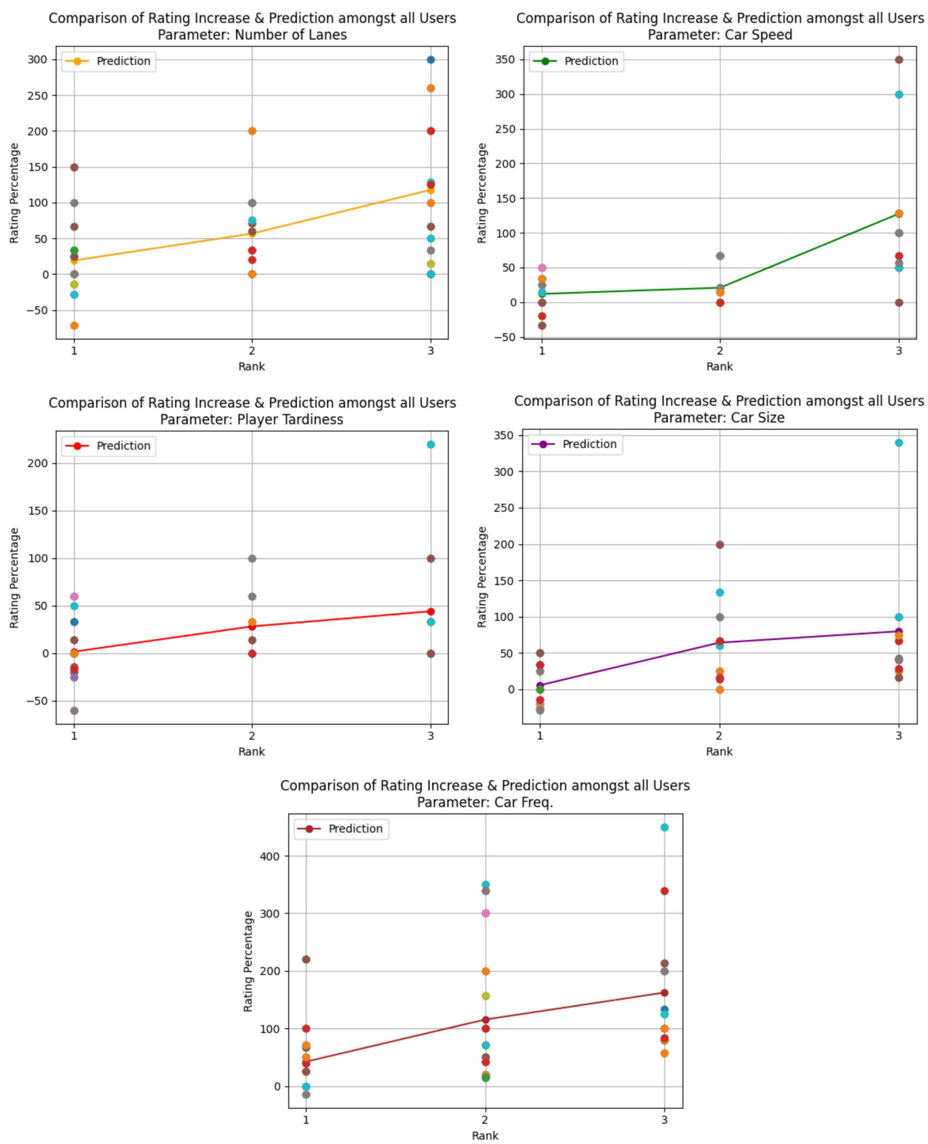

Figure 3.2: Rating Percentage Prediction in-comparison to user results

Chapter 4: Discussion

4.1 Why Use Machine Learning?

Machine learning allows for a systematic approach to understanding data and using it for prediction. This can allow for a better understanding of how different parameters affect user experience. An alternate option would be to simply use average ratings for each level but that isn't systematic and is vulnerable to outliers, especially with this limited dataset. To a certain extent, the average rating for all levels is able to show which levels are more difficult. However, it is susceptible to outliers. Machine learning is a feasible solution for future research as the dimensions of data are hoped to be increased (i.e., with health data, fun/stress rating, etc.). A further discussion on future work unlocked by machine learning can be found in chapter 5.

4.2 Prediction Error

While a prediction error of 53% might seem high, this can be considered adequate given this was done on limited and subjective user data. Visually, the random forest regression predictions also make sense. All rating percentages show an increase with each rank. The difficulty predictions also resemble the actual difficulty rating trends. For example, most user ratings also showed car frequency to be the hardest parameter, while player tardiness was the lowest in rank 3. Given other algorithms had even higher prediction errors, this suggests that the utilized machine learning algorithm was not a reason for the error. To better understand this prediction error, it would be useful to examine the different elements that produced this result.

Figure 4.2 shows the distribution of prediction errors amongst users. This figure suggests that two users were outliers as they showed an unusually high prediction error. This prompts the following questions:

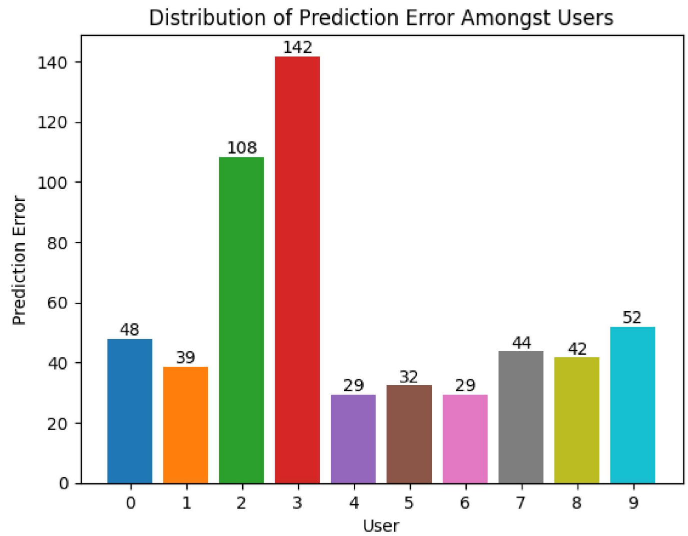

Figure 4.1: Prediction Error Distribution Amongst Users

- What are the differences in rating percentage that made them so far away from the prediction?
- How do they compare to the rest?
- What could be a solution?

4.2.1 Comparison of Rating Percentages and Prediction

To compare and visualize the differences in ratings, I have plotted the rating percentages of the two players with the highest prediction error (players 3 and 2, labeled H1 and H2 respectively), the two players with the lowest prediction error (players 6 and 4, labeled L1 and L2 respectively), and the actual prediction. Above each player's rating percentage, the difference between the rating and the corresponding prediction is listed, showing the value directly corresponding to the prediction error. This plot can be seen below in figure 4.2

Figure 4.2 shows that the user rating percentages are vastly different amongst the two groups of players. However, this can be explained by the difference in each user's difficulty perception. Figure 4.3 helps explain this. Both users H1 and H2 gave ratings that were similar in order to the prediction by the regressor, they regarded car distance as the most difficult parameter, and all difficulty parameters but one show a consistent trend in increasing difficulty. This is useful in explaining that some users perceive an increase in difficulty differently than others and that this doesn't mean their data shows a differing understanding of what is difficult, it shows that the difference is the way they view just how much difficulty is impacted. For example, all users found the number of cars 3 level to be the most difficult level, however, players L1 and L2 found it to be approximately 110% harder than the base level, while players H1 and H2 both found it to be over 300% harder. While the data of these users are still valid, their difference in difficulty perception reflects the reality that subjective data is bound to have variance uncommon to typical machine learning data. This data is still valuable in training the machine learning algorithm as including such variances will only get us a more comprehensive model. However, a larger dataset is also likely to improve the prediction error that was present as a result of these two users. Without these two users, the average prediction error would drop to 26%, which is significantly lower than the initially calculated 53%.

4.3 Explanation for Randomness and Outliers

As can be seen in previous figures, not all user ratings were uniformly increasing through each rank, and there were some outliers. For example, the visualizations in figure 4.3 shows that not all ranks were rated by users to be more difficult than the previous. While the typical trend was that higher ranks received higher ratings (as similarly reflected in the ML prediction in figure 3.1), there were still some outliers. As a result, there are three explanations that have been devised.

First, some users commented that events where they died as a result of being between two big cars made them perceive the level as being harder. As mentioned in a previous section, there is an element of randomness in many games and this was one kept for this game. As this trend was noted for multiple users during the user studies, it is likely to be the case for a portion of the outliers. For example, this might be an explanation for why user H1 rated car size 3 to be higher than car size 1, but rated car size 2 higher than both. This random event might have altered the trend of perception of increasing difficulty through ranks and caused this anomaly.

Second, the user might be getting desensitized to the difficulty of the game as they keep playing.

This is a common trend in video games as users also find a constant challenge to be more rewarding [18]. At the start of the experiment, users played a test round to reduce this feeling, but ultimately it is impossible to stop users from getting desensitized to difficulty when they play a game long enough. Looking at player H2's data, I found that out of all car speed ranks, they played car speed 2 last, which suggests that this might be why they gave this lower rating. Thus, this is another likely explanation for outliers as it is in line with previous research.

Third, some ranks do not vary enough in difficulty. As the prediction model also shows, the prediction of amongst some ranks are more subtle. This is likely because the difficulty between two levels does not vary enough to cause a vast difference in rating.

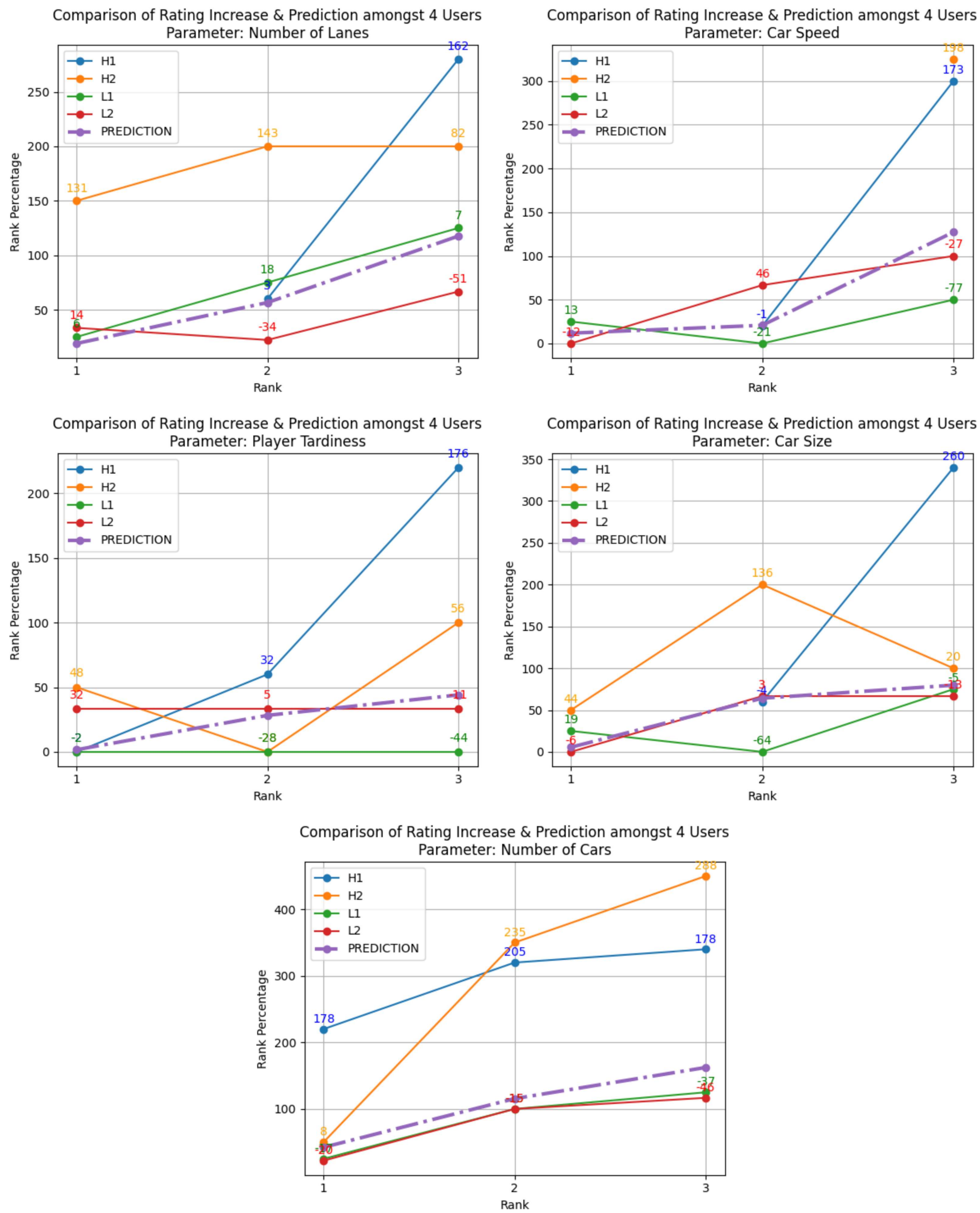

Figure 4.2: ML Prediction and Rating Percentages of 4 Users with prediction error above plot.

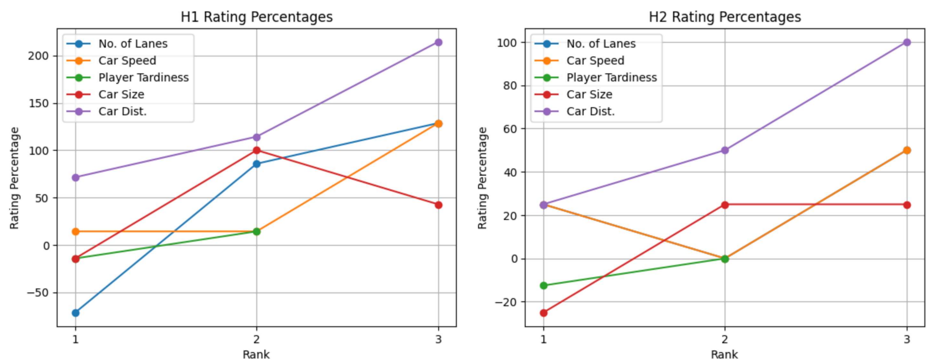

Figure 4.3: H1 and H2 User Rating Percentages

Chapter 5: Limitations and Future Work

5.1 Main Limitation

The main limitation of this paper is that it was done using a limited dataset. Because of this, outliers were able to affect my results and I had to make compromises in taking some non-methodical approaches. I recognize that I have made an assumption regarding difficulty amongst rankings. For the purposes of this thesis, I found this compromise to be acceptable as it only removes the most edge-case ratings, which skew the results the most. Approximately 90% of collected data was not removed.

5.2 Contribution

The main contribution of this paper is its address and approach to a problem in VR development that may be regarded as a setback for the industry. The approach to this flaw in design can be critical in determining the permanency of users in the VR space. As game studios and developers enter the market with new creations, it is the responsibility of researchers to discover new development techniques for the mutual benefit of the field. As such, I, the author hope that my contribution will be an adequate first address to a relatively new issue. While the work in this paper was not aimed at finding a definite answer, it was done to draw attention to a prevalent issue with a creative and methodical design to guide future solutions. The various proposed approaches are aimed at a problem that is complex. The solution requires comprehensive studies. Therefore, the main potential of this paper lies within the proposed future work using machine learning, data collection, and optimization, which, is further unlocked through this proof-of-concept.

5.3 Expansion of Data

As was stated in the earlier chapters, there was more data collected in this study than was used. This was done with the hope that this work can be expanded on in future research. All data collection and its purposes were determined through extensive research and in-depth analysis. The usage of only one collected data parameter (difficulty rating) was a purposeful approach to verify the reliability of the proposed method. Now that the proposed approach is shown to be sufficient, future research can be discussed in more detail with regard to these parameters.

5.3.1 Fun and Stress Rating

As we are trying to provide the machine learning algorithm with additional data about the user's experience during game levels, data related to emotion can assist the machine learning algorithm in gaining additional context. The fun factor is a vital part of understanding how the user views a given level. As was discussed, users find a continuous challenge to be the most fun. In this case, finding a balance that will keep the user challenged but able to complete levels is likely to keep users entertained and having fun. This will also improve the automation of levels as we will be able to better understand how the given difficulty affects the player, and which they prefer.

The stress rating is another factor that can help our automation system better understand human emotion. If the emotion of stress is better understood in relation to other factors, it can help the system in delivering experiences tailored to the trigger of specific emotions. For example, if the game wants to invoke a feeling of stress for players at an advanced level, it can pick parameters accordingly. Or, if a player is feeling too stressed and is unable to enjoy the game, the game can pick up on this and reduce the parameters that lead to this stress.

5.3.2 E4 Wristband Medical Data

While the data discussed so far consists of subjective qualitative data, is important that we attempt to incorporate quantitative user data. One of the ways we can do this is by analyzing medical data extracted from users during gameplay. As people cannot control their body's reactions, it will lead to a more objective result. As long as data is appropriately analyzed, it may provide very valuable training data. The data recorded by the E4 wristband [19] is as follows:

- Acceleration - Acceleration of wristband in 3DoF.
- PPG Sensor, Bloody Volume Pulse (BVP) - Measures change in volume of blood.
- Continuous electrodermal activity (EDA) - Change in electrical skin conductance (or sweat).
- Heart Rate - Heart beats per minute.
- Interbeat intervals (IBIs) - Intervals between heart beat.
- Temperature - Measures user body temperature.

Figure 5.1 shows the EDA data of a player as an example of medical data collected during gameplay. The timestamps of the recorded data can be aligned with gameplay data to extract measurements from the user at each gameplay instance. While this data might lead viewers to initially think that it directly portrays valuable data, this might need some manipulation to deduce a better understanding of data through each level. For example, a user's EDA might show to be high in an easy level because they had previously played a hard level. Thus, a more methodical approach like percentage calculation similar to the rating percentage can be taken to separate data on each level and make it relevant to the given level only.

Additionally, previous studies [20] have been able to use the E4 wristband and its data to train a machine-learning algorithm to detect motion sickness in VR. Another study using the same wristband studied the EDA results of users to examine emotions in a scary and immersive VR roller-coaster experience [21]. This resulted in the forming of meaningful relations between the wristband data, emotions, and events happening in the VR roller-coaster experience. These findings reaffirm the notion that medical data from wearable devices (like the E4 wristband) can be used to understand and detect user emotions in VR. Lastly, this wristband is also a useful tool to extract additional medical data. A study found that breath rate [22] can be extracted from the PPG signal. Breath rate is a vital sign that is monitored in clinical settings [23], thus, it might similarly be useful when monitoring user experiences in VR.

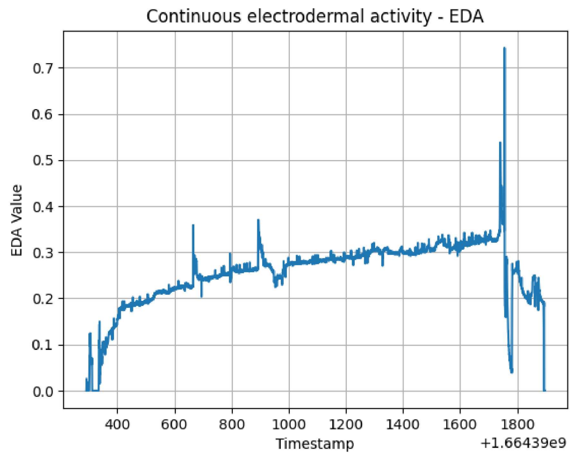

Figure 5.1: Example data (EDA) collected by E4 Wristband during gameplay

5.3.3 Positional, Perspective and other Gameplay Data

Positional and perspective gameplay data were also collected during user studies. Each user's gameplay recording had the position and rotation of the user's body (Unity body collider), headset (head), and controllers (hands) at each frame. This can provide useful information when regarding the actions of the player. It can be examined if the body position, view, or speed/approach of the player suggests comfort or ease/difficulty with a given level. The difficulty parameters, car positioning, and car type were also recorded to allow for the collection of all relevant gameplay data. This allowed me to generate a 3D playback of the game scene in Unity to visualize the gameplay of each user. Figure 2.1 is an annotated screenshot taken in this replay functionality, while the player was playing the game. Figure 2.2 is a pair of screenshots from the user's POV during the same gameplay, as the playback functionality was designed to let the user alternate between both views. Separately, timestamps of each level start, finish, and result (win/lose) are also recorded which can provide information on player performance like timing and result. Given my system saves detailed data that contains all gameplay events, it can be used to derive contextual information on each scene. For example, if a user gave an outlier rating because they were sandwiched between two large vehicles, this can be confirmed by the recorded data. If the user struggled to complete the level and took their time, this can show that. If users play better or worse when they gaze at certain parts of the environment, this can show that as well as where they looked at.

5.4 Optimization, Automation, and Generation of New Levels

Following data collection, testing and predictions, I wanted to devise a solution to a problem I faced as a researcher. I designed a system that has five parameters but only one of the parameters is manipulated at each level. This results in an optimization of difficulty accordingly to the 16 predefined levels with automation based on ranking. While this is adequate for the aim of the research conducted, it is not comprehensive enough to be a final solution for an impactful implementation. This is because parameters in many games can simultaneously and dynamically change, often irrespective of each other. Thus, a more comprehensive solution would account for multiple parameters changing at once. In the layout of this game, two parameters changing at once could be considered a new level. As there were no intermediate levels generated or tested, it lead me to question if these levels could instead be generated and predicted by machine learning.

Figure 5.2 shows the predicted rating percentage for three intermediate levels created, where only two parameters are increased, along with predictions for the 16 levels. This prediction looks reasonable as the regression model was able to predict that it would mostly be harder than the levels where only one of the parameters was increased. However, there is no way of verifying the accuracy of this as user data was not collected for the intermediate levels. The problem with this is that having all 16 levels be tested by users was time-consuming, and the creation of intermediate levels would make user testing more challenging. My solution for this in future studies is that players can play the default 16 levels and play an extra 5-10 intermediate levels. In the user studies, players replayed some levels upon completing 16 levels for the collection of additional data and to decrease the prevalence of outliers. This was a useful approach for the limited number of user data. But, in a system with more data collected, this solution can further assist the algorithm to better understand the relationship between each parameter. While there will be less data collected on intermediate levels, machine learning can help cover this gap as the model will have training on the default levels to assist it. This will be a much closer application to how difficulty parameters are adjusted in most games.

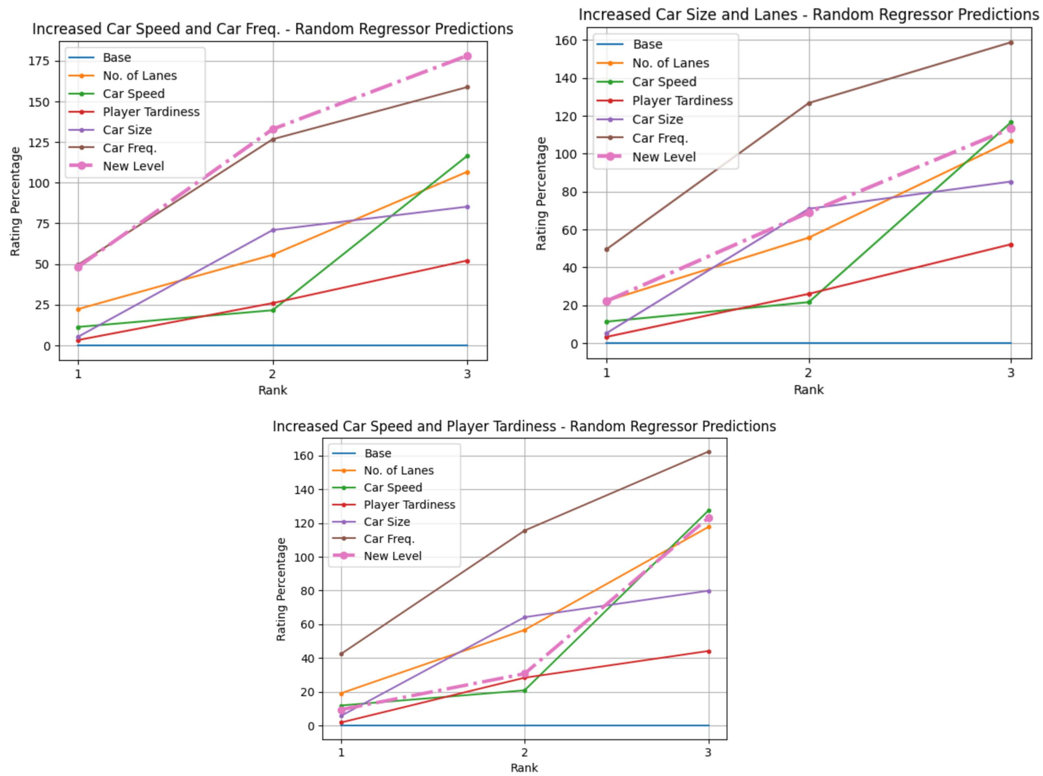

Figure 5.2: Intermediate Levels and Predictions in Comparison to Default Game Levels

For the foreseeable future research, a final model trained with adequate user data and intermediate levels could provide a system with greater freedom in parameter adjustment and depth of difficulty. As was discussed in previous sections, the main issue comes from parameters that are not in the control of the developer. To be able to create a system that can overcome this problem, automatic optimization and adjustment of individual parameters are needed. The main scenario in which this would be used is when there is a parameter that cannot be changed, and the system has to understand this and manipulate other parameters to create the intended experience, tailored to the situation. For example, if the user runs into a level where the car size is too big and can't advance to the next level, the game can't reduce the size of the cars mid-game as this would be an obvious and distracting adjustment. Instead, the game can slightly lower car speed and player tardiness to adjust the game level to the intended difficulty. The optimization to this can be found using the Monte Carlo Markov chain optimizer (MCMC) method. The MCMC optimizer is a method that takes parameters as inputs and uses an iterative formula to find the most optimal solution to balance the parameter values to achieve the intended final combined value [24]. In this case, the parameters would be the game difficulty parameters and the final combined value would be the intended level of difficulty.

Chapter 6: Conclusion

User experiences are harder to measure and predict in VR as it is a platform that is relatively new to developers and more difficult to control due to its immersive nature. To enhance user experience, a better understanding is needed, especially in scenarios that are unplanned by the developer. In this thesis, I proposed a solution that would use MCMC optimization to automate level difficulty generation using data from machine learning algorithms that use game parameters to form a prediction. While the work presented doesn't implement a definite solution, it addresses an important issue in VR development and proposes a key approach with a substantial understanding of the problem, along with valuable suggestions for future research. The work implemented uses user data from tests to train a machine-learning algorithm. This algorithm then presents a reasonable understanding of the game difficulty. The presented work has a sole reliance on subjective qualitative user data which has resulted in a model with 53% prediction error. Through discussions and further analysis, it was found that the results of the model may be considered adequate as predictions were realistic, anomalies were explainable, and prediction error could be reduced to 26% through the removal of some outliers. Additionally, an increase in the number of user data is likely to improve my results further as the size of the training data was limited. Furthermore, the incorporation of data like the 'fun' / 'stress' rating, medical wristband data, and/or contextual gameplay data is also likely to further improve the proposed model.

Bibliography

[1] L. Lee, T. Braud, P. Zhou, L. Wang, D. Xu, Z. Lin, A. Kumar, C. Bermejo, and P. Hui, "All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda," *CoRR*, vol. abs/2110.05352, 2021. [Online]. Available: https://arxiv.org/abs/2110.05352

[2] F. Pallavicini, A. Pepe, and M. E. Minissi, "Gaming in virtual reality: What changes in terms of usability, emotional response and sense of presence compared to non-immersive video games?" *Simulation & Gaming*, vol. 50, no. 2, pp. 136–159, 2019.

[3] M. Zyda, "From visual simulation to virtual reality to games," *Computer*, vol. 38, no. 9, pp. 25–32, 2005.

[4] M. Katzourin, D. Ignatoff, L. Quirk, J. J. Laviola, and O. C. Jenkins, "Swordplay: Innovating game development through vr," *IEEE Computer Graphics and Applications*, vol. 26, no. 6, pp. 15–19, 2006.

[5] J. J. LaViola Jr, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev, *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[6] M. R. Mine, F. P. Brooks, and C. H. Sequin, "Moving objects in space: Exploiting proprioception in virtual-environment interaction," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '97. USA: ACM Press/Addison-Wesley Publishing Co., 1997. [Online]. Available: https://doi.org/10.1145/258734.258747

[7] D. A. Bowman, E. Kruijff, J. J. L. Jr., and I. Poupyrev, *3D User Interfaces*. Boston: Addison-Wesley, 2005.

[8] E. Chang, H.-T. Kim, and B. Yoo, "Virtual reality sickness: A review of causes and measurements," *International Journal of Human-Computer Interaction*, vol. 36, pp. 1–25, 07 2020.

[9] H. Stockdale, "7 great non-vr games with must-play vr support," *UploadVR*, 2020. [Online]. Available: https://uploadvr.com/great-games-vr-support/

[10] G. Lees, "The best vr games on pc in 2022," *PC Games N*, 2022. [Online]. Available: https://www.pcgamesn.com/best-vr-games

[11] S. Hollister, "Meet the modder making $20,000 a month bringing the biggest games to vr," *The Verge*, 2022. [Online]. Available: https://www.theverge.com/23190201/luke-ross-vr-real-mod-gta-v-elden-ring-horizon-red-dead

[12] A. M. Research, "Global virtual reality content creation market research report: By content type (videos, 360-degree photos and games), by component (software and services), by end user (real estate, travel & hospitality, media & entertainment, healthcare, gaming, automotive and others), by region (north america, europe, asia-pacific, middle east & africa, south america) - forecast till 2030," *Allied Market Research*, 2022. [Online]. Available: https://www.marketresearchfuture.com/reports/virtual-reality-content-creation-market-9552

[13] C_G_Walker, "What's the deal with resident evil 4 vr's difficulty?" *Reddit r/Oculus*, 2021. [Online]. Available: https://www.reddit.com/r/oculus/comments/rfnmov/whats_the_deal_with_resident_evil_4_vrs_difficulty/

[14] MemeIsDrugs, "Why is difficulty changing in stride vr by itself?" *Reddit r/Oculus*, 2022. [Online]. Available: https://www.reddit.com/r/oculus/comments/s6ethg/why_is_difficulty_changing_in_stride_vr_by_itself/

[15] supercooljoe01, "Actually difficult vr games?" *Reddit r/Virtual Reality*, 2022. [Online]. Available: https://www.reddit.com/r/virtualreality/comments/y0hu1z/actually_difficult_vr_games/

[16] failatgood1, "Anyone else finding re4vr way too easy?" *Reddit r/OculusQuest*, 2021. [Online]. Available: https://www.reddit.com/r/OculusQuest/comments/qgyndy/anyone_else_finding_re4vr_way_too_easy/

[17] A. Isaksen, D. Gopstein, and A. Nealen, "Exploring game space using survival analysis," in *FDG*, 2015.

[18] A. Corcos, "Being enjoyably challenged is the key to an enjoyable gaming experience: an experimental approach in a first-person shooter game," *Socioaffective Neuroscience & Psychology*, vol. 8, no. 1, p. 1474668, 2018, pMID: 29785247. [Online]. Available: https://doi.org/10.1080/20009011.2018.1474668

[19] Empatica, "E4 sensors," *Empatica Website*, 2022. [Online]. Available: https://www.empatica.com/research/e4/

[20] N. Martin, N. Mathieu, N. Pallamin, M. Ragot, and J.-M. Diverrez, "Virtual reality sickness detection: an approach based on physiological signals and machine learning," in *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2020, pp. 387–399.

[21] M. Bastiaansen, M. Oosterholt, O. Mitas, D. Han, and X. Lub, "An emotional roller coaster: Electrophysiological evidence of emotional engagement during a roller-coaster ride with virtual reality add-on," *Journal of Hospitality & Tourism Research*, vol. 46, no. 1, pp. 29–54, 2022. [Online]. Available: https://doi.org/10.1177/1096348020944436

[22] A. Fusco, D. Locatelli, F. Onorati, G. C. Durelli, and M. D. Santambrogio, "On how to extract breathing rate from ppg signal using wearable devices," in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2015, pp. 1–4.

[23] C. Chourpiliadis and A. Bhardwaj, "Physiology, respiratory rate (pmid: 30725991)," *StatPearls*, 01 2019.

[24] M. Jerrum and A. Sinclair, *The Markov Chain Monte Carlo Method: An Approach to Approximate Counting and Integration*. USA: PWS Publishing Co., 1996, p. 482–520.

Curriculum Vitae

Erdem Murat is an international student from Ankara, Turkey, and a Computer Science Master's Student at George Mason University. He received his Bachelor of Science in Computer Science from George Mason University in 2021. He is a member of the Design Computing and eXtended Reality (DCXR) research group where he is mentored and advised by Prof. Craig Yu. He is also the Secretary and a leading member of the Computer Science Graduate Student Association (CSGSA). Outside of graduate school, Erdem is a volunteer for the Arlington-based non-profit the Global Co Lab Network as the Virtual Reality Education Director.