PLUME: Enregistrer, Visualiser et Analyser le comportement utilisateur lors d’expériences XR

Charles Javerliat∗† Sophie Villenave∗‡ Pierre Raimbaud§ Guillaume Lavoué¶

Univ Lyon, Centrale Lyon, CNRS, INSA Lyon, UCBL, LIRIS, UMR5205, ENISE, F-42023 Saint Etienne, France

ABSTRACT

From education to medicine to entertainment, a wide range of industrial and academic fields now utilize eXtended Reality (XR) technologies. This diversity and growing use are boosting research and leading to an increasing number of XR experiments involving human subjects. The main aim of these studies is to understand the user experience in the broadest sense, such as the user cognitive and emotional states. Behavioral data collected during XR experiments, such as user movements, gaze, actions, and physiological signals constitute precious assets for analyzing and understanding the user experience. While they contribute to overcome the intrinsic flaws of explicit data such as post-experiment questionnaires, the required acquisition and analysis tools are costly and challenging to develop, especially for 6DoF (Degrees of Freedom) XR experiments. Moreover, there is no common format for XR behavioral data, which restrains data-sharing, and thus hinders wide usages across the community, replicability of studies, and the constitution of large datasets or meta-analysis. In this context, we present PLUME, an open-source software toolbox (PLUME Recorder, PLUME Viewer, PLUME Python) that allows for the exhaustive record of XR behavioral data (including synchronous physiological signals), their offline interactive replay and analysis (with a standalone application), and their easy sharing due to our compact and interoperable data format. We believe that PLUME can greatly benefit the scientific community by making the use of behavioral and physiological data available for the greatest, contributing to the reproducibility and replicability of XR user studies, enabling the creation of large datasets, and contributing to a deeper understanding of user experience.

1 INTRODUCTION

A wide range of industrial and academic fields now utilize eXtended Reality (XR). The accessibility and popularity of XR technologies have witnessed notable advancements in the past decade, resulting in a substantial increase in experimental human studies in XR environments. Whatever their final application, these research studies mostly aim to understand the user experience in a broad sense, e.g., for evaluating the quality of a new device or improving our global knowledge of human beings in a specific context.

To understand the user experience, researchers mostly rely on self-reported data, i.e., questionnaires or semi-structured interviews. However, these data have known limitations: they are post-experiment, meaning that users assess the memory of their experience rather than their actual experience, questionnaires easily become outdated or irrelevant to study-specific content, and biases can be introduced by the users e.g., by trying to guess the right answer. To avoid these limitations, researchers are increasingly turning towards behavioral data (e.g., body motions, gaze, actions) and physiological signals (e.g., electrodermal activity, heart rate) to assess the quality of experience. However, these data are more tedious to acquire, analyze, and share, especially in the case of 6DoF (Degrees of Freedom) XR studies. The required acquisition and analysis tools

Figure 1: The standalone PLUME Viewer application allows for the interactive visualization and analysis of behavioral and physiological data recorded with the PLUME Recorder Unity plugin. From left to right: hierarchy of objects in the virtual environment; heatmap of user position and 3D trajectory, highlighting of the most interacted objects; gaze direction; analysis control panel for visualizations; custom event markers; timeline with synchronized physiological signals tracks and markers.

∗These authors contributed equally and should be seen as joint first authors.
†e-mail: charles.javerliat@ec-lyon.fr
‡e-mail: sophie.villenave@ec-lyon.fr
§e-mail: pierre.raimbaud@ec-lyon.fr
¶e-mail: guillaume.lavoue@enise.ec-lyon.fr
are costly and challenging to develop. Moreover, there is no common format for XR behavioral data, which restrains data-sharing, and thus hinders wide usage across the community and replicability of studies.

To resolve the aforementioned issues and democratize the use of behavioral and physiological data for XR research, the PLUME software toolbox has been introduced [4]. This open-source software toolbox (containing PLUME Recorder, PLUME Viewer, and PLUME Python) allows for the exhaustive record of XR behavioral data (including synchronous physiological signals), their offline interactive replay and analysis (with a standalone application), and their easy sharing due to its compact and interoperable data format.

2 Background

In the scientific literature as well as in the industry, several software tools have been already proposed to collect and possibly visualize behavioral or/and physiological data of XR users. Tobii Ocumen commercial software [6] allows for the capture of VR user’s gaze, hand, and head motions, as well as its contextualized replay. LSL-Unity [3] enables the recording of physiological sensor signals synchronously with virtual objects’ position. XREcho [7] is a plugin for Unity to record XR sessions and replay them in the virtual environment (VE), integrating visualizations inside the VE of the position of virtual objects. It does not offer support for the recording of physiological signals except for eye-related ones. VRSTK [2] offers similar functionalities to XREcho but it can also record signals from a limited selection of physiological hardware as well as showing gaze directions inside the VE. Vizard [5] proposes a fully integrated VR system with eye-tracking and the possibility of integrating other physiological sensors. However, this proprietary solution is expensive and incompatible with Unity. Cognitive3D [1] is the most complete and versatile solution, available for Unity and Unreal Engine and compatible with standalone HMDs. It can record and replay XR sessions together with eye-tracking and physiological signals and offers analysis tools and post-experience dashboards. Unfortunately, it is a paid proprietary solution, and the free version is limited as it does not offer a record explorer in the context of the XR scene. It should be noted that experience data are stored on the company servers. These existing toolkits did not reach their public for several reasons: (i) too complicated to use or configure, (ii) lack of compatibility or versatility, (iii) incomplete set of features, (iv) proprietary format or (v) expensiveness. PLUME [4] solves these issues by gathering existing separate features plus proposing additional ones such as compatibility with standalone HMDs, compatibility with built applications, LSL integration to support a large range of sensors, interaction highlighting, and accurate gaze heatmaps. Beyond this large set of functionalities, the key feature of PLUME is its ease of use: a simple drag-and-drop of the Unity plugin in a project allows for the recording; PLUME Viewer is a standalone application that does not require Unity nor any VR device to be run; PLUME Python makes easy the export toward usual data formats. Lastly, PLUME was specifically optimized to minimize its impact on performance, a question not tackled in existing frameworks.

3 PLUME: record, replay, analyze, and share XR experiments

The PLUME software toolbox is composed of three main components, i) PLUME Recorder, a plug-n-play package to record virtual experiments in Unity, ii) PLUME Viewer, a standalone application to replay the record files without the need for the original Unity project, iii) PLUME Python, a module to extract data from the record file for external analysis and conversion to other formats (LSL, CSV, pandas dataframe). The features presented are listed in the following sections.

GitHub Page: https://www.github.com/liris-xr/PLUME

PLUME Recorder The PLUME Recorder is the cornerstone of the toolbox. It allows for continuously recording the state of the virtual environment in a Unity application with minimal impact on performances. By default, the recorder will record as much data as possible, namely object position, appearance, sound, interactions, and physiological signals (through a LabStreamingLayer integration). The recorder also allows for custom data recording, such as event markers or custom-defined data structures in Google Protocol Buffer files. This format is the one we use for all of our samples as it allows for a fast and frugal serialization, is platform-neutral and can be de-serialized in any language. The PLUME Recorder is compatible with Windows, Android, and iOS and with standalone devices. For the VR use case, we use the functionalities of OpenXR as much as possible so that it is compatible with many headsets. As the record files all follow the same serialization process and data format, they can be used interoperably across devices. For example, one could record an experiment on a standalone Android device and analyze the record file on a Windows machine, without the need for the original Unity project.

PLUME Viewer The PLUME Viewer is a standalone application for viewing and analyzing the record files independently of the Unity project. It offers analysis modules such as interactions analysis, 3D trajectories, in-context physiological signals tracks, position and eye gaze heatmaps for which export is available as point clouds with the scalar field embedded. PLUME Viewer is useful to observe a recorded experiment like a video in a media player, as it does not require anything more than the record files to reconstruct the VE.

PLUME Python The interoperability of the record files allows for other languages to load those files for external analysis. PLUME Python is a module that can load record files using the Protobuf package to filter and convert the data into more commonly used formats in data analysis, eg., pandas dataframe or CSV files. Embedded data such as LabStreamingLayer’s samples can be exported to XDF files for external use in tools such as SigViewer, EEGLAB or MoBI-LAB.

Acknowledgments

This work was supported by Auvergne-Rhône-Alpes region as part of the PROMESS project and by the French National Research Agency as part of the RENFORCE project (ANR-22-CE31-0023-03).

References