Walk in Place Based VR Locomotion System

Master Thesis

Sanket Suresh Kokane

Aalborg University

Electronics and IT

AALBORG UNIVERSITY

STUDENT REPORT

Electronics and IT

Aalborg University

http://www.aau.dk

Title:

Walk in Place Based VR Locomotion System

Theme:

Computer Graphics, Computer Vision, Virtual Reality

Project Period:

Spring Semester 2023

Project Group:

VGIS 1049A

Participant(s):

Sanket Suresh Kokane

Supervisor(s):

Claus Brøndgaard Madsen

Copies: 1

Page Numbers: 39

Date of Completion:

2nd June 2023

Abstract:

Virtual Reality (VR) users typically spend only 15 to 20 minutes per session due to simulation sickness. This sickness is often caused by locomotion systems that don't allow natural physical movement, leading to a mismatch between visual and vestibular systems and also disrupt the immersive experience. Walk in place (WiP) locomotion systems offer a controlled and natural way to move in VR by simulating walking while remaining stationary. However, existing WiP solutions such as expensive treadmills or tracking sensors are not practical for average users. Software-based WiP solutions lack freedom of movement and struggle with occluded body parts. Also none of these solutions have explored a deep learning approach for motion identification. This thesis proposes a webcam-based WiP VR locomotion solution using Mediapipe Pose estimation and deep learning. The solution aims to provide intuitive and flexible walking-based locomotion. The solution was tested on flat and sloped terrains to assess user acceptance, experience, and cybersickness compared to traditional controller-based locomotion.

Preface

This is a Masters Thesis report, written by Sanket Suresh Kokane during the tenth semester of Masters in Vision, Graphics and Interactive Systems at Aalborg University. This thesis examines walk in place solution for locomotion in VR using mediapipe pose estimation. This report is the documentation of the work carried out for the duration of the development of this solution. Any material in this report, which is not cited, is composed by the author. The author would like to express their gratitude towards supervisor Claus B. Madsen for his continued guidance and support.

Aalborg University, June 2, 2023

Sanket Suresh Kokane

<skokan21@student.aau.dk>

Contents

Preface **v**

1 Introduction **2**

1.1 Locomotion in VR 2

2 Literature Review **5**

2.1 Hardware Based Solutions 5

2.2 Software Based Solutions 7

2.3 Pose Estimation 8

2.4 Activity Tracking 9

2.5 Cybersickness 10

3 Methodology **12**

3.1 Preliminary Solutions 12

3.1.1 Manual landmark processing 12

3.1.2 LSTM based solutions 13

3.2 Unity Mediapipe Connection 18

3.3 Testing Scene 18

4 Testing **22**

4.1 Test Setup 22

4.2 Discussion 24

4.2.1 Cybersickness 24

4.2.2 Simulation Experience 25

5 Limitations and Future Work **29**

6 Conclusion **32**

Bibliography **33**

A CSQ-VR Questionnaire **36**

B User Experience Questionnaire **39**

Chapter 1

Introduction

The purpose of this chapter is to provide a brief overview of VR locomotion systems, their drawbacks and the proposed walk in place solution.

1.1 Locomotion in VR

Locomotion in VR implies the method of simulating movement. VR simulations and games provide users with a virtual world to explore, requiring them to engage in physical actions. Locomotion systems facilitate this movement in the virtual world. The most natural way of moving in a simulation would be to physically walk which would require a large enough space which is seldom available. Thus, locomotion systems are designed to facilitate motion in the simulation while being physically stationary.

A plethora of locomotion systems exists with varying philosophies and usability. A widely used locomotion system in games and other simulations is a controller-based movement. It is simple, intuitive and physically less intense. However, its biggest drawback is being cognitively intense and can result in severe cybersickness. This limits users ability to stay in the scenario for long and breaks the

proverbial immersion in VR[1].

Another commonly used VR locomotion technique is point and teleport. It is used in simulations that require less physical interaction and is a more visual experience. This method mitigates the problem of cybersickness as users don't actually feel the movement but teleport to the desired location[14]. This, however, isn't very immersive and provides minimal interactive motion. Its not efficient enough for two of the most popular VR usecases, video games and training simulators. It is also not intricate enough to lead to research[11].

Walk in place is a very intuitive method of locomotion in VR that requires the user to perform a walking motion while being stationary in the physical world. WiP reduces the risk of cybersickness to a great degree while maintaining immersion and providing quite an interactive experience of the virtual world[11][1]. WiP can be implemented using a multitude of technologies each of which can be hardware based, software based, or a blend of both. Chapter 2 explores existing WiP solutions in detail. The hardware based solutions require specialised hardware like a treadmill or a platform that can be space consuming and expensive and therefore isn't the most reasonable solution for most home users. Cheaper and compact specialised hardware like sensors can be used to track users motions. However, these sensors need constant maintenance, charging and also may not be affordable. To reduce the dependency on specialised hardware, computer vision based WiP solutions used pose tracking and a simple webcam. These methods generally would use human pose tracking models like openpose[2] or YOLOv7[19] and then calculate difference between joints to identify walking. However, these methods tend to fail with the drawbacks of their underlying pose tracking algorithms. Simple calculation based methods also do not account for varying walking patterns among different people.

The proposed solution in this thesis explores Mediapipe Pose[4] estimation as a base for tracking in a WiP solution for locomotion in VR. Mediapipe is known to be lightweight, CPU-accelerated, can work at high fps, and provides optimized pre-trained models making it an ideal solution for detecting the walking motion for VR locomotion. Mediapipe Pose provides landmarks or key points which are then passed through an LSTM model to predict walking motion. This information is then communicated to the VR locomotion system. This solution was tested to assert the advantages of WiP locomotion including reduced cybersickness, intuitive control and increased immersion. The tests required candidates to navigate through a linear passage and an inclined surface using WiP and a controller-based locomotion system. The candidates were also asked to fill out questionnaires to rate both locomotion systems.

Chapter 2

Literature Review

This chapter dives into the existing WiP solutions, exploring both software and hardware-based approaches. It also examines the different pose estimation algorithms and evaluates their effectiveness for WiP locomotion. Furthermore, it also explains the concept of cybersickness, how to measure and mitigate it.

2.1 Hardware Based Solutions

Omni by Virtuix[17], Kat-Walk C2[5], and many more treadmill based solutions exist to provide the best WiP solutions as it relates users walking, directly to their VR avatar's motion in the simulation. These solutions, however, are very expensive and not reasonable for home use. They also need their own power source which tend to be quite exhaustive. Omni One can be seen in fig 2.1 and Kat-Walk C2 can be seen in fig 2.2.

Figure 2.1: Omni One[17]

Figure 2.2: KatWalk C2[5]

Compact hardware based WiP solutions with remarkable accuracy are also available like a wrist worn sensor suggested by Park et al.[13] or a set of up to five sensors by WalkOVR[18] seen in figure 2.3 that can track more than just walking. These solutions also however require constant charging and generic upkeep of specialised hardware. They also tend to be quite expensive.

Figure 2.3: WalkOVR Mocap[18]

2.2 Software Based Solutions

Software based solutions for WiP are affordable, customized to usecases and require generic or no additional hardware.

A WiP solution based on movements of HMD proposed by Lee et al tracks the VR HMD(Head Mounted Display) and identifies walking based on a pattern of movement created as the user walks in place[10]. However, to get a proper range of values, the user needs to jog in place to generate a series of values that can be identified as a step. This can be quite exhausting and not really feel like walking. Further, its requires significant speed in movement to identify fast walking which not only causes fatigue but also increases the risk of cybersickness.

Kim et al[7] proposed an OpenPose[2] based WiP solution that requires a Microsoft Kinect camera to calibrate users position which is then fixed. The camera can then calculate the movement of user's ankle and identifies a walking pattern. The research tests out multiple Force Integration Driven(FID) based solutions where one unit of force is applied to push the user in the simulation based on movements of their ankles as seen in figure 2.4. The motion of ankles is identified using OpenPose pose tracking and the force is applied on the avatar in the simulation as the ankle is moved over a predefined threshold. The walking speed is kept in check by an opposing resistant force.However this method only considers movements of ankles as features to identify walking. This could attribute to the reduced accuracy when the user weren't looking straight at the camera as OpenPose does not perform well in case of occlusion. This method also requires users to take some steps to calibrate the camera and the users also need to stand on a mat thus making it a very rigid system to deploy.

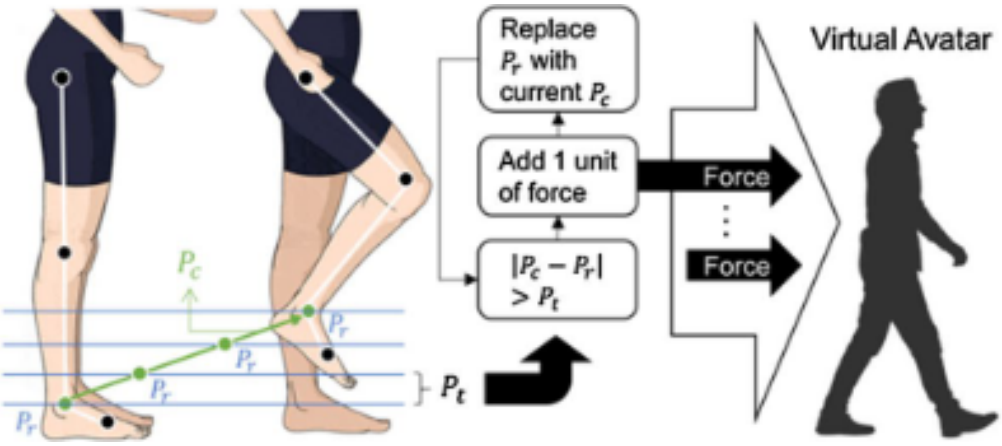


Figure 2.4: Visualization of FID WiP[7]

2.3 Pose Estimation

The motivation behind this thesis is to find a low cost WiP solution that doesn't require specialised hardware, is accurate enough and adaptable for most VR use-cases. No existing solutions have explore MediaPipe Pose Estimation[4] to track walking as other pose estimation methods like OpenPose and YOLO have shown sufficient performance. However, with recent advancements, MediaPipe supports CPU-acceleration, has less computational complexity and yet provide better FPS. Even though OpenPose and MediaPipe have identical accuracy, MediaPipe performs better when a part of body is occluded and has better accuracy for 3D pose estimation[3]. YOLOv7[19] is another similarly accurate pose estimation technique but also lacks behind MediaPipe in the same metrics as OpenPose. YOLOv7 tracks pose in each frame of the video feed resulting in low frame rates. It is also CPU accelerated but uses significantly more resource. MediaPipe outperforms YOLOv7 in low light conditions and with occluded body parts.[9]. Table 2.1 compares YOLOv7, OpenPose and MediaPipe pose tracking.

Feature	OpenPose	YOLOv7	MediaPipe
Resource use	CPU and GPU	GPU	CPU
FPS	15-20	25-30	30-40
Accuracy on COCO dataset	70%	80%	90%
Latency	100-200 ms	50-100 ms	20-50 ms
Memory usage	1 GB	2 GB	512 MB
Model size	100 MB	200 MB	50 MB
Inference time	100 ms	50 ms	20 ms
Batch size	1	8	16

Table 2.1: Comparison of pose estimation models[2][4][9][19][3]

Thus, the drawbacks of OpenPose based solution by Kim et al.[7] can be overcome by using MediaPipe instead. The low accuracy in side poses can be due occluded ankles which can are tracked better by MediaPipe. MediaPipe will also enable the solution to work without GPU and still churn out significant FPS.

2.4 Activity Tracking

Unlike pose estimation, detecting walking motion requires identifying a series of poses that collectively count as a step. Thus, pose estimation algorithms, by themselves cannot identify a walking motion. Walking being a very humane activity, the patterns can vary person to person drastically. Therefore, pose estimation needs to be paired with another component that can process the results of the pose estimation over time to predict walking and similar complex motions.

Putra et al. presented a comparison of different LSTM architectures for human activity tracking using MediaPipe pose landmarks.[15] All of the compared LSTM architectures yielded highly accurate results with VA-LSTM-SYSU performing the best out of the five models. The goal was to recognise multiple human activities

like run, skip, jump and also walk. Thus, MediaPipe pose estimation along with an LSTM is a feasible solution for walk in place detection.

2.5 Cybersickness

Cybersickness has existed in the relms of VR locomtion since its inception and persists to hold back large scale adaption of VR. When moving in VR simulations, users experience multiple undesirable effects like fatigue, nausea, strained eyesight, breathlessness, disorientation, and more. These symptoms and their intensity can vary based on factors such as age, gender, and pre-existing physical conditions or disabilities.

Cybersickness, similar to motion sickness is the result of sensory mismatch termed as Vection. Vection is the sensation of experiencing physical motion due to visual percetion of motion while being stationary. This is attributed to the mismatch of what users see to the signals perceived by user's vestibular and proprioceptive systems, which contribute to their sense of balance and body position[6].

Simulations that involve locomotion often try to minimise the cybersickeness based on user tests. The severity of cybersickness experienced in a VR simulation can be quantified using a cybersickness quotient, calculated using a survey that assess symptoms like nausea and disorientation. These symptoms are indicative of visual strain, fatigue, dizziness, and other related experiences.

To measure cybersickness in a coherent manner, Kourtesis et al. proposed the Cybersickness in Virtual Reality Questionnaire (CSQ-VR)[8]. CSQ-VR consists of six questions, with two questions for each of nausea, vestibular and oculomotor senses. Users can rate their feeling of those symptoms from a range of 1 to 7, representing the absence to extreme presence of the symptom. The cybersickenss

score is just the sum of all the values assigned to each response. For further reference, Appendix A contains the CSQ-VR questionnaire.

A cybersickness study can help understand the underlying drawbacks of the simulation and locomotion system. These drawbacks can be mitigated to some extent through simple measures like surround audio, speed control, turn control and other visual aids. However, some locomotion systems might further stray away in terms of an immersive experience trying to reduce cybersickness. Thus, the CSQ-VR will be used to measure the cybersickness score of the proposed WiP solution to find the trade off between minimal cybersickess and maximum immersion.

Chapter 3

Methodology

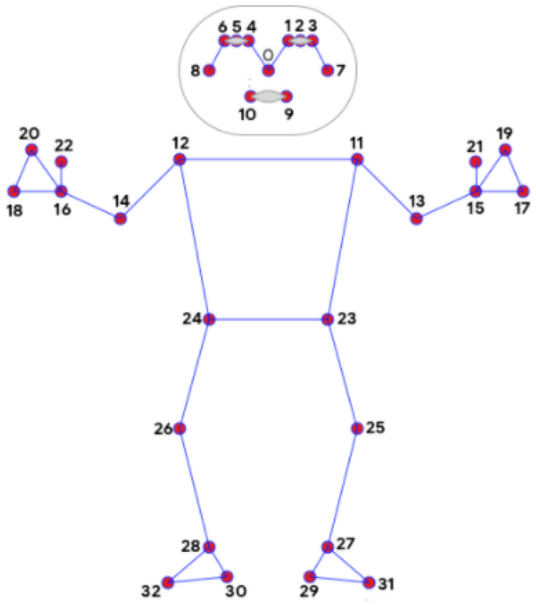
This chapter delves into the various solutions explored in order to accomplish the set objectives. The drawbacks encountered during the implementation of each solution are highlighted along with how they were resolved by adopting different approaches. Section 3.1 explains MediaPipe and LSTM based solutions and progression through various LSTM architectures and datasets. Section 3.2 explains programmatic coupling of MediaPipe with Unity to relay WiP to the user in simulation. Section 3.3 describes the simulation development.

3.1 Preliminary Solutions

3.1.1 Manual landmark processing

A real-time system such as a motion based locomotion requires light weight solutions. So as part of having minimal processing, an initial solution tried involved getting MediaPipe pose landmarks seen in figure 3.1and calculating the difference between knee landmark y values to detect walking. This solution would detect any motion of the knees as walking and didn't account for difference in height of the users and produced false positive errors at a very high rate. Thus, for robustness, ankle and heel landmarks were also considered which didn't make a huge differ-

ence. To account for relative movement of the legs, difference of either ankle's y value with knee's y value under a threshold was used to detect walking. Since the threshold was set manually, this solution was also sensitive and produced false positive results.



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

Figure 3.1: Mediapipe Pose landmarks

3.1.2 LSTM based solutions

With no signs of coherent results from simply processing landmarks, a more engineered solution based on LSTM was explored. MediaPipe pose estimation was used to extract joint landmarks from videos of people walking in place. Workout videos from YouTube where a person was walking in place were used to extract landmarks. Screenshots from these videos can be seen in figure 3.2. The videos were edited to only include walking in place movements, labeled as 'walking' and rendered at 1920x 1080 at 60fps. Videos where people did other activities were also used and labeled as 'not_walking'. The label also included a three digit number as identifier.

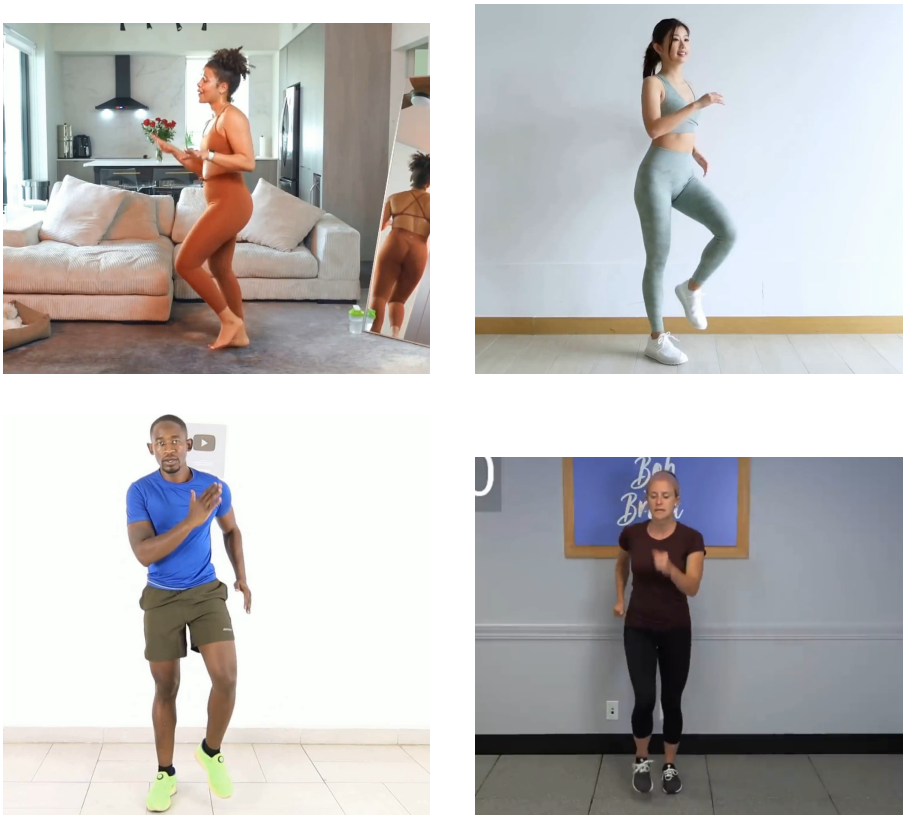


Figure 3.2: Screenshots of workout videos used for training

The preprocessing involved reading each video, one frame at a time and extracting the mediapipe pose landmarks, then dropping the z coordinate and visibility of each landmark and then adding the video identifier, frame number, the x and y coordinates of each landmark and label from the video name and writing it to a file. 10 percent of this data was used as validation and rest as training. This data was used to train a LSTM based on the VA-LSTM-SYSU as described in [15]. This LSTM was trained with a batch size of 32 for 350 epochs at a learning rate of 0.01. The resulting training accuracy was 0.97 and 0.084 loss along with 0.88 validation accuracy but a 0.65 validation loss. These metrics pointed to a overfit model and upon testing the results were far from expected and didnt work well on live camera feed. The architecture was then modified with various combinations of parameters of learning rate, batch size and dropout rates. A better accuracy rate of

0.98 for training and 0.92 over validation data was achieved, however, this model also overfit the training data and performed equally worse on live camera feed.

Another notable problem with this LSTM was it wouldn't just focus on the movement of the legs but the upper body as well and the testing over video feed showed quite low fps than expected. Thus, to reduce data load and make the system invariant to upper body movements, landmarks for joints above the hips were dropped and only hips, knees, ankles, heels and toes were considered. Dropping 22 out of 32 reduces the training time and could provided better fps on live camera feed. This new data, preprocessed in the same way, was used to train the LSTM with 0.01 learning rate, 32 batch size, 0.2 dropout rate on the 3 dropout layers and two dense layers for 350 epochs. The training and validation accuracy for this model was identical to the one before with more landmark data, however, the camera feed was able to predict steps improved minimally but the lack of fps persisted.

The workout videos where people would walk in place were quite similar to one another in the way that people would lift their legs and the time to take a step was identical. Thus It could be hypothesised that the data didn't have enough variety or balance. The ROSE Labs action recognition dataset "NTU RGB+D" and "NTU RGB+D 120" contains videos for "A99: run on the spot" as seen in figure 3.3which could be used to extend the training dataset[16][12]. 50 videos from this class were added to the training dataset and 4 to the validation set. These videos were also edited to be 1920x1080p at 60fps and the LSTM model was trained with the last mentioned parameters. Adding these videos also didn't change the training or validation accuracy as these videos were only 1 to 2 seconds long while the workout videos ranged from 20 to 60 seconds. Thus the aggregate frame count of the new videos is negligible to the older workout videos.

Figure 3.3: Screenshots from video with label A99: run on the spot from ROSE Dataset[16][12]

To find even more diversity in training data, 5 subjects were recorded walking and standing in place. The subjects were recorded using a webcam with 1920x1080 at 60fps. The subjects would first walk and then stand facing the camera, then at a 45°, looking to the left side then 135°, facing backwards, at 225°, looking to the right side of the camera and finally at 315°(All the rotation values in degrees are approximations). This was done to account for omnidirectionality. These videos were also edited to only have walking parts or not walking parts and renamed as such. To add more variety to this dataset, the videos were passed through a augmentation pipeline. The videos would be left shifted or right shifted at random without moving the subject out of the frame. The total size of this new dataset came out to be 527 videos comprising of 236 not walking and 291 walking videos all 1920x1080 at 60fps. The validation set contained similar 23 such videos with 10 not walking and 13 walking videos.

A new LSTM was defined to be trained on this larger dataset. The model has 3 LSTM layers, 2 dense layers with dropout at each layer. The dropout rate for each layer is 0.2. The step size for the LSTM was 10. The model used Adam optimizer with learning rate on 0.001 and binary cross-entropy loss. This model

trained for 350 epochs and batch size of 64. The Resulting accuracy was 0.97 for training and 0.91 for validation. The model can be seen in Fig 3.4.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 10, 128)           76288

 dropout (Dropout)           (None, 10, 128)           0

 lstm_1 (LSTM)               (None, 10, 64)            49408

 dropout_1 (Dropout)         (None, 10, 64)            0

 lstm_2 (LSTM)               (None, 32)                12416

 dense (Dense)               (None, 64)                2112

 dropout_2 (Dropout)         (None, 64)                0

 dense_1 (Dense)             (None, 1)                 65

=================================================================
Total params: 140,289
Trainable params: 140,289
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.4: Model Architecture

This model had good accuracy and also predicted well with live camera feed. Using MediaPipe Pose without the holistic landmarks significantly improved the FPS on the pose tracking. Thus, this model was considered as the final tracking part of the proposed WiP solution.

3.2 Unity Mediapipe Connection

The MediaPipe pose estimation is performed by a python script that communicates with a C# script over a socket connection. The C# script establishes a socket connection as a server and looks for clients that send messages. The C# script also controls the movement of a XR Rig in a Unity Scene that represents the user. Thus, as the python scripts predicts walking, it will send a message to the C# script which in turn moves the user in the simulation.

Initial solution involved translating the XR Rig as the C# receives a walking prediction and stop when it receives a not walking prediction. This was done using a simple transform.Translate() function. This method would move the user at uneven intervals and felt out of sync. The transform.Translate() moved the XR Rig in a linear way that didn't feel natural. Thus, a rigid body approach to move the XR Rig was explored.

A rigid body component is attached to the XR rig to give it physical properties. The addForce() method is used to apply a force on this rigid body to make it move. This force is calculated in the forward direction of the camera multiplied with a step size. For each walking prediction by the python script, the user will move one step discarding any predictions while the user is walking. This approach made the walking even, smooth and close to natural movement.

3.3 Testing Scene

Two Unity scenes were created to demonstrate the WiP system. The scenes were configured with XR Interaction toolkit to run the scene on a Meta Quest 2. The scenes contain an alley with walls on both side to ensure walking in a straight line. The walls are quad game objects with detailed textures on it. The path underneath

has a cobble stone texture. Trees of varying heights are placed behind the walls to exude a feeling of being in a different position when the user is actually moving. The scenes use a spotlight to act as a light source placed at a narrow angle to allow shadows to fall onto the path. Global illumination along with a skybox was used to create baked lightmaps and baked shadows on the scene to reduce GPU load.

Each scene had two XR Rigs one for WiP and the other for controller based locomotion. The XR Rig for WiP uses the socket connection in C# script to enable locomotion. The other rig uses assets provided by the XR Interaction toolkit to enable controller based locomotion. It uses the Locomotion System script to allow the rig to move and the Continuous Move Provider script that controls and configures the actual motion. The Continuous Move Provider script configures the controllers and read the joystick button on the left controller to move the XR rig. A Character Controller component is also added to the XR rig that handles the physics of motion based on controller input. The flat surface scene can be seen in figure 3.5.

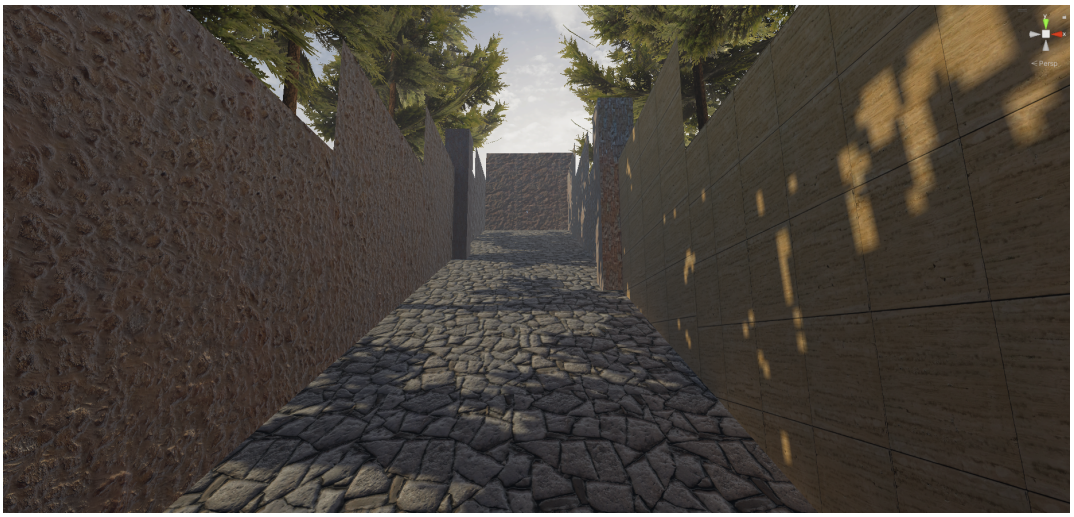
(a)

(b)

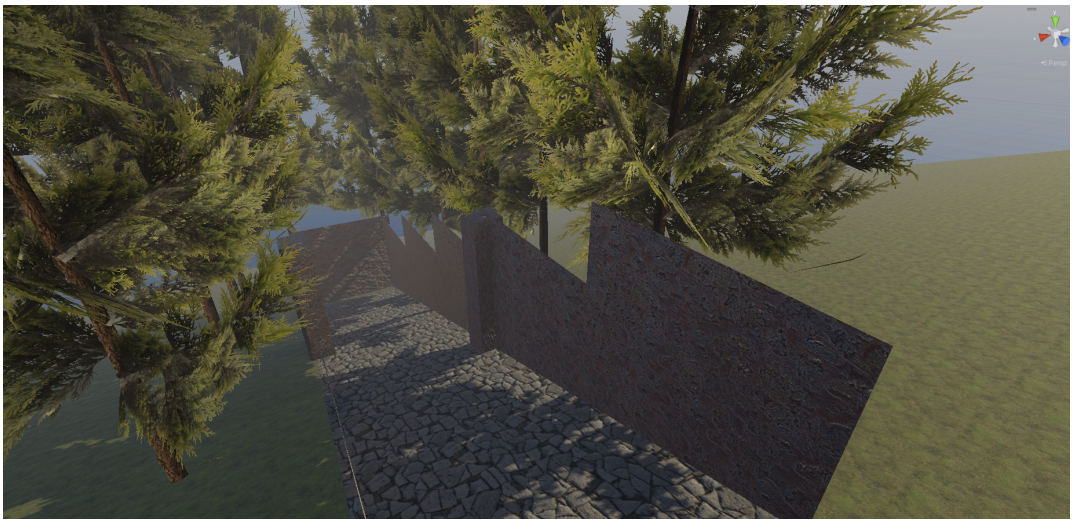
Figure 3.5: Flat surface alley scene used to test controller and WiP solutions

The same scene was used to also test locomotion on an inclined surface with the floor being a downwards slope. The inclination of the floor was 20°with the ground and the user would stand at the higher point of this slope. The inclined alley scene used for the test can be seen in figure 3.6. To ensure the force applied was with the slope and not in a straight line, a raycaster was used. This raycaster is placed at the base of the player and and checks for a hit with the ground collider. For every

hit, the Y position of the hit location is used to move the player along the slope.



(a)



(b)

Figure 3.6: Inclined surface alley scene used to test controller and WiP solutions

Chapter 4

Testing

To test the usability and and accuracy of the proposed WiP solution, user tests were conducted. The test involved 6 participants of ages ranging from 16 to 26 with median age of 24 years. Out of the six participants, 2 were female and rest were male. The test involved 4 tasks of moving from one end of the test scene to the other. Two scenes with a flat walking path and another with an inclined walking path were used for the test. These scenes are described with details in section 3.3. The participant would use the proposed WiP and a controller based locomotion system to complete the tasks in both the scenes. To minimize potential bias, the order in which the users performed the task was randomized. After completing the tasks, the participants were requested to complete the CSQ-VR to evaluate and compare the cybersickness scores associated with both locomotion methods. Furthermore, a general questionnaire was administered to gather feedback on the participants' experiences with the controller-based locomotion system and the proposed WiP locomotion system. For reference, the questionnaire can be found in Appendix B.

4.1 Test Setup

The tests were conducted indoors in a controlled environment. The simulation was running on Meta Quest 2 and developed in Unity Game Engine. A logitech

C920 webcam was used for pose tracking coupled with a python program that performed the prediction and communicated with the Unity scene. The webcam was mounted on a tripod 3 meters away from the general position of the user and 1.2 meters high from the ground. The Unity scene, python program and Quest were all running on Asus TUF Dash 15 supported by an Nvidia RTX 3060. The participants were asked to stand behind a taped line so that their body was completely visible to the camera. A room scale guardian was set on the Quest 2 which was only large enough such that the participants would see the passthrough if they stepped out of the view of the camera. Figure 4.1 shows the test setup and the view on the screen with a participant during test.

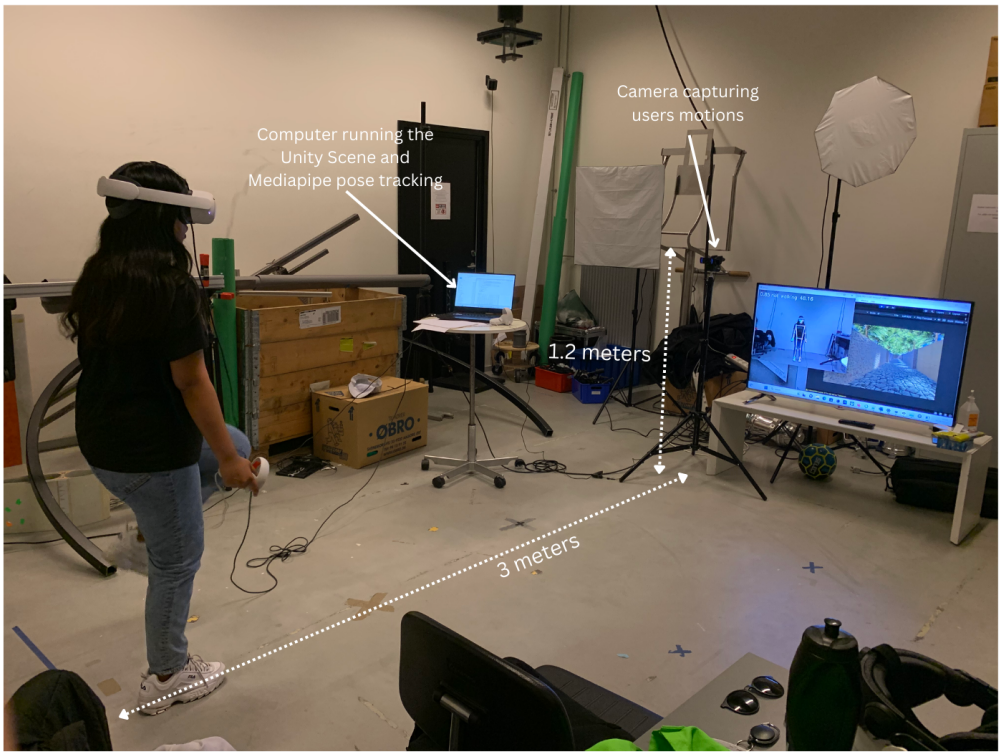


Figure 4.1: Test setup

4.2 Discussion

This section discusses the observations made during the tests in terms of usability and adaptability. Further WiP and controller based solutions are compared based on user's cybersickness scores and overall experience. This section also addresses the shortcomings and potential areas of improvements of the WiP solutions.

4.2.1 Cybersickness

After completing the four tasks, the participants were asked to fill out the CSQ-VR one for each locomotion system. The total cybersickness scores were calculated and compared for each method and for each user. Table 4.1 shows the participants age, gender and respective cybersickness scores for both methods. The score for each participant for each locomotion system can be minimum 6 and maximum 72. Overwhelmingly the cybersickenss score for controller based locomotion system was worse off for all the participants. Average score for controller system is almost twice more than its counter part.

Participant	Age	Gender	CSQ-VR Score(Controller)	CSQ-VR Score(WiP)
1	23	M	22	9
2	16	M	11	7
3	18	F	6	6
4	25	M	11	7
5	26	F	19	10
6	26	M	12	8
Average	-	-	13.5	7.8

Table 4.1: User CSQ-VR scores

Participant 2 had previous experience of playing VR games with controller and demonstrated tolerance to the cybersickness. Participants 1 and 5 had their first experience in VR simulations and suffered severe cybersickness however, their score

for the WiP based system was comparable to others. For controller based locomotion system, 5 out of 6 participants felt some sense of instability with 2 participants reporting moderate feeling of postural instability. 4 participants also reported some feeling of disorientation, 2 having very mild and other 2 having moderate effects. 5 participants also reported having visual discomfort ranging from very mild to moderate feeling. Thus, postural instability, disorientation and visual discomfort were the commonly experienced symptoms in controller based locomotion. These symptoms can be very much attributed to the 'vection' effect.

In terms of WiP method, 3 participants reported experiencing visual discomfort, 2 participants reported having very mild to mild feeling of postural instability. These are common symptoms with respect to locomotion in VR and can be dealt with visual and audio aids. However, 4 participants also experienced very mild feeling of visually induced fatigue and 5 participants experienced some form of oculomotor uneasiness. This could be a result of continuous physical motion causing lack of concentration and leading to visual strain. This maybe further worsen by the amount of leg movement required to move in the simulation. This could be mitigated with a smoother walking motion. On the contrary the cybersickness score stays quite close to the minimum value throughout the column and thus can be deemed to superior locomotion system for a more enjoyable simulation experience.

4.2.2 Simulation Experience

The express purpose of the WiP locomotion is to have the most immersive and realistic movement in the simulation. The test participant thus were asked to judge the locomotion systems based on their usability and general viability. Figure 4.2 shows how many participant preferred either of the locomotion systems based on certain criteria. When asked which system they found better, 4 participants preferred WiP over controlled based locomotion on account of a intuitive and immersive experi-

ence. Similar consensus was found when asked about which felt better walking on flat surface. However, the opposite was true in terms of walking on a downwards slope as 5 participants preferred the controller based system. A reasonable point of contention being the movement for WiP solution on flat surface was not similar to going down a slope. This could also be another reason as to why participants felt visual discomfort and postural instability. Stabilizing this slope movement along with a minor vertical movement to account for the different between the flat surface of the physical space and the slope in the simulation, could provide a better experience.

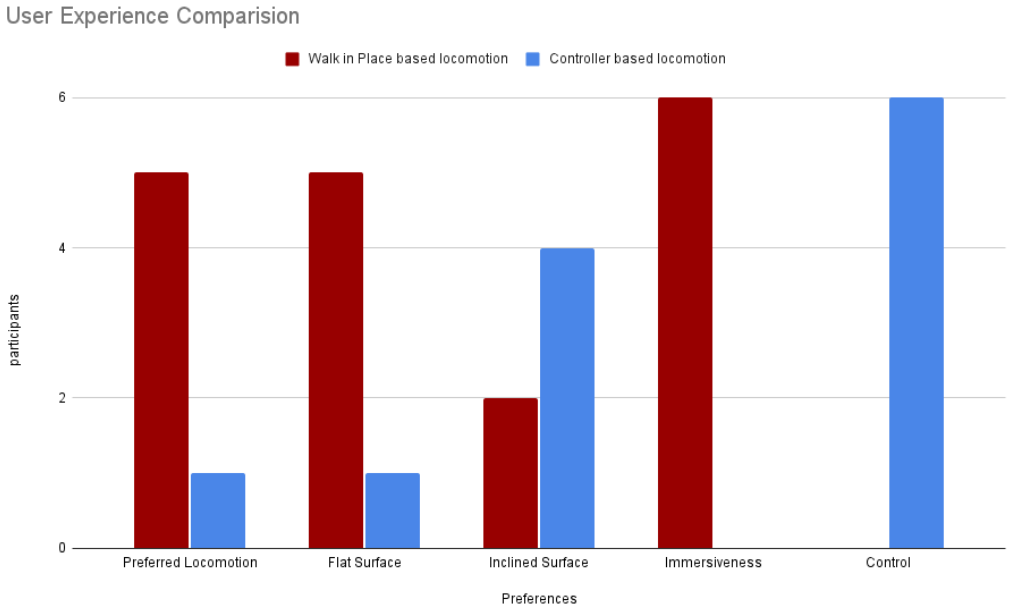


Figure 4.2: Comparison of users preferrences for controller vs WiP based locomotion system

A major problem with development of the walking prediction system was the need for larger knee movement due to lack of diversity in the dataset. This might tire off users quickly and defeat the purpose of an longer enjoyable VR experience. Thus when asked how long would users stay in each of the locomotion systems, 5 participants said for a moderate amount of time in WiP but 4 participants said the

same for controller based locomotion. This doesn't necessarily justify any method over other as current WiP solution might incurs fatigue overtime. However, based on amount of time people can use VR treadmills and how less people can stay in a controller based locomotion, it is not implausible to say a WiP system that doesn't require users to raise their legs as high, would overall be better than a controller based system.

The steps taken by the participants during the tests were recorded to measure the false negative steps. It requires 15 steps to reach from one end of the path to the other and no user could complete it within that. Table 4.2 shows the number of steps taken by each participant to complete the task for both the scene. On average the participants needed 20.8 steps on the flat surface and around 23 steps on the slope. The average miss rate for flat surface was 0.28 and 0.35 on the slope. The excess steps needed were generally at the start of each task when user would need to find the threshold of how high they needed to lift their legs. The higher miss rate on the slope was the result of participants often looking downwards and thus not lifting their legs up higher for the system to work. However, within a few steps the participant got used to the slope as well.

Participant	Steps in Flat Scene	Steps in Inclined Scene
1	19	20
2	18	24
3	25	27
4	21	20
5	22	26
6	20	22

Table 4.2: Participant Steps in Flat and Inclined Scenes

This was also evident as when the participants when asked which system provided them with a more immersive experience, everyone had the same opinion in favour of the proposed WiP locomotion system. Some reasoning for it being, that it felt 'natural', more realistic than using a controller and was 'fun and amazing' to use.

Some noticeable drawbacks to having over the top realism in movement was the complete removal from the physical world prompted participants to actually walk instead of walk in place. This caused tracking errors as the legs wouldn't be visible in the camera frame but was handled as the locomotion system would just pause if all the MediaPipe pose landmarks weren't visible. This could also explain the slight discomfort for the slope motion as this disassociation from physical world also meant reduced awareness of the physical terrain.

Another drawback of the LSTM based system is that the motion can never be as prompt as mirroring. This of course reduces the amount of control user can have on the motion as it would take a certain amount of physical motion to be reflected in the simulation especially something as natural and continuous as walking. To find how far off the proposed system is to one based on button press in terms of control, the users were asked to which system they found to provide better control over their movement. All the participants agreed on the controller based system being the better in terms of control. This being the most redeeming quality of controller based system and makes it so much easier to use than other locomotion systems. With more training data and a reliable omnidirectional locomotion, WiP could achieve better control.

Chapter 5

Limitations and Future Work

This chapter discusses the limitations of the proposed solution and possibilities for future improvements that could lead to a more robust and usable solution.

Numerous approaches were explored for every part of the proposed solution especially in motion tracking and prediction part. Sparse availability of training data and time and resource extensive models held back the proposed solution to meet some of its expectations.

The major of expectation that could have elevated the solution was omnidirectionality. Although snap turning in Unity could have aided the solution in terms of maneuvering abilities, the perception of natural movement would have been lost. As the training data used was mostly videos of people walking while facing the camera, the model wasn't accurate enough to handle omnidirectionality. The system was thus constraint to not predict when the user's shoulder and knees came closer as they would when the user turns to their side. MediaPipe pose does hold the capability to accurately detect pose landmarks omnidirectionally as discussed in section 2.3 given it tracks and predicts human poses on every frame.

Another commonly reported problem during the tests was the amount of leg movement that was required to move the user's VR avatar, didn't feel as natural as walking and incurred fatigue. This could also be attributed to the movement that the people in the training videos would do as the videos were physical exercise related. The training data later recorded did to some effect reduce the amount of leg lifting required but ideally the training data should only include natural walk in place movement. Videos of people simply walking were also at some point considered for training however, the MediaPipe pose landmarks couldn't match that pattern to walking in place. Although this didn't deter test participants from moving in the solution once they figured it out and overwhelmingly were in favour of using WiP instead of controller based solution.

An odd problem in the system that further enhanced the impact of the previous point was height variance of the test participants. Test participant shorter that 170 cms had a really hard time getting the pose estimation model to identify their walking patters. However, adjustments to the test setup and camera placements enabled them to get moving but with unnaturally high leg lifts. Further work in this area to relatively have variety in the training data with respect to height of people and in turn distances between their joints is a possible solution.

For an even better experience, test participant repeatedly mention lack of control over the step size and speed which they found adequate in the controller based locomotion. This certainly is a necessary feature to make the system actually applicable to VR simulation usecases.

The LSTM architecture used in the proposed solution could also amount to the above mentioned drawbacks. Thus, other architectures could also be explored for better accuracy and control over movement. The current model has excellent FPS throughput ranging from 40 to 55 fps. However, anywhere around 25 to 30 FPS is

sufficient for continuous motion. Therefore, a model that can trade off the accuracy and FPS for smoother movement is still desirable.

Chapter 6

Conclusion

The objective of this thesis was to create a deep learning based WiP locomotion system using MediaPipe that addresses cybersickness while also providing an intuitive and immersive user experience. Existing software based walk in place solution are very rigid in terms where and how the user stands and doesn't provide adequate omnidirectionality. The proposed solution certainly provides some freedom of movement as long as tracking is possible and has possibilities for omnidirectionality and speed control. It doesn't require any specialised hardware and only uses a common RGB camera thus being most cost effective compared to VR treadmills, tracking sensors and even specialised camera based WiP solutions. This solution was successful in identifying its cybersickness effects which are minor at worst. Based on user tests, it can also be asserted that the proposed solution does provide an intuitive locomotion technique and an immersive experience. The proposed solution also promotes use of deep learning as an activity tracker instead of basic keypoint analysis for a sophisticated and more robust solution.

Bibliography

[1] Costas Boletsis **and** Dimitra Chasanidou. "A Typology of Virtual Reality Locomotion Techniques". **in***Multimodal Technologies and Interaction*: 6.9 (2022). ISSN: 2414-4088. URL: https://www.mdpi.com/2414-4088/6/9/72.

[2] Zhe Cao **andothers**. *OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields*. 2019. arXiv: 1812.08008 [cs.CV].

[3] Jen-Li Chung, Lee-Yeng Ong **and** Meng-Chew Leow. "Comparative Analysis of Skeleton-Based Human Pose Estimation". **in***Future Internet*: 14.12 (2022). ISSN: 1999-5903. URL: https://www.mdpi.com/1999-5903/14/12/380.

[4] Google. *Mediapipe Pose*. Accessed: 05 11, 2023. Year of publication. URL: https://developers.google.com/mediapipe/solutions/vision/pose_landmarker.

[5] katvr. *katwalkc2*. Accessed: 05 08, 2023. Year of publication. URL: https://www.kat-vr.com/products/kat-walk-c2.

[6] Juno Kim **andothers**. "Effects of linear visual-vestibular conflict on presence, perceived scene stability and cybersickness in the oculus go and oculus quest". **in***Frontiers in Virtual Reality*: 2 (2021), **page** 582156.

[7] Woojoo Kim, Jaeho Sung **and** Shuping Xiong. "Walking-in-place for omnidirectional VR locomotion using a single RGB camera". **in***Virtual Reality*: 26.1 (2022), **pages** 173–186. ISSN: 1434-9957. DOI: 10.1007/s10055-021-00551-0. URL: https://doi.org/10.1007/s10055-021-00551-0.

[8] Panagiotis Kourtesis **andothers**. "Cybersickness in virtual reality questionnaire (csq-vr): A validation and comparison against ssq and vrsq". **in***Virtual Worlds*: **volume** 2. 1. MDPI. 2023, **pages** 16–35.

[9] Vikas Gupta Kukil. *YOLOv7 vs MPP*. Accessed: 05 08, 2023. Year of publication. URL: https://learnopencv.com/yolov7-pose-vs-mediapipe-in-human-pose-estimation/.

[10] Juyoung Lee, Sang Chul Ahn **and** Jae-In Hwang. "A walking-in-place method for virtual reality using position and orientation tracking". **in***Sensors*: 18.9 (2018), **page** 2832.

[11] Sungkil Lee **andothers**. "Effects of Locomotion Technique on Motion Sickness, Presence, and User Performance in a Virtual Environment". **in***Journal of Human-Computer Interaction*: 35.4 (2019), **pages** 299–311.

[12] Jun Liu **andothers**. "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding". **in***IEEE Transactions on Pattern Analysis and Machine Intelligence*: 42.10 (2020), **pages** 2684–2701.

[13] Chanho Park, Kyungho Jang **and** Junsuk Lee. "Walking-in-place for vr navigation independent of gaze direction using a waist-worn inertial measurement unit". **in***2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*: IEEE. 2018, **pages** 254–257.

[14] Aniruddha Prithul, Isayas Berhe Adhanom **and** Eelke Folmer. "Teleportation in Virtual Reality; A Mini-Review". **in***Frontiers in Virtual Reality*: 2 (2021). ISSN: 2673-4192. DOI: 10.3389/frvir.2021.730792. URL: https://www.frontiersin.org/articles/10.3389/frvir.2021.730792.

[15] Ichsan Arsyi Putra, Oky Dwi Nurhayati **and** Dania Eridani. "Human Action Recognition (HAR) Classification Using MediaPipe and Long Short-Term Memory (LSTM)". **in***TEKNIK*: 43.2 (2022).

[16] Amir Shahroudy **andothers**. "NTU RGB+D: A large scale dataset for 3D human activity analysis". **in***Proceedings of the IEEE conference on computer vision and pattern recognition*: 2016, **pages** 1010–1019.

[17] virtuix. *Omnibyvirtuix*. Accessed: 05 08, 2023. Year of publication. URL: https://www.virtuix.com/.

[18] walkovr. *walkovr*. Accessed: 05 08, 2023. Year of publication. URL: https://walkovr.com/.

[19] Chien-Yao Wang, Alexey Bochkovskiy **and** Hong-Yuan Mark Liao. *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. 2022. arXiv: 2207.02696 [cs.CV].

Appendix A

CSQ-VR Questionnaire

CyberSickness in Virtual Reality Questionnaire (CSQ-VR)

A brief tool for evaluating the Virtual Reality Induced Symptoms and Effects (VRISE)

Please, from 1 to 7, **circle** the response that better corresponds to the presence and intensity of the symptom.

Nausea A: Do you experience nausea (e.g., stomach pain, acid reflux, or tension to vomit)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

Nausea B: Do you experience dizziness (e.g., light-headedness or spinning feeling)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

Vestibular A: Do you experience disorientation (e.g., spatial confusion or vertigo)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

Vestibular B: Do you experience postural instability (i.e., imbalance)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

CyberSickness in Virtual Reality Questionnaire (CSQ-VR)

A brief tool for evaluating the Virtual Reality Induced Symptoms and Effects (VRISE)

Oculomotor A: Do you experience a visually induced fatigue (e.g., feeling of tiredness or sleepiness)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

Oculomotor B: Do you experience a visually induced discomfort (e.g., eyestrain, blurred vision, or headache)?

1	2	3	4	5	6	7
Absent Feeling	Very Mild Feeling	Mild Feeling	Moderate Feeling	Intense Feeling	Very Intense Feeling	Extreme Feeling

Please write below any additional comments relevant to the question above:

Category	Symptom	Symptom Intensity	Category Score
Nausea	Nausea (Nausea A)		
	Dizziness (Nausea B)		
Vestibular	Disorientation (Vestibular A)		
	Imbalance (Vestibular B)		
Oculomotor	Fatigue (Oculomotor A)		
	Discomfort (Oculomotor B)		
CSQ-VR Score =			

Symptom Intensity = the score provided by the responder.

Category Score = Score A + Score B

CSQ-VR Score = Nausea score + Vestibular score + Oculomotor score

CyberSickness in Virtual Reality Questionnaire (CSQ-VR) derives from the VR Neuroscience Questionnaire (VRNQ).

Both were developed by Panagiotis Kourtesis.

Appendix B

User Experience Questionnaire

Which locomotion system would you prefer to use?

Which system was better for a flat surface?

Which system was better for the slope?

How long can you stay in
Controller based locomotion system:
Walk in Place based locomotion system:

Which locomotion system provided a greater sense of immersion in the virtual environment?

Which locomotion system do you think gave you better control on the movement?