

A playback tool for reviewing VR experiences

Thomas Lopez
Technicolor, France

Olivier Dumas
Technicolor, France

Fabien Danieau
Technicolor, France

Bertrand Leroy
Technicolor, France

Nicolas Mollet
Technicolor, France

Jean-François Vial
Technicolor, France

ABSTRACT

Designing a VR content is a challenging task due to the complexity of the analysis of a user's experience. We propose a tool allowing a reviewer to record a user's movements when consuming a VR content, and to playback this experience. During the playback, the reviewer can visualize in real-time the visible objects to the user (within the field of view) or the interactive ones (within the workspace). Besides, the field of view and workspace of numerous devices can be simulated in order to predict how the content would fit. In the demonstration, a user is first invited to experience a VR scene. Then the recorded experience can be played back from a reviewer's point of view.

CCS CONCEPTS

• **Computing methodologies** → **Virtual reality**; • **Social and professional topics** → *Technology audits*; • **Human-centered computing** → Collaborative content creation;

KEYWORDS

VR, review, user's experience

1 INTRODUCTION AND RELATED WORK

With the growing VR market, more and more content is going to be created. Yet it is still challenging to review an immersive content. As a user is free to look around, a VR designer would hardly predict one's behavior. Besides, the user's live point of view cannot be directly experienced, otherwise cyber sickness would be felt. It appears then a need for a tool to analyze a user's experience in order to be able to iterate during the design process.

This issue relates to the challenges encountered in Collaborate Virtual Environments (CVE). Goebbels et al. defined an interactive system as a 3-step loop composed by awareness, action and feedback [2]. In the case of a collaborative system, the awareness includes the user's own perception and the perception of the other users. More precisely the authors defined this awareness in five points: proprioception; perception of his own physical and virtual input devices and of the virtual data set; perception of co-presence; perception of co-physical and virtual input devices and of shared virtual data set; perception of co-knowledge and co-status. The three last points are particularly important in our context. As a VR content is being reviewed, the reviewer needs to be aware of the user's status. In their paper, the authors also provided guidelines to design a proper CVE. They advise to represent the other users and their input devices to ensure a good understanding of their actions and limitations. Also, visual or audio latency decreases the quality of the awareness.

The reviewing of a user's VR experience can be partially done thanks to eye-tracking systems. Used for decades with traditional video content, such systems are now embedded in head-mounted displays (see [1] for a detailed review of the state-of-the-art on the visualization of eye-tracking data). Stellmach et al. have shown how to compute heatmap for 3D models [5]. 3D heatmaps are however more costly to produce than the 2D versions. To tackle this, Maurus et al. proposed a method to generate this data in real-time [4].

In our case, we need a system allowing to monitor the users behavior in real-time, which includes their gaze and their movements. Also we need to simulate the experience with other devices. Today there is no standard VR device and content creators may want to know what their content would look like with different systems.

2 THE REVIEWING TOOL

Our proposed tool allows to record a user's movement within a VR experience, and to play it back for reviewing purposes. In order to better understand the user's experience, the field of view (FoV) is displayed in real-time as well as the workspace (i.e. tracking space). Besides, numerous devices with their own FoV and workspace can be simulated (see [3] for the list of devices). The reviewer can also control the playback speed to better examine interesting events.

This tool was developed as a plugin for the Unity engine. The user's movements are recorded as an animation which can then be easily synchronized with the scene. During the playback, an avatar representing the user is thus animated thanks to an inverse kinematics algorithm (FinallIK) driven by the recorded points (i.e. head and hands if a standard Oculus Rift configuration was used).

2.1 User's Field of View and Workspace

The rendering of the user's field of view and workspace requires a specific compositing performed by a custom shader (see Figure 2). From the reviewer's point of view, a RGB image is captured as well as six masks (depth maps) which are composited into the final image I . The masks are the scene depth map D , the depth map of the inside of the workspace WI , the depth map of the outside of the workspace WO , the depth map of the inside of the FoV FI , the depth map of the outside of the FoV FO , and a depth map representing the occlusions O . To generate the depth map of each workspace, two meshes representing the space are rendered: one with the normals pointing toward the outside and another one with normals toward the inside. Then the reviewer's camera captures the two depth maps of these meshes in the scene. The same procedure applies for the masks associated to the FoVs. For the occlusion image, a pointing light is activated at the user's position in a simplified duplicated scene (only the meshes). As the light field has the same shape than the user's FoV, only the objects visible by the user appear lighted from the reviewer's point of view. The captured image, stored as



Figure 1: Left: Reviewer's view without additional information. Right: User's FoV (colored), workspace (black and white), and timeline controller (layout on top of the controller) are visible. The user, in grey, looks at the man and cannot see the woman.

a depth map, represents the occlusion map. The final image can be formalized as follows, where for each pixel i_{xy} , the ones in the user's FoV and not occluded appear colored (1), the ones in the workspace are in black and white (2) and the others are the depth map (3):

$$\forall i_{xy} \in I, i_{xy} = \begin{cases} i_{xy}, & d_{xy} > f_{xy}^o \wedge d_{xy} < f_{xy}^i \wedge o_{xy} > 0 \\ f(i_{xy}), & d_{xy} > w_{xy}^o \wedge d_{xy} < w_{xy}^i \\ d_{xy}, & \text{otherwise} \end{cases} \quad (1) \quad (2) \quad (3)$$

with $d_{xy} \in D$, $f_{xy}^o \in FO$, $f_{xy}^i \in FI$, $w_{xy}^o \in WO$, $w_{xy}^i \in WI$ and $o_{xy} \in O$. f is a function rendering a rgb pixel into a black and white pixel (averaging of the components). Figure 1 shows an example of a reviewer's point of view. Of course other representations of the FoV and the workspace are possible.

2.2 Timeline and Devices Control

As depicted in Figure 1, the reviewer has a dedicated interface to control the timeline of the scene. The interface allows to start and pause the animation, to change its speed, and to seek forward or backward. All the objects manipulated by the user or by the physics engine were recorded and saved as animations. Thus during playback, all animations are synchronized to ensure that the reviewer has the exact user's experience.

From this interface, the reviewer can also change the device simulated by the shader. The user's FoV and workspace is then modified according to the device characteristics.

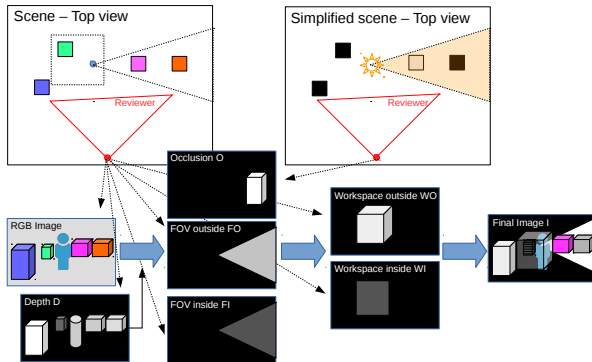


Figure 2: Overview of the compositing method

3 CONCLUSIONS & PERSPECTIVES

A tool for reviewing the user's experience with VR content is presented here. The user moving in the content and his or her field of view and workspace are displayed. This way the reviewer has a good understanding of what is seen and what is at the user's reach.

We designed a real-time scene where a couple is arguing in the living room. In the demonstration, users are first invited to explore the content with an Oculus Rift, and then to examine their experience as reviewers. From there, they may simulate the use of another VR device (i.e. StarVR, HTC Vive, Hololens, etc.).

Many features are going to be added to this tool. An annotation system will be implemented for highlighting flaws or unwanted artifacts in the content. Embedded eye-tracking systems will also be supported to refine the visualization of the FoV. Finally, we plan to enable the review of content experienced by multiple users.

REFERENCES

- [1] Tanja Blascheck, Kuno Kurzahls, Michael Raschke, Michael Burch, Daniel Weiskopf, and Thomas Ertl. 2014. State-of-the-art of visualization for eye tracking data. In *Proceedings of EuroVis*, Vol. 2014.
- [2] Gernot Goebbels, Vali Laloti, and Martin Göbel. 2003. Design and evaluation of team work in distributed collaborative virtual environments. In *Proceedings of the ACM symposium on Virtual Reality Software and Technology*. ACM, 231–238.
- [3] Will Mason. 2015. VR HMD Roundup: Technical Specs. <https://uploadvr.com/vr-hmd-specs/>. (2015). [Online; accessed 15-September-2017].
- [4] Michael Maurus, Jan Hendrik Hammer, and Jürgen Beyerer. 2014. Realistic heatmap visualization for interactive analysis of 3D gaze data. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 295–298.
- [5] Sophie Stellmach, Lennart Nacke, and Raimund Dachsel. 2010. Advanced gaze visualizations for three-dimensional virtual environments. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 109–112.