Cybersickness assessment framework(CSAF): An Open Source Repository for Standardized Cybersickness Experiments

Adriano Viegas Milani*
École Polytechnique Fédérale de Lausanne

Nana Tian†
École Polytechnique Fédérale de Lausanne

Ronan Boulic‡
École Polytechnique Fédérale de Lausanne

ABSTRACT

Studies of cybersickness (CS) often require a significant investment in creating the primary VR environment and other experiment-related features. In addition, minor differences in VR content in independent studies may lead to opposite results. This motivates us to introduce a standardized CS assessment framework to facilitate the development of CS-related experiments. This paper describes the concept of the framework, the current progress of the technical implementation, and future goals.

1 INTRODUCTION

Cybersickness is a complex, distressing, and yet inadequately understood phenomenon arising from immersive media [5]. Recent studies have demonstrated a growing interest in investigating the impact of content factors and the efficacy of cybersickness reduction strategies [1, 6]. However, the neural mechanisms underlying cybersickness remain elusive, and the approaches to mitigate this phenomenon exhibit significant variations in effectiveness.

One of the major challenges facing cybersickness research is the lack of standardization in the development of virtual reality (VR) environments, locomotion styles, and reduction techniques. This leads to prolonged development cycles and inconsistencies across studies on the same topic. Furthermore, essential information such as translational and rotational speed/acceleration, and duration of acceleration is often missing from the literature, making it challenging to evaluate and compare results from different studies. Additionally, the VR content sources used in research papers (e.g., 360 videos or VR game source code) are rarely made publicly available, hindering replicability.

To address these issues and promote future cybersickness research, we propose the Cybersickness Assessment Framework (CSAF). Our framework aims to overcome these challenges by providing a comprehensive environment for experiments that emphasizes ease of use, extensibility, and simple sharing of parameters. The proposed CSAF currently includes various locomotion methods, visual cybersickness reduction techniques, and customizable experiment settings with a centralized interface, enabling easy expansion and sharing of selected parameters through a preset system. The project, built in Unity, is available at `https://github.com/AdrianoVM/CSAssessmentFramework`.

2 RELATED WORK

There are multiple projects attempting to simplify different aspects of VR related research. If we first look at the issue of environment creation, VREX by Vasser et al. [12] is an example of such a project. It is a Unity project containing an intuitive menu system to create indoor environments, specializing in furnished rooms for psychological experiments. the Locomotion Usability Test Environment for Virtual Reality (LUTE) by Sarupuri et al. [9] is another codeless environment creation tool, facilitating the time-consuming task of creating different types of environment to test locomotion techniques. The situations proposed do not include any activities and are mostly variations of straight roads with some turns, which might not be sufficient for cybersickness research.

In terms of tools related to the experiment design, bmlTUX by Bebko et al. [3] is a toolkit presented in the form of a Unity Package, focusing on facilitating the technical aspects of experiment organization and evaluation. It handles variables, randomization, counterbalancing, and more, while proposing a preset system for different experiment designs. Ubiq-exp by Steed et al. [11], built upon the Ubiq library, is another package that simplifies experiment creation, but focusing on the multiplayer aspects, with avatars, voice communication, remote experiments and more. Both tools are in the form of packages meant to be added to existing projects, and as such do not include a complete bundle with environments, techniques, or even cybersickness related experiments.

This brings us to cybersickness reduction techniques. Samuel Ang and John Quarles proposed GingerVR [2], a repository of popular visual cybersickness reduction techniques, which were inspired by existing literature. These techniques were summarized and developed using Unity, in order to reduce the time spent on re-implementing commonly used techniques from scratch and to address the inconsistencies between customized implementations in the literature.

Cannavò et al. proposed the Locomotion Evaluation Testbed VR (LET-VR) as a solution to evaluate locomotion techniques in virtual reality environments [4]. The LET-VR testbed offers a variety of locomotion techniques and experiments, and implements a protocol and scoring system to rank them based on custom requirements.

While the LET-VR testbed is focused primarily on locomotion techniques, it does not have the same focus as GingerVR and our proposed CSAF framework on cybersickness. The LET-VR lacks certain features that are specifically designed for cybersickness studies, such as a user-friendly interface and a system for reducing cybersickness. However, these features could potentially be added through a fork of the project.

CSAF is designed to be a separate framework with a focus on these specific features, offering maximum flexibility and customization. Table 1 provides a comparison of the different frameworks and their features, highlighting the important aspects for a cybersickness framework. While all of the related works aim to simplify VR research, they differ in their scope and focus.

*e-mail: adriano.viegasmilani@epfl.ch
†e-mail: nana.tian@epfl.ch
‡e-mail: ronan.boulic@epfl.ch

	VREX [12]	bmlTUX [3]	Ubiq-exp [11]	GingerVR [2]	LUTE [9]	LET-VR [4]	CSAF
Environment Customization	✓				✓		✓
Multiple environments	✓				✓	✓	>
Environment Creation	✓						>
CS Reduction Techniques				✓			✓
Locomotion Techniques	✓					✓	✓
GUI	✓	✓			✓		✓
Minimal-Coding	✓	✓			✓	✓	✓
Tutorials	✓	✓	✓			✓	>
Preset System		✓					✓
Designed for Expansion		✓	✓				✓
CS related Experiment						✓	✓
Advanced Experiment Control		✓				✓	>
Multiplayer Experiment			✓				X
Logging	✓	✓	✓			✓	✓
Grading system						✓	X
Tool Format	Project	Package	Package	Repository	Repository	Project	Project

Table 1: Comparison of features in related work. ✓: implemented partially or completely, >: planned for next version, X: Not planned

3 CSAF CONCEPTUAL BASIS

Based on the background, we have identified a list of requirements for the primary version of the Cybersickness (CS) assessment framework. These requirements are as follows:

R1 **A Unity Project**. The entire framework must be implemented in Unity, as it is a widely used platform for VR development.

R2 **A Complete Game Environment**. The CS Assessment Framework must include a complete package for experiments with adequate and customizable VR environments.

R3 **A Set of Manipulable Factors of CS**. The experiment in the CSAF should include the most common content factors that could have an impact on cybersickness and allow for the manipulation of these factors (e.g. locomotion style, navigation path, and other visual stimuli).

R4 **Compatibility with CS Reduction Techniques**. The CSAF should be compatible with any Unity implementation of cybersickness reduction techniques and be easily expandable.

R5 **A Data Saving System**. The CSAF must be able to log real-time user data such as head movement (translation and rotation) and physiological signals (e.g. eye blinks) and save them into CSV files as required.

R6 **Preset System for Easy Sharing**. The CSAF should contain a preset system that enables researchers to generate, import, and export JSON files that describe all the information in the VR environment, allowing for easy sharing of experiment presets with lightweight files.

4 CS ASSESSMENT FRAMEWORK

The Cybersickness (CS) assessment framework (CSAF) was developed with the defined requirements in mind, and its design is centered around these requirements.

0

In order to meet the first requirement R1, CSAF is based on Unity version 2021.3 LTS. The project format was chosen over a repository of tools, as it would not be capable of offering the complete bundle requested in requirement R2. While it is possible to convert it into a Unity Package, it would not be the first choice, as the core objective is to provide a framework that contains most of the necessary tools for research, which works without additions, but is still modifiable and expandable, instead of being a tool that enhances other projects.

From the end-user perspective, the framework is separated into two main parts: the Setup Window and the Experiment Application.

4.1 Setup Window

The Setup Window serves as the central hub of the framework, outside the experimental context. It comprises four tabs, each corresponding to a distinct aspect of the experiment that may be modified: *Experiment*, *Environment*, *Vision*, and *Movement*. These tabs, or managers, contain multiple references to components present in the scene or within the assets. Components are the functional elements that impart behavior to GameObjects, and thus dictate the overall execution of the game. By organizing these components in one location, the Setup Window facilitates ease of use and organization, as the user does not need to remember the specific location of each script or its associated functionality. The appearance of the Setup Window can be observed in the accompanying figure 2.

Each manager provides a variety of options that begin to satisfy the experimental requirement R3. Options refer to the components that possess parameters that can affect the experiment. As can be seen in the figure 1, the different options available under each manager are readily apparent.

- Below *Experiment*, we have the `Game Handler`, responsible for the customization and control of the overall experiment, `Data Saver` for logging headset position, partly covering requirement R5, and `Game Assets` for sound management.

- The *Environment* tab contains a `Path Creator` and `Collectible Placer` to create a customizable path of collectibles, more detail on the activity in section 4.2, and

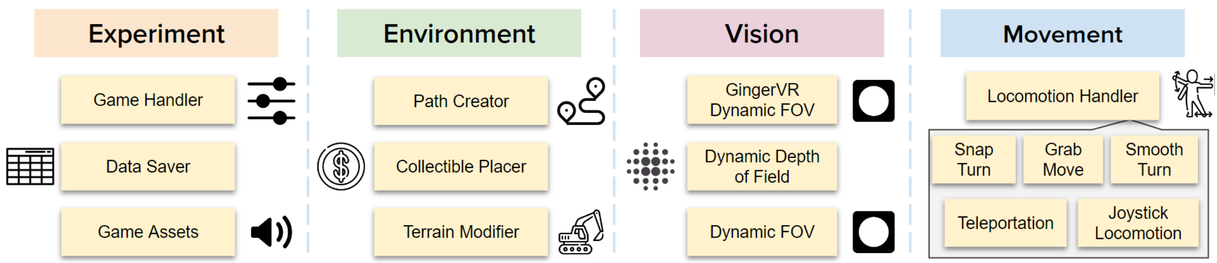

Figure 1: Visual Summary of Available Options in the Setup Window[1]. These four categories will be expanded in future versions.

`Terrain Modifier` which applies Perlin noise and smaller scale customizable repeating patterns to the terrain.

- For *Vision*, we first have a slightly modified version of Ginger VR's implementation of dynamic Field of View reduction [2], as well as our own version which operates differently. In addition, we have a dynamic Depth of Field effect capable of also functioning like a blur effect.
- Lastly, the *Movement* manager holds `Snap Turn` and `Smooth Turn` for orientation, as well as `Teleportation`, `Joystick Locomotion`, and `Grab Move` for locomotion. All these techniques use implementations from the XR Interaction Toolkit provided by Unity, and are grouped together for ease of use in our `Locomotion Handler`.

A key objective in future versions of the framework is to increase the quantity and degree of customization of options available. By expanding the range of options, researchers will be able to tailor the experimental experience to a greater extent.

When a component is frequently modified in order to personalize the experiment, it can be added to the appropriate tab of the Setup Window, without the need for any further modification. This approach ensures that any external script can be integrated in this manner, thereby fulfilling requirement R4.

Another Utility of this window is the capacity to load and save presets for all components in all tabs, or on a per-tab basis. They function similarly to unity's preset system for MonoBehaviours[2], so it is possible to do local changes to parameters and then load back the original values, or save the new changes. The main difference is that this creates a single JSON file for all components that can be shared between researchers working on the same project, instead of creating multiple presets for each MonoBehaviour, as per requirement R6.

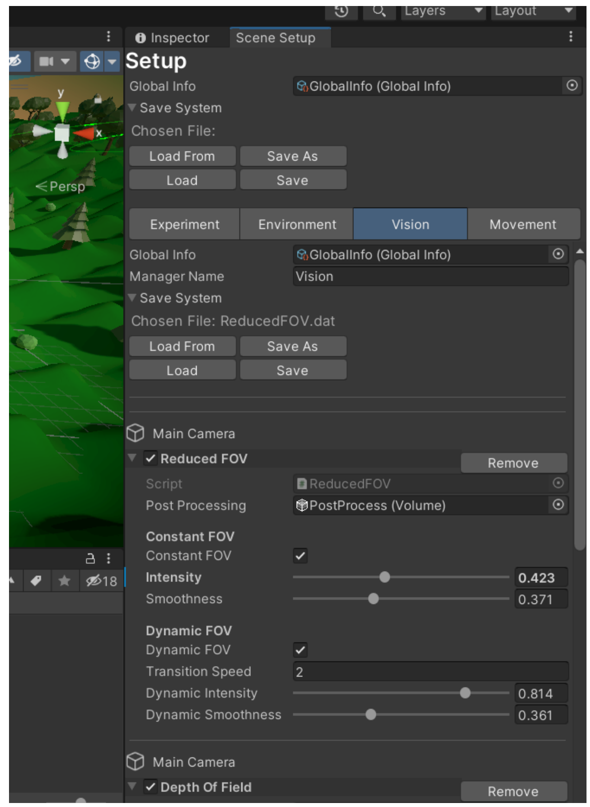

Figure 2: Setup Window with *Vision* tab/manager selected

4.2 Experiment Application

In order to fulfill requirement R2, an experimental application has been developed. This application consists of two distinct components: the menu and the experiment. The menu, as depicted in the first image of figure 3, enables the researcher to adjust certain locomotion related parameters, select the data to be recorded during the ensuing session, and exert control over experimental parameters such as duration.

The second component is the actual experiment, which is presented to and experienced by the participant. To ensure that the participant remains engaged for the duration of the experiment, a task or activity must be implemented that is both engaging and entertaining. One such task that has been implemented is a collectible game, in which the participant must navigate a path and collect various items, as illustrated in the second image of figure 3. It is important to note that this task may be insufficiently stimulating for longer sessions.

The environment in which the experiment takes place must also be carefully considered. It should be relatively simple to implement, not overly distracting, not excessively resource-intensive, and not likely to induce anxiety. This last consideration is particularly important, as anxiety is known to exacerbate symptoms of cybersickness [8]. The chosen environment, depicted in figure 3, is a toy-like forest. In future versions of the framework, a variety of environments may be made available for different types of experiments.

When the participant has collected enough items, or the timer has expired, or the researcher has chosen to end the experiment, the data that has been recorded is saved to a CSV file and the menu is re-displayed. The activity and environment can both be customized by utilizing the options presented in section 4.1, thus fulfilling requirement R2.

[2] https://docs.unity3d.com/Manual/Presets.html

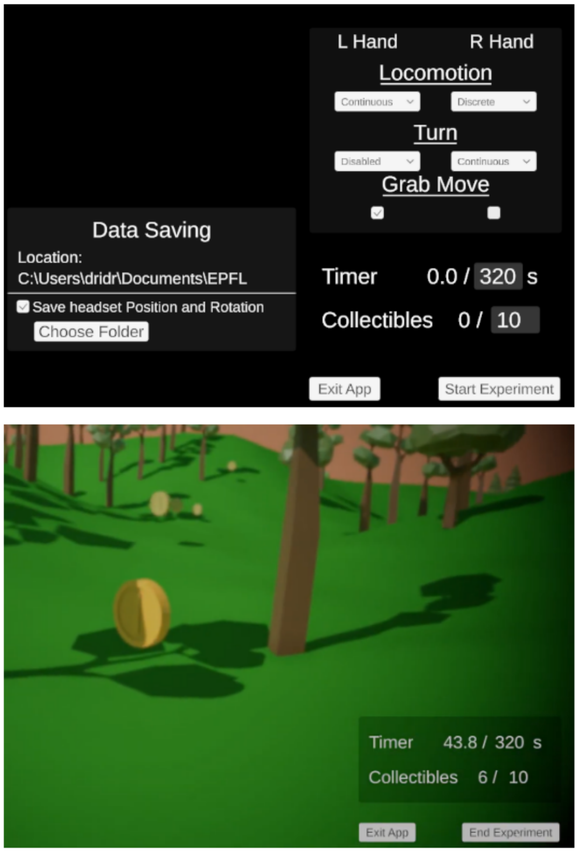

Figure 3: Menu showing selected options, and Experiment where the participant is following the path of coins. The second image is blurry as a dynamic depth of field effect is active.

5 Future study plan

The assessment of the proposed framework, referred to as CSAF, will center on three primary stages:

Usability: The framework will be evaluated for its ease of use by both potential users and researchers. User feedback will be obtained to determine the effectiveness of the framework and to identify areas for improvement in terms of user experience.

Functionality: The framework will be tested through pilot experiments to assess its functionality. These experiments will involve comparison of different techniques for reducing cybersickness and replication of previous studies.

Creation of innovative experiments: The framework will be utilized in full-scale novel experiments to evaluate its robustness and potential for expansion. For example, the framework will be used to compare innovative cybersickness reduction techniques with established methods, such as dynamic FOV reduction.

As CSAF is specifically designed for cybersickness research, additional features will be integrated to address previously unexplored areas. This may include incorporating balance tests, such as the rod-frame test [10] and postural instability tests [7], to better understand the impact of visual and vestibular senses on cybersickness susceptibility, and to track individual susceptibility during the experiment. Other general future plans include updating the current user interface settings to enhance usability in line with feedback from users, and incorporating additional scenes to increase diversity, as well as adding more locomotion and cybersickness reduction techniques.

6 Conclusion

In this study, we introduce a novel framework for the assessment of cybersickness (CS) concepts. The framework, implemented as an open-source Unity project, features an integrated Virtual Reality (VR) environment, a range of manipulable CS-related variables, various techniques for reducing cybersickness, and experiment-related utilities, including in-game questionnaires and data logging capabilities. This framework provides researchers with a streamlined and efficient means of conducting customizable experiments, with centralized interfaces, obviating the need to devote significant time and effort to the implementation of a VR environment from scratch. Furthermore, the framework enables the seamless expansion and dissemination of selected parameters. Researchers can distribute their experimental configurations and VR game data through a compact JSON file, enabling other individuals to effortlessly obtain the specifics of a VR experiment by importing the JSON file, with minimal requirement for additional dependencies.

7 Acknowledgment

This research is supported by the SNF Sinergia grant CR-SII5 180319.

References

[1] M. Al Zayer, P. MacNeilage, and E. Folmer. Virtual locomotion: a survey. *IEEE transactions on visualization and computer graphics*, 2018.

[2] S. Ang and J. Quarles. Gingervr: An open source repository of cybersickness reduction techniques for unity. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 460–463. IEEE, 2020.

[3] A. O. Bebko and N. F. Troje. bmltux: Design and control of experiments in virtual reality and beyond. *i-Perception*, 11(4):2041669520938400, 2020.

[4] A. Cannavò, D. Calandra, F. G. Pratticò, V. Gatteschi, and F. Lamberti. An evaluation testbed for locomotion in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):1871–1889, 2020.

[5] E. Chang, H. T. Kim, and B. Yoo. Virtual reality sickness: a review of causes and measurements. *International Journal of Human–Computer Interaction*, 36(17):1658–1682, 2020.

[6] J. Kim and T. Park. The onset threshold of cybersickness in constant and accelerating optical flow. *Applied Sciences*, 10(21):7808, 2020.

[7] S. Litleskare. The relationship between postural stability and cybersickness: it's complicated–an experimental trial assessing practical implications of cybersickness etiology. *Physiology & Behavior*, 236:113422, 2021.

[8] R. Pot-Kolder, W. Veling, J. Counotte, and M. Van Der Gaag. Anxiety partially mediates cybersickness symptoms in immersive virtual reality environments. *Cyberpsychology, Behavior, and Social Networking*, 21(3):187–193, 2018.

[9] B. Sarupuri, S. Hoermann, M. C. Whitton, and R. W. Lindeman. Lute: A locomotion usability test environmentfor virtual reality. In *2018 10th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*, pp. 1–4, 2018. doi: 10.1109/VS-Games.2018.8493432

[10] D. Spinelli, D. Spinelli, G. Antonucci, R. Daini, and P. Zoccolotti. Local and global visual mechanisms underlying individual differences in the rod-and-frame illusion. *Perception & Psychophysics*, 57(6):915–920, 1995.

[11] A. Steed, L. Izzouzi, K. Brandstätter, S. Friston, B. Congdon, O. Olkkonen, D. Giunchi, N. Numan, and D. Swapp. Ubiq-exp: A toolkit to build and run remote and distributed mixed reality experiments. *Frontiers in Virtual Reality*, 3, 2022.

[12] M. Vasser, M. Kängsepp, M. Magomedkerimov, K. Kilvits, V. Stafinjak, T. Kivisik, R. Vicente, and J. Aru. Vrex: an open-source toolbox for creating 3d virtual reality experiments. *BMC psychology*, 5(1):1–8, 2017.