This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TG.2022.3178539, IEEE Transactions on Games

Real-time Prediction of Simulator Sickness in Virtual Reality Games

Jialin Wang[1], Hai-Ning Liang[2,*], Diego Monteiro[3], Wenge Xu[4], Jimin Xiao[5]

***Abstract*—Virtual Reality (VR) technology has progressed rapidly and is used in various domains, particularly games. Simulator Sickness (SS) still represents a significant problem for its wider adoption. The most common way to detect SS is using the Simulator Sickness Questionnaire (SSQ). SSQ is a subjective measurement and is inadequate for real-time applications such as VR games. This research aims to develop a model to predict SS in real time using in-game characters' movement and users' eye motion data during gameplay in VR games. To achieve this, we designed an experiment to collect such data with three types of games. We trained a Long Short-Term Memory neural network with the eye-tracking and character movement data to predict SS. Our model can predict SS in real time with an accuracy of 83.4% for players who suffer from severe sensitivity to SS. Our results indicate that, in VR games, our model is an accurate and efficient method to predict SS in real time.**

***Index Terms*—Virtual Reality, Simulator Sickness, Gaming, Real-time Prediction, Machine Learning, In-game Character and Eye Movement Data.**

I. Introduction

Virtual Reality (VR) technology has been growing in the last decade, especially in the last few years, with the proliferation of inexpensive consumer Head-Mounted Displays (HMDs). Despite the advances, Simulator Sickness (SS) remains a constraint and challenge and has a negative effect on the broader adoption of VR [1], [2]. Many people cannot use VR devices for a long time due to SS-related symptoms [3]–[5]. As such, there are significant benefits in finding methods to predict, minimize, and eliminate SS in VR applications, especially in games.

The most common method to assess SS is the Simulator Sickness Questionnaire (SSQ) [6], [7]. It can be used to quantify SS for activities that could lead to SS symptoms. However, it is not possible to quantify real-time SS for VR environments with SSQ. Although new SS assessment methods have been proposed to address this [8], [9], they require special sensors, for example, to capture Electro Dermal Activity (EDA), Heart Rate (HR), and electroencephalogram (EEG) data. These sensors are often not integrated into current consumer-level VR HMDs.

[1]J. Wang is with the Department of Computing, Xi'an Jiaotong-Liverpool University, Suzhou, China 215123, and the Department of Computer Science, University of Liverpool, Liverpool, UK L69 3BX.

[2]H.-N. Liang is with the Department of Computing, Xi'an Jiaotong-Liverpool University, Suzhou, China 215123.

[3]D. Monteiro is with ESIEA, Laval, France 53000.

[4]W. Xu is with the School of Computing and Digital Technology, Birmingham City University, Birmingham, UK B4 7XG.

[5]J. Xiao is with the Department of Intelligent Science, Xi'an Jiaotong-Liverpool University, Suzhou, China 215123.

*Corresponding author (haining.liang@xjtlu.edu.cn).

In this research, we proposed a simple and low-cost method that could achieve real-time SS prediction with current consumer-level VR HMDs that have built-in eye trackers (e.g., FOVE, HTC VIVE Pro Eye, and PICO NEO 2 Eye). Eye-tracking has now been widely used in consumer-level VR HMDs because it provides additional possibilities for VR devices like gaze interaction and facial expression detection [10]–[12].

SS is a type of motion sickness caused by movement in the environment perceived by the visual system [13]–[15]. The etiology, or cause of motion sickness, involves three possible factors: reflexive eye movements (EM), sensory conflict (SC), and postural instability (PS) [13]. The eyes receive most of the stimulation when users interact with VR applications like games. Character movement data (e.g., position, velocity, Euler angle, and angular velocity data from character and HMD) also relates to SC since the motion in the perceived environment is apparently one of its causes [13]. In our approach, we record character movement and eye-tracking data from users playing three different VR games. The data is then used to train and develop models to predict SS during gameplay in real time.

One of the most significant challenges in SS prediction is to quantify it objectively and extract its features. In this research, we propose using new features to identify SS in VR environments. Based on our review of the literature, we hypothesize that two in-game features (intense character movement and negative eye movement (i.e., longer average blink and fixation durations [16])) are highly linked to SS: (1) $\boldsymbol{H_1}$: Eye tracking data can be used to predict SS since SS may cause negative eye movement; and (2) $\boldsymbol{H_2}$: Character movement data can be used to predict SS since intense character movement may cause SS [13], [17]. We used a novel labeling method to break real-time gaming events into classifiable events, which in turn can be used to train a model and subsequently predict SS in VR games. Analysis of time series data often requires high throughput machine learning (ML) [18]. Real-time SS prediction based on sequential data also requires sequential modeling, which is also a problem of multi-step forecasting for multivariate time series for the eye and character movement data in our case. Therefore, we used a Long Short-Term Memory (LSTM) neural network [19] for training our dataset because it is an efficient method for sequential modeling.

Games are important applications in VR. However, unlike in typical 2D displays, VR games can commonly bring symptoms of SS in many users [20], [21]. For this study, we developed three different VR games that can stimulate SS during gameplay and used them to collect the data used to train our model. We chose these VR games because they can produce different SS levels depending on changes in viewing perspective, movement trajectory, and speed during gameplay [22].

Our game recordings (dataset) of participants' gameplay sessions have been used to link character movement and eye-tracking data and place them along the same timeline. This is helpful for further analysis to detect salient events during gameplay across sessions and for different participants. Part of the game data recordings was used to train our LSTM neural network.

The main contribution of this paper is that we proposed a novel method for real-time SS prediction in VR games based on players' eye motion and in-game character movement data during gameplay. We used an LSTM neural network to train our model using eye and character movement data. To the best of our knowledge, we are the first to use both eye movement and character movement data together. This model can be used to improve the gaming experience in real time whenever SS is predicted during gameplay.

II. Related Work

In this section, we review previous research on Simulator Sickness (SS) and approaches to detect SS.

A. Simulator Sickness

SS affects a significant number of VR users at different levels. Its causes are still being investigated. Theories suggest that it is caused by eye movements and sensory conflicts [13]. However, it is difficult to pinpoint when exactly people start feeling sick in VR, especially in games. Works that try to identify the cause of sickness in VR rely on correlating results from after playing the game with activities performed during gameplay [23], [24].

The problem of the most widely used method to assess SS (i.e., SSQ) is that it cannot be applied to detect SS in real time. Hence, studies have been conducted to allow real-time detection of SS using various psychophysiological data. Most approaches use machine learning techniques to analyze such data recorded from VR environments [8], [9]. However, they can be cumbersome and expensive because they rely on equipment that is not naturally integrated into VR (e.g., EEG, EDA, and HR sensors).

Even though many theories suggest that visual input is closely related to SS [13], [16], [25], to the best of our knowledge, no study has looked into using eye movement and character movement to infer SS in VR HMDs. The closest is [26], in which the authors investigated changes in people's pupils to detect SS and emotional changes.

Eye tracking has become popular in VR due to the possibility of using the data in applications [10]–[12]. There are now many consumer-level VR HMDs that come integrated with eye trackers (e.g., FOVE, HTC VIVE Pro Eye, and PICO NEO 2 Eye). So, using eye-tracking to infer SS requires no additional hardware and may yield positive results, looking directly into one of the sources of SS.

B. Simulator Sickness Detection

1) Internal Psychophysiological Data: Sensory conflict is one of the possible causes of SS and can affect the state of the brain [13]. Research has shown that EEG can be used to assess SS levels, similar to using SSQ data [27]. EEG data can show the different states of the brain [28]. Jeong et al. reported an approach to detect SS with EEG data using Deep Learning algorithms, which could achieve accuracies of above 90% [8]. They found and detected SS patterns from the image of raw EEG data and calculated band powers of EEG signals. Other techniques to detect SS involve the use of several psychophysiological features. For example, Nam et al. used EEG, Electrooculogram (EOG), Electrocardiogram (ECG), Fingertip skin temperature (SKT), Photoplethymogram (PPG) and Skin conductance (SCL) as input in their artificial neural network, which can achieve a minimum mean square error of 0.09) [29]. Gardé et al. used EDA and ECG to demonstrate the vibro-kinetic seat can cause lower SS [30]. Garcia-Agundez et al. used EOG, ECG, respiratory effort, galvanic skin response (GSR) to detect SS with support vector machines (SVM), which can achieve a SS detection accuracy of 81.8% [31]. Even though their results are positive, their data collection requires individual sensors, which are difficult and expensive to be included in today's consumer-level VR HMDs.

Other research that used composite psychophysiological data (e.g., [9], [32]) had low success rates when using fewer features (i.e., only EDA and Blood Pulse Volume). Their model can explain only 48% of the SS in VR, which has a much lower accuracy than other models. This means that using such methods is currently inefficacious, inefficient, or expensive. Thus, we want to investigate alternatives that require fewer devices and sensors (i.e., low-cost) and that are closer to the problem, so that results are more straightforward and inexpensive to achieve. Our method can achive an accuracy of 83.4% without the use of extra sensors that can be expensive and difficult to include in today's HMDs and can bring inconveniences to users.

2) Eye-tracking Data: Nowadays, many VR HMDs have eye trackers integrated into them to capture and record users' eye movements. SS can produce changes in users' eye movements as a natural response [13]. Research has shown that there is a correlation between SS and specific eye movements (e.g., number of blinks) [33]. More prolonged VR HMD exposure can cause a significantly greater number of blinks (in contrast to using a desktop monitor, for example). There are significant correlations between the average number of blinks and SS [33]. This means eye movement data may contain some features of SS.

The correlation between eye behavior and SS further supports the use of eye-tracking devices to detect early onset SS. If these user behaviors can be extracted as features to be fed to a Machine Learning model, it can potentially identify if someone is getting sick in real time. However, VR games are complex and involve a series of other behaviors that might interfere with the results. As such, their result cannot be immediately applied [16]. We propose to investigate associations of eye movements and other in-game behaviors for results with higher

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TG.2022.3178539, IEEE Transactions on Games

Fig. 1. Screenshots of the three VR games used in our experiment for data collection. (A) Racing Car: driving around along the track. (B) Parkour: collecting all gold coins by jumping from building to building. (C) Space Miner: mining gold asteroids in space.

prediction accuracy.

Our method can achieve SS prediction in real time using a commercial VR HMD with eye-tracking capabilities. To our knowledge, our dataset is the first one to analyze both eye movement data and character movement data. Our results show that this combination can help the analysis and prediction of SS for at least one kind of application, VR games.

III. Data Collection

A. The Three VR Games

We developed three VR games using Unity3D to collect eye motion and in-game character movement data during gameplay (see Figure 1). We created a series of games and ran some pilot studies to arrive at the three games. The results of the pilot studies showed that they could produce character movement that is intense and at a level that can stimulate SS in users. Our pilot run showed that these three VR games could produce enough SS stimulation with different levels during gameplay. To record possible SS symptoms, no SS reduction techniques have been applied to these VR games. Racing Car (see Figure 1 a) is a first-person perspective (1PP) VR game in which a player needs to drive around a track. One round takes about 4 minutes to complete. The primary SS stimulation was the 2D (x and y) motion in the environment. Parkour game (see Figure 1 b) is a 1PP VR game, where a player needs to jump from roof to roof to collect gold coins. One round takes about 4.5 minutes to finish. Space Miner (see Figure 1 c) is a third-person perspective (3PP) VR game, and a player needs to pilot a starship to mine gold asteroids. One round takes about 17 minutes to finish. The primary SS stimulation of Parkour and Space Miner is 3D (x, y, and z) motions when moving in the two environments (e.g., Parkour requires the player to jump from roof to roof while Space Miner requires movement with three dimensions: roll, pitch, and yaw; see player paths in Figure 2). Racing Car and Parkour both use a typical humanoid avatar (in 1PP), while Space Miner is based on a 3PP of a spaceship. Although 3PP VR games (e.g., *Lucky's Tale* and *Chronos*) are not as popular as 1PP VR games (e.g., *Beat Saber*), the former types of games are still common in VR and represent an important category of games. As such, we decided to include at least one such game so that both 3PP and 1PP VR games are explored. Table I summarizes the features of the three games for comparative purposes. In our experiment, the participants were told to play a specific VR game for 7 minutes, excluding pause time. Based on our pilot runs, 7 minutes of gameplay can produce enough SS stimulation because the exposure time of similar research ranges from 1 to 5 minutes [8].

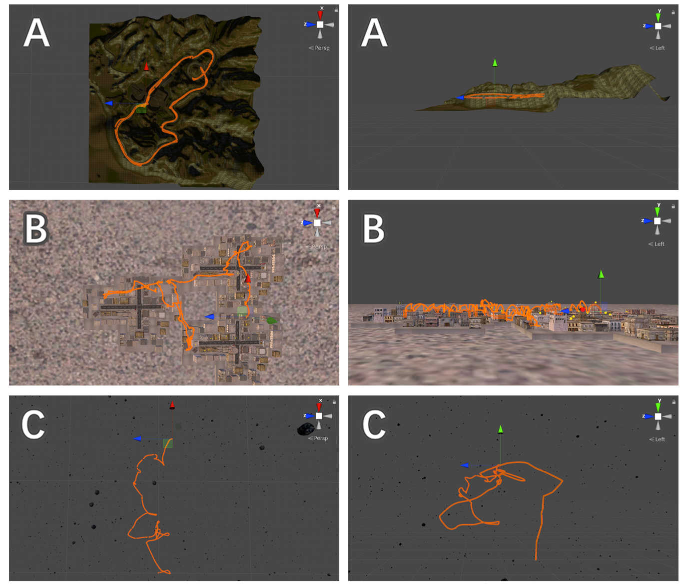


Fig. 2. Player paths (orange lines) in z-x (left figures) and z-y (right figures) plane recorded from one participant. (A) Racing Car. (B) Parkour. (C) Space Miner.

B. Experiment

1) Participants: We recruited 20 volunteers from a local university campus. There were 14 males and 6 females whose ages ranged between 18 and 27, with an average age of 20.4±1.93. They all had normal or corrected-to-normal eyesight and had no history of color blindness or mental/physical issues. They all consented to participate in the experiment voluntarily. The experiment was conducted in accordance with the guidelines and regulations of the University Ethics Committee at Xi'an Jiaotong-Liverpool University.

2) Apparatus: The HTC VIVE Pro Eye was used as the VR HMD for the experiment. It supports eye-tracking and full VR functions. The eye trackers have a spatial accuracy of 0.5° to 1.1° and support capturing gaze data with a frequency of 120 Hz, a trackable field of view of 110°, and 5-point calibration [1]. It was connected to a computer with 16 GB RAM, a GeForce GTX 1080 Ti GPU, and an Intel Core i7-7700k CPU. The input device used in the experiment was two HTC VIVE controllers. We collected all movement data of the VR HMD and character, such as position, velocity, Euler angles, and angular velocity. The character's movement data can be used in the playback of the gameplay sessions.

3) Procedure: Each participant was assigned a specific order of VR games in which he or she would play. The orders had been formed through a 3×3 Latin Square to mitigate carry-over effects. Participants needed to calibrate the eye tracking device and fill out a questionnaire to collect demographic and past VR and gaming experience information before the first VR game. A dataset that contains eye-tracking and character movement data would be recorded during gameplay (see the gameplay time in subsection III-A).

Previous research about SS has shown that closed eyes can subside sickness [34]–[36]. Therefore, the participants were

[1]https://www.vive.com/uk/product/vive-pro-eye/specs/

TABLE I
FEATURES OF THE THREE VR GAMES USED IN THE DATA COLLECTION. 1PP: FIRST-PERSON PERSPECTIVE; 3PP: THIRD-PERSON PERSPECTIVE.

Game	Character	Player Task	Control Method	SS Stimulation
Racing Car	Humanoid avatar (in 1PP)	Driving around along a track	Left joystick on the y axis: acceleration Right joystick on the x axis: steering	2D motion
Parkour	Humanoid avatar (in 1PP)	Jumping from roof to roof to collect gold coins	Left joystick on both x and y axis: walking left trigger: jumping right joystick on the x axis: rotation	3D motion
Space Miner	Spaceship (in 3PP)	Piloting a starship to mine gold asteroids	Left joystick on both x and y axis: flight attitude Left trigger: acceleration Right trigger: laser mining	3D motion

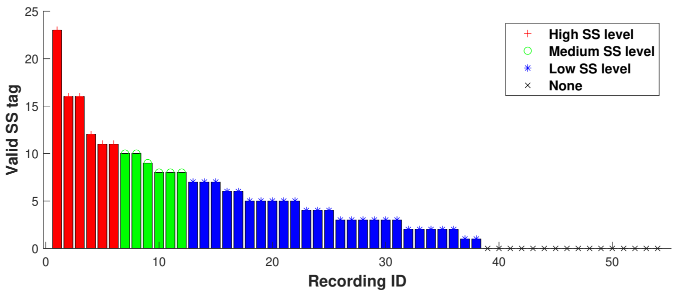


Fig. 3. Game recordings sorted by SS level (number of valid SS tags). Each bar represents how many valid SS tags that session had.

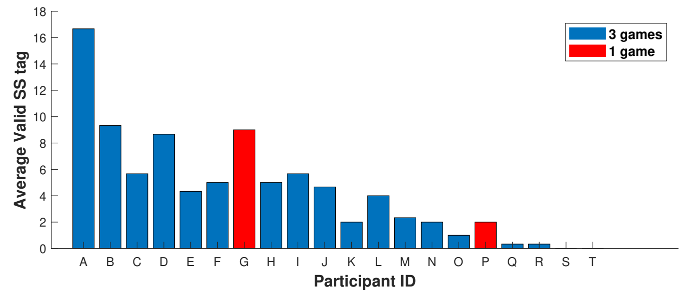


Fig. 4. Participants sorted by the maximum number of valid SS tags. The letter represents the SS sensitivity level of all participants. For example, Participant A has the highest SS sensitivity level (the highest number of valid SS tags).

told to close their eyes immediately when they felt any SS symptoms; we explained SS symptoms to our participants before the experiment. These symptoms include nausea, sweating, vertigo, and dizziness. The game was then paused if the participants closed their eyes for at least 0.5 seconds. The game would continue by clicking a button on the controller after the participants opened their eyes for 3 seconds. The participants were told not to continue the game unless they felt better. After each game, the participants were required to rest for at least one night to prevent the accumulation of SS. Each participant needed at least 3 days to complete the experiment.

C. Results of Data Collection

54 game recordings (dataset) were collected from all 20 participants. Three male participants failed to complete the whole experiment and only played one VR game. They could not finish all VR games due to their strong reaction to SS stimulation. As mentioned in subsubsection III-B3, the trigger for pausing was 0.5 seconds of closed eyes, and the game could continue after the participants opened their eyes for 3 seconds in each pause. Thus, the theoretical minimum pause was 3.5 seconds. However, if the participants paused the game due to a strong SS stimulation, they would need longer to recover from the symptoms of SS. Therefore, we use 5 seconds as the threshold for a valid pause caused by a strong SS stimulation. To improve the quality of SS tags, each pause longer than 5 seconds was considered a valid SS tag.

Figure 3 shows the results of valid SS tags from the 54 game recordings. 15 (27.8%) game recordings had no valid SS tags. As expected, participants with higher SS sensitivity would produce more valid SS tags (that is, game pauses due to a high level of SS). In our case, SS sensitivity was from the SS symptoms of participants displayed during the experiment and their interview after the experiment. Therefore, the SS levels of game recordings were classified by the number of valid SS tags. High, medium, and low SS levels means more than 10 tags, between 8 to 10 tags, and less than 8 tags, respectively. SS level is used as a metric of dataset quality to choose a better dataset for model training. Only high and medium SS level recordings (top 12 game recordings) would be used as the dataset for neural network training because of the higher ratio of SS tags and more evenly distributed SS events. Pause tags are also important to maintain the integrity of the time series data used in model training and input data feeding. The other reason is that we recommended participants to pause and get some rest after predicting SS events.

All 20 participants were sorted by the maximum number of valid SS tags and assigned a letter of the alphabet. Participant A had both the maximum number of valid SS tags and the highest average number of valid SS tags. Participants with a high maximum number of valid SS tags would usually have a high average number of valid SS tags also. The trend observed in Figure 4 is similar to the pattern in Figure 3 (i.e., participants with higher SS sensitivity produced more valid SS tags).

For the 17 participants who completed all 3 games, the average SS tag numbers of Parkour, Space Miner, and Racing Car were 6.64, 3.71, and 3.24, respectively. Parkour produced more SS stimulated symptoms than the other two games. For the 12 high and medium SS level recordings, the ratio of Parkour, Space Miner, and Racing Car was 6:3:3 and was related to the average SS tag numbers. We collected demographics and past VR and gaming experiences from the pre-questionnaire. Most participants (40%) had some experience with VR (that is, had seen it or interacted with it before). Most participants (50%) played video games twice or more a week. Most participants (40%) preferred to pause the game and take a break when SS symptoms happened during the VR experience.

TABLE II
SUMMARY OF THE EYE-TRACKING AND CHARACTER MOVEMENT DATA.

Data	Number	Unit
Euler angle of HMD (x y z)	3	$^\circ$
Euler angle of character (x y z)	3	$^\circ$
Angular velocity of HMD (x y z)	3	$^\circ/s$
Angular velocity of character (x y z)	3	$^\circ/s$
Position of HMD (x y z)	3	m
Position of character (x y z)	3	m
Velocity of HMD (x y z)	3	m/s
Velocity of character (x y z)	3	m/s
Raw eye movement data	13	unitless(0 to 1)
Magnitudes of eye acceleration speed	2	unitless
Magnitude of character acceleration speed	1	m/s^2
Magnitude of character angular acceleration	1	$^\circ/s^2$
sum	**41**	

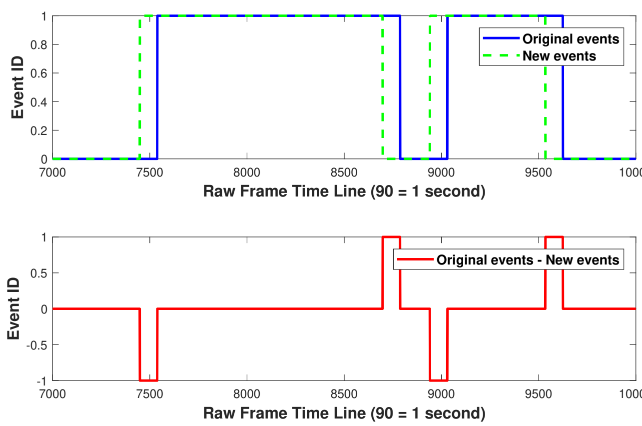


Fig. 5. Labelling method for model training. The original events are real-time events. The new events are left shift (future) original events which are the final input events for model training.

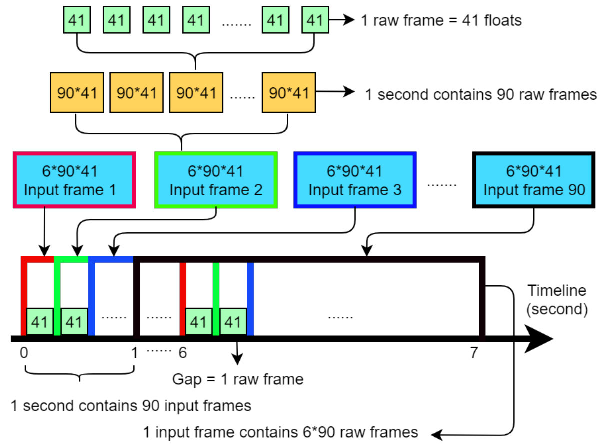


Fig. 6. Example of an input frame for LSTM model training. The size of sliding window is 6 seconds. The final input frame consists of 6 * 90 stacked raw frames.

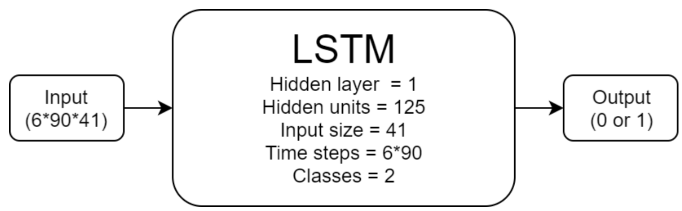


Fig. 7. Model architecture of our model for SS prediction

IV. Modelling Using Eye and Character Movement Data

A. Dataset and Model

The dataset contains 41 types of data from eye tracker, HMD (the motion data of the HMD was captured from the character camera), and character: Euler angles (6), angular velocities (6), positions (6), velocities of HMD and character (6); raw eye movement data (13), magnitudes of eye acceleration speed (2), magnitudes of character acceleration speed and character angular acceleration (2) (see Table II). The magnitude of eye acceleration speed was calculated from 13 types of raw eye movement data recorded from the HTC VIVE Pro Eye SDK [2]. Magnitudes of character acceleration speed and character angular acceleration were calculated using the virtual sensor script in the VR games.

The range of values of different raw data types varies widely. Therefore, min-max normalization was applied to the dataset to rescale the data within the range of 0 to 1 before inputting the model [37].

1) Labelling Method: The aim of this research is to predict SS in real time. However, there is not an accurate method to label SS tags since SS is not a discrete state. Therefore, we turn it into a problem of multi-step forecasting for multivariate time series for the eye and character movement data [38]. There are two event IDs in the raw dataset: 0 denotes normal gameplay, and 1 represents the pause state. Long blinks (which are the trigger of pauses) are classified into event 0 to prevent mixed patterns between events 0 and 1. For instance, the original events in Figure 5 mean two pause state periods among three normal gameplay periods. The predicted events are left shift (future) original events shown in Figure 5. Therefore, the model tries to predict future pause states, which represent the consequence of SS and game states. The red line in Figure 5 represents the difference between the original events and the predicted events.

2) Model Architecture and Configuration: Figure 6 shows the structure of input frames for training the model. One second contains 90 raw frames of raw data. Each input frame contains 6 seconds of raw data with an interval of 1 raw frame. In other words, the throughput is 90 input frames per second. A 6 seconds input frame (6 * 90 raw frames) also means the input buffer cannot get enough raw frames to fill one input frame in the first 6 seconds in real-time applications. However, the length of the input frame is also an adjustable parameter.

Figure 7 shows the whole architecture of our LSTM model, which is a special kind of recurrent neural network that can learn long-term patterns from continuous data [19]. Our model has 1 hidden layer and 125 hidden units (decided by parameter optimization); its input is shown in Figure 7. Its output is one of the two event IDs cited in subsubsection IV-A1. Given that different players and games would have different SS patterns on the eye and character movement dataset, each game recording

[2]https://vr.tobii.com/sdk/develop/unity/getting-started/vive-pro-eye/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TG.2022.3178539, IEEE Transactions on Games

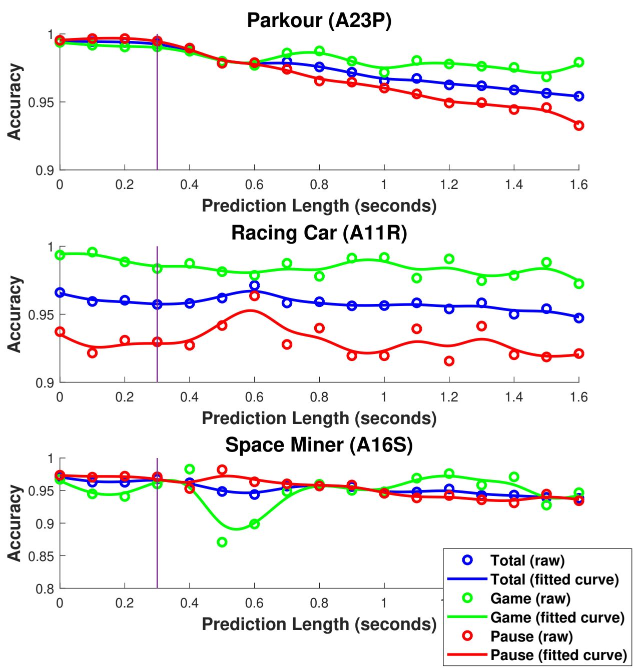


Fig. 8. Results of prediction length tuning on game recordings from Participant A who has the highest number of valid SS tags in all the 3 different VR games. They show that the accuracy did not drop in the first 0.3 seconds (indicated by the vertical line).

was trained separately.

The dataset would not be split randomly to prevent non-existent new time series patterns which did not exist in the actual time series data since we are using LSTM to forecast time series data. Training and testing datasets were the first 70% and the last 30% of data from the recordings of the same game, respectively.

B. Evaluation of the Model for SS Prediction

1) Results of Prediction Length Tuning: Figure 8 shows the results of prediction length tuning on A23P, A11R, and A16S—letter A means Participant A, the number next represents the number of valid tags, and the last letter means the game (P for Parkour, R for Racing Car, and S for Space Miner). The fitted curves are used to clarify the scatter points and show the trends. As can be observed in Figure 4, Participant A contributed the best (most valid SS tags) game recording in all 3 VR games. Accuracy is defined as the percentage of correct predictions in all predictions for the corresponding events. The results show that the accuracy did not drop sharply in the first 0.3 seconds. Therefore, we try to use 0.3 seconds as the prediction length for all other game recordings. Figure 8 also shows that the model can get higher and tighter accuracy lines on game recordings with a high SS level (A23P and A16S). Although the accuracy lines of A11R are not as tight as the lines of A23P and A16S, their 3 accuracy lines are still above 0.9.

There are two event IDs (normal gameplay and pause state), as mentioned in subsubsection IV-A1. The aim of a prediction

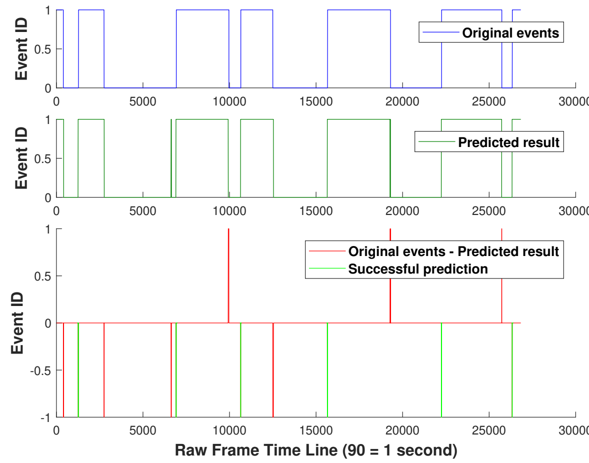


Fig. 9. Testing results of the model trained on Participant A, with 23 valid tags, using the Parkour game and a prediction length of 0.3 seconds.

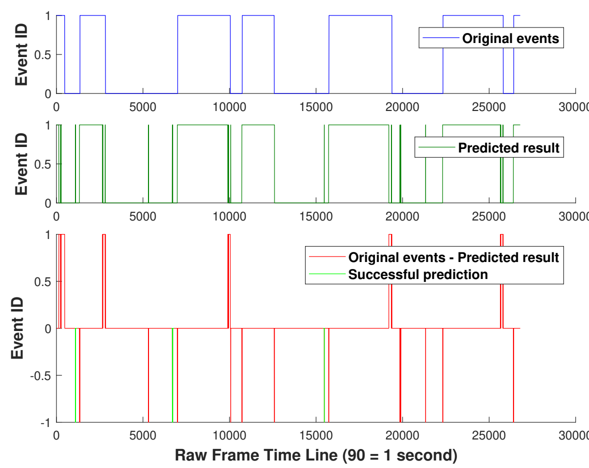


Fig. 10. Testing results of the model trained on Participant A, with 23 valid tags, using the Parkour game and a prediction length of 1.6 seconds.

length of x seconds is to predict future states in the next x seconds. For both Figure 9 and Figure 10, the middle green line denotes the predicted result. The red line is the visualization of the difference between original events and predicted results. The green line segments below the red line represent a successful prediction of future pause states in the next x seconds, which also shows how to use the model to predict and respond to pause states in games. For example, the developers can design predefined actions in response to the prediction result when the green line segments come up.

Figure 9 shows the testing results of the model trained on A23P with a prediction length of 0.3 seconds. It shows that the model can predict future pause states, which are consequences of SS, in 0.3 seconds. Although there are a few false positives predicted results, it still can predict all 6 pauses before they happen. Figure 10 shows the testing results of the model trained on A23P with a prediction length of 1.6 seconds. It reveals the distribution of false positives and the reason for the drop

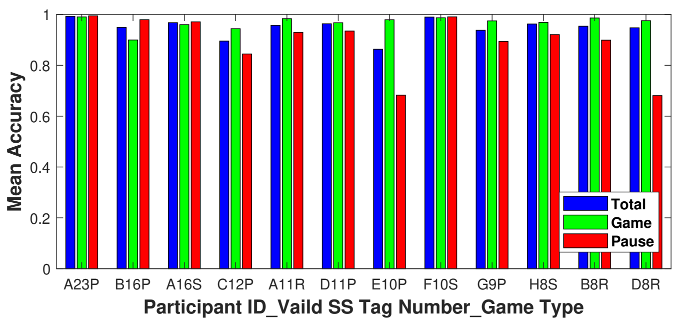
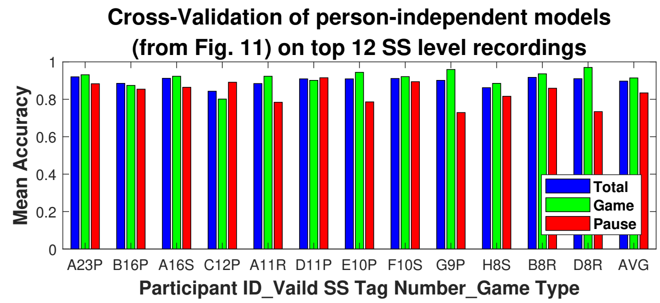


Fig. 11. Testing dataset results of recording-independent models trained on the top 12 game recordings from Participants A to H with a prediction length of 0.3 seconds.

in accuracy in longer prediction lengths. There are more false positives in-game states. In addition, only half of 6 pauses can be predicted in 1.6 seconds. Although a longer prediction length is possible, the accuracy drops with prediction length. This is the secondary reason for using 0.3 seconds as the prediction length for all other game recordings.

2) Results on All Datasets: The model trained on the whole dataset has worse performance than the model trained on a single game recording (recording-independent model) due to the non-existent new patterns among different game recordings in the whole dataset. Therefore, we will focus on evaluating the recording-independent model in the following section. Figure 11 (A23P to D8R are the names of the top 12 game recordings in Figure 3) shows the testing dataset results of the model trained with a prediction length of 0.3 seconds. The model has a good performance on all events from all three VR games. The model can be used to predict SS for different games and players. It has a better performance on game recordings with a high SS level (A23P to D11P). Because A23P has the most valid SS tags, it gives us also best testing results.

3) Generalization of Recording-independent Models: Figure 12 shows the overall result of k-fold Cross-Validation (CV) on the model using the same model configuration from Figure 11 [39]. Due to space constraints, it is difficult to put all 38 columns of the CV results in one figure. For readability, only the results on the datasets with high and medium SS levels (top 12 recordings) are shown in Figure 12. Figure 12 reveals a poor performance of the model on datasets with low SS levels (the rest 26 recordings). Therefore, recording-independent models trained on datasets with high and medium SS levels are better. Datasets with high and medium SS levels are sufficient for model training. In other words, recordings with low SS levels can be excluded from the dataset. In general, k-fold cross-validation means that the training is done on k-1 sub samples; the testing is done on the k-th sub sample (the larger fold was used for training and the smaller for validation). However, we decided to use a recording-independent model rather a model trained on datasets from different recording sessions (see also subsubsection IV-B2). Therefore, we take each game recording as one fold of the dataset, and the larger fold was used for validation and the smaller for training (the training is done on k-th sub sample; the testing is done on the rest k-1 sub samples).



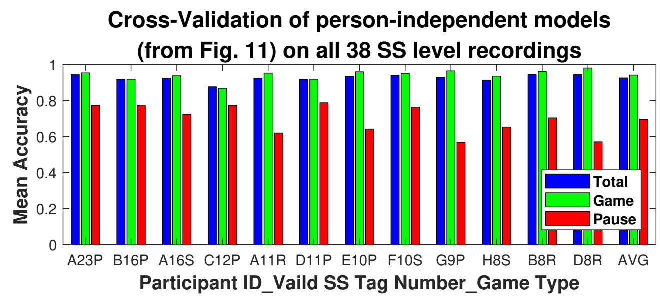


Fig. 12. K-fold Cross-Validation of the model (using the same model configuration from Figure 11) on different datasets. The X axis denotes the average accuracies from recording independent models (AVG means the k-fold result of the model).

The average accuracy from recording independent models of 12 and 38 fold datasets indicates that the recording-independent model has better generalization results on worse cases from participants with higher SS sensitivity. The model has a SS prediction accuracy of 83.4% on the 12 fold dataset and 69.6% on the 38 fold dataset. This difference was caused by the different number of SS tags between the 12 and 38 fold datasets. Our model training method tends to have higher accuracy on datasets with more SS tags and more evenly distributed SS events (see subsection III-C). Moreover, the k-fold CV results show that the model has worse generalization results if the game recordings with lower SS levels are included in the testing dataset. This means the model has difficulty predicting SS for players with lower SS sensitivity. In other words, it is also good news for players with high SS sensitivity since they can contribute better datasets and get higher accuracies of SS prediction. Previous research on SS detection based on internal psychophysiological data shows weak generalization results from person-independent models (models trained on independent datasets consisting of recorded data from each user rather than the whole dataset) [32]. Our results show that participants with higher SS sensitivity can contribute data with higher quality in their worst cases to improve the model's generalizability. Moreover, the model works better on players with higher SS sensitivity. However, it was difficult to collect such data since high SS level recordings were relatively rare (6 of the total 54 recordings in this experiment).

Figure 13 indicates that models trained on one participant's dataset can also be used to predict SS for other participants in different games. Parkour produced twice as many valid

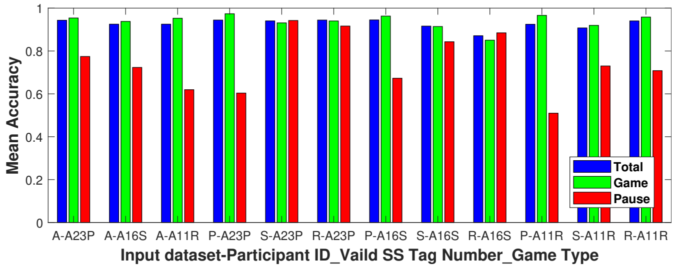


Fig. 13. Cross-Validation of 3 recording independent models (trained on one participant's dataset (A23P, A16S, and A11R) using the same model configuration from Figure 11) on all 38 recordings grouped by input dataset type: All dataset (A), Parkour (P), Space Miner (S), and Racing Car (R).

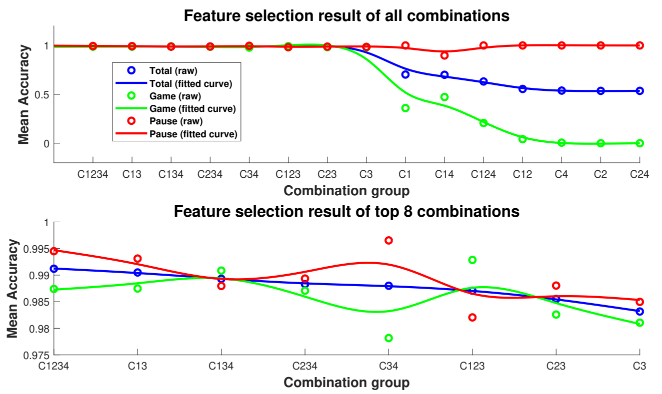


Fig. 14. Feature selection result (sorted by average (total) accuracy) among all 15 possible combinations of 4 groups of raw data: Group 1: position and angle group; Group 2: velocity and angular velocity group; Group 3: eye-tracking group; Group 4: character acceleration group.

SS tags as the other two VR games. All 3 models have a low pause accuracy for the Parkour game (P-A23P, P-A16S, P-A11R—letter P means the Parkour dataset, the part after - represents the independent model trained on the corresponding dataset using the same model configuration from Figure 11) compared with the other two games. The pause accuracy of all 3 models for Space Miner and Racing Car drops as the number of SS tags (S-A23P, R-A23P, S-A16S, R-A16S, S-A11R, R-A11R). A23P shows the best performance, even higher than models trained on A16S and A11R recorded in the Space Miner and Racing Car games. This also proves that the model trained on participants with higher SS sensitivity in VR games that can cause more SS could also achieve high performance on participants with lower SS sensitivity in VR games that can cause lower SS.

4) Feature Selection: For now, LSTM does not support feature importance calculation. Instead, we used the dataset combination instead of feature importance. The current dataset contains 41 types of data in Table I. The training and running of the model on such a dataset can result in high GPU usage which affects the performance of VR games. However, Figure 14 shows that it is possible to reduce the number of data types in the dataset. The initial 41 data types can be split into 4 different groups: Group 1: position and angle group (HMD Euler angle, character Euler angle, HMD position, and character position); Group 2: velocity and angular velocity group (HMD angular velocity, character angular velocity, HMD velocity, and character velocity); Group 3: eye-tracking group (13 raw eye-tracking data, left eye acceleration, and right eye acceleration); and, Group 4: character acceleration group (character acceleration speed and character angular acceleration). The 4 groups have 15 possible combinations as shown in Figure 14. In general, C1234 $>$ C3 $>$ C1 $>$ C4 $>$ C2. C3 is the most important feature group since all top 8 combinations contain C3, and the performance of the rest of the other combinations is much lower than the top 8 combinations. C3 also contributes most to the accuracy in all single group combinations. However, C1234 (combination of all 4 groups) still has the best performance, which supports our hypotheses mentioned in the introduction. Although C3 is much better than all other single group combinations, different eye-tracking devices may provide different eye-tracking data types, which means the performance of C3 may vary on different eye-tracking devices. In contrast, all VR games can use the same data types in C1, C2, and C4.

V. Discussion and Future Work

The above results provide further support for our hypotheses mentioned in the introduction: $\boldsymbol{H_1}$: Eye tracking data can be used to predict SS since SS may cause negative eye movement; and $\boldsymbol{H_2}$: Character movement data can be used to predict SS since intense character movement may cause SS [13], [17]. These two in-game features (negative eye movement and intense character movement) exist in our dataset of eye movement data and character movement data. In addition, the results show that the model trained on such a dataset can be used to predict SS. To further explain the above results, we can also assume that there are continuous SS patterns that exist in eye and character movement data since subsubsection IV-B1 shows that the prediction length can be adjustable.

Our results described above lead to a promising conclusion: the accuracies are related to the number of valid SS tags. In other words, the model can achieve higher accuracy for players with higher SS sensitivity (that is, those whose data have more valid SS tags). This observation is especially interesting and valuable since those players are the ones who are more likely to quit or feel discouraged from using VR because of their SS symptoms. It also means that a better dataset for this model requires longer game recordings from players with higher SS sensitivity. However, Figure 3 shows that game recordings with high SS levels are rare in 7 minutes of gameplay. Although a longer gameplay time may produce more valid SS tags, it is not suitable since it is not ideal for players with high SS sensitivity to have prolonged exposure to VR games. In general, a better dataset needs more game recording sessions and participants than a longer gameplay time. This is one aspect that can be explored in the future.

Figure 15 shows one of the main applications of the real-time SS detection model during gameplay. It can be used as the basis of an AI-based negative feedback system in VR gaming environments. Our negative feedback system of a real-time SS detection model is similar to the Enhanced Affective Game Loop [40], which also shows the relationship

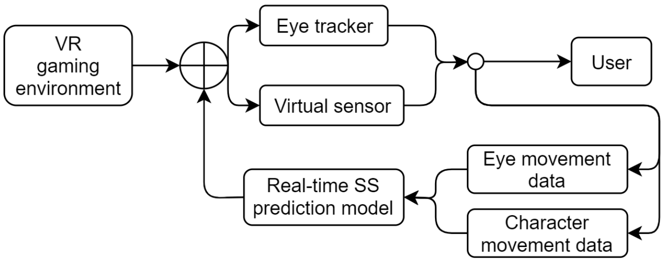


Fig. 15. Negative feedback system of real-time SS detection to be used to keep balance and prolong the flow experience in the game.

between the player, physiological input, and game. The real-time SS detection model can detect SS from a user's real-time eye movement plus in-game character movement data during gameplay. Then, the VR environment can respond to the SS feedback, possibly by reducing SS stimulation in the game. The reduction can be made by adding SS mitigating techniques like Rotation Blurring [41], Depth-of-Filed Blur Effects [42], [43], 2D/3D views [44], and Gazing Cues [45] among others [46]–[48]. Alternatively, if the situation is severe and recurrent during gameplay, it is possible to automatically suggest users to pause and rest using a predefined transition mechanism.

The model has an excellent performance in the prediction of game and pause states. It can be used to reduce SS stimulation during the development phase of a VR game. For example, when compared to existing real-time SS detection methods in the literature [8], [9], the model provides developers with a low-cost, simple, and efficient solution to locate and remove possible SS stimulation aspects during game development or adding ways to mitigate them in the game. It can predict real-time SS events for players who are highly sensitive to SS in some VR games. Using the model, developers can design predefined actions or features in response to SS feedback to improve user experience and gameplay. For example, the game difficulty can be automatically adjusted to a lower level when SS is predicted.

Prior research has suggested that 1PP VR games can produce more stimulation than 3PP VR games [22], [24]. Therefore, it is helpful to explore 1PP games in more depth and collect data from other types of 1PP VR games to see if the same findings can be replicated and further insights discovered. Further, to improve the efficiency and performance of SS prediction, we need to compress current SS features to reduce hardware requirements for real-time applications and attempt to test different neural networks. Different length of input time series is also worth further exploration due to the difficulty of predicting SS in users with milder symptoms. One aspect that may be useful is to consider using a simplified in-game SSQ to get feedback from participants instead of using long blink since it cannot reflect the actual level of SS, like the dynamic, in-game approach used in [32], [49]. For example, in [49], the researchers used an on-screen sliding scale (from 0 to 10) to collect participants' current level of sickness dynamically while the game would still be running in the background. Their results show that this collection approach is accurate enough to reflect players' SS to a large extent.

Furthermore, as the number of male and female participants are not equal, we cannot draw any conclusion about the influence of gender in SS and VR games. In the future, we plan to extend our work and involve more female participants and participants from other groups, like the elderly, to develop models that are more gender-neutral and can be applied to various population groups [50]. Moreover, the official SDK of HTC VIVE Pro Eye does not mention or provide details about the quality of the data capture. We plan to use more professional eye tracking devices in future experiments. For the model training aspect, we fed the data into the model at the frame level to avoid missing parts of SS events for continuous prediction. However, the predicted errors at the frame level can mislead the users. In the future, we will apply a filter to the results from the frame-level prediction or use a longer interval for the input data. For the model evaluation aspect, feature selection was done by combining different data types due in part to the difficulty of determining the features' importance for LSTM (see subsubsection IV-B4). In the future, a more standard feature selection strategy will be helpful in a new model architecture that can support feature importance calculation. Furthermore, subjective experiments will be conducted to test the effect of the model in actual applications.

VI. Conclusion

In this paper, we presented an experiment and model performance for real-time Simulation Sickness (SS) prediction based on players' eye movement plus in-game character movement data collected during gameplay in VR games. We posed two hypotheses from our literature review: (1) $\boldsymbol{H_1}$: Eye tracking data can be used to predict SS since SS may cause negative eye movement; and (2) $\boldsymbol{H_2}$: Character movement data can be used to predict SS since intense character movement may cause SS. We used three different VR games to produce enough SS stimulation during gameplay in the collected data. The evaluation of our model supports our hypotheses and shows that it can achieve high levels of performance and accuracy in SS prediction for players who are highly sensitive to SS. The model can predict SS in real time with an accuracy of 83.4% for players who suffer from severe sensitivity to SS. Our results suggest that our prediction method can be used to predict SS for VR games in real time. The approach and model are simple yet effective in predicting SS during gameplay for VR games.

Acknowledgments

The authors thank the participants who joined the study and the reviewers for their insightful comments and useful suggestions that helped improve our paper. This work was supported in part by Xi'an Jiaotong-Liverpool University Key Special Fund (#KSF-A-03), National Natural Science Foundation of China (#61972323), and the Future Network Scientific Research Fund (#FNSRFP-2021-YB-41).

REFERENCES

[1] M. McGill, A. Ng, and S. Brewster, "I am the passenger: How visual motion cues can influence sickness for in-car VR," in *Conference on Human Factors in Computing Systems - Proceedings*, vol. 2017-May, no. May, 2017, pp. 5655–5668.

[2] S. Weech, J. Moon, and N. F. Troje, "Influence of bone-conducted vibration on simulator sickness in virtual reality," *PLoS ONE*, vol. 13, no. 3, pp. 1–21, 2018.

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

[4] S. Von Mammen, A. Knote, and S. Edenhofer, "Cyber sick but still having fun," in *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*, 2016, pp. 325–326.

[5] C. Yildirim, "Cybersickness during VR gaming undermines game enjoyment: A mediation model," *Displays*, vol. 59, no. February, pp. 35–43, 2019. [Online]. Available: https://doi.org/10.1016/j.displa.2019.07.002

[6] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal, "Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness," *The International Journal of Aviation Psychology*, vol. 3, no. 3, pp. 203–220, jul 1993. [Online]. Available: https://doi.org/10.1207/s15327108ijap0303_3

[7] S. A. Balk, M. A. Bertola, and V. W. Inman, "Simulator sickness questionnaire: twenty years later," in *Proceedings of the Seventh International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design*, 2013, pp. 257–263.

[8] D. Jeong, Y. Sangbong, and Y. Jang, "Cybersickness Analysis with EEG Using Deep Learning Algorithms," in *IEEE VR*, 2019, pp. 827–835.

[9] N. Martin, N. Mathieu, N. Pallamin, M. Ragot, and J.-M. Diverrez, "Automatic recognition of virtual reality sickness based on physiological signals," *Ibs 2018*, 2018.

[10] S. Hickson, N. Dufour, A. Sud, V. Kwatra, and I. Essa, "Eyemotion: Classifying Facial Expressions in VR Using Eye-Tracking Cameras," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019, pp. 1626–1635.

[11] K. Olszewski, J. J. Lim, S. Saito, and H. Li, "High-fidelity facial and speech animation for VR HMDs," *ACM Transactions on Graphics*, vol. 35, no. 6, 2016. [Online]. Available: https://doi.org/10.1145/2980179.2980252

[12] T. Piumsomboon, G. Lee, R. W. Lindeman, and M. Billinghurst, "Exploring natural eye-gaze-based interaction for immersive virtual reality," in *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, 2017, pp. 36–39.

[13] M. B. Flanagan, J. G. May, and T. G. Dobie, "The role of vection, eye movements and postural instability in the etiology of motion sickness," *Journal of Vestibular Research: Equilibrium and Orientation*, vol. 14, no. 4, pp. 335–346, 2004.

[14] J. F. Golding, "Motion sickness susceptibility," *Autonomic Neuroscience*, vol. 129, no. 1-2, pp. 67–76, 2006.

[15] R. S. Kennedy, J. E. Fowlkes, K. S. Berbaum, and M. G. Lilienthal, "Use of a motion sickness history questionnaire for prediction of simulator sickness." *Aviation, Space, and Environmental Medicine*, vol. 63, no. 7, pp. 588–593, 1992.

[16] G. Menshikova, A. Kovalev, O. Klimova, A. Chernorizov, and S. Leonov, "Testing the Vestibular Function Development in Junior Figure Skaters Using the Eye Tracking Technique," *Procedia - Social and Behavioral Sciences*, vol. 146, pp. 252–258, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.sbspro.2014.08.123

[17] T. Wada and K. Yoshida, "Effect of passengers' active head tilt and opening/closure of eyes on motion sickness in lateral acceleration environment of cars," *Ergonomics*, vol. 59, no. 8, pp. 1050–1059, 2016. [Online]. Available: http://dx.doi.org/10.1080/00140139.2015.1109713

[18] Y. Li, F.-X. Wu, and A. Ngom, "A review on machine learning principles for multi-view biological data integration," *Briefings in Bioinformatics*, vol. 19, no. 2, pp. 325–340, dec 2016. [Online]. Available: https://doi.org/10.1093/bib/bbw113

[19] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, nov 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[20] M. L. Ibáñez and F. Peinado, "Walking in VR: Measuring presence and simulator sickness in first-person virtual reality games," *CEUR Workshop Proceedings*, vol. 1682, pp. 49–60, 2016.

[21] W. Xu, H.-N. Liang, Z. Zhang, and N. Baghaei, "Studying the Effect of Display Type and Viewing Perspective on User Experience in Virtual Reality Exergames," *Games for Health Journal*, 2020.

[22] D. Monteiro, H.-N. Liang, W. Xu, M. Brucker, V. Nanjappan, and Y. Yue, "Evaluating enjoyment, presence, and emulator sickness in VR games based on first-and third-person viewing perspectives," *Computer Animation and Virtual Worlds*, vol. 29, no. 3-4, p. e1830, 2018.

[23] W. Xu, H.-N. Liang, Y. Yu, D. Monteiro, K. Hasan, and C. Fleming, "Assessing the effects of a full-body motion-based exergame in virtual reality," in *Proceedings of the Seventh International Symposium of Chinese CHI*, ser. Chinese CHI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–6. [Online]. Available: https://doi.org/10.1145/3332169.3333574

[24] D. Monteiro, H.-N. Liang, A. Abel, N. Bahaei, and R. de Cassia Monteiro, "Evaluating Engagement of Virtual Reality Games Based on First and Third Person Perspective Using EEG and Subjective Metrics," in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, 2018, pp. 53–60.

[25] J. Wang, R. Shi, Z. Xiao, X. Qin, and H.-N. Liang, "Effect of render resolution on gameplay experience, performance, and simulator sickness in virtual reality games," *Proc. ACM Comput. Graph. Interact. Tech.*, vol. 5, no. 1, may 2022. [Online]. Available: https://doi.org/10.1145/3522610

[26] B. John, "Pupil diameter as a measure of emotion and sickness in VR," *Eye Tracking Research and Applications Symposium (ETRA)*, pp. 4–6, 2019.

[27] E. S. Pane, A. Z. Khoirunnisaa, A. D. Wibawa, and M. H. Purnomo, "Identifying Severity Level of Cybersickness from EEG signals using CN2 Rule Induction Algorithm," *2018 International Conference on Intelligent Informatics and Biomedical Sciences, ICIIBMS 2018*, vol. 3, pp. 170–176, 2018.

[28] G. Thut and C. Miniussi, "New insights into rhythmic brain activity from TMS-EEG studies," *Trends in Cognitive Sciences*, vol. 13, no. 4, pp. 182–189, 2009.

[29] Y. H. Nam, Y. Y. Kim, H. T. Kim, H. D. Ko, and K. S. Park, "Automatic detection of nausea using bio-signals during immerging in a virtual reality environment," *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, vol. 2, no. February, pp. 2013–2015, 2001.

[30] A. Gardé, P.-M. Léger, S. Sénécal, M. Fredette, E. Labonté-Lemoyne, F. Courtemanche, and J.-F. Ménard, "The Effects of a Vibro-Kinetic Multi-Sensory Experience in Passive Seated Vehicular Movement in a Virtual Reality Context," in *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI EA '18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1–6. [Online]. Available: https://doi.org/10.1145/3170427.3188638

[31] A. Garcia-Agundez, C. Reuter, H. Becker, R. Konrad, P. Caserman, A. Miede, and S. Göbel, "Development of a Classifier to Determine Factors Causing Cybersickness in Virtual Reality Environments," *Games for Health Journal*, vol. 8, no. 6, pp. 439–444, 2019.

[32] N. Martin, N. Mathieu, N. Pallamin, and M. Ragot, "Virtual reality sickness detection : an approach based on physiological signals and machine learning," in *ISMAR*, 2020, pp. 508–520.

[33] M. S. Dennison, A. Z. Wisti, and M. D'Zmura, "Use of physiological signals to predict cybersickness," *Displays*, vol. 44, pp. 42–52, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141938216301081

[34] O. L. Bock and C. M. Oman, "Dynamics of subjective discomfort in motion sickness as measured with a magnitude estimation method," *Aviation Space and Environmental Medicine*, vol. 53, no. 8, pp. 773–777, 1982.

[35] J. Reason, "Motion sickness: Some theoretical and practical considerations," *Applied Ergonomics*, vol. 9, no. 3, pp. 163–167, 1978.

[36] A. Brainard and C. Gresham, "Prevention and treatment of motion sickness," *American Family Physician*, vol. 90, no. 1, pp. 41–46, 2014.

[37] S. Patro and K. K. Sahu, "Normalization: A preprocessing stage," *arXiv preprint arXiv:1503.06462*, 2015.

[38] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A comparison of ARIMA and LSTM in forecasting time series," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018, pp. 1394–1401.

[39] T. Fushiki, "Estimation of prediction error by using K-fold cross-validation," *Statistics and Computing*, vol. 21, no. 2, pp. 137–146, 2011.

[40] R. Robinson, K. Wiley, A. Rezaeivahdati, M. Klarkowski, and R. L. Mandryk, ""Let's Get Physiological, Physiological!": A Systematic Review of Affective Gaming," in *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, ser. CHI PLAY '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 132–147. [Online]. Available: https://doi.org/10.1145/3410404.3414227

[41] P. Budhiraja, M. R. Miller, A. K. Modi, and D. Forsyth, "Rotation Blurring: Use of Artificial Blurring to Reduce Cybersickness in Virtual Reality First Person Shooters," *arXiv preprint arXiv:1710.02599*, oct 2017.

[42] S. Hillaire, A. Lecuyer, R. Cozot, and G. Casiez, "Depth-of-Field Blur Effects for First-Person Navigation in Virtual Environments," *IEEE Computer Graphics and Applications*, vol. 28, no. 6, pp. 47–55, nov 2008.

[43] R. Shi, H.-N. Liang, Y. Wu, D. Yu, and W. Xu, "Virtual Reality Sickness Mitigation Methods: A Comparative Study in a Racing Game," *Proc. ACM Comput. Graph. Interact. Tech.*, vol. 4, no. 1, 2021. [Online]. Available: https://doi.org/10.1145/3451255

[44] D. Monteiro, H.-N. Liang, J. Wang, H. Chen, and N. Baghaei, "An in-depth exploration of the effect of 2D/3D views and controller types on first person shooter games in virtual reality," in *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2020, pp. 713–724.

[45] L. Pomarjanschi, M. Dorr, P. J. Bex, and E. Barth, "Simple gaze-contingent cues guide eye movements in a realistic driving simulator," *Human Vision and Electronic Imaging XVIII*, vol. 8651, p. 865110, 2013.

[46] M. J. Cevette, J. Stepanek, D. Cocco, A. M. Galea, G. N. Pradhan, L. S. Wagner, S. R. Oakley, B. E. Smith, D. A. Zapala, and K. H. Brookler, "Oculo-Vestibular Recoupling Using Galvanic Vestibular Stimulation to Mitigate Simulator Sickness," *Aviation Space and Environmental Medicine*, vol. 83, no. 6, pp. 549–555, 2012.

[47] E. Medina, R. Fruland, and S. Weghorst, "VIRTUSPHERE: Walking in a human size VR "hamster ball"," *Proceedings of the Human Factors and Ergonomics Society 52nd Annual Meeting, New York, NY, USA*, no. 27, pp. 2102–2106, 2008.

[48] A. S. Fernandes and S. K. Feiner, "Combating VR sickness through subtle dynamic field-of-view modification," *2016 IEEE Symposium on 3D User Interfaces, 3DUI 2016 - Proceedings*, pp. 201–210, 2016.

[49] D. Monteiro, H.-N. Liang, X. Tang, and P. Irani, "Using Trajectory Compression Rate to Predict Changes in Cybersickness in Virtual Reality Games," in *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2021, pp. 138–146.

[50] T. C. Peck, L. E. Sockol, and S. M. Hancock, "Mind the gap: The underrepresentation of female participants and authors in virtual reality research," *IEEE transactions on visualization and computer graphics*, vol. 26, no. 5, pp. 1945–1954, 2020.