University of Padova

Department of Information Engineering

Master Thesis in ICT For Internet and Multimedia

VR User Identification from Movement Analysis

Supervisor
Prof. Federico Chiariotti
University of Padova

Co-supervisor
Prof. Sara Baldoni

Master Candidate
Bedriye Gulce Kaya

Academic Year
2023-2024

I dedicate this thesis to the most important two people in my life, my mother, Beyhan Kaya, and my father, Ali Kaya. I consider myself the luckiest person in the world to have you.

Abstract

With the rapid advancement and widespread adoption of virtual reality (VR) technologies, the importance of accurate user identification within these platforms has gained importance for the security and privacy content. This thesis explores the potential of using human motion data as a biometric identifier within the games played in VR environments. Extensive studies involving 60 users were conducted to understand if head and hand movements can be distinguishing to identify the participants within multiple VR sessions. This research demonstrated that we can reliably identify participants with up to 90% accuracy using head and hand motion data as biometric markers.

In this thesis, the movement data of 60 VR users were separated into two groups by playing one slow and one fast game with two different orders between four different VR games: Forklift Simulator, Beat Saber, Medal of Honor, and Cooking Simulator. The slow games that each participant played were the Forklift Simulator or Cooking Simulator and the fast games Beat Saber or Medal of Honor. While group one was playing Cooking Simulator and Beat Saber, group two played Forklift Simulator and Medal of Honor. The order has also changed; order one played the slow game first and order two played the fast game first. We achieved high identification accuracy with the movement data recordings thanks to this dual-game approach which allowed us to capture a wide range of movement patterns.

Contents

Listing of Figures

Listing of acronyms

VR . . . . Virtual Reality

SVM . . . . Support Vector Machine

RBF . . . . Radial Basis Function

HMD . . . . Head Mounted Display

kNN . . . . k-Nearest Neighbors

IS . . . . Information Systems

1
Introduction

The emergence of affordable standalone virtual reality (VR) devices, such as the Meta Quest 2, has allowed VR to reach mass-market adoption in recent years, with nearly 10 million VR headsets sold in 2022 alone[8].The usage of tools such as Head-mounted displays (HMDs) and joysticks helps the user move within the virtual world and also thanks to these tools we can record the movement data from the users.

The movement of a person can be used as a biometric identifier[9] since human behavior patterns change remarkably from person to person and this makes the movement data a reliable biometric identifier. These movement data also known as kinetic signatures allow us to say who the person is. This research was studied with 60 participants' head and hand movement data with various machine learning models and different window sizes and obtained up to 90% of identification accuracy.

Behavioral identification methods offer significant potential for enhancing security and privacy. These methods enable a system to recognize a user based on their unique kinetic signature, allowing automatic access to content or interfaces without requiring permissions or passwords. This streamlined access not only makes the system more user-friendly but also reduces the risk of unauthorized access. However, there are several disadvantages to consider. These methods may raise privacy concerns, as they require the collection and storage of sensitive behavioral data. Additionally, the accuracy of behavioral identification can be affected by changes in the user's behavior or physical condition, potentially leading to errors in user recognition. Lastly, the technology might be susceptible to sophisticated spoofing attacks, where imposters mimic the behavioral traits of legitimate users.

In this thesis, the motion data from four different games was studied. These games are divided into two categories: slow games which are Cooking Simulator and Forklift Simulator and fast games which are Beat Saber and Medal of Honor. The study was carried out with 60 participants and 30 of them played one slow and fast game while the other 30 played the other slow and fast game. The head and hand movement data was analyzed to identify the user with various machine learning models such as Linear Support Vector Machine (SVM), Non-Linear Support Vector Machine (RBF Kernel), AdaBoost, Random Forest, and ExtraTrees. We analyzed the data with window sizes of 1,3,5, and 10 seconds, and the highest accuracy we got was from a window size of 1 second and the least accuracy was from a window size of 10 seconds. In slow games, the prediction performance was higher than the fast games' accuracy. In both Cooking Simulator and Forklift Simulator data we obtained 90% test accuracy in window size 1 while in Beat Saber as a fast game, the test accuracy in window size 1 was 85%, and in Medal of Honor test accuracy was 90% but the best performing training model ExtraTrees with average accuracy was 85%.

This thesis has 5 chapters in total. After the Introduction, the second chapter is a Literature Overview of the history of VR, VR in practice, and related works in the identification of people from VR. In the third chapter, the Data Acquisition process is introduced. In data acquisition; participants, games, VR headsets, data structures, and hand and head movement data are introduced. In the fourth part, the Evaluation Methodology and Results are explained. In this chapter, all four games' accuracy outputs are shown with the graphs and confusion matrices according to changing window sizes. The last chapter is the Conclusion where all these results are discussed.

2
Literature Overview

2.1 History of VR

VR technology immerses users in virtual environments, creating a sense of "being there" [10]. While traditionally studied from a technological perspective, there is growing interest in its behavioral and organizational impacts within Information Systems (IS) research [11].

Contemporary VR technology primarily uses head-mounted displays (HMDs), also known as VR headsets, to immerse users in a virtual world by blocking out the real world[12]. The first HMD, The Sword of Damocles, was developed in the 1960s by Ivan Sutherland, followed by Eric Howlett's LEEP system in the 1970s [13]. In the 1980s, VPL Research introduced several VR devices like the EyePhone, AudioSphere, DataGlove, and DataSuit [14]. Early VR was used mainly for specialized training applications, such as flight simulation and military training. The first consumer HMDs, including Sega VR and Nintendo's Virtual Boy, appeared in the 1990s but were not successful due to low graphic capabilities and motion sickness issues [15]. The CAVE system improved resolution and latency but required dedicated rooms and expensive projectors, limiting its use to professional fields [16].

Around 20 years later, gaming HMDs like Oculus Rift, HTC VIVE, and PlayStation VR brought VR into private households. Modern VR technology includes headphones for sound and controllers for haptic feedback, with advanced systems featuring haptic gloves, suits, and multi-dimensional treadmills[17]. The advent of wireless, stand-alone VR systems like Oculus Quest and HTC VIVE Focus has further facilitated home use. Modern VR technologies replace real-world sensory information with synthetic stimuli, including 3D visual imagery, spatialized sound, and tactile feedback [18].

2.2 VR in Practice

Virtual Reality (VR) has received significant attention recently, with its global market size projected to grow from $7.3 billion in 2018 to $120.5 billion by 2026 [19]. The gaming sector is a significant driver of this growth, with VR headsets like Oculus Quest and HTC VIVE revolutionizing gaming and entertainment.

VR is also being adopted by companies in various other industries, apart from gaming and entertainment, VR is increasingly used in education, particularly for corporate training and university students [20], as well as in schools. Companies like VR Immersive Education and Google Expeditions offer VR applications for subjects such as anatomy, geography, history, physics, and chemistry [21].

Leading companies such as IKEA use VR for onboarding, Macy's enhances the shopping experience, and Verizon trains clerks for emergencies. Deutsche Bahn, Germany's national railway company, uses VR to conduct emergency training for its staff. Volkswagen uses VR for prototyping, allowing designers and engineers to visualize and interact with new vehicle models before they are physically built. This accelerates the design process and reduces costs. Similarly, Tata Motors offers customers the ability to configure cars in VR, providing a personalized and immersive buying experience that helps customers make more informed decisions. Energy and manufacturing sectors are also benefiting from VR technology. E.ON, a leading energy company, trains its substation workers using VR, ensuring they can perform their duties safely and eficiently. Shell leverages VR for safety training, preparing its workforce to handle hazardous situations without real-world risks. MHI Vestas uses VR to showcase wind turbines, allowing stakeholders to explore and understand their technology in a virtual setting. Educational institutions and medical facilities are integrating VR for advanced training. Columbia University and Harvard Medical School train surgeons using VR, allowing them to practice complex procedures in a risk-free environment. Ivoclar Vivadent uses VR to distract dental patients during procedures, reducing anxiety and improving the patient experience. Other industries like real estate, architecture, tourism, military, law enforcement, construction, manufacturing, journalism, and media also utilize VR for various applications. This includes VR-based marketing, shopping, consulting, prototyping, and remote work. Major VR providers, such as Oculus for

Business and HTC VIVE Enterprise, now offer enterprise editions of their devices to cater to these diverse business needs. This expansion prompts IS research to explore VR design and practical use.[22]

Figure 2.1: Surgeons can ftratin for complticafted operafttions tin a safe envtironmenft usting VR applticafttions[1]

2.3 Related Work

The exploration of user identification in Virtual Reality (VR) environments has yielded diverse approaches, as demonstrated by numerous studies focusing on various biometric and behavioral techniques.

In 2018, Mustafa et al. designed and evaluated a head and body movement-based continuous authentication system for VR applications. Based on a dataset of 23 users interacting with a VR application over two sessions, they obtained mean equal error rates as low as 7%. This study highlights the potential of head and body movement patterns for continuous user authentication, offering a promising solution for security-sensitive VR applications. [23] Similarly, Pfeuffer et al. (2019) investigated body motion as behavioral biometrics for VR, examining which behaviors are suitable for user identification. In a study with 22 participants performing tasks such as pointing, grabbing, walking, and typing, they monitored head, hand, and eye motion data. They found that relative distances between body parts showed the highest accuracy for user identification, with overall accuracies of about 40% across sessions by using the Random

Forest model. Their findings highlight the potential of proprioception and head motion as reliable biometric features for secure and adaptive VR environments.[24] In 2020, Kupin et al. introduced a method for authenticating users in VR by tracking their behavior during tasks like throwing a ball at a target. This approach, crucial for mission-critical applications, relies on matching the 3D trajectory of the dominant hand gesture controller to a library of trajectories, rather than using PINs or passwords. The system handles variations in actions using a symmetric sum-squared distance metric. In a pilot study with 14 subjects, the method achieved up to 92.86% accuracy with a library of 10 trajectories per subject, and 90.00% accuracy with 6 trajectories per subject[25]. The same year, Li et al. explored the authentication of users based on head nodding in response to music. By analyzing the nodding patterns, they achieved mean Equal Error Rates (EERs) ranging from 4.43% to 24.94% on a dataset of 30 subjects[26]. Another study in 2020 by Olade et al. explores the use of kinesiological data for biometric user identification within VR systems. Their research demonstrates that individual behavioral and movement characteristics, unique to each person, can serve as effective biometric discriminates. In their study, 15 participants' hand, head, and eye gaze data within the VR environment were captured and by using machine learning classification methods such as kNN and SVM, the study achieved a high confidence in identifying users, with an average identification confidence value of 0.98 and a classification accuracy of 98.6%. [27] Moreover, Miller et al. (2020) conducted a lab study of 511 users, whose telemetry was captured while they watched a series of 360-degree videos in VR. Using a Random Forest model, they succeed in identifying the users with 95% accuracy from 5 minutes of telemetry data. [28] In 2021, Yi et al. investigated user authentication through six specific head gestures, including shapes such as circles, triangles, squares, and lines. Participants executed these gestures by using their noses as pointers, effectively tracing the shapes in the air. On a dataset of 18 users, Yi et al. reported authentication accuracies of up to 92% [29]. Tricomi et al. (2022) demonstrated the profiling of AR and VR users with laboratory studies of 34 and 35 users, respectively. They uniquely identify 30 users in VR with 95% accuracy using a logistic regression model[30]. Hu et al. (2023) contribute to this field by collecting the eye and head movement data from 30 participants performing four different tasks (Free viewing, Visual search, Saliency, and Track) in 15 360-degree VR videos. With EHTask Method, state-of-the-art task recognition methods derived from 2D viewing conditions, achieved an accuracy of 84.4% on their dataset and 61.9% on a real-world dataset. [31] Recently, in 2024, Liebers et al. investigated kinetic signatures—spatiotemporal movement data that is unique to each individual. Their study involved 24 participants performing various VR sports and exercises over two sessions and examined how static (muscular activity to hold joints in place) and dynamic (muscular activity to change positions) components influence the identifiability of kinetic signatures. They found that the identifiability of a kinetic signature depends on its static and dynamic components, achieving up to 90.91% identification accuracy.[9]

3
Data Acquisition

In this thesis, users are identified through their head and hand movements recorded during gameplay using the Meta Quest 2 VR headset. This section explains in detail how the data was acquired. First, the participants are introduced by their ages, genders, their previous general gaming experience, and their previous VR gaming experience. Participant information is followed by a description of the games and the VR headset used. The data structure is then explained with a graph, and the data folders are shown with screenshots. Lastly, the hand and head movement data are described using graphs and heat maps.

3.1 Participant Information

The users who managed to finish the games were 60 people. Their ages are between 19 to 35. Figure 3.1 shows the age distribution of the participants. The majority of the participants are between 24 to 29 years old, with a peak at 25 years old. This age distribution is important as it represents a young demographic that is likely familiar with gaming environments, potentially impacting their performance and interaction in VR settings. Understanding the age distribution helps in analyzing the adaptability and learning curves of different age groups in VR environments.

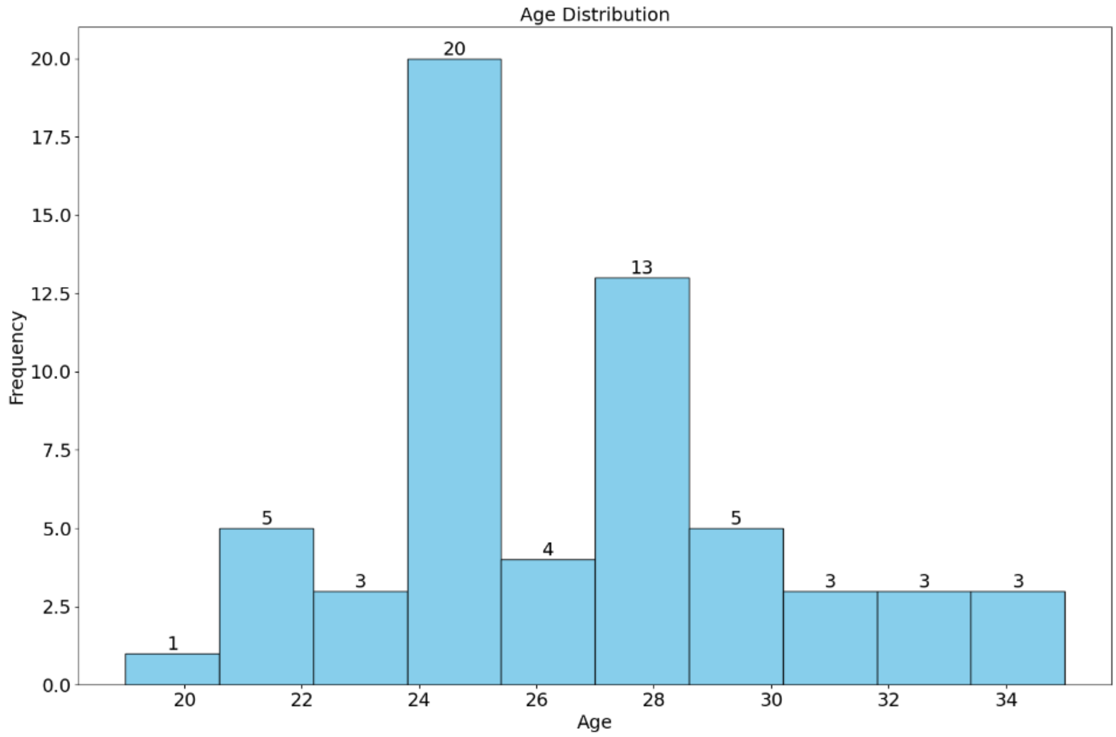

Figure 3.1: Age Dtisftrtibufttion of fthe Parfttictipanfts

The games were played by 38 male and 22 female users. This distribution is crucial as it may influence the results due to potential gender-specific differences in interaction with VR technology. Gender diversity in the study ensures a more comprehensive understanding of user interactions and helps in designing more inclusive VR systems.

The gaming backgrounds of the users are various also. Most of the users have previous gaming experience while when it comes to VR gaming experience, most of the users had the first experience in VR.

Figure 3.2 shows the distribution of previous general gaming experience among the participants. The data reveals that a significant portion of the participants have moderate to high levels of general gaming experience. This information is pertinent as prior gaming experience can influence how quickly participants adapt to VR environments and how they perform in VR tasks. Participants with higher gaming experience might find it easier to navigate and complete VR tasks compared to those with little to no gaming experience.

Figure 3.3 highlights the participants' previous VR gaming experience. Notably, most participants had minimal to no prior VR experience. This lack of VR experience is important to consider when analyzing the results, as it could impact the initial learning curve and per-

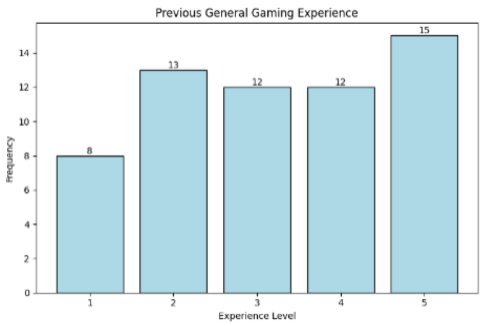

Figure 3.2: Prevtious General Gamting Expertience Dtisftrtibufttion of fthe Parfttictipanfts

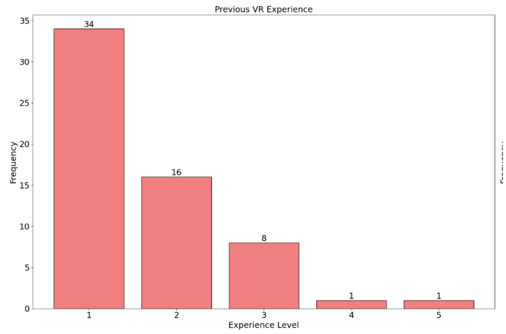

Figure 3.3: Prevtious VR Gamting Expertience Dtisftrtibufttion of fthe Parfttictipanfts

formance in the VR environment. It underscores the necessity of designing user-friendly and intuitive VR interfaces to accommodate both novice and experienced users.

The demographic data including age, gender, and gaming experience provides a comprehensive overview of the participant pool. Understanding these distributions is crucial for interpreting the results of the study, as they can affect user performance and interaction within the VR environment. The variability in participants' backgrounds ensures that the study can provide insights applicable to a broad range of users, thereby enhancing the generalizability of the findings.

3.2 Games

Four different games were played during the study: Beat Saber, Forklift Simulator, Medal of Honor, and Cooking Simulator and movement data was collected for each participant while using four VR commercial games selected based on the type of required movement (i.e., fast and slow) and the type of content. Beat Saber and Medal of Honor are categorized as fast games, whereas Forklift Simulator and Cooking Simulator are considered slow games. To examine the impact of game pace and the order of play on user performance and interaction, users are separated into two groups, with each group playing the games in different orders. Specifically, each group played one slow and one fast game, but the order varied: half of the participants played the slow game first followed by the fast game, and the other half played the fast game first followed by the slow game.

The decision to switch the order in which the games were played is rooted in understanding how different paces of gameplay affect user behavior and performance in VR environments.

3.2.1 Cooking Simulator

Cooking Simulator is a slow type of game that is played in a huge kitchen environment with all kinds of utensils and ingredients to prepare 80 recipes available in the game.[2] In this thesis, participants only prepared two recipes.

Figure 3.4: Cookting Stimulaftor Game[2]

Tasks of the Cooking Simulator game require precision and attention to detail. Players interact with various kitchen tools and ingredients to create specific dishes. This game emphasizes fine motor skills and careful planning, making it a slow and methodical experience compared to the other games.

3.2.2 Beat Saber

Beat Saber is a virtual reality rhythm game set in various surrealistic neon environments. Players slice blocks representing musical beats with brightly-colored sabers, using VR controllers. Each song presents a stream of approaching blocks laid out in sync with the song's beats and notes, located in one of 12 possible positions of a 4x3 grid. For this study, we considered three Beat Saber play options: mono-directional (0°), bi-directional (90°), and omnidirectional (360°) at three different levels (easy, medium, and hard).[32][33]

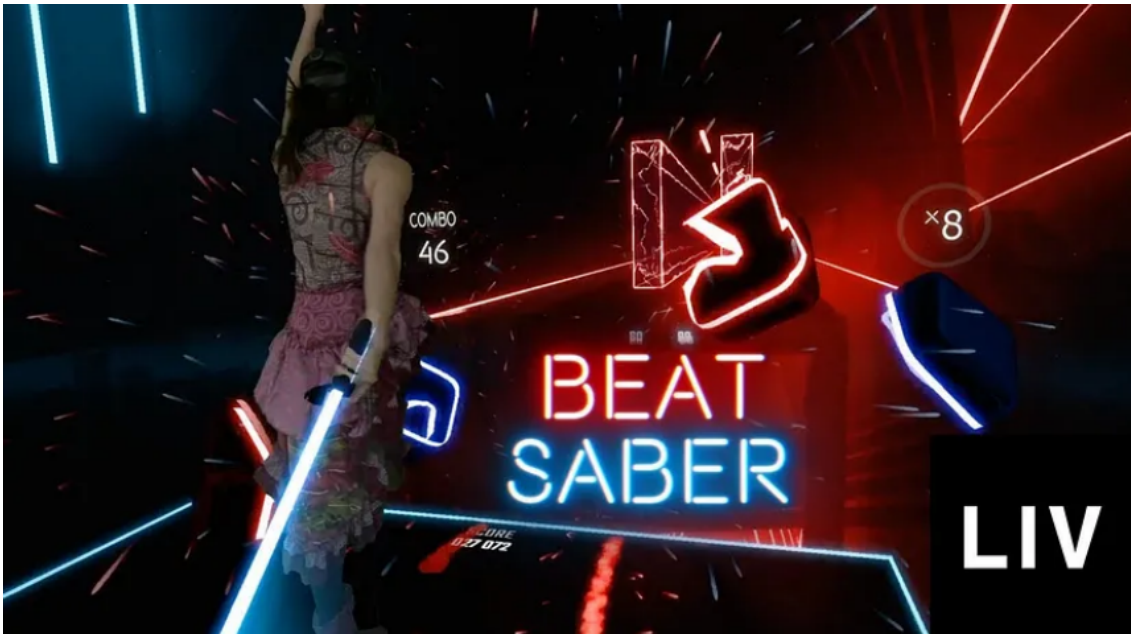

Figure 3.5: Beaft Saber Game[3]

Beat Saber is a fast-paced rhythm game that requires quick reflexes and precise timing. Players must slice blocks in sync with the music, often at a rapid pace, which creates an exhilarating and immersive experience. Each block corresponds to a specific beat, and players use virtual lightsabers to cut through them, matching their direction and color. This game emphasizes fast motor skills, hand-eye coordination, and rhythm, as players must constantly adjust their movements to keep up with the tempo and complexity of the tracks. The high-intensity gameplay is both physically and mentally demanding.

3.2.3 Forklift Simulator

Forklift Simulator 2019 is developed to train qualified forklift operators in a realistic learning environment cost-effectively and reliably. The simulator provides individuals with comprehensive training using real forklift equipment and rich scenario content. Forklift Simulator enables operators to learn operational techniques such as mast controls, forklift maneuvers, and accident risks in different work areas and periods. Participants engaged in levels with increasing dificulty on the four-wheel sit-down counterbalanced forklift. If participants failed a level, they had to repeat it until they passed or the experiment concluded.[34][33]

Figure 3.6: Forkltifft Stimulaftor Game[4]

Forklift Simulator is a slow-paced training game focused on operating a forklift with precision and accuracy. It emphasizes realistic controls and scenario-based learning, requiring players to manage the forklift with careful attention to detail. The game covers various operational aspects, including loading and unloading materials, navigating tight spaces, and adhering to safety protocols. By simulating real-world conditions, the game helps trainees build confidence and competence in handling a forklift, reducing the risk of accidents and improving overall workplace safety.

This game is more about accuracy and operational understanding, offering a stark contrast to the fast-paced action of games like Beat Saber and Medal of Honor. While Beat Saber challenges players' reflexes and rhythm through rapid block-slicing to the beat of the music, and Medal of Honor immerses players in intense, fast-moving military combat scenarios, Forklift Simulator takes a more deliberate and methodical approach. It requires players to think strategically, plan their movements, and execute tasks with precision.

3.2.4 Medal of Honor

Medal of Honor: Above and Beyond is a first-person shooter virtual reality game. The game takes place in North Africa, France, Norway, and Germany during World War II, taking the franchise back to its roots. It was released for the Oculus Rift and Steam VR on December 11, 2020. In this study, participants undertook a survival task, aiming to live as long as possible, repeated for at least 10 minutes.[35][33]

Figure 3.7: Medal of Honor Game[5]

Medal of Honor is a fast-paced first-person shooter that immerses players in World War II scenarios. It involves intense combat situations, requiring quick decision-making, fast reflexes, and strategic thinking. This game contrasts with the more structured and slower-paced tasks in Forklift Simulator and Cooking Simulator, offering a dynamic and high-energy gameplay experience.

The four games chosen for this study provide a diverse range of gameplay experiences, from the fast-paced, reflex-driven action of Beat Saber and Medal of Honor to the slow, precision-focused tasks of Cooking Simulator and Forklift Simulator. This diversity is essential for examining how different types of gameplay affect user performance and interaction in VR environments. By varying the order of gameplay, the study can analyze the impact of game pace and transition effects on users, providing valuable insights into user adaptation and performance across different VR scenarios.

As the information about the games is detailed above, these games are divided into two categories based on the velocity of movement during play. This classification will be more evident in sections 3.4.1 Hand Movement Data and 3.4.2 Head Movement Data, where heat maps and average values of user interactions for each game will be presented. For instance, in the Forklift

Simulator and Cooking Simulator, it is clear that the movement is limited due to the nature of the tasks involved. These slow-paced games involve specific, goal-oriented tasks that require precision and careful manipulation of virtual objects, resulting in concentrated areas of activity in the heat maps.

On the other hand, fast-paced games like Beat Saber and Medal of Honor require rapid, continuous movements and quick reflexes, leading to more varied and widespread movement patterns in the heat maps.

3.3 Meta Quest 2

The Meta Quest 2, initially released as the Oculus Quest 2 on October 13, 2020, and rebranded in 2022, is a state-of-the-art standalone virtual reality (VR) headset developed by Meta (formerly Facebook). This headset is designed to offer an immersive VR experience without the need for a PC or external sensors, making it accessible and user-friendly for a wide range of applications from gaming to professional training.

The Quest 2 features significant hardware improvements over its predecessor, the Oculus Quest, including a higher resolution display, a more powerful processor, and an increased refresh rate. These enhancements provide users with a more visually engaging and smoother VR experience.[36]

Figure 3.8: Quesft 2 Headseft[6]

The included hardware also comprises the third-generation Oculus Touch controllers. These controllers are ergonomically designed for comfort and precision, featuring improved haptic feedback and hand-tracking capabilities. Each controller has an analog thumbstick, two face buttons (A/B on the right, X/Y on the left), a menu button, and grip and trigger buttons. The controllers are tracked by the headset's built-in cameras, providing accurate positional data and responsive interactions in the virtual environment.

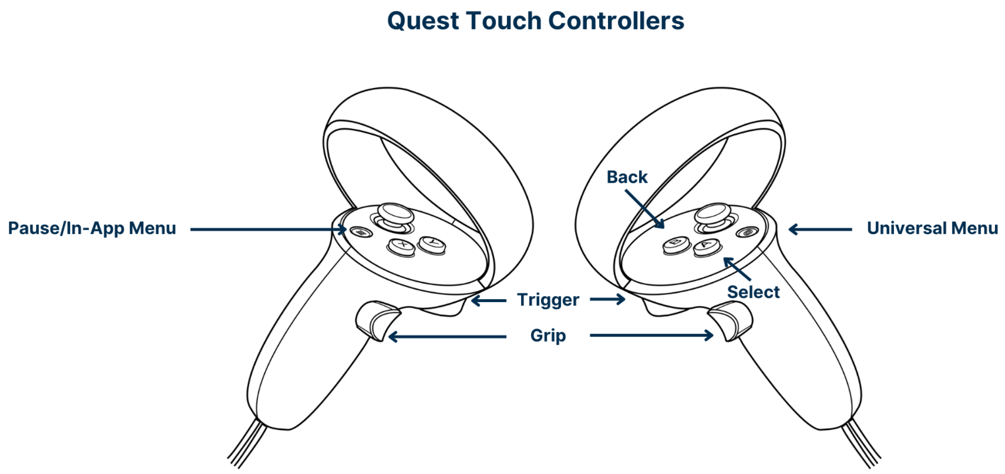

Figure 3.9: Quesft Touch Conftrollers[7]

Movement data were captured from 60 users as they interacted with the VR games. The Quest 2 headset and Touch controllers recorded hand and head movements. The headset's built-in cameras and internal sensors captured positional and rotational data, while the controllers recorded hand movements and button presses. Table 3.1 shows the details of the data columns.

Remote Button	This column refers to buttons on an additional remote device, which will always be 0 and can be ignored.
Touch Buttons	These refer to the A, B, and Thumb buttons on the right controller and the X, Y, Menu, and Thumb buttons on the left controller. Each button is associated with a number, and if it is pressed, that number appears in the CSV file. If multiple buttons are pressed simultaneously, their sum appears in the CSV file.
Touch Touches	This column refers to the touch of one of the above-mentioned buttons. Since this mainly depends on how the controllers were held and not on button pressing, they are not directly related to user interaction with the game. Moreover, it is not clear how these touches are related to the numbers in the CSV file, so this column can be discarded in the initial analysis.
Index Trigger Columns (Left and Right)	These refer to the pressing of the trigger button, usually employed for clicking. Its value varies between 0 and 1, depending on the level of pressure applied.
Hand Trigger Columns (Left and Right)	These refer to the pressing of the grip button, usually employed for grabbing. Its value varies between 0 and 1, depending on the level of pressure applied.
Position and Orientation of Controllers	Captures the spatial data (x, y, z coordinates) and rotational data (quaternions) of both the left and right controllers.
Position and Orientation of the Headset	Captures the spatial data and rotational data of the headset.
Additional Columns	There are additional columns that are constant and can be ignored.

Table 3.1: Deftatils of Dafta Columns

3.4 Data Structure

The dataset is organized in 60 users' folders including 4 different CSV files inside. The organization of the data is as in Figure 3.10.

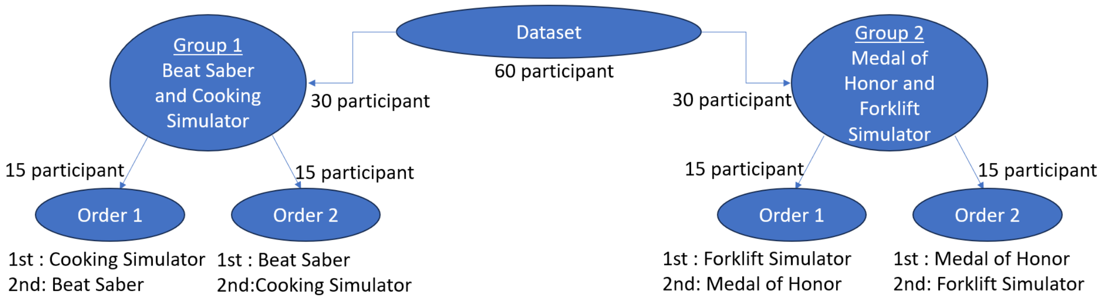

Figure 3.10: Daftaseft Organtizafttion

There are 60 participants in total. These 60 participants were divided into two groups; Group 1 played the games Beat Saber and Cooking Simulator, and Group 2 played the games Medal of Honor and Forklift Simulator. Each group is playing one slow and one fast game in two different order. In Order 1, first, the slow game was played and after that, the fast game was played. In Order 2, first, the fast game was played and after that, the slow game was played. The users' folders are the same as in Figure 3.11.

Figure 3.11: User Folders

For instance, the user ”group2_order2_user14” first played the fast game in Group 2 which is Medal of Honor, and second played the slow game Forklift Simulator. Each participant has 4 CSV files in their folders.

Figure 3.12: CSV ftiles tinstide fthe users’ folders

As in Figure 3.12, the user ”group2_order2_user14” has the CSV files for slow trafic, slow movement, fast trafic, and fast movement. In this thesis, we studied only movement data. So the data was analyzed by the games’ names. For instance, we analyzed the ”group2_order2_user14_slow_movement” CSV file as the user ”group2_order2_user14” in-side the Forklift Simulator.

Inside the movement CSV files the columns include the information about the headset and controllers. The controllers’ position and orientation feature columns, the headset’s position and orientation feature columns, and the column of ’time.

Additionally, we have the ’Initial Survey’ CSV file which has, Gender, Previous VR Experience, and Previous General Gaming Experience information for each user as shown in Figure 3.13.

ID	Age	Gender	Previous VR experie	Previous general ga
group1_order1_user0	22	Female	1	2
group1_order1_user1	33	Female	1	2
group1_order1_user2	34	Female	2	2
group1_order1_user3	24	Male	1	3
group1_order1_user4	23	Female	2	2
group1_order1_user5	35	Male	1	1
group1_order1_user6	24	Male	2	5
group1_order1_user7	24	Male	1	2
group1_order1_user8	26	Male	2	4
group1_order1_user9	25	Male	2	5
group1_order1_user1	27	Male	1	5

Figure 3.13: Intifttial Survey Ftile

The last data file that we have was the Dataset Information file which has the information of the user's group, order, Game Speed type, Game Name, User number, Duration of trafic and movement, and Sample number of trafic and movement as shown in Figure 3.14.

Traffic filepath	Movement filepath	Group	Order	Game Speed	Game Name	User	Duration (Traffic)	Duration (Movement	N Samples (Traffic)	N Samples (Moveme
group2_order1_user1	group2_order1_user1	2	1	slow	Forklift Simulator	12	1506.259	1506.261	3256229	90309
group2_order1_user1	group2_order1_user1	2	1	fast	Medal of Honor	12	1030.054	1030.054	2462182	61750
group1_order1_user4	group1_order1_user4	1	1	slow	Cooking Simulator	4	654.11	654.108	1821437	39217
group1_order1_user4	group1_order1_user4	1	1	fast	Beat Saber	4	1170.307	1170.305	4416866	70167
group1_order1_user1	group1_order1_user1	1	1	slow	Cooking Simulator	11	1239.796	1239.799	3340600	74327
group1_order1_user1	group1_order1_user1	1	1	fast	Beat Saber	11	1296.704	1296.704	4752357	77739
group1_order1_user8	group1_order1_user8	1	1	slow	Cooking Simulator	8	1269.314	1269.348	3447555	76102

Figure 3.14: Daftaseft Informafttion Ftile

We can divide the movement data into two Head and Hand Movement data.

3.4.1 Hand Movement Data

In this part, the average values of each hand trigger and position data in the columns of the CSV files for each user in four games are presented first. The metrics used in the graphs are the position of the left controller on the vertical/lateral/frontal axis: LeftTouchPosX, LeftTouchPosY, LeftTouchPosZ, the position of the right controller on the vertical/lateral/frontal axis: RightTouchPosX, RightTouchPosY, and RightTouchPosZ and the trigger of both hands: LeftHandTrigger, and RightHandTrigger.

Figure 3.15 displays the average values of various hand movement metrics for each user during their interaction with the Cooking Simulator.

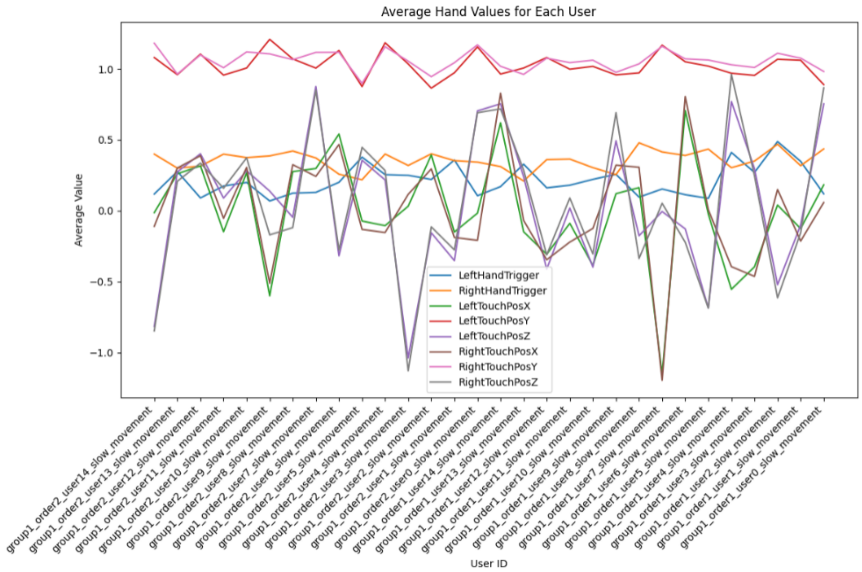

Figure 3.15: Cookting Stimulaftor Average Hand Values by User

LeftTouchPosY and RightTouchPosY show a relatively stable pattern across users. The LeftTouchPosX and RightTouchPosX have noticeable fluctuations, indicating variations in horizontal hand movements across different users.LeftHandTrigger and RightHandTrigger show less variation, suggesting that vertical movements are relatively more consistent. The Z coordinates (LeftTouchPosZ and RightTouchPosZ) also show some variability, implying changes in depth movements, possibly related to reaching or interacting with objects at varying distances. Certain users show spikes or dips in specific metrics. For example, user group1_order2_user3_slow_movement shows significant variations in LeftTouchPosZ and RightTouchPosZ. Such anomalies could be due to specific gameplay behaviors or interaction styles unique to those users.

Generally, the users tend to use their left and right hands differently, as seen by the distinct patterns in the LeftTouchPos and RightTouchPos metrics. This could be due to the tasks in the Cooking Simulator, where users might prefer using one hand over the other for certain actions.

Users are grouped into two categories (order 1 and order 2), and each group's members show varying degrees of movement. Despite the variations, there are not that much of clear distinctions in movement patterns solely based on group categorization.

Figure 3.16 shows the average values of various hand movement metrics for each user during their interaction with Beat Saber.

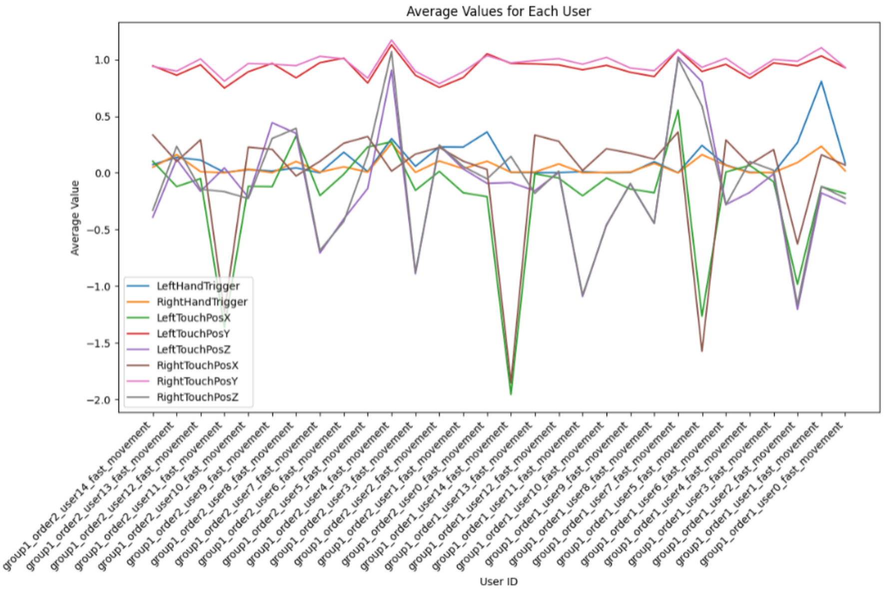

Figure 3.16: Beaft Saber Average Hand Values by User

LeftTouchPosY and RightTouchPosY exhibit relatively stable patterns, similar to other VR games. The LeftTouchPosX and RightTouchPosX show fluctuations, indicating dynamic horizontal hand movements typical in a rhythm game where players need to reach different blocks.LeftHandTrigger and RightHandTrigger also show variations. The Z coordinates (LeftTouchPosZ and RightTouchPosZ) exhibit pronounced variability, reflecting the depth movements required to align with incoming blocks and perform the slicing action.

Some users show significant spikes or dips in specific metrics, highlighting individual differences in gameplay behavior. For instance, group1_order2_user4_fast_movement shows a pronounced peak in RightTouchPosZ, indicating an intensive forward hand movement, possibly reflecting an aggressive playstyle.

The trends suggest different usage patterns for left and right hands, with both showing similar but individually distinctive variations which is expected as Beat Saber requires both left and right hand movements to hit blocks coming from various directions.

Individual playstyles and engagement levels likely influence these observed variations more than group categorization.

Figure 3.17 shows the average values of various hand movement metrics for each user during their interaction with the Forklift Simulator.

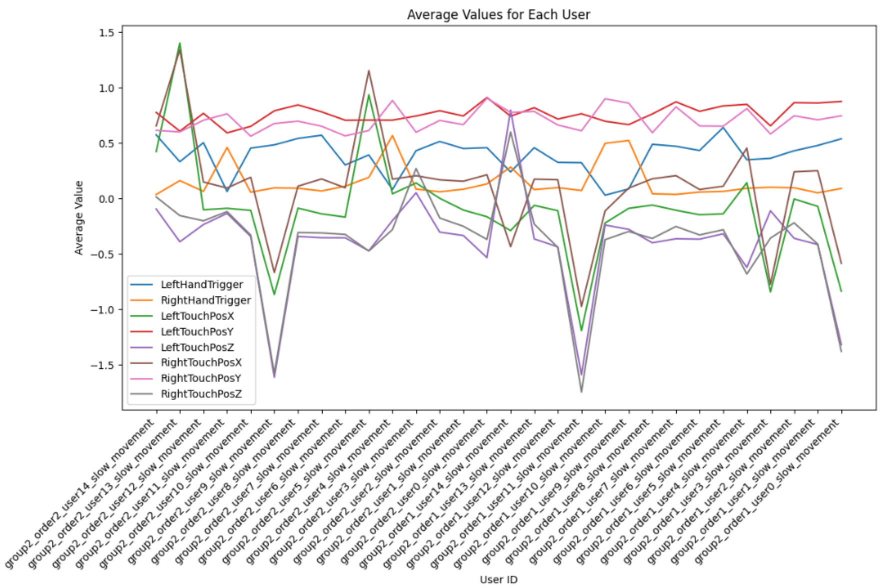

Figure 3.17: Forkltifft Stimulaftor Average Hand Values by User

Both LeftTouchPosY and RightTouchPosY exhibit less variation and relatively stable patterns compared to other metrics implying vertical movements are minimal. The LeftTouchPosX and RightTouchPosX show less stability with more pronounced fluctuations, indicating more horizontal hand movement. LeftHandTrigger and RightHandTrigger also show less variation, this suggests that trigger usage in the Forklift Simulator is consistent and less dynamic. LeftTouchPosZ and RightTouchPosZ exhibit some variability. Some users exhibit spikes or dips in specific metrics, such as LeftTouchPosZ and RightTouchPosZ for certain users, indicating individual differences in gameplay behavior. For example, user group2_order1_user11_slow_movement shows a significant dip in RightTouchPosZ, which could indicate a unique interaction or an anomaly in data collection.

The overall trend suggests slight differences in the use of left and right hands, but both show similar patterns of stability and less variability. This uniformity indicates that the task requirements in the Forklift Simulator equally consistently engage both hands.

No distinct differences are observed solely based on order categorization, reinforcing that the nature of the game dictates a more standardized interaction pattern across different users. The data highlights consistent and less dynamic interaction metrics typical of a seated driving simulation game.

Figure 3.18 shows the average values of various hand movement metrics for each user during their interaction with the Medal of Honor.

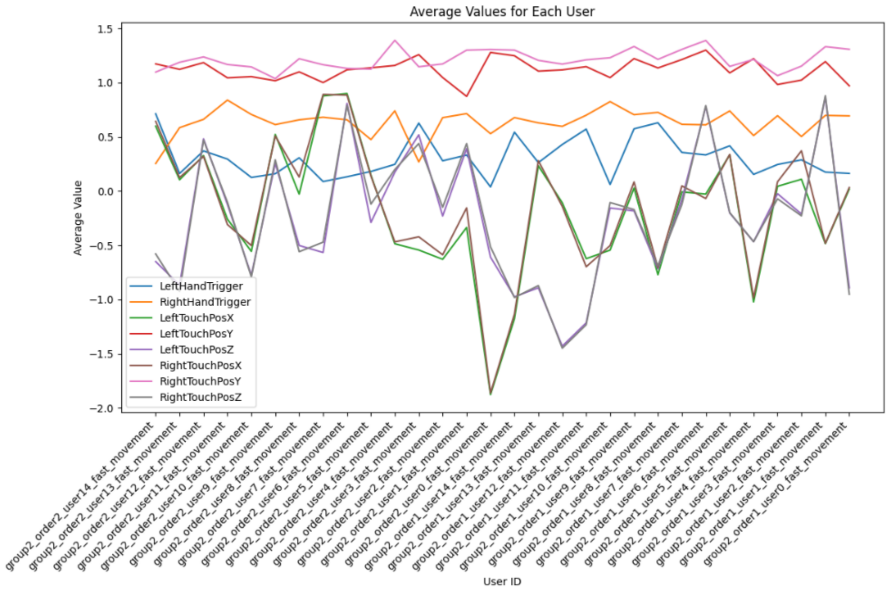

Figure 3.18: Medal of Honor Average Hand Values by User

LeftTouchPosY and RightTouchPosY values show relatively stable patterns across different users, indicating consistent vertical adjustments.LeftTouchPosX and RightTouchPosX show some variability, indicating horizontal hand movements as players aim and maneuver in the game. LeftHandTrigger and RightHandTrigger also exhibit variations. LeftTouchPosZ and RightTouchPosZ show pronounced fluctuations, indicating dynamic forward and backward movements typical in an action game requiring constant repositioning.

Certain users exhibit significant spikes or dips in specific metrics, reflecting individual differences in gameplay behavior. For instance, group2_order1_user1_fast_movement shows noticeable peaks in RightTouchPosZ, indicating aggressive forward hand movements, possibly due to intense gameplay or frequent forward aiming.

The trends suggest different usage patterns for left and right hands, with both showing distinctive variations. This difference aligns with the nature of first-person shooters where players use both hands independently for aiming, shooting, and other interactions.

No significant distinctions are based solely on order categorization, reinforcing that individual playstyle and engagement levels influence the observed variations.

The provided graph effectively captures the average hand movement values for each user while playing Medal of Honor VR. The data highlights dynamic interaction metrics typical of a first-person shooter, with noticeable individual variations reflecting different playstyles.

These insights can be valuable for understanding user engagement and optimizing game design to enhance the immersive experience.

Across all games, both LeftTouchPosY and RightTouchPosY show relatively stable patterns, indicating consistent use. Individual variations are noticeable across all games. Group 1 users have the LeftTouchPosY and RightTouchPosY values very stable and higher than the other trigger and position data values in both the Cooking Simulator and the Beat Saber games. Other parameters are between -0.5 and 0.5 mostly, but of course there are certain spikes for some users. When Group 2 users' graphs are checked, the average values are more close to each other. They also have the LeftTouchPosY and RightTouchPosY values very stable and higher than the other trigger and position data values. The Forklift Simulator data is below 1, while the Medal of Honor has values over 1. There are also certain spikes in both games, but generally, the values are not far from each other. In Medal of Honor, Left Touch Pos X and Z, and Right Touch Pos X and Z have the most spikes while in Forklift Simulator Right and Left Touch Pos Z values have more spikes than the Pos X values.

In this part, the right and the left touch position heatmaps in the XZ plane are presented. The right and left touch position heatmaps in the XZ plane for various VR games highlight differences in user hand movements. In the Cooking Simulator (Figure 3.19 and Figure 3.20), the spread along the X-axis for the right touch position suggests users frequently move their right hand horizontally to interact with different objects or controls, while the Z-axis spread indicates depth movements, showing users reaching out or pulling back. Multiple high-density areas suggest specific tasks that require precise right-hand positioning. Similarly, the left touch position shows significant horizontal spread and depth movements, indicating dynamic interactions with in-game elements and a wider range of tasks compared to the right hand.

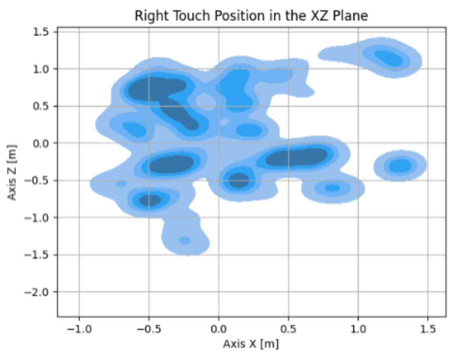

Figure 3.19: Cookting Stimulaftor Rtighft Touch Heaft Map tin X-Z Plane

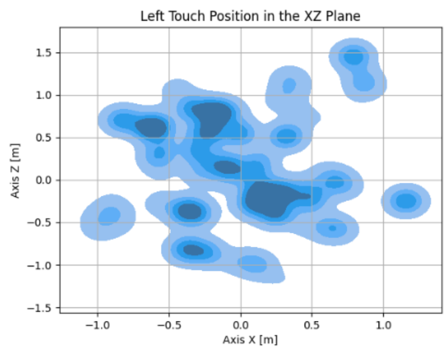

Figure 3.20: Cookting Stimulaftor Lefft Touch Heaft Map tin X-Z Plane

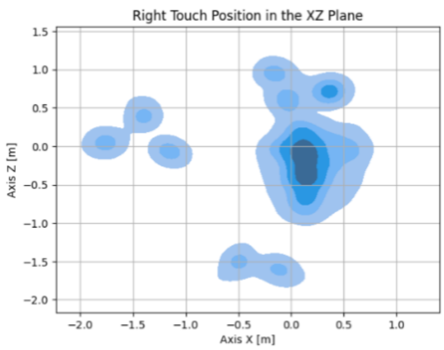

Figure 3.21: Beaft Saber Rtighft Touch Heaft Map tin X-Z Plane

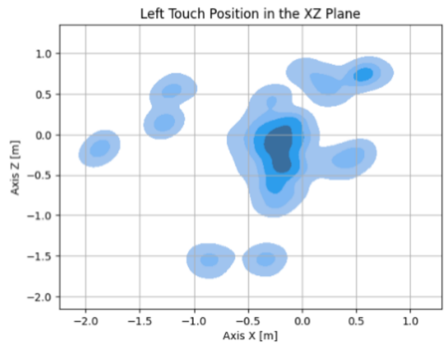

Figure 3.22: Beaft Saber Lefft Touch Heaft Map tin X-Z Plane

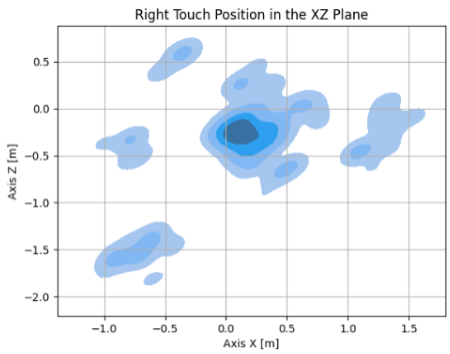

Figure 3.23: Forkltifft Stimulaftor Rtighft Touch Heaft Map tin X-Z Plane

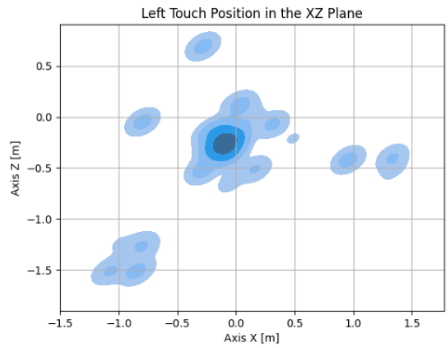

Figure 3.24: Forkltifft Stimulaftor Lefft Touch Heaft Map tin X-Z Plane

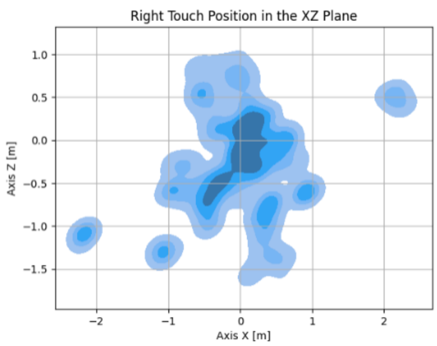

Figure 3.25: Medal of Honor Rtighft Touch Heaft Map tin X-Z Plane

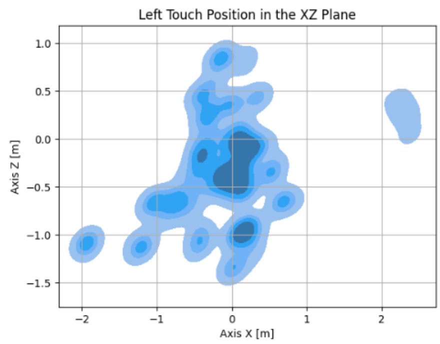

Figure 3.26: Medal of Honor Lefft Touch Heaft Map tin X-Z Plane

In Beat Saber (Figure 3.21 and Figure 3.22), the spread along both the X-axis and Z-axis for the right touch position reflects the fast-paced, rhythmic slicing actions required. A high-density area around the central region indicates a common hand position during idle moments, with a concentrated area around (0, 0) implying a central resting position. The left touch po-sition heatmap shows significant movement along both axes, highlighting the dynamic nature of hand movements required for slicing blocks. Central and spread-out clusters indicate the frequent and varied reaches needed, reflecting quick and repetitive hand movements.

In Forklift Simulator (Figure 3.23 and Figure 3.24), the majority of right hand movement is concentrated around the central region, indicating less dynamic and more stable positioning, typical for the seated driving environment. The main cluster around the origin suggests frequent and consistent use of the right hand in a central position. The left touch position heatmap also shows a central high-density area with minimal spread, indicating stable and controlled interactions typical for driving and operating controls.

In Medal of Honor (Figure 3.25 and Figure 3.26), multiple dense regions with significant spread indicate varied and dynamic right hand movements, typical of an action-packed first-person shooter. Users move their right hand dynamically to aim, shoot, and interact with the environment, with central and scattered dense areas implying frequent and varied hand positions. The left touch position heatmap shows similar dynamics, with significant movement along both axes highlighting varied hand movements required for aiming and other actions. Central and scattered clusters indicate frequent use for different actions like aiming and interacting with objects.

The provided heatmaps indicate that the Medal of Honor requires dynamic and varied hand movements, reflecting the active and adaptive nature of the game. In contrast, the Forklift Simulator shows stable and consistent hand positions, reflecting the controlled and less dynamic gameplay. The Cooking Simulator shows more dynamic interaction patterns compared to the Forklift Simulator. These insights are valuable for understanding user interaction and optimizing VR game design to enhance the immersive experience.

3.4.2 Head Movement Data

In this part, the average head movement values for each user in four games are presented first. The metrics used in the graphs are the position of the HMDs on the vertical/lateral/frontal axis: 'HeadPosX', 'HeadPosY', 'HeadPosZ', and the orientation of the HMDs: 'HeadOrientationW', 'HeadOrientationX', 'HeadOrientationY', 'HeadOrientationZ'.

Figure 3.27 displays the average values of various head movement metrics for each user during their interaction with the Cooking Simulator.

HeadPosX, HeadPosY, and HeadPosZ show moderate variability across users. Vertical head movements (HeadPosY) are relatively stable compared to horizontal (HeadPosX) and depth (HeadPosZ) movements.

HeadOrientationX and HeadOrientationZ are consistently high and stable, suggesting minimal rotation around the X and Z-axis. HeadOrientationW and HeadOrientationY show more variability, indicating varied head orientations during gameplay.

Significant variability among users, with some showing pronounced peaks and dips in head movement metrics. Reflects the interactive nature of the game, requiring head movements to look around and interact with different elements in the kitchen.

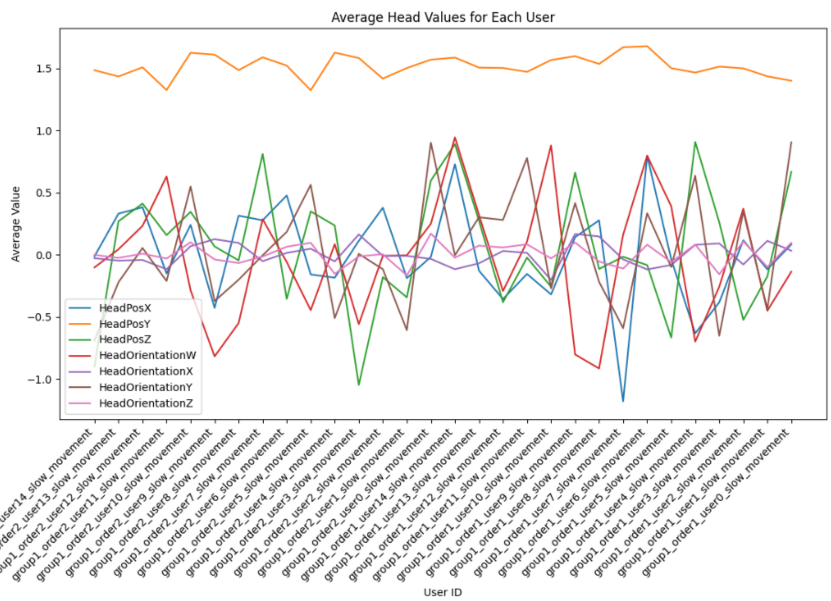

Figure 3.27: Cookting Stimulaftor Average Head Values by User

Figure 3.28 displays the average values of various head movement metrics for each user during their interaction with the Beat Saber. HeadPosX and HeadPosZ show higher variability compared to the Cooking Simulator. Reflects the dynamic and fast-paced nature of the game, requiring frequent and rapid head movements to track incoming blocks. HeadOrientationZ and HeadOrientationX remain consistent, indicating stable rotational positioning around the Z-axis. HeadOrientationW and HeadOrientationY exhibit more fluctuations, indicating dynamic head movements to follow the rhythm and blocks.

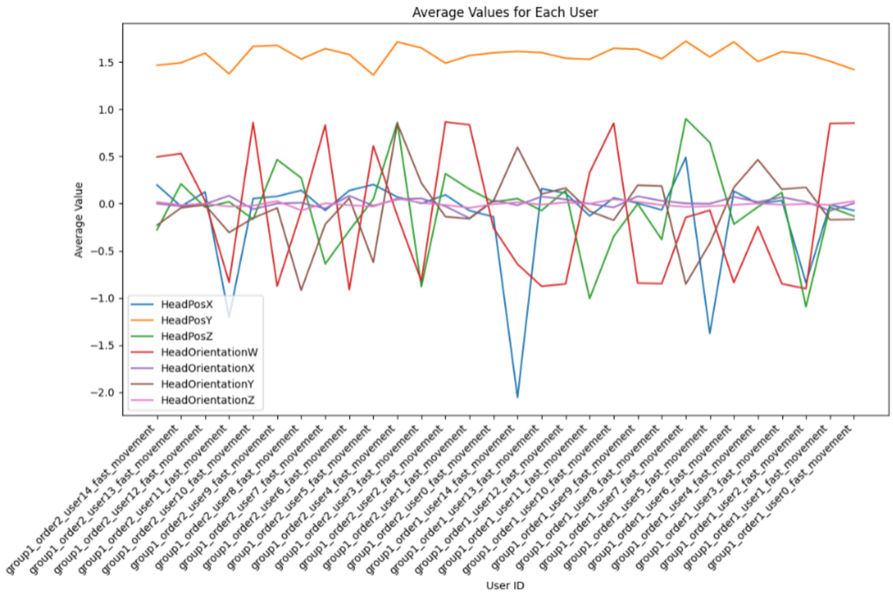

Figure 3.28: Beaft Saber Average Head Values by User

Figure 3.29: Forkltifft Stimulaftor Average Head Values by User

Figure 3.29 displays the average values of various head movement metrics for each user during their interaction with the Forklift Simulator.

HeadPosY exhibits less variability, reflecting the controlled and seated nature of the game.

HeadOrientationX, Y, Z, and W are similar to other games. Variability among users is minimal, reflecting consistent and controlled head movements required for operating the forklift.

Figure 3.30 displays the average values of various head movement metrics for each user during their interaction with the Medal of Honor. HeadPosX and HeadPosZ show significant variability indicating dynamic head movements typical of a first-person shooter. Reflects active engagement in aiming, shooting, and navigating the game environment.

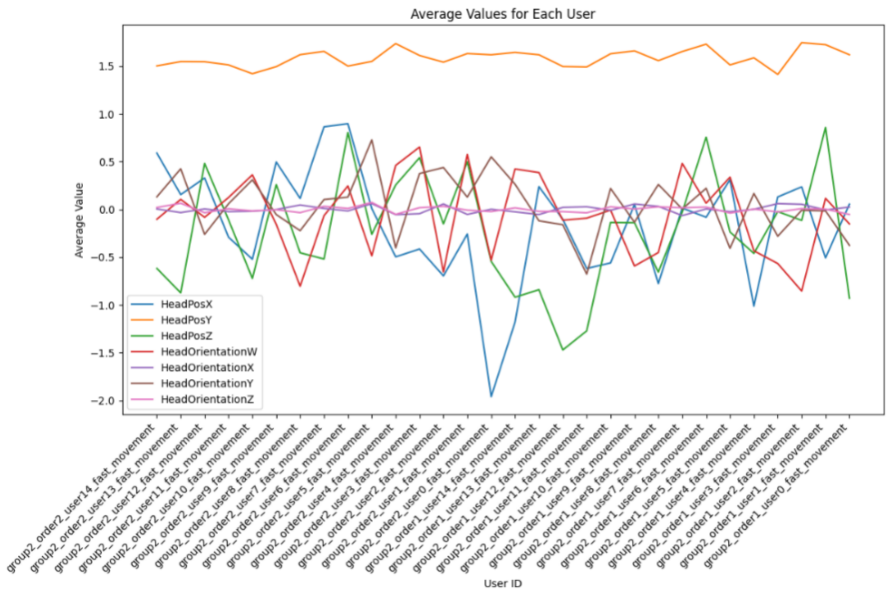

Figure 3.30: Medal of Honor Average Head Values by User

Cooking Simulator and Forklift Simulator (slow movement games) show more stable and less variable head positions compared to Beat Saber and Medal of Honor (fast movement games). Beat Saber and Medal of Honor exhibit higher variability in head positions, reflecting dynamic and frequent head movements. HeadPosX and HeadPosZ show significant variability since HeadPosY is generally stable. HeadOrientationX and Z are stable across all games, indicating minimal rotational movements around the X and Z axis.

In this part, the heat maps of the HMD position in the XZ plane of each game are presented.

In Cooking Simulator (Figure 3.31), the heatmap shows multiple dense regions indicating varied head positions. The significant spread along both X and Z axes suggests that users frequently move their heads to interact with different kitchen elements. Diverse head movements reflect tasks like looking at different ingredients and utensils, while central dense regions imply a default head position that users often return to.

For Beat Saber (Figure 3.32), the heatmap reveals a central high-density region with fewer spread-out areas. The prominent main cluster indicates frequent head positioning around the center, suggesting that users maintain a stable head position while slicing blocks. Some spread indicates quick adjustments to follow the rhythm and blocks coming from different directions.

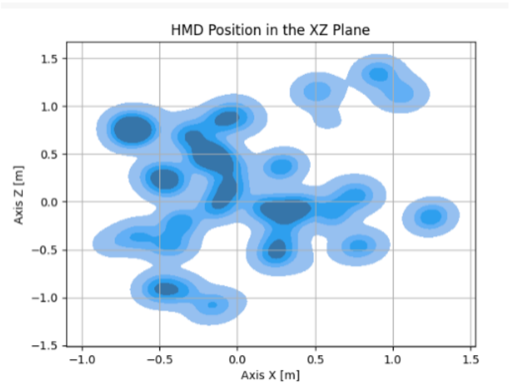

Figure 3.31: Cookting Stimulaftor HMD Postifttion Heaft Map tin fthe X-Z Plane

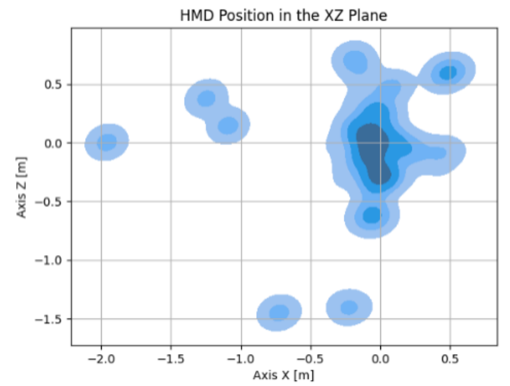

Figure 3.32: Beaft Saber HMD Postifttion Heaft Map tin fthe X-Z Plane

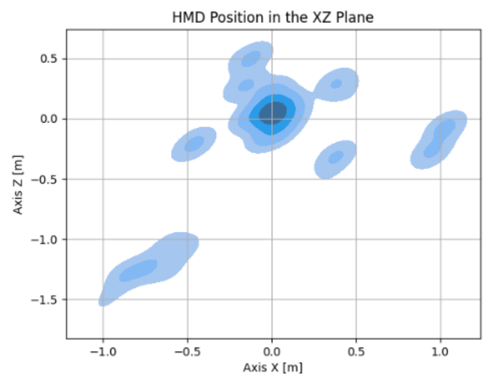

Figure 3.33: Forkltifft Stimulaftor HMD Postifttion Heaft Map tin fthe X-Z Plane

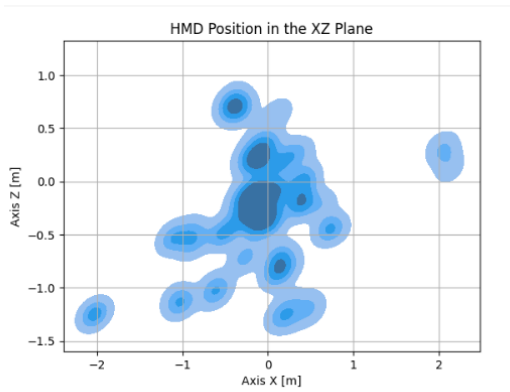

Figure 3.34: Medal of Honor HMD Postifttion Heaft Map tin fthe X-Z Plane

In Forklift Simulator (Figure 3.33), the heatmap shows a central high-density region with minimal spread. This suggests that users maintain a relatively stable head position, typical of a seated driving environment. Limited head movement reflects the controlled nature of steering and driving tasks, with central clustering indicating consistent head positioning.

For Medal of Honor (Figure 3.34), the heatmap displays multiple dense regions with significant spread, indicating varied and dynamic head movements. This reflects active engagement in aiming, shooting, and navigating the game environment. The significant spread along both the X and Z axes highlights extensive head movements typical of a first-person shooter. Multiple high-density areas suggest frequent repositioning of the head to engage with targets and navigate.

Comparing these games, Cooking Simulator and Medal of Honor show significant spread in head positions, reflecting dynamic and interactive tasks. In contrast, Beat Saber and Forklift Simulator exhibit more centralized high-density regions, indicating more stable head positions.

Cooking Simulator demonstrates diverse and interactive head movements, reflecting tasks that require looking around the kitchen. The forklift Simulator features stable and controlled head movements, consistent with a seated driving environment and slow-paced gameplay. Beat Saber shows centralized head movements with quick adjustments, reflecting rhythmic and repetitive slicing actions. Medal of Honor displays dynamic and varied head movements, indicating active engagement in shooting and navigating.

4

Evaluation Methodology And Results

This chapter presents and analyzes the evaluation methodology and results generated from the machine learning models used in Python. The models analyze four different games with different window sizes compare the accuracy with best-performing models and test the data by applying the best-performing model. The applied models are presented with confusion matrices for each game and all the window sizes. The performance graphs are given for each game according to the changing window sizes. Aimed to find out what accuracy performance was achieved in which window sizes and how reliable the identification performance could be.

4.1 Learning Methods

As the feature engineering process, to derive insightful features from the data, the dataset was divided into windows of time: 1, 3, 5, and 10 seconds with each window generating statistical features including the mean, minimum, and maximum values, as well as the trend (slope) and average differences over time.

After collecting the data correctly and organizing them, several Machine Learning algorithms were applied to train and test the data for identification. The used models are listed below:

Support Vector Machine It is a widely utilized Supervised Learning algorithm serving both Classification and Regression tasks.

The central objective of SVM is to craft an optimal line or decision boundary capable of partitioning an n-dimensional space into distinct classes. This delineation ensures the accurate categorization of new data points in subsequent instances. The pivotal construct in this process is the hyperplane, representing the optimal decision boundary.

SVM identifies critical points, known as support vectors, strategically positioned to contribute to the formation of the hyperplane. The algorithm's nomenclature, Support Vector Machine, is derived from the emphasis on these significant support vectors in defining the best possible decision boundary.[37]

We used two types of SVM:

1. Linear SVM: This SVM is used when the data is linearly separable.

2. Non-Linear SVM (RBF Kernel): This SVM is used when the data is not linearly separable.

Random Forest Classifier: A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.[38]

AdaBoost Classifier: A meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on dificult cases.[39]

Extra Trees Classifier: This class implements a meta estimator that fits several randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.[40]

Each model was evaluated using K-fold cross-validation to find the models' average training accuracy.

Model performance was primarily assessed using accuracy metrics and confusion matrices. The accuracy metric provided a straightforward indication of the model's overall effectiveness, while the confusion matrices offered detailed insight into the types of errors made by the models, such as false positives and false negatives.

The best-performing model was then identified based on the average accuracy across the cross-validation sets. This model was further trained on the entire training dataset and finally evaluated on a held-out test set to assess its performance on unseen data.

After the analysis, we obtained various types of visual outputs such as plots of rotation angles, position coordinates over time, head and hand movement values for each user, and heat maps of head and hand movement data.

The fundamental libraries used in these analyses are;

Pandas: For data manipulation and aggregation.

Numpy: For numerical operations.
Scikit-learn: For machine learning models and preprocessing.
Matplotlib and Seaborn: For plotting and visualizations.

4.2 Cooking Simulator Results

In this section, the results of the study on various machine learning models applied to the Cooking Simulator data with different window sizes are presented. The performance of the following models: Linear SVM, RBF SVM, Random Forest, AdaBoost, and Extra Trees were evaluated. The analysis was conducted with window sizes of 1, 3, 5, and 10. Table 4.1 provides an overview of the average accuracies of these models across different window sizes.

Models	1 second window	3 second window	5 second window	10 second window
Linear SVM	0.5153	0.5069	0.5029	0.4913
SVM RBF Kernel	0.6005	0.5708	0.5525	0.5246
Random Forest	0.8578	0.7981	0.7691	0.7299
AdaBoost	0.1791	0.1501	0.1462	0.1291
Extra Trees	0.8678	0.8074	0.7748	0.7431
Best Model	Extra Trees	Extra Trees	Extra Trees	Extra Trees
Test of Best Model	0.9050	0.8338	0.8040	0.7670

Table 4.1: Cookting Stimulaftor Accuracy Scores wtifth Wtindow Stizes

As illustrated in Table 4.1, the performance of each model varies with changes in window size. The Extra Trees model consistently outperformed the others, maintaining high accuracy across all window sizes, followed closely by the Random Forest model. The RBF SVM and Linear SVM models showed moderate performance, while the AdaBoost model had the lowest accuracy. Other than the Linear SVM, all the models had a big decrease, when the window size was increasing. For instance, Extra Trees on window size 1 had a 14.37% accuracy decrease on window size 10 while Linear SVM had a 4.657% decrease. With a larger window size, the model's complexity increases as it tries to split based on more features. This can lead to overfitting, where the model captures noise rather than the underlying patterns, resulting in a larger decrease in accuracy. Extra Trees work by creating multiple trees using random subsets of features. When the window size increases, there are more features, and many of them might be redundant. Extra Trees might overfit these redundant features, leading to a significant drop in performance. while Linear SVMs assume a linear relationship between features and the target variable. Increasing the window size adds more features, but as long as the relationship remains approximately linear, the SVM can handle this increase better.

4.2.1 Cooking Simulator Window Size 1

First, we applied the machine learning models on window size 1. In this window size, we observed the best accuracy among the different window sizes. The Extra Trees model emerged as the best-performing model with an average accuracy of 0.8678. When applied to the test data, this model achieved an accuracy of 0.9050.

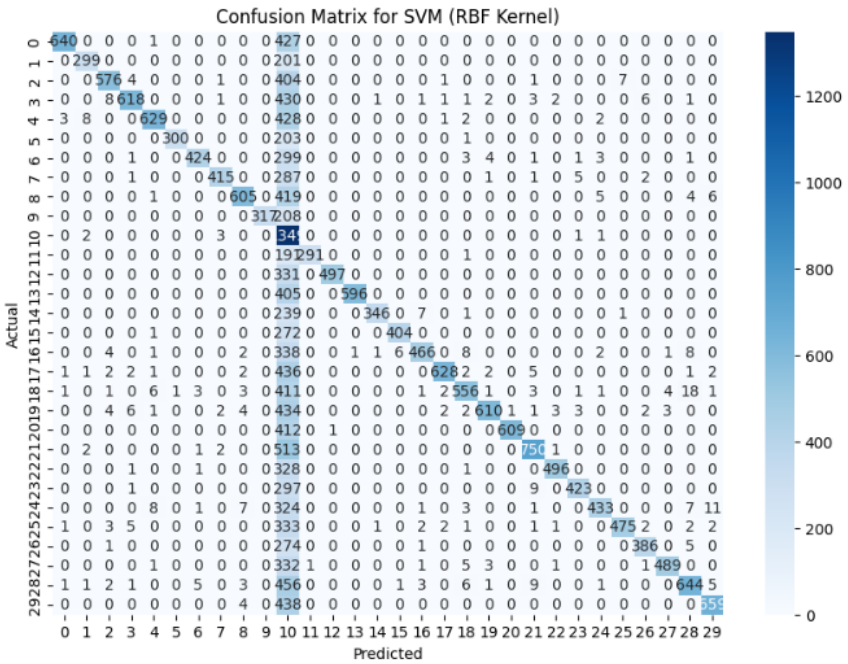

Figure 4.1: Cookting Stimulaftor Wtindow Stize 1 Confustion Maftrtix for SVM RBF

Figure 4.1 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Cooking Simulator data with window size 1.

Correctly classified instances in confusion matrices for each class are represented by the diagonal elements. When the model is performing well in predicting those users, they are indicated with high values on the diagonal. For instance, the highest accuracy is on user 10 with correctly classified 1349 instances. But also all the users were predicted as user 10 with a high rate, the reason for this is SVM focuses on finding the optimal hyperplane that maximizes the margin between classes. If one class (e.g., user 10) has more support vectors or is closer to the decision boundary, SVM might misclassify other classes as this dominant class.

Misclassifications are the non-zero values off the diagonal. There are some classes with higher misclassification, such as class 28, which indicates possible confusion between similar classes.

This model shows very high accuracy for some classes but still doesn't show strong performance for many classes.

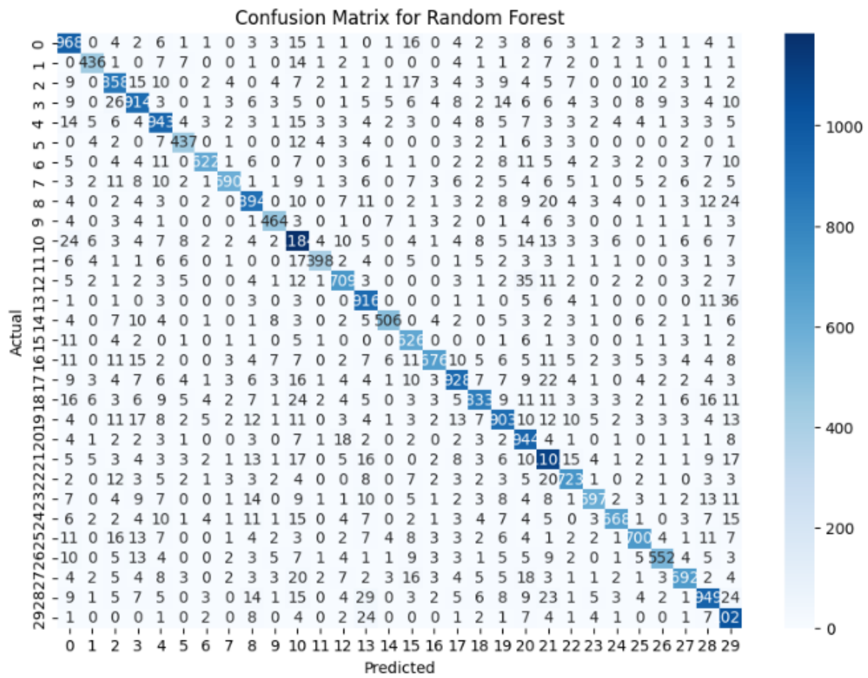

Figure 4.2: Cookting Stimulaftor Wtindow Stize 1 Confustion Maftrtix for Random Foresft

Figure 4.2 shows the confusion matrix for the Random Forest model applied to the Cooking Simulator data with window size 1. The Random Forest model generally shows better performance than both SVM models, as seen in the confusion matrix by higher counts on the diagonal and fewer off-diagonal misclassifications. The Random Forest model performs better than the SVM models, indicating its robustness in handling the complexities of the data.

Figure 4.3 shows the confusion matrix for the AdaBoost model applied to the Cooking Simulator data with window size 1. The AdaBoost model appears to be lower compared to the other models, as indicated by the relatively lower counts along the diagonal. There are so many non zero values off the diagonal and the values on the diagonal are very low to show a good performance. The ensemble nature of AdaBoost, while powerful in some contexts, does not seem to handle the complexities of the Cooking Simulator data as effectively as the Random Forest model. The absence of a prominent vertical line in the AdaBoost confusion matrix compared to the SVM indicates that AdaBoost is better at balancing predictions across different classes. This can be attributed to its ensemble nature, which corrects misclassifications iteratively and reduces the impact of any single class dominating the predictions. This balanced performance is a key strength of AdaBoost in handling diverse and potentially imbalanced datasets.

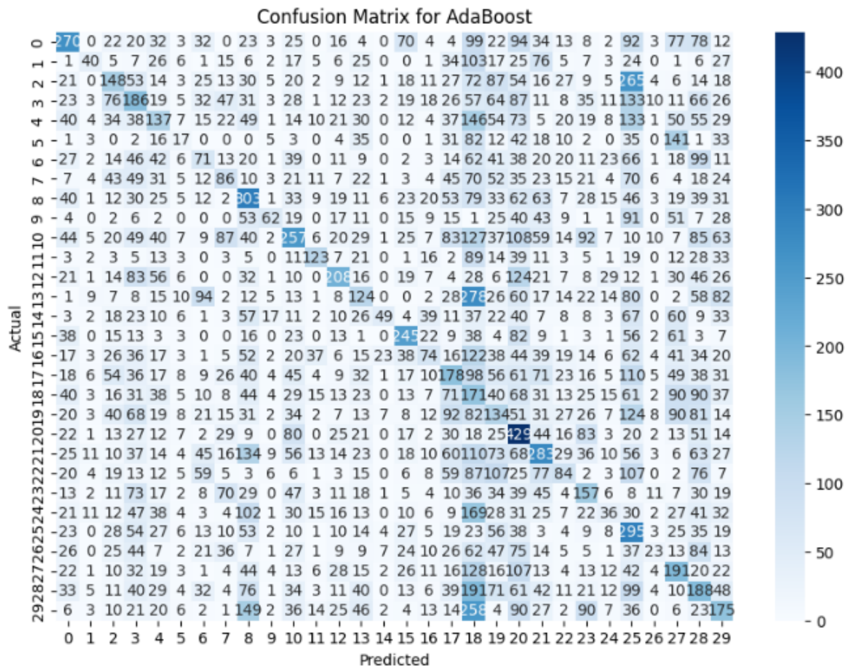

Figure 4.3: Cookting Stimulaftor Wtindow Stize 1 Confustion Maftrtix for AdaBoosft

Figure 4.4 shows the confusion matrix for the Extra trees model applied to the Cooking Simulator data with window size 1. For instance the classes 10, 21 show excellent performance with high values on diagonal. Most of the classes have high diagonal values that show the robustness of the Extra Trees model in predicting the majority of the classes. In Cooking Simulator data with the window size 1, the best performing model is Extra Trees.

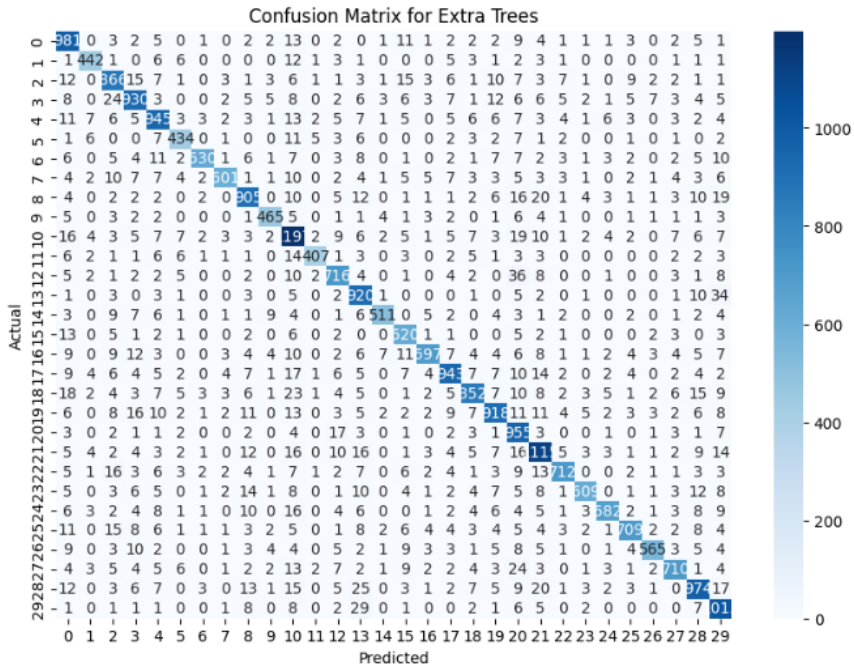

Figure 4.4: Cookting Stimulaftor Wtindow Stize 1 Confustion Maftrtix for Exftra Trees

4.2.2 Cooking Simulator with Longer Windows

Secondly we applied the machine learning models on window size 3. In this part, the best-performing model is Extra Trees with an average accuracy of 0.8074 and the test accuracy when it is applied to Extra Trees is 0.8338.

Confusion matrices are very similar to window size 1. As shown in Table 4.1, all the models are performing better in window size 1 than in window size 3. As same in window size 1, Extra Trees and Random Forest models are performing well, SVM RBF Kernel is slightly better than Linear SVM and the AdaBoost model is not a suitable model for this data and the window size.

After window size 3 we applied the machine learning models on window size 5. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7748 and the test accuracy when it is applied to Extra Trees is 0.8040.

Confusion matrices are very similar to window size 1 as well as window size 3. As shown in Table 4.1, all the models are performing better in window size 1 than in window size 3 and window size 5. The linear SVM model performance in window size 5 is almost the same as in window size 3. As in window sizes 1 and 3, Extra Trees and Random Forest models are performing well, SVM RBF Kernel is slightly better than Linear SVM and the AdaBoost model is not a suitable model for this data and the window size.

Finally, we applied the machine learning models on window size 10. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7431 and the test accuracy when applied to Extra Trees is 0.7670.

Figure 4.5 shows the confusion matrix for the Ada Boost model applied to the Cooking Simulator data with window size 10. Lowest performance among others. In this confusion matrix, user 19 is predicted more frequently and is different from the one in window size 1. When the window size increases from 1 to 10, the number of features grows significantly. This can lead to a higher dimensional feature space where the relationships between the features become more complex. If user 19's feature vectors are particularly distinct or dominant in this high-dimensional space, the model might be biased toward predicting user 19 more frequently.

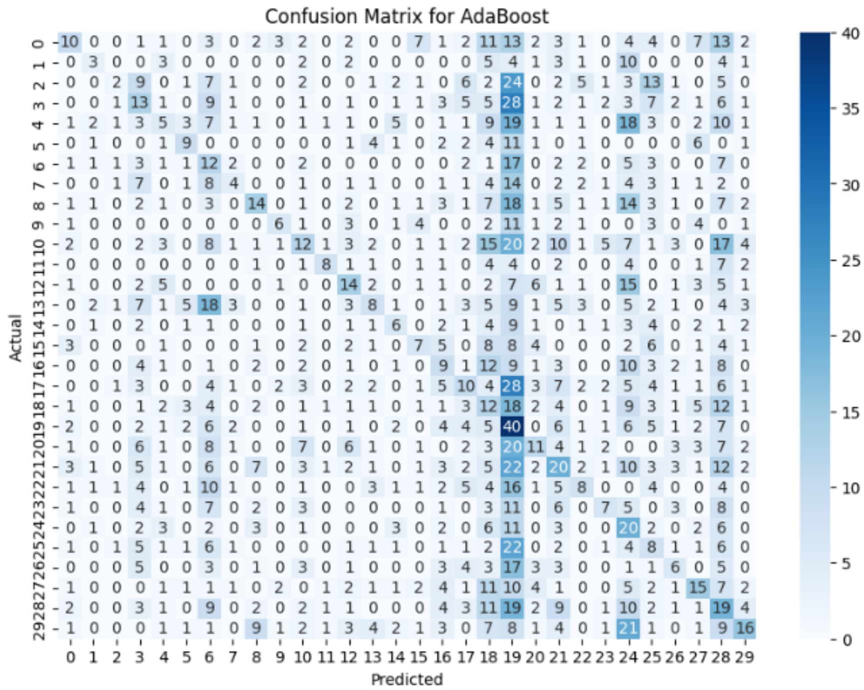

Figure 4.5: Cookting Stimulaftor Wtindow Stize 10 Confustion Maftrtix for AdaBoosft

4.3 Beat Saber Results

In this section, the results of the study on various machine learning models applied to the Beat Saber data with different window sizes are presented. The performance of the following models: Linear SVM, RBF SVM, Random Forest, AdaBoost, and Extra Trees were evaluated. The analysis was conducted with window sizes of 1, 3, 5, and 10. Table 4.2 provides an overview of the average accuracies of these models across different window sizes.

As illustrated in Table 4.2, the performance of each model varies with changes in window size. The Extra Trees and the Random Forest model consistently outperformed the others, maintaining high accuracy across all window sizes. But except the window size 10, the best performing model for window sizes 1, 3, and 5, was the Random Forest model. The RBF SVM and the Linear SVM models showed moderate performance, while the AdaBoost model had the lowest accuracy.

Models	1 second window	3 second window	5 second window	10 second window
Linear SVM	0.4996	0.5009	0.4990	0.4919
SVM RBF Kernel	0.5668	0.5489	0.5354	0.5100
Random Forest	0.8300	0.7581	0.7345	0.7039
AdaBoost	0.2505	0.2153	0.2141	0.2045
Extra Trees	0.8263	0.7550	0.7318	0.7089
Best Model	Random Forest	Random Forest	Random Forest	Extra Trees
Test of Best Model	0.8499	0.7753	0.7416	0.7147

Table 4.2: Beaft Saber Accuracy Scores wtifth Wtindow Stizes

4.3.1 Beat Saber Window Size 1

First we applied the machine learning models on window size 1. In this window size, we took the best accuracy among the other window sizes. The best-performing model is Random Forest with an average accuracy of 0.83 and the test accuracy when applied to Extra Trees is 0.8499.

Figure 4.6 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Beat Saber data with window size 1. The model shows strong performance with good classification accuracy for several classes. Misclassifications are noted in classes like 0, 3, and 14. Some classes like 2 and 18 have high precision. The picked user is user 4. The reason for this the RBF kernel in SVM aims to find a decision boundary that maximizes the margin between classes. If the support vectors for user 4 are positioned in a way that they are close to the feature spaces of other users, the decision boundary might be skewed, leading to frequent misclassification as user 4.

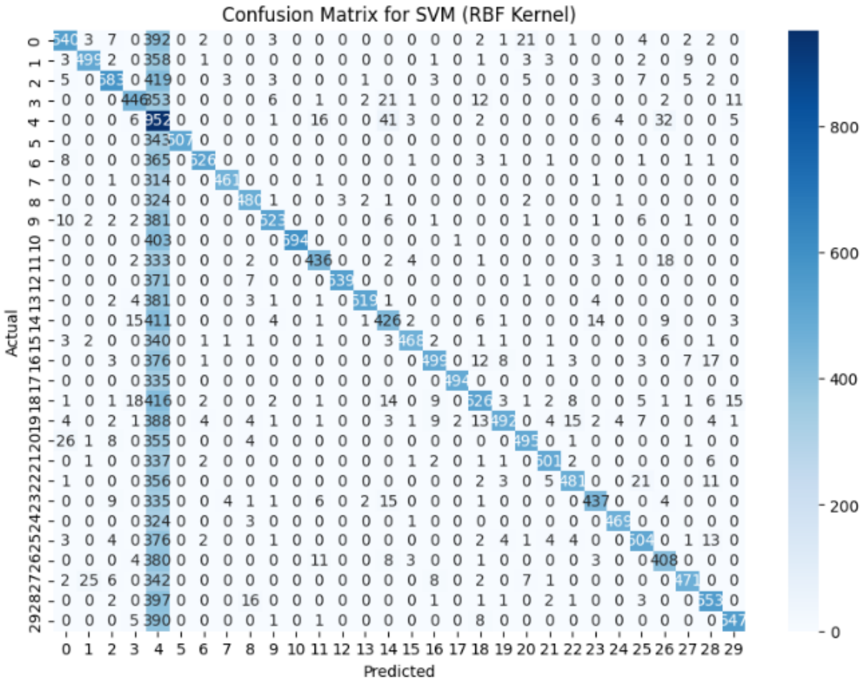

Figure 4.6: Beaft Saber Wtindow Stize 1 Confustion Maftrtix for SVM RBF

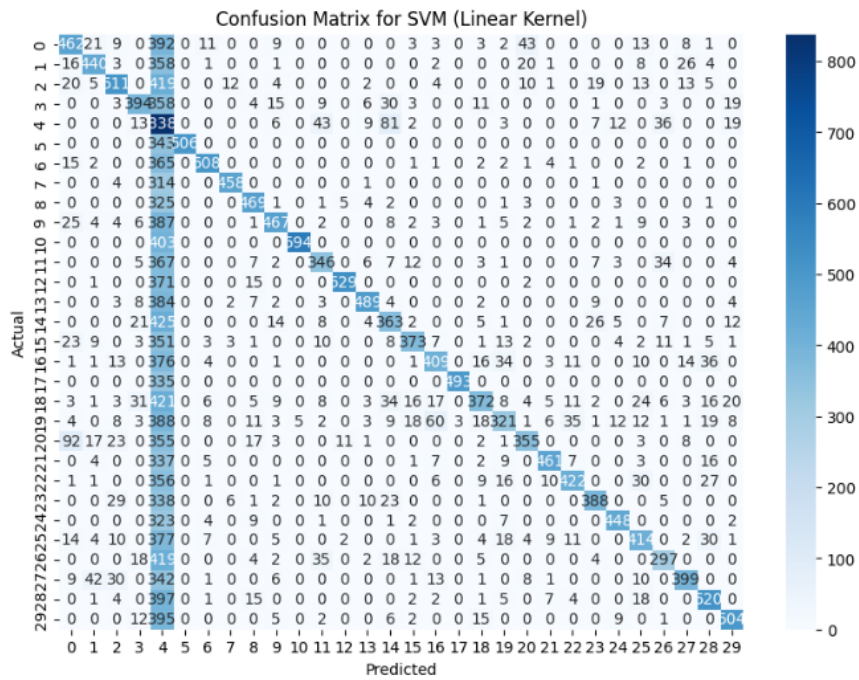

Figure 4.7: Beaft Saber Wtindow Stize 1 Confustion Maftrtix for SVM

Figure 4.7 shows the confusion matrix for the SVM Linear Kernel model applied to the Beat Saber data with window size 1. Reasonable performance with good classification accuracy for several classes. Misclassifications and high performing classes were noted in the same classes as RBF Kernel. The confusion matrix is almost the same as SVM RBF. User 4 is picked as in SVM RBF kernel.

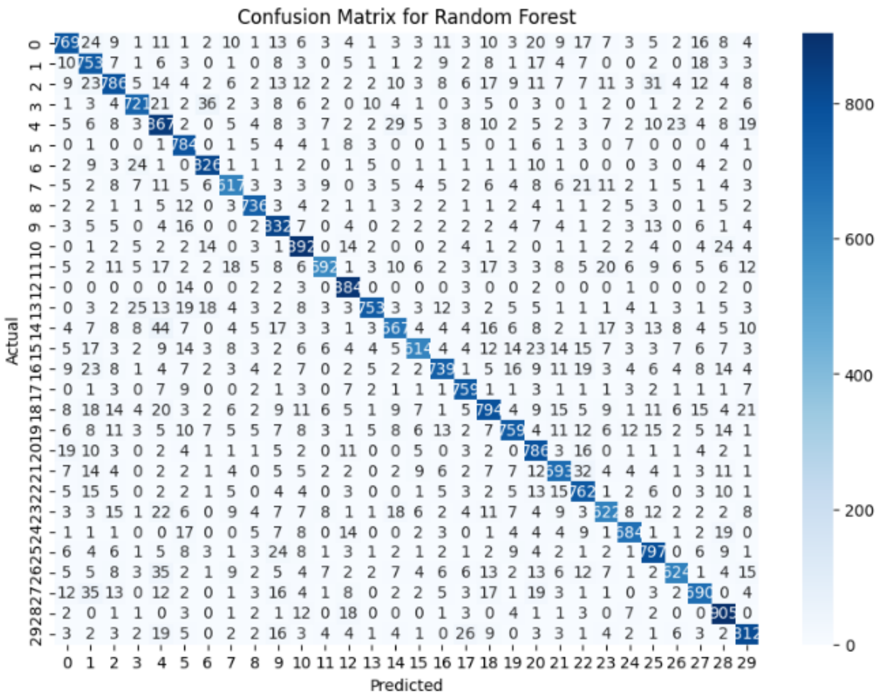

Figure 4.8: Beaft Saber Wtindow Stize 1 Confustion Maftrtix for Random Foresft

Figure 4.8 shows the confusion matrix for the Random Forest model applied to the Beat Saber data with window size 1. High accuracy for classes 10, 12, and 28, with minimal misclassification.

Figure 4.9 shows the confusion matrix for the Ada Boost model applied to the Beat Saber data with window size 1. Lower overall performance compared to other models. It has higher misclassification rates across multiple classes. Struggles significantly with classes like 6, 11, and 20. This model has limited strengths, showing good performance in very few classes. Significant weaknesses, with many misclassifications suggesting that AdaBoost may not be well-suited for this dataset.

Figure 4.9: Beaft Saber Wtindow Stize 1 Confustion Maftrtix for AdaBoosft

4.3.2 Beat Saber with Longer Windows

Secondly we applied the machine learning models on window size 3. In this part, the best-performing model is Random Forest with an average accuracy of 0.7581 and the test accuracy when applied to Random Forest is 0.7753.

Figure 4.10 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Beat Saber data with window size 3. Classes like 2, 28, and 29 show high accuracy with minimal misclassifications and the overall performance is good. In this confusion matrix. different than the window size 1, user 19 is picked too. It is not misclassified as much as user 4, but the reason for this can be the increase in the window size increases the number of features, leading to a more complex feature space. If user 19's features dominate this high-dimensional space, the model might be more likely to predict user 19 in undefined cases.

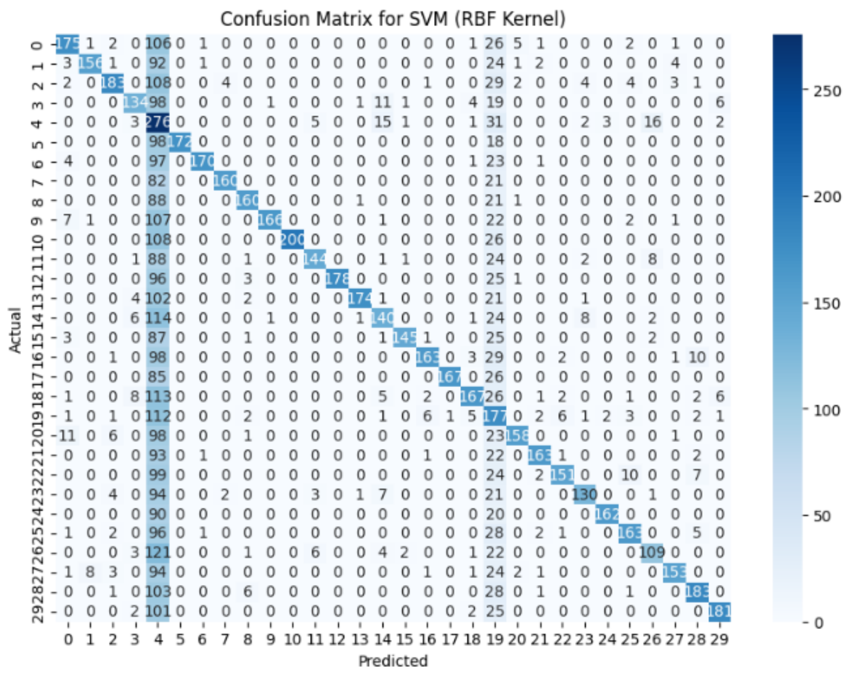

Figure 4.10: Beaft Saber Wtindow Stize 3 Confustion Maftrtix for SVM RBF

After window size 3 we applied the machine learning models on window size 5. In this part, the best-performing model is Random Forest with an average accuracy of 0.7345 and the test accuracy when applied to Random Forest is 0.7416.

Figure 4.11 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Beat Saber data with window size 5. Classes like 10, 28, and 29 show high accuracy with minimal misclassification but the overall performance is good. As well as in SVM Linear Kernel, in this window size, this time user 18 is the most misclassified after user 4, not user 19. The model might have overfitted to the patterns of user 18 during training. This can happen if user 18's data has distinctive but not unique characteristics that the model learned too well, leading to generalization issues.

Figure 4.12 shows the confusion matrix for the Ada Boost model applied to the Beat Saber data with window size 5. The lowest overall performance compared to other models. Struggles with most of the classes. In this window size, user 19 is predicted the most even not the true positivity rate is low in AdaBoost. The model might have overfitted to patterns in user 19's data during training. Overfitting can occur if the model learns specific characteristics of user 19 too well, leading to poor generalization to other users.

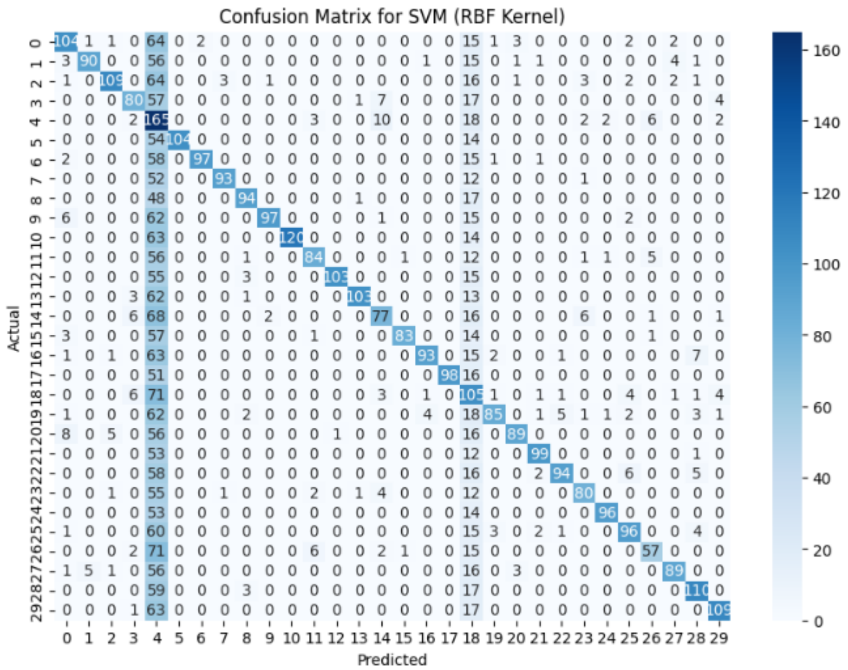

Figure 4.11: Beaft Saber Wtindow Stize 5 Confustion Maftrtix for SVM RBF

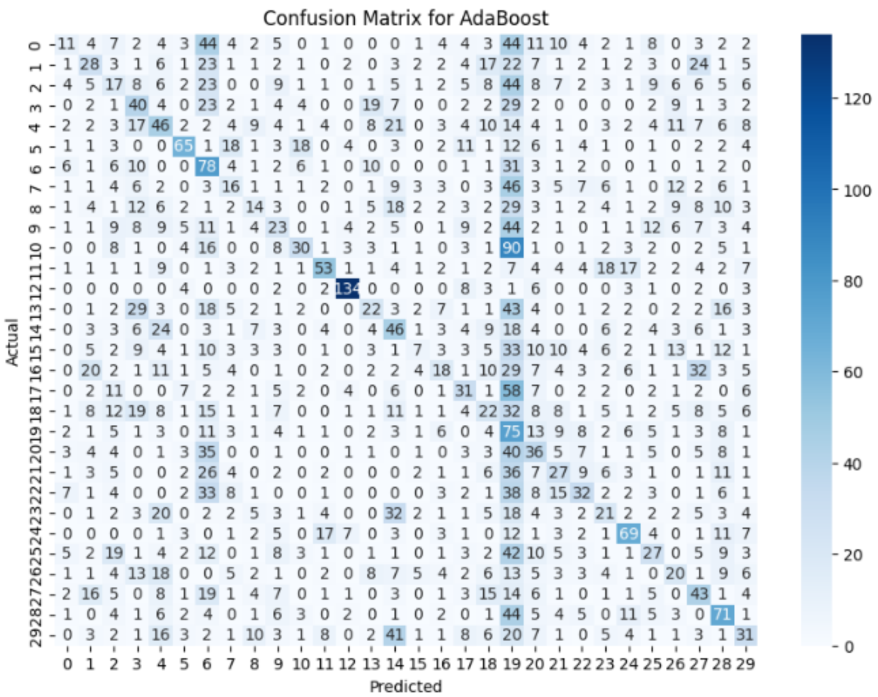

Figure 4.12: Beaft Saber Wtindow Stize 5 Confustion Maftrtix for AdaBoosft

Finally, we applied the machine learning models on window size 10. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7089 and the test accuracy when applied to Extra Trees is 0.7147.

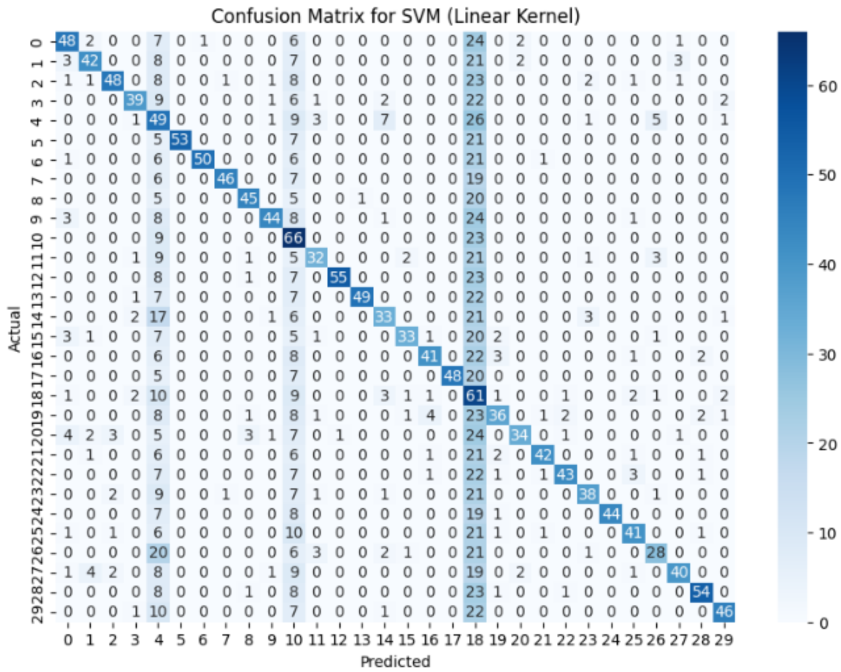

Figure 4.13: Beaft Saber Wtindow Stize 10 Confustion Maftrtix for SVM

Figure 4.13 shows the confusion matrix for the SVM Linear Kernel model applied to the Beat Saber data with window size 10. The same classes have similar high values and high misclassification rates with SVM Kernel. Performs lower than the RBF SVM. In this window size user 18 is the most predicted and after 18, 4, and 10 are predicted incorrectly. The reason for this, with a window size of 10, the feature space becomes more complex as it captures more temporal dependencies and patterns.   If user 18's, 10's, and the 4's patterns overlap significantly with those of other users, the models might be biased towards predicting these users.

Figure 4.14 shows the confusion matrix for the Ada Boost model applied to the Beat Saber data with window size 10. The lowest overall performance compared to other models. Struggles with most of the classes. User 19 is not as much predicted as in window size 5.

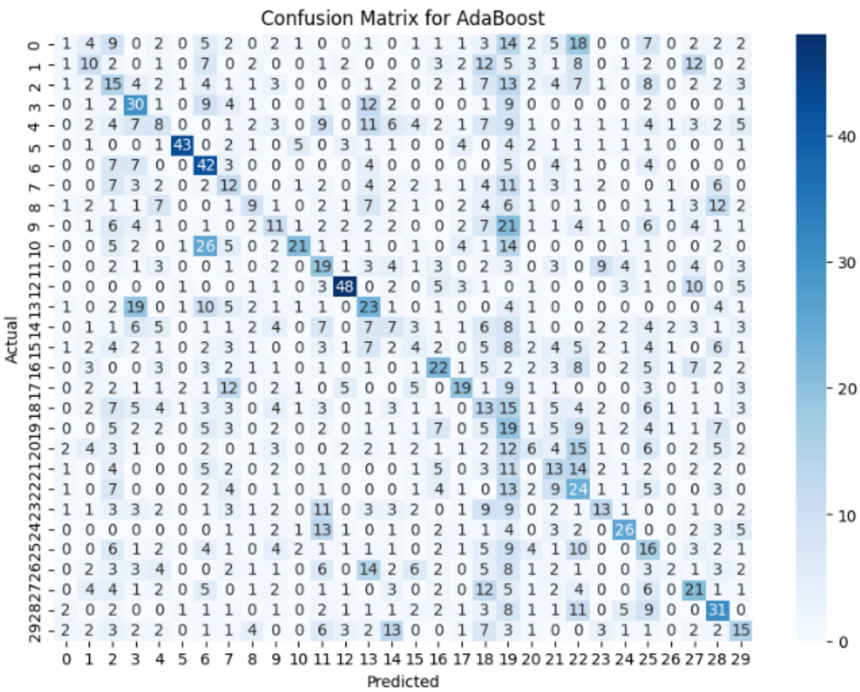

Figure 4.14: Beaft Saber Wtindow Stize 10 Confustion Maftrtix for AdaBoosft

4.4 Forklift Simulator Results

In this section, the results of the study on various machine learning models applied to the Forklift Simulator data with different window sizes are presented. The performance of the following models: Linear SVM, RBF SVM, Random Forest, AdaBoost, and Extra Trees were evaluated. The analysis was conducted with window sizes of 1, 3, 5, and 10. Table 4.3 provides an overview of the average accuracies of these models across different window sizes.

As illustrated in Table 4.3, the performance of each model varies with changes in window size. The Extra Trees and the Random Forest model consistently outperformed the others, maintaining high accuracy across all window sizes. But except the window size 3, the best performing model for window sizes 1, 5, and 10, was the Extra Trees model. The RBF SVM and the Linear SVM models showed moderate performance and also almost the same performance on every window size, while the AdaBoost model had the lowest accuracy.

Models	1 second window	3 second window	5 second window	10 second window
Linear SVM	0.6043	0.5992	0.5942	0.5914
SVM RBF Kernel	0.6084	0.5988	0.5911	0.5833
Random Forest	0.8756	0.8335	0.8106	0.7920
AdaBoost	0.1550	0.1484	0.1313	0.1228
Extra Trees	0.8786	0.8323	0.8136	0.8002
Best Model	Extra Trees	Random Forest	Extra Trees	Extra Trees
Test of Best Model	0.9033	0.8491	0.8431	0.8138

Table 4.3: Forkltifft Stimulaftor Accuracy Scores wtifth Wtindow Stizes

4.4.1 Forklift Simulator Window Size 1

First we applied the machine learning models on window size 1. In this window size, we took the best accuracy among the other window sizes. The best-performing model is Extra Trees with an average accuracy of 0.8786 and the test accuracy when it is applied to Extra Trees is 0.9033.

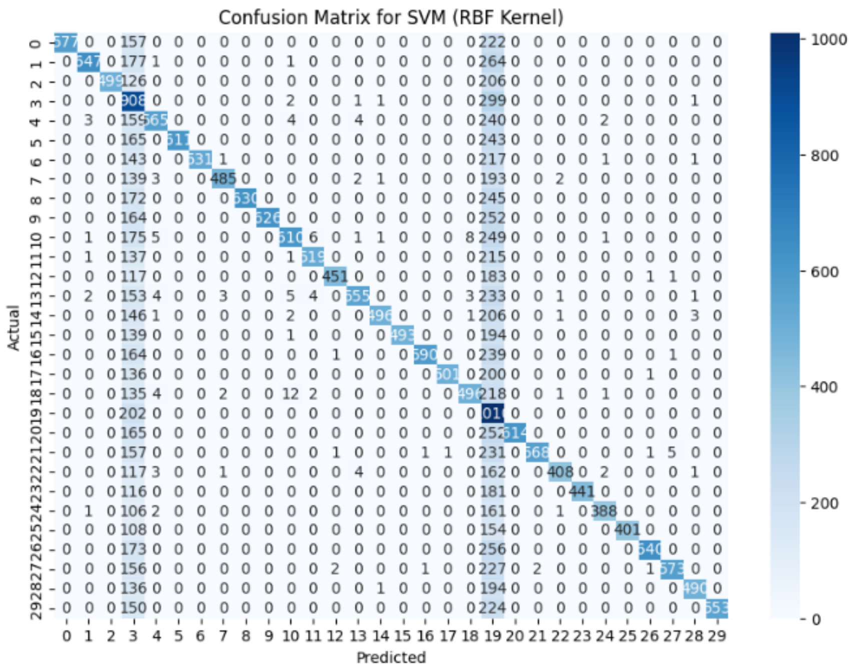

Figure 4.15: Forkltifft Stimulaftor Wtindow Stize 1 Confustion Maftrtix for SVM RBF

Figure 4.15 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Forklift Simulator data with window size 1. Generally, it performs well for window size 1. The most misclassified users are user 3 and the user 19. The reason for this can be the placement of support vectors near the decision boundaries and they can significantly influence the predictions. If many support vectors belong to specific users, the model may skew towards these users in ambiguous cases. The Linear SVM confusion matrix also has the same features.

4.4.2 Forklift Simulator with Longer Windows

Secondly we applied the machine learning models on window size 3. In this part, the best-performing model is Random Forest with an average accuracy of 0.8335 and the test accuracy when it is applied to Random Forest is 0.8491. All the model confusion matrices have the same features as in window size 1.

After window size 3 we applied the machine learning models on window size 5. In this part, the best-performing model is Extra Trees with an average accuracy of 0.8136 and the test accuracy when applied to Extra Trees is 0.8431.

Finally, we applied the machine learning models to window size 10. In this part, the best-performing model is Extra Trees with an average accuracy of 0.8002 and the test accuracy when applied to Extra Trees is 0.8138. The features of the confusion matrices for all models are not very different from each other in all window sizes.

4.5 Medal of Honor Results

In this section, the results of the study on various machine learning models applied to the Medal of Honor data with different window sizes are presented. The performance of the following models: Linear SVM, RBF SVM, Random Forest, AdaBoost, and Extra Trees were evaluated. The analysis was conducted with window sizes of 1, 3, 5, and 10. Table 4.4 provides an overview of the average accuracies of these models across different window sizes.

As illustrated in Table 4.4, the performance of each model varies with changes in window size. The Extra Trees and the Random Forest model consistently outperformed the others, maintaining high accuracy across all window sizes. But for all the window sizes the best performing model was the Extra Trees model. The RBF SVM and the Linear SVM model showed moderate performance, while the AdaBoost model had the lowest accuracy.

Models	1 second window	3 second window	5 second window	10 second window
Linear SVM	0.5036	0.4916	0.4869	0.4765
SVM RBF Kernel	0.5864	0.5495	0.5280	0.4997
Random Forest	0.8451	0.7725	0.7378	0.6925
AdaBoost	0.1521	0.1179	0.0979	0.0727
Extra Trees	0.8559	0.7883	0.7565	0.7157
Best Model	Extra Trees	Extra Trees	Extra Trees	Extra Trees
Test of Best Model	0.8952	0.8058	0.7767	0.7292

Table 4.4: Medal of Honor Accuracy Scores wtifth Wtindow Stizes

4.5.1 Medal of Honor Window Size 1

First we applied the machine learning models on window size 1. We took the best accuracy in this window size among the other window sizes. The best-performing model is Extra Trees with an average accuracy of 0.8559 and the test accuracy when applied to Extra Trees is 0.8952.

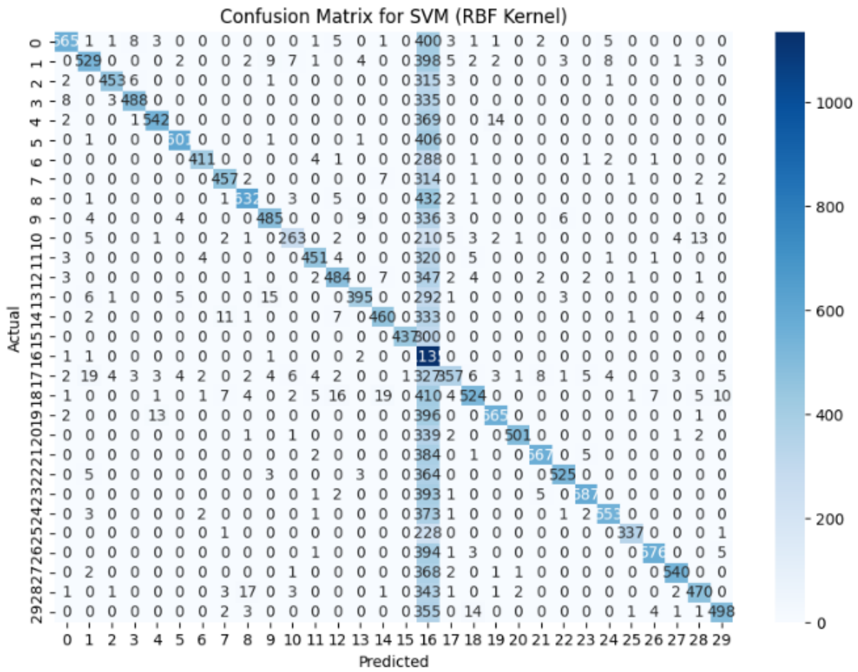

Figure 4.16: Medal of Honor Wtindow Stize 1 Confustion Maftrtix for SVM RBF

Figure 4.16 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Medal of Honor data with window size 1. Same as Liner SVM, user 16 was predicted with a high rate, the reason for this is SVM focuses on finding the optimal hyperplane that maximizes the margin between classes. User 16 might have more support vectors or is closer to the decision boundary, SVM might misclassify other classes as this user.

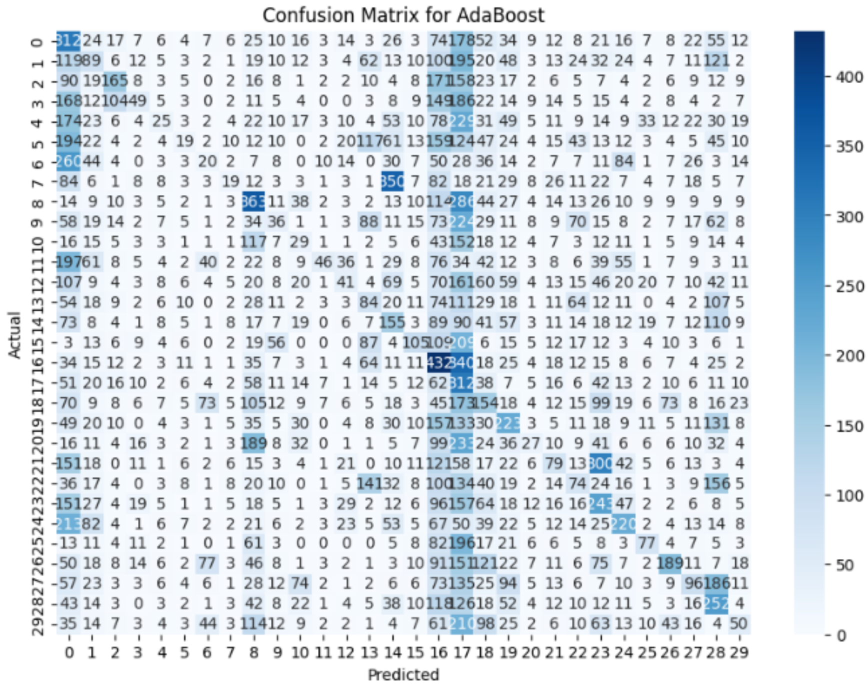

Figure 4.17: Medal of Honor Wtindow Stize 1 Confustion Maftrtix for AdaBoosft

Figure 4.17 shows the confusion matrix for the Ada Boost model applied to the Medal of Honor data with window size 1. There are many incorrectly predicted users, such as users 0, 16, and 17. Users may have feature vectors that are highly similar, leading to confusion by the model. When features of different users overlap significantly, the model struggles to distinguish between them.

4.5.2 Medal of Honor with Longer Windows

We applied the machine learning models on window size 3. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7883 and the test accuracy when it is applied to Extra Trees is 0.8058.

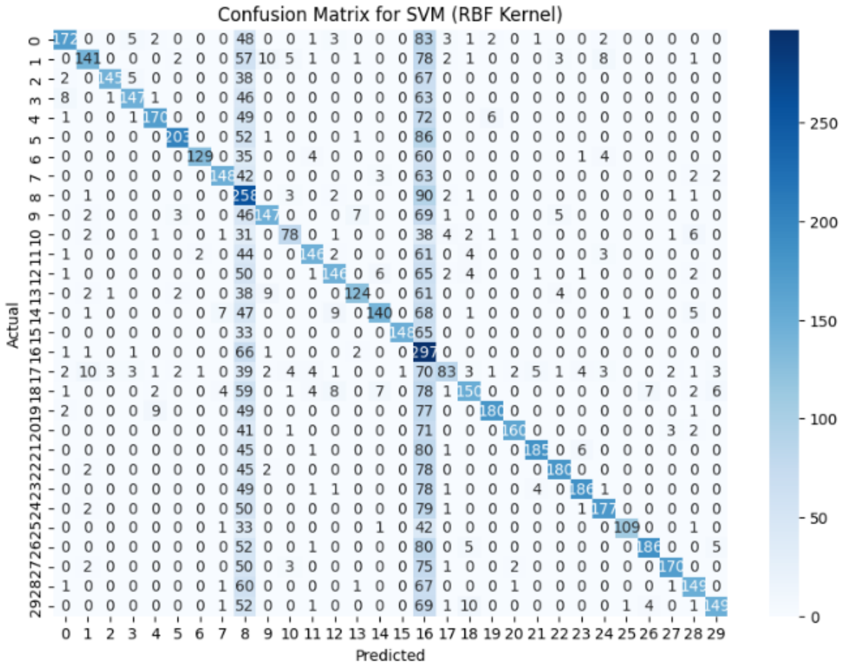

Figure 4.18: Medal of Honor Wtindow Stize 3 Confustion Maftrtix for SVM RBF

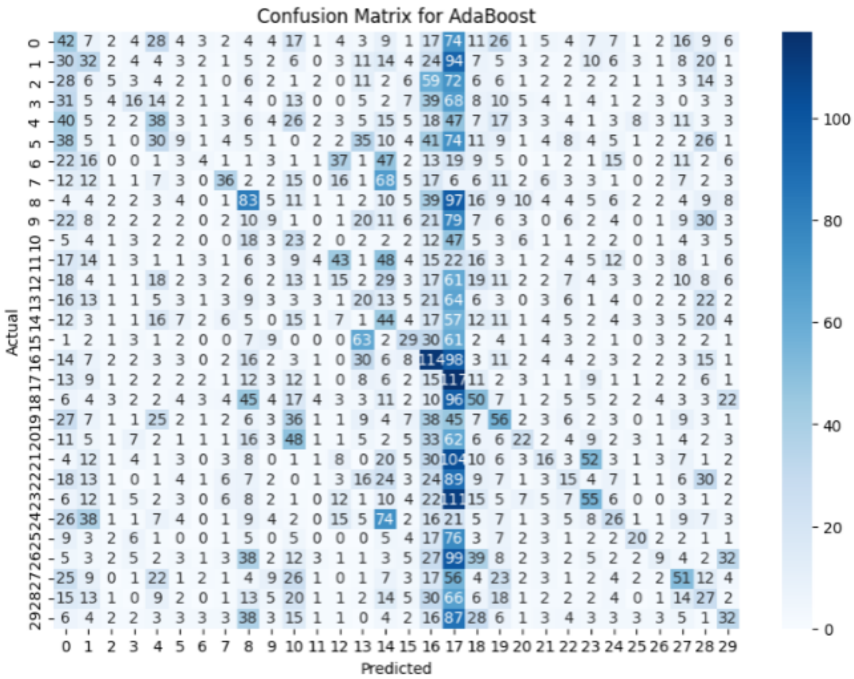

Figure 4.19: Medal of Honor Wtindow Stize 3 Confustion Maftrtix for AdaBoosft

Figure 4.18 shows the confusion matrix for the SVM (RBF Kernel) model applied to the Medal of Honor data with window size 3. In window size 3, other than user 16, user 8 is also misclassified. With a window size of 3, more temporal data is captured, which increases the number of features. If the added features do not help in distinguishing users better but instead increase the overlap between feature vectors of different users, this can lead to more misclassification.

Figure 4.19 shows the confusion matrix for the Ada Boost model applied to the Medal of Honor data with window size 3. Now user 17 is misclassified more clearly than window size 1. The reason for this can be a larger window size captures more temporal dependencies and patterns. If user 17's behavior exhibits strong, distinctive patterns over this larger window, the model might generalize these patterns across other users.

After window size 3 we applied the machine learning models on window size 5. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7565 and the test accuracy when applied to Extra Trees is 0.7767. There is no eye-catching difference in this window size.

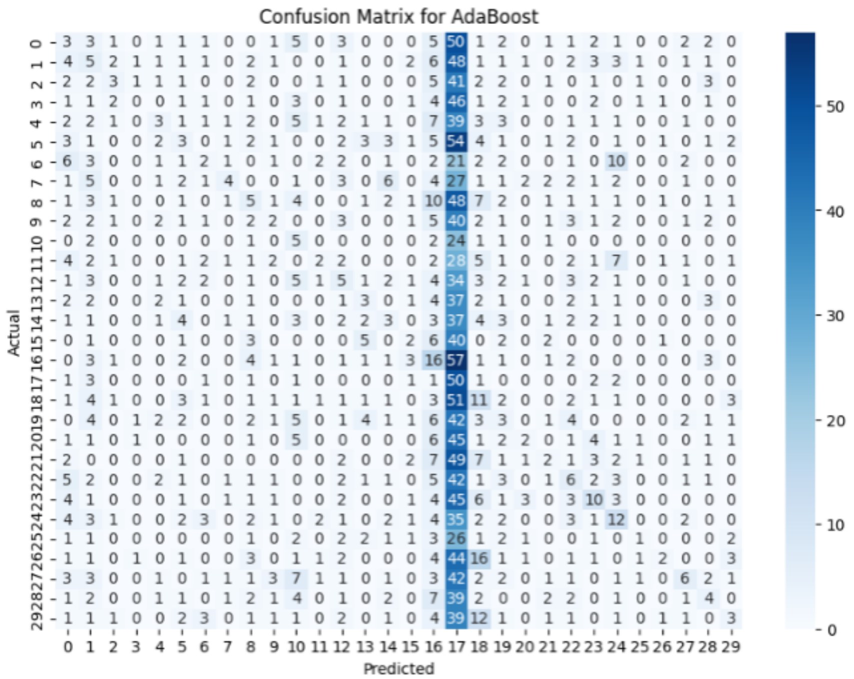

Figure 4.20: Medal of Honor Wtindow Stize 10 Confustion Maftrtix for AdaBoosft

Finally, we applied the machine learning models to window size 10. In this part, the best-performing model is Extra Trees with an average accuracy of 0.7157 and the test accuracy when it is applied to Extra Trees is 0.7292.

Figure 4.21 shows the confusion matrix for the Ada Boost model applied to the Medal of Honor data with window size 10. This time, different than the window size 1, user 17 is misclassified very clearly. Users may have feature vectors that are highly similar, when the window size is increasing, the model struggles to distinguish between them.

Across all datasets and window sizes, ensemble Models (Extra Trees and Random Forest) consistently showed the best performance. These models leverage the power of multiple decision trees, which helps in capturing complex patterns and relationships within the data. SVM Models (Linear and RBF Kernel) displayed moderate performance. The SVM models are effective in high-dimensional spaces and when the number of features is greater than the number of samples. AdaBoost was consistently the worst-performing model across all datasets and window sizes. This suggests that AdaBoost may not handle the complexity in the gaming datasets well, potentially due to its iterative nature which might amplify misclassifications.

For almost all the models, increasing the window size led to a decrease in accuracy. This is likely due to the increase in the number of features, which adds complexity and can result in overfitting. In some cases, such as with the SVM RBF model, certain classes were frequently misclassified as a dominant class. This suggests that the model's decision boundary may be influenced heavily by certain users whose data points are closer to the margin. Certain users were frequently misclassified, indicating that the models might be overfitting to specific users' data patterns. This is evident in the confusion matrices where specific users are predicted more often than others.

5

Conclusion

This thesis explores the potential of using VR user movement data as a biometric identifier by analyzing head and hand movements. The study involved sixty participants divided into two groups, each experiencing two different orders of four games: Cooking Simulator, Beat Saber, Forklift Simulator, and Medal of Honor. The movement data was processed with vari-ous machine learning models, including Linear Support Vector Machine (SVM), Non-Linear SVM (RBF Kernel), AdaBoost, Random Forest, and ExtraTrees, and was evaluated across window sizes of 1, 3, 5, and 10 seconds. Group 1 played Cooking Simulator and Beat Saber as one slow and one fast game, while Group 2 played Forklift Simulator and Medal of Honor. Order 1, first played the slow game in their group, while Order 2 played the fast game first. After the analysis, the results showed that the order difference didn't show a big difference in movement analysis. Different groups also didn't make a big difference in the identification process. The identification accuracy results are not very different from each other. However, the slow and fast games differ in identification. When we analyzed the head and hand movement data, slow games were easier to predict. Because the users tend to move more deliberately and consistently. These deliberate actions lead to more stable and predictable patterns, making it easier for machine learning models to learn and identify unique user signatures. Slow movements reduce the likelihood of sudden, unpredictable changes in direction or speed. This results in cleaner data and typically they have longer durations for each interaction. This extended time allows for the collection of more data points, providing a richer dataset for training and identifying unique user behaviors. In fast games, participants do not generally change their positions, especially in Beat Saber heat map results show that the participants are mostly located in the central resting position. So as the lowest identification accuracy, Beat Saber is obtained, and after the Beat Saber, the second lowest accuracy is obtained from the other fast game Medal of Honor. The Forklift Simulator has the highest accuracy and the second is the Cooking Simulator.

The machine learning models that are used in this thesis are Linear Support Vector Machine (SVM), Non-Linear SVM (RBF Kernel), AdaBoost, Random Forest, and ExtraTrees.The ExtraTrees model consistently outperformed others across all window sizes, Random Forest followed the Extra Trees as the second-best-performing model. Their performances were not far away from each other, Extra Trees was just slightly higher than Random Forest. Linear and RBF Kernel SVM models performed the best in the Forklift Simulator identification process. SVM RBF Kernel always performed better than the Linear SVM for all the games. But both of the SVM models' accuracy scores for all the window sizes changed between 49% and 60%, so after Extra Trees and Random Forest models, SVM models are not the best models to choose for the identification process for this data. AdaBoost had the lowest accuracy for all the games and in all the window sizes. So AdaBoost is not a suitable model for this data.

Smaller window sizes (1 second) yielded higher accuracy compared to larger window sizes (10 seconds), likely due to the ability to capture movement details. Smaller window sizes provide more granular data and a larger number of samples, both of which contribute to higher accuracy. Except for the Linear SVM, there wasn't a change in Beat Saber accuracy while the window sizes were changing. In other games too, the most stable model for changing window sizes was Linear SVM.

In conclusion, slow games (Cooking Simulator and Forklift Simulator) generally result in higher accuracy. This is likely due to more predictable and stable movement patterns, making it easier for models to distinguish between users. Fast games (Beat Saber and Medal of Honor) had more dynamic and varied movements, which increased the complexity of distinguishing between users but still maintained relatively high accuracy. The best-performing models for this study are Extra Trees and Random Forest with a window size of 1. Based on this study, it can be concluded that head and hand movements have the potential to serve as reliable biometric identifiers, achieving 90% accuracy in user identification. With further model enhancements and a broader study, the reliability of these identifiers can be significantly improved.

References

[1] Antoinette Price. The role of international standards in virtual education and training systems. [Online]. Available: https://etech.iec.ch/issue/2020-03/the-role-of-international-standards-in-virtual-education-and-training-systems

[2] Eurogamer.net. Cooking Simulator VR serves up a huge portion of fun. [Online]. Available: https://www.eurogamer.net/articles/2021-08-08-cooking-simulator-vr-serves-up-a-huge-portion-of-fun

[3] Gillies, M. Beat Saber. Virtual Reality | Virtual People. [Online]. Available: https://medium.com/virtual-reality-virtual-people/beat-saber-63a96e198f63

[4] Nintendo. Forklift Simulator. Nintendo. [Online]. Available: https://www.nintendo.com/it-ch/Giochi/Giochi-scaricabili-per-Nintendo-Switch/Forklift-Simulator-2078578.html

[5] Valve Corporation. Medal of Honor: Above and Beyond. [Online]. Available: https://store.steampowered.com/app/1402320/Medal_of_Honor_Above_and_Beyond/.

[6] Thomas Knight. Oculus Quest 2 / Meta Quest 2 – Hardware Review | A Not Too Technical VR Review. [Online]. Available: https://www.nookgaming.com/oculus-quest-2-hardware-review/

[7] learn gienc. Meta Quest 2 User Guide. [Online]. Available: https://learn.gienc.org/course/meta-quest-2-user-guide/meta-quest-2-user-guide

[8] Q. ai-Powering a Personal Wealth Movement. VR Headset Sales Underperform Expectations, What Does It Mean For The Metaverse In 2023? Section: Money. [Online]. Available: https://www:forbes:com/sites/qai/2023/01/06/vr-headset-sales-underperformexpectations-what-does-it-mean-for-themetaverse-in-2023/.

[9] e. a. Liebers, Jonathan, “Kinetic signatures: A systematic investigation of movement-based user identification in virtual reality.” p. 1–19, 2024. [Online]. Available: https://dl.acm.org/doi/full/10.1145/3613904.3642471

[10] D. A. Bowman and R. P. McMahan, “Virtual reality: how much immersion is enough?” *Computer*, vol. 40, no. 7, pp. 36–43, 2007.

[11] H. Cavusoglu, A. R. Dennis, and J. Parsons, “Immersive systems,” *Journal of Management Information Systems*, vol. 36, no. 3, pp. 680–682, 2019.

[12] F. P. Brooks, “What’s real about virtual reality?” *IEEE Computer graphics and applications*, vol. 19, no. 6, pp. 16–27, 1999.

[13] W. R. Sherman and A. B. Craig, “Understanding virtual reality,” *San Francisco, CA: Morgan Kauffman*, 2003.

[14] D. Raitt, “The electronic library manager’s guide to virtual reality,” *Electronic library*, vol. 9, no. 1, pp. 3–5, 1991.

[15] L. Rebenitsch, “Managing cybersickness in virtual reality,” *XRDS: Crossroads, The ACM Magazine for Students*, vol. 22, no. 1, pp. 46–51, 2015.

[16] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, “Surround-screen projection-based virtual reality: the design and implementation of the cave,” in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 51–58.

[17] V. Mach, J. Valouch, M. Adámek, and J. Ševčík, “Virtual reality–level of immersion within the crime investigation,” in *MATEC Web of Conferences*, vol. 292. EDP Sciences, 2019, p. 01031.

[18] D. A. Bowman and R. P. McMahan, “Virtual reality: how much immersion is enough?” *Computer*, vol. 40, no. 7, pp. 36–43, 2007.

[19] Fortune Business Insights (2019). Virtual reality market size, share industry analysis. [Online]. Available: https://www.fortunebusinessinsights.com/industry-reports/virtual-reality-market-101378.

[20] L. Freina and M. Ott, “A literature review on immersive virtual reality in education: state of the art and perspectives,” in *The international scientific conference elearning and software for education*, vol. 1, no. 133, 2015, pp. 10–1007.

[21] A. Brown and T. Green, "Virtual reality: Low-cost tools and resources for the classroom," *TechTrends*, vol. 60, pp. 517–519, 2016.

[22] I. Wohlgenannt, A. Simons, and S. Stieglitz, "Virtual reality," *Business & Information Systems Engineering*, vol. 62, pp. 455–461, 2020.

[23] T. Mustafa, R. Matovu, A. Serwadda, and N. Muirhead, "Unsure how to authenticate on your vr headset? come on, use your head!" in *Proceedings of the Fourth ACM International Workshop on Security and Privacy Analytics*, ser. IWSPA '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 23–30. [Online]. Available: https://doi.org/10.1145/3180445.3180450

[24] S. P. L. M. D. B. K. Pfeuffer, M. J. Geiger and F. Alt, "Behavioural biometrics in vr: Identifying people from body motion and relations in virtual reality," p. 12, 2019. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3290605.3300340

[25] A. Kupin, B. Moeller, Y. Jiang, N. K. Banerjee, and S. Banerjee, "Task-driven biometric authentication of users in virtual reality (vr) environments," in *MultiMedia Modeling: 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8–11, 2019, Proceedings, Part I 25*. Springer, 2019, pp. 55–67.

[26] S. Li, A. Ashok, Y. Zhang, C. Xu, J. Lindqvist, and M. Gruteser, "Whose move is it anyway? authenticating smart wearable devices using unique head movement patterns," in *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2016, pp. 1–9.

[27] C. F. Olade, Ilesanmi and H.-N. Liang., "Biomove: Biometric user identification from human kinesiological movements for virtual reality systems." vol. 20, no. 10, p. 2944, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/10/2944

[28] H. J. J. A. L. Mark Roman Miller, Fernanda Herrera and J. N. Bailenson., "Personal identifiability of user tracking data during observation of 360-degree vr video." vol. 10, no. 1, October 2020. [Online]. Available: https://www.nature.com/articles/s41598-020-74486-y

[29] S. Yi, Z. Qin, E. Novak, Y. Yin, and Q. Li, "Glassgesture: Exploring head gesture interface of smart glasses," in *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 2016, pp. 1–9.

[30] P. P. Tricomi, F. Nenna, L. Pajola, M. Conti, and L. Gamberini, "You can't hide behind your headset: User profiling in augmented and virtual reality," *IEEE Access*, vol. 11, pp. 9859–9875, 2023.

[31] S. L. Zhiming Hu, Andreas Bulling and G. Wang, "Ehtask: Recognizing user tasks from eye and head movements in immersive virtual reality." vol. 29, no. 4, pp. 1992–2002, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9664291

[32] Wikipedia contributors. Beat Saber. [Online]. Available: https://en.wikipedia.org/wiki/Beat_Saber

[33] S. Baldoni, F. Battisti, F. Chiariotti, F. Mistrorigo, A. B. Shofi, P. Testolina, A. Traspadini, A. Zanella, and M. Zorzi, "Questset: A vr dataset for network and quality of experience studies," in *Proceedings of the 15th ACM Multimedia Systems Conference*, 2024, pp. 408–414.

[34] Sanlab Learning. Forklift Simulator. [Online]. Available: https://www.sanlablearning.com/forklift-simulator/#:~:text=The%20simulator%20provides%20individuals%20with,work%20areas%20and%20time%20periods.

[35] Wikipedia contributors. Medal of Honor: Above and Beyond. [Online]. Available: https://en.wikipedia.org/wiki/Medal_of_Honor:_Above_and_Beyond

[36] ——. Quest 2. [Online]. Available: https://en.wikipedia.org/wiki/Quest_2

[37] misbahanwar. Support Vector Machines (SVM) -Linear and Non-Linear SVMs. [Online]. Available: https://medium.com/@misbah.anwar/support-vector-machines-svm-linear-and-non-linear-svms-4715398ab66c

[38] sklearn.ensemble. RandomForestClassifier. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[39] ——. AdaBoostClassifier. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html#r33e4ec8c4ad5-1

[40] ——. ExtraTreesClassifier. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html

Acknowledgments

I am deeply grateful to my supervisors, Prof. Federico Chiariotti and Prof. Sara Baldoni, for their contributions, teachings, and personal interest in my thesis writing process. I would also like to extend my sincere thanks to all the professors who generously shared their deep knowledge in the field of Information and Communications Technology. Many thanks to my colleagues, and friends in Padova, Turkiye, and all over the world who always stood by me during this journey.
Finally, I'd like to thank my family. First, to my beloved parents, Beyhan Kaya and Ali Kaya, for their boundless love and support, making me feel truly fortunate to have them in my life. I owe my most sincere gratitude from my heart to my uncles, Cem Güler and Selçuk Güler, who have been role models in intelligence, wisdom, and diligence throughout my educational and professional journey. A special thanks to my uncle Kaya for enriching my vision. My appreciation also extends to my brother, Eren Çağlar Kaya, and his dear wife, Güneş Biçer Kaya, for their unwavering support. Lastly, I cherish the joy that my sweet nephew, Can Kaya, brings into my life with his delightful presence.