frontiers in Virtual Reality

ORIGINAL RESEARCH
published: 14 June 2021
doi: 10.3389/frvir.2021.676895

6DIVE: 6 Degrees-of-Freedom Immersive Video Editor

Ruairi Griffin[1], *Tobias Langlotz*[2] *and Stefanie Zollmann*[1]*

[1]Department of Computer Science, University of Otago, Dunedin, New Zealand, [2]Department of Information Science, University of Otago, Dunedin, New Zealand

Editing 6DoF videos using standard video editing tools is challenging, especially for non-expert users. There is a large gap between the 2D interface used for traditional video editing and the immersive VR environment used for replay. In this paper, we present 6DIVE, a 6 degrees-of-freedom (DoF) immersive video editor. 6DIVE allows users to edit these 6DoF videos directly in an immersive VR environment. In this work, we explored options for a timeline representation as well as UI placement suitable for immersive video editing. We used our prototypical implementation of an immersive video editor to conduct a user study to analyze the feasibility and usability of immersive 6DoF editing. We compared 6DIVE to a desktop-based implementation of a VR video editor. Our initial results suggest that 6DIVE allows even non-expert users to perform basic editing operations on videos in VR. While we did not find any statistically significant differences for the workload between the VR and the desktop interface, we found a statistically significant difference in user preference, with a preference for the VR interface. We also found higher ratings for the user experience metrics in VR captured by the user experience questionnaire.

Keywords: VR, VR video, video editing, 6DOF, VR headset

OPEN ACCESS

Edited by:
Ruofei Du,
Google, United States

Reviewed by:
Xiaoxu Meng,
Tencent Holdings Limited, China
Julian Frommel,
University of Saskatchewan, Canada

***Correspondence:**
Stefanie Zollmann
stefanie.zollmann@otago.ac.nz

Specialty section:
This article was submitted to Technologies for VR, a section of the journal Frontiers in Virtual Reality

Received: *06 March 2021*
Accepted: *25 May 2021*
Published: *14 June 2021*

Citation:
Griffin R, Langlotz T and Zollmann S (2021) 6DIVE: 6 Degrees-of-Freedom Immersive Video Editor. Front. Virtual Real. 2:676895. doi: 10.3389/frvir.2021.676895

1 INTRODUCTION

VR head-mounted displays (VR HMDs) have seen major developments in recent years. Nowadays, VR HMDs such as the Oculus Quest integrate tracking, controllers, and computation in a consumer-priced package[1]. Because of these recent developments in accessibility and in price, they can be increasingly found in normal households. Here, they cannot only be used for gaming but also for media consumption. In particular, immersive videos (either spherical, RGBD video, or a combination of both) are expected to become more relevant and have been the focus of recent research (Broxton et al., 2020; Attal et al., 2020; Zollmann et al., 2020). Immersive videos allow filmmakers to capture real-life dynamic scenes while providing the viewer the freedom to move and view these scenes from different perspectives (either by rotating the head or with a full 6 Degrees-of-Freedom, 6DoF). Contrary to traditional videos or monocular 360° videos, 6DoF videos are characterized by having depth information for each frame (Richardt et al., 2019). While there is an increasing number of state-of-the-art approaches for directly capturing immersive videos or recovering depth data, there is only a little work focusing on editing such material. In practice, such video material is often edited using tools designed for traditional 2D videos or a combination of several traditional tools. Some of these traditional tools have support for monocular 360° videos, for example by allowing viewing in an immersive VR display. Unfortunately, this support is rudimentary as one still needs to switch between

[1]Oculus: https://www.developer.oculus.com/learn/oculus-device-specs/

desktop editing and VR viewing as immersive editing is usually not supported. In addition, producing and editing content for VR is still challenging, and requires specialized knowledge similar to games programming. For example, producers are required to use tools like Unity3D (Nebeling and Speicher, 2018; Vert and Andone, 2019). Thus, producing edited content for VR is still cumbersome, especially for 6DoF videos. To our best knowledge, there is neither support for direct editing or even immersive editing of 6DoF videos nor research that investigates immersive editing for 6DoF or other volumetric videos.

Thus, working with 6DoF video requires a cumbersome process to preview changes, render the video, transfer it between different applications to finally view the resulting 6DoF video in an immersive VR HMD. On top of that comes the rendering time that is typically proportional to the length of a video and in the case of high-resolution footage needed for spherical videos, could take more than a minute for every minute of video, creating a very long feedback loop. Given the expected prevalence of 6DoF video, we believe that an approach designed to give real-time feedback for editing decisions is needed for 6DoF video which includes the support of immersive editing in a VR headset. In this work, we work towards closing that gap by presenting an approach for immersive editing of 6DoF videos. Here, we see in particular three main contributions that are described within this work:

- We develop and present a concept for editing 6DoF immersive videos in VR.
- We develop interaction techniques for this paradigm.
- We evaluate this paradigm, and the interaction techniques by developing a prototype, and running an initial user study.

In our explorative work, we draw upon existing UI paradigms and metaphors from traditional video editing tools, such as iMovie[2] or Adobe Premiere[3]. We also built on the work of Nguyen et al. (2017), which explores some aspects of immersive editing for monocular 360 videos.

Addressing the identified research gap and providing a more usable and efficient approach for editing VR videos will benefit both, expert users as well as more casual creators (Richardt et al., 2019). Better designed, and easier-to-use tools for editing will allow for both better and more VR experiences through eventually better and more immersive content. This is particular of relevance as immersive videos have many important applications whether for education, science communication, business training, or entertainment (Radianti et al., 2020; Reyna, 2018; Elmezeny et al., 2018).

2 RELATED WORK

In this paper, we investigate new methods for immersive 6DoF video editing. Our work is based on previous research and standards in video editing, VR video formats as well as VR video editing.

2.1 Video Editing

Video editing is the process of taking video material from a camera, reviewing it (Video Browsing), selecting which parts to keep, which to leave out (Cutting/Slicing), ordering the clips (Timeline placement), marking areas of interest for collaborators or future reference (Bookmarking) and the addition of effects and titles (Augmentation/Titling). There are other aspects of video editing, such as adding sound effects and audio mixing, color correction/color grading which we do not explore in this work.

Video editing applications, known as non-linear editors (NLEs) maintain an internal representation, a data structure that encodes the transformations from the source videos and assets to the final result. Typically when the editor is finished with editing, they render the result out to a standard video format that can be played in a standard video player. This is done by combining the footage and re-encoding it in an optimized format for delivery. Video editing applications are large, complex pieces of software developed over many years by teams of engineers. While there are some open-source NLEs[4], they deemed not suitable for being used in a VR headset due to complex codebases, tightly coupled view logic, or non-portable design. In this work, we focus on 6DoF video editing in an immersive VR environment. In order to investigate aspects of user experience and performance, we decided to develop our own editor but focus only on a small subset of features of typical NLEs.

The primary UI paradigm used in video editing to manipulate the state (transformations from source video to the final result) is the timeline. The timeline is a visual representation of all video material arranged temporally, often also displaying other metadata of the video clips. There are two common formats for timelines in NLEs. The basic format is a single track that represents a list of videos, these can be rearranged, transitions can be added between them and the length of each video can be changed. This type of timeline is more common in casual video editors. However, the most common format in professional non-linear editors consists of several tracks (layers) where the editor can freely place videos (small blocks representing the length and in/out points of each video clip). This allows, for example, for multiple videos to overlap and play at the same time, allowing for effects like Picture-in-Picture (where a one video is overlaid on another video).

2.2 VR Video

There are several choices of video content for VR, from the basic and now fairly ubiquitous monocular 360-Degree video to 6DoF light-field capture. Each type represents trade-offs between ease and cost of capture, ability to edit, and visual fidelity as well as immersion. While there are a small number of previous works that focus on video editing and compositing in AR (e.g., Langlotz et al., 2012), we focus on VR video content and video editing.

[2]iMovie: https://www.apple.com/imovie/

[3]Premiere: https://www.adobe.com/products/premiere.html

[4]Kdenlive: https://kdenlive.org/en/, OpenShot: https://www.openshot.org

2.2.1 Monocular 360-Degree Video

Monocular 360-Degree videos are videos that capture a full 360-degree perspective. Typically, they are captured by stitching the output from multiple cameras together using panoramic techniques (Brown and Lowe, 2007; Szeliski, 2007). These videos can be displayed in a VR headset by rendering the 360 panoramas for each frame onto a spherical or cylindrical representation. This allows users to rotate their heads when using a VR headset for viewing and provides three DoF, rotations around the X, Y, and Z-axis. However, translational movements are not supported in monocular 360 videos. Any translational movements by the user of the VR headset will not have any effect on the rendered content. There are no binocular depth cues such as convergence or binocular disparity, or motion parallax.

2.2.2 360-Degree Stereo Video

360-Degree Stereo videos take monocular 360-Degree videos and add a second viewpoint to add binocular disparity cues. The two separate views are captured a fixed distance apart (usually corresponding to an average inter-ocular distance). One view is used for the left eye, and the other for the right eye. Capturing videos with the correct disparity for every direction for 360-Degree video is a non-trivial problem. Early work by Ishiguro et al. (1992) proposed Omni-directional stereo (ODS) to create stereo panoramic views. ODS (Richardt (2020)) has been used to create stereo panoramas from cameras moving along a circular path (Richardt et al., 2013; Baker et al., 2020). Anderson et al. (2016) proposed the Jump system that uses a multi-camera rig to produce ODS video. Schroers et al. (2018) proposed a pipeline capturing and displaying VR videos, based on ODS panoramas. These approaches allow for mostly correct disparity for all directions except the poles (top and bottom of the sphere). Adding stereo restores binocular depth cues. In particular, binocular disparity gives users a much greater perception of depth for objects in their close to medium range vision. However, it is important to note that ODS is displayed with a fixed convergence, which can cause discomfort in some people. Furthermore, the user is still restricted to 3DoF since the stereo images are captured for an ideal viewing spot (close to the original capture position).

2.2.3 6 Degrees-of-Freedom Video

6DoF videos are a type of volumetric video, video data that consists of a 3D scene and a temporal component (Schwarz et al., 2018). For 6DoF videos, the user can move with six degrees-of-freedom. This includes the rotation around the X, Y, and Z-axis with the addition of translation on the X, Y, and Z-axis (Richardt et al., 2019). In this work, we focus on editing 6DoF video. There is a large body of existing work in capturing and playing back volumetric video. Volumetric video can be captured using RGB-Depth cameras (Maimone and Fuchs, 2011), synthesized from multiple cameras[5], generated from a single perspective using a deep neural network (Rematas et al., 2018), or using light-field approaches (Broxton et al., 2020).

The capturing of 6DoF video is an active research problem, though recent advancements in deep neural networks and low-cost high-quality head-mounted displays have accelerated the progress. Recent work by Broxton et al. (2020) demonstrate impressive results. Broxton et al. (2020) present an approach that synthesizes 6DoF videos from 46 action cameras placed on a sphere. They use a neural network to create a 3D representation of the scene, which they compress and view with a layered sphere representation. This approach captures dynamic lighting and atmospheric effects (effects that change based on the viewpoint). However, the approach does require complex multi-camera setups and produces significant volumes of data requiring 28 CPU hours processing time per frame. There are commercial solutions that capture 360-Degree RGBD videos (Kandao Obsidian R $\approx$ $4200 USD (Tan et al., 2018), as well as 360-Degree RGBD can be synthesized from 360 ODS video using stereo matching. These 360-Degree RGBD videos can be rendered efficiently in real-time on mobile hardware. There are some drawbacks in absolute fidelity when compared to other state-of-the-art methods. In particular, the draped-canopy geometry (Richardt et al., 2019) can cause some *uncanny valley* type effects, where regions of high depth gradient can create streaky triangles, but we believe it captures the essential elements of the 6DoF video.

Given the continued progress in 6DoF video capturing and playback, we have designed our system in a highly flexible way such that the viewer is a *black box* and could be swapped out to another 3D geometry-based method, for example, an MSI (multi-sphere image) or a voxel-based method, given it satisfies the performance requirements of our system (Serrano et al., 2019; Attal et al., 2020; Regenbrecht et al., 2019). This can even be done per clip, for example, mixing flat 360-Degree footage with 6DoF. We have used this approach to render non-360-Degree RGB-D videos alongside 6DoF video, demonstrating the versatility of our system. To avoid the challenges of complex capture setups and processing/playback pipelines, we decided to build our 6DoF editing framework based on 360-Degree RGB-D video.

2.3 VR Video Editing

While there is a large body of research into capturing VR content, there is relatively little research into the tools needed for editing this content. Nguyen et al. (2017) proposed VRemiere, a VR-based video editing interface for editing monocular panoramic VR videos, using an HMD (head-mounted display). The editor interface is controlled by a keyboard and mouse. They conducted a user study, which collected both quantitative and qualitative evidence to suggest expert VR video editors prefer the VR-based interface. While Nguyen et al. (2017)'s work is the first that investigated video editing in VR, there are still open gaps. One gap is that the interaction techniques used by VRemiere are based on traditional desktop input and as such are keyboard and mouse-driven. This creates major challenges for VR editing as the editor wearing the head-mounted display, still has to be seated at a desk holding a mouse. This means the user can not face the opposite direction and perform editing operations. We believe

[5]Intel true view: https://www.intel.com/content/www/us/en/sports/technology/true-view.html

FIGURE 1 | The immersive video editor is used to render a 6DoF video in VR. The editor contains UI elements for editing and replaying the video. Using video material publicly available (http://pseudoscience.pictures) processed with Stereo2Depth (Gladstone and Samartzidis, 2019).

that incorporating interaction with 6DoF controllers provides the editor more freedom of movement. The work by Grubert et al. (2018) showed that there is an advantage of using standard desktop keyboard input in VR for text entry. However, they also mention that this type of input is only suitable for "future information" workers that sit in front of a desk in a virtual environment.

Another gap of the work by Nguyen et al. (2017) is that it focused on editing monocular 360-Degree footage. Aspects of editing of 6DoF footage have not been investigated so far. When editing virtual reality video, one has to consider several new aspects compared to 2D video. One is the differences in storytelling between traditional 2D video and virtual reality video. This includes aspects of how the editor can direct the viewers' attention throughout the video and varies significantly compared to traditional 2D videos (Brown et al., 2016). For example, VR videos force viewers to be in the scene with the content of the video. This means viewers can be highly sensitive to things like action close to the camera, or movement of the camera. It is difficult to gauge these elements from an equirectangular projection of 360-Degree video, and even more difficult once we add depth cues that cannot be perceived on a traditional display. An immersive approach to editing allows the editor to experience the video as the viewer would, and make adjustments and choices appropriately with real-time feedback.

3 IMMERSIVE EDITING CONCEPT

When editing VR video material, one has to consider several aspects that differ compared to traditional 2D video. This includes attention direction, ego-motion, and depth perception. We considered these aspects when designing 6DIVE, our immersive editing concept.

The fundamental idea of our immersive editing approach is to narrow the gap between the 2D interface where video content is typically edited, and the immersive VR experience in which 6DoF video will be replayed. For this purpose, we developed a VR application in which the user can interactively edit 6DoF videos (**Figure 1**). We believe that the immersive VR environment can serve as an effective place to produce content with the right application design. To map the video editing paradigms into the immersive environment, we have to overcome two conceptual challenges.

- How to place both, the content and UI in the same immersive space, while minimizing placement and depth conflicts between them.
- How to design the UI to be usable, even with the lower accuracy of pointing input from the 6DoF controllers (compared to a mouse).

3.1 Video Replay

We use RGBD 360 videos as input for our immersive video editor and apply a mesh-based approach that uses the depth map channel of the video (aligned to the 360-Degree video) as a displacement map. We create a spherical 3D mesh geometry and displace each of the vertices. This creates an entirely convex geometry (from the perspective of the center of the sphere), which we call a draped canopy. The video replay differs compared to the replay of stereoscopic 360 VR replay where two panoramic images are rendered, each for one eye.

3.2 UI Placement

For video editing interfaces there are a number of UI elements required. One important aspect of immersive VR video editing is how to place both, the content and UI in the same immersive space without any conflicts. Nguyen et al. (2017) investigated these aspects for monocular 360 videos and render the interface at infinity and composite it over their 360-Degree video. In addition, they allow the user to move position of the UI by dragging it using the mouse. For 6DoF videos, we cannot render the interface at infinity as this will conflict with the depth cues of our scene

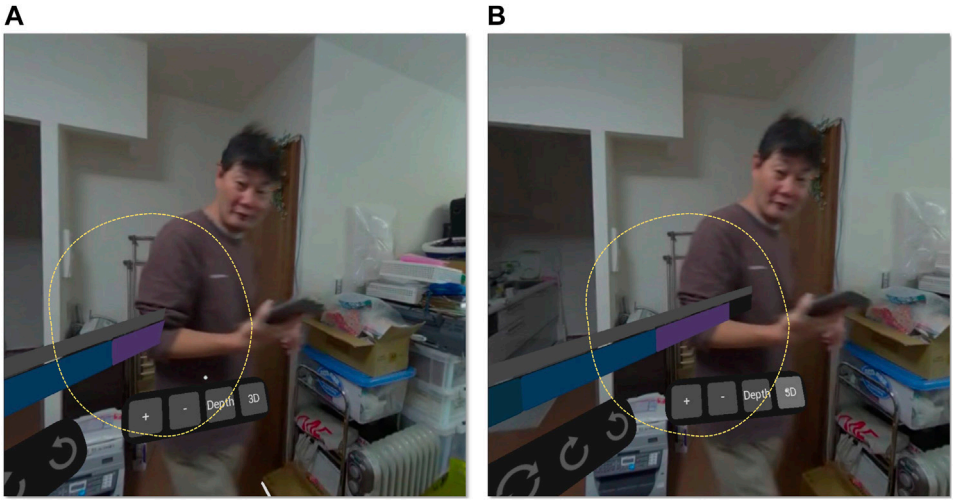

FIGURE 2 | (A) A 6DoF video is rendered in 3D, and interferes with the 3D UI elements as the draw order is specified by the depth and the UI elements disappear behind the video elements. **(B)** The same video rendered as monoscopic 360-degree video does not exhibit this issue. Using video material from https://www.kandaovr.com/gallery/available *via* http://pseudoscience.pictures/

(**Figure 2**). Instead, we render the interface at a fixed depth of 1.2 m. Though this still leaves the possibility of depth conflicts with other elements in the scene, these conflicts are in practice not very frequent and we did not experience them in our experiments. Other options to address these conflicts were proposed by Nguyen et al. (2018) by dynamically adjusting the depth of the UI or blurring the video with a halo around UI elements for stereoscopic videos. We also investigated other options for dynamically placing UI elements such as having UI elements following the user. For example, buttons and timelines could rotate so that they are always within the participants' reach. Another option is to use UI elements that disappear when not needed to allow the user to view more of the scene when they are not being used. However, we decided to have permanent UI elements in a fixed location with regard to the user to make it easier for them to find the elements and avoid any confounding factors from these dynamics.

3.3 Timeline

The timeline is one of the main visual metaphors in video editing. It is used to represent the temporal order of video clips in a video editing project. A playhead element is used to indicate the current play position within the video editing project and represents a frame within the currently selected video clip. We investigated two options for placing a timeline in the VR video editor. The first option is a cylindrical timeline wrapped around the user. The idea is that the user can interact with single video clips on the timeline while rotating themselves to select a video clip of interest. The 360 degrees relate to the full length of the edited video sequence. The advantage of this approach is that the video clips would always stay in the same position with regard to the user location in VR. However, the approach presented also a couple of non-obvious problems such as what to do when the length of the videos is longer than the circumference of the cylinder. In order to address this, the full video length would be mapped onto the 360 degrees available. However, if the overall video length changes, this mapping would need to be updated and could mean that video clips are presented in different locations. Eventually, we decided against this approach as it would confuse the user.

The option we finally used implements a planar timeline, located in 3D space. This is similar to what editors would expect from traditional 2D video editing. The planar timeline sits in front of the user and allows them to interact with single video clips. One challenge that we experienced with this option is the inaccuracy of pointing in VR. Two 6DoF controllers are used to control the UI, by casting a ray from the controller as a pointing device. This ray-casting approach suffers from the lever problem. When the user points at elements further away, even small movements in the user's hands can result in large movements in the resulting pointing location. In order to reduce this effect, we decided to move the timeline and keep the timeline playhead static. This means that the current video is always as close as possible to the user, reducing this levering effect.

In addition to the geometric representation of the timeline, there are different options for the complexity of the timeline itself. In traditional 2D video editing, we have the option for a single-track or multi-track editing interface. While a single-track interface only allows to edit videos sequentially, multi-track interfaces allow for more complex effects such as overlays, blending as well as picture-in-picture effects. In our prototypical implementation, we developed both options. However, we focused on the single-track approach as it has a couple of advantages for VR editing. Firstly, a single-track interface takes up less space in the scene. Thus, more of the content, and less of the interface will be visible to the user. Secondly, a multi-track interface, because it allows arbitrary movements for each video to any point within the track,

FIGURE 3 | Arranging video clips. **(A)** The user hovers over and selects the second clip in the timeline using the VR controller (highlighted in turquoise). **(B)** The user hovers over and selects the last clip (highlighted in turquoise). **(C)** The user presses the swap button. f) The two clips are now swapped. Video material: http://pseudoscience.pictures/

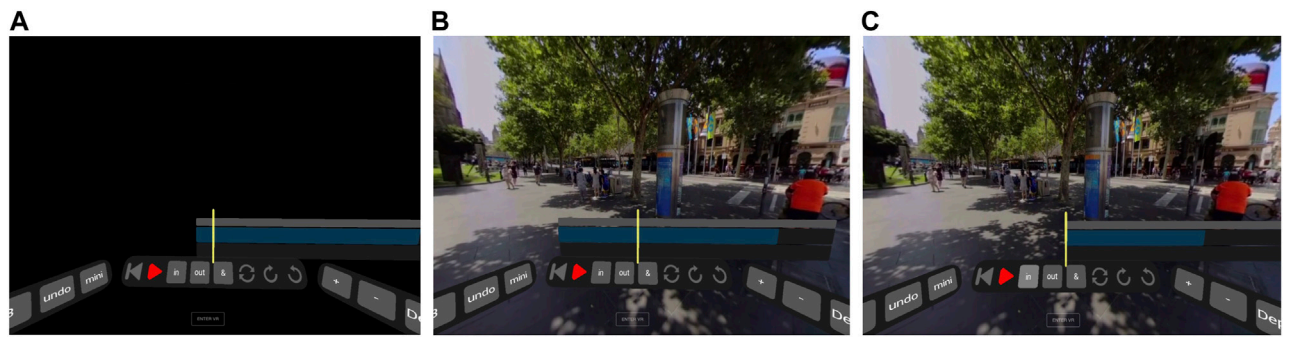

FIGURE 4 | Video clip cutting. **(A)** The user plays through the introduction section of the video, which includes a black screen. **(B)** The user wants to cut the video, so it starts at the current point (to remove the introduction section), the user presses the "in" button. **(C)** The video is cut down and now it plays without the introduction section. Using video material publicly available (http://pseudoscience.pictures/) processed with Stereo2Depth (Gladstone and Samartzidis (2019))

requires finer control. This could be mitigated with "snapping" tools, which intelligently snap videos to sensible edges, or with tools that allow for more precise movements by translating larger hand movements into smaller actual interface movements. However, these approaches complicate the problem from both user experience and implementation point of view.

3.4 Arranging and Cutting Video Clips

Arranging and cutting video clips are part of the main functions of video editing. Arranging video clips as part of a larger video editing project allows the user to change the order in which video clips are replayed. While traditional 2D video editing interfaces support drag-and-drop for arranging video clips, we decided to implement a more basic option to avoid problems with using the VR controllers for drag-and-drop operations. Our arrangement options are simplified to swapping the position of two selected clips. For this option, the user selects two clips on the timeline, then swaps the positions of both clips by pressing a button in the UI (**Figure 3**). This allows for any ordering of the given clips on the timeline.

The purpose of cutting video clips is to make them shorter to remove unneeded video material. In order to support the cutting of video clips, we implemented UI elements in our immersive video editor that allow the user to set the "in" and "out" points of a video to the current playhead position (**Figure 4**). The user can then use the timeline and the UI elements ("in" and "out", **Figure 4**) to play through the video to select a new starting point of the video and to select a new ending point of the video.

3.5 Rotation

Rotating operations in video editing are editing operations that are specific to 360-Degree videos. The viewer/consumer of a 360-Degree video has the full freedom to look in any direction during replay. However, editors often want to choose an initial orientation for storytelling purposes, for instance by putting the focus on important elements of the scene. Particularly, this is a problem when there are multiple clips as a result of a video editing project. In this case, editors might want to either re-orientate the videos at every cut or leave all the videos fixed in an absolute position for the entire sequence. Thus, we need to provide user interface elements that support changing the orientation of each video clip in immersive 360-Degree video editing.

Pavel et al. (2017) proposed different techniques for aligning multiple 360 video clips. They investigated these techniques and found that viewers spend 6–10% more time viewing labeled

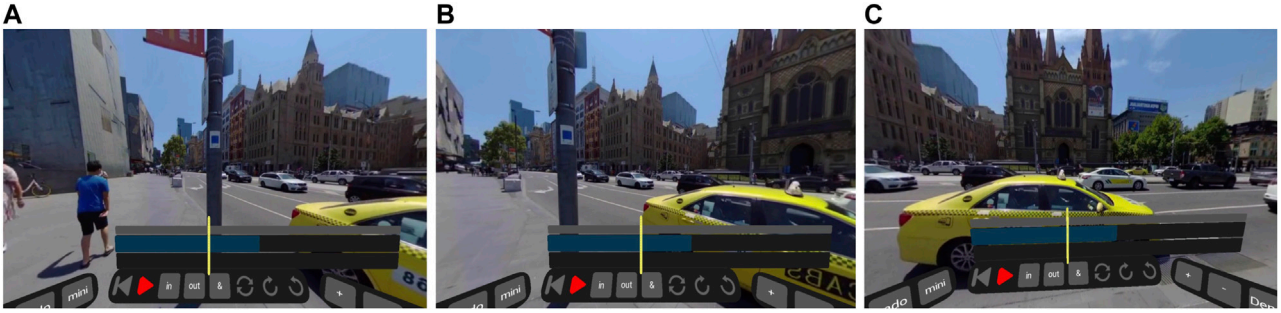

FIGURE 5 | Adjust the rotation of a VR video clip. **(A)** The user selects a clip to rotate. **(B,C)**: The user presses the "Rotate Clockwise" button several times to rotate the video so that the yellow cab is at the center of the frame. This position can either be absolute or can be reset at each cut (http://pseudoscience.pictures/).

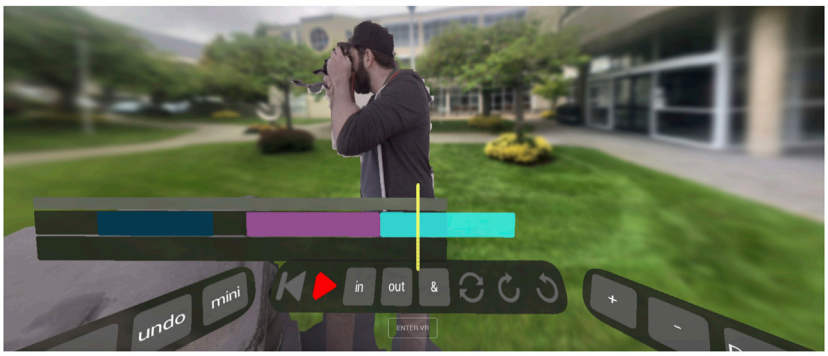

FIGURE 6 | Spatial Editing: Combining two 6DoF videos. One video shows a person with a camera, the other video clip displays a courtyard. The first video clip is combined with the second video clip using a two-track timeline. Please note that the artifacts visible around the person are due to depth map inaccuracies and not a problem specific to the immersive video editor.

FIGURE 7 | Temporal editing vs. spatial editing. The left figure shows temporal editing: Using the timeline to align two VR video clips embedded in a multilayer timeline. The right figure shows the manipulation interface for compositing of two VR video clips. The interface allows the editor to move one VR video clip in virtual space. Please note that the artifacts visible around the person are due to depth map inaccuracies and not a problem specific to the immersive video editor. In this example the video clip with the person is captured using a mobile phone, in such cases, the artifacts can become more noticeable.

important points in 360-Degree videos that reset the position on each cut to a predefined rotation compared to using traditional fixed-orientation cuts. Absolute positioning however has the advantage of providing the same viewing experience independent of the viewer and does not require a replay application that is aware of the different clips in the video. Given either paradigm, the editor must have an interface to rotate clips when editing a sequence of VR videos. We decided

to use a UI element for implementing the rotation interface. The UI element allows editors to rearrange the orientation of each video clip by pressing a button for rotating clockwise or counterclockwise (**Figure 5**). The advantage of using a button over using input from the VR controllers includes that it does not require fine movements of the 6DoF controllers.

3.6 Spatial Editing

VR and volumetric film-making may require the editor to control spatial aspects of the VR experience. For example, the editor may need to modify depths in the scene to highlight certain elements or to make others less noticeable. Allowing the editor to combine multiple volumetric videos and to replay them at the same time is useful for compositing. For example, this can be used to combine multiple 6DoF videos of the same scene but captured from different viewpoints or to composite multiple 6DoF videos from different scenes (e.g. compositing a person from one video into a video from a different environment, **Figure 6**). It is highly important to avoid depth conflicts in these cases. We implemented a basic operation for spatial editing that allows to spatially blend two video clips (**Figure 6**) as well as support these operations by a manipulation interface that allows changing the spatial arrangement of a VR video clip in the virtual space (**Figure 7**).

3.7 Transitions

Transitions are used in traditional 2D video editing, both to reduce the visual discontinuity between two shots, e.g., a cross-fade, or to emphasize them for effect, e.g., a zoom transition. 6DoF videos are typically filmed from a static point of view to reduce motion sickness, and the mean time between single cuts is typically longer [17s vs. 3–5 s for typical video (Pavel et al., 2017)]. Simple cuts can be abrupt and disorientating, as the viewer is abruptly transported to a new immersive scene. A smooth transition between the two videos may help to reduce this effect.

We added basic transitions to our VR video editor by applying several traditional 2D transitions such as blending of video frames and as well as blending between depth maps. In order to implement the transition techniques, we used GLSL shader programming to mix between two subsequent video clips. For this purpose, we use the time variable from the two video clips and pass them to the shader program. We then use this time variable to implement three types of transitions: 1) alpha blending, 2) depth map blending, and 3) alpha and depth map blending combined. For the alpha blending, we use a traditional transparency blending where we modify the transparency by calculating a weight for the alpha channel of each video frame based on the time variable. For the depth map blending, we use the weights in order to blend the depth maps only. And for the combined alpha and depth map blending, we use the computed weight to mix the depth maps and the alpha map at the same time.

4 USER STUDY

We developed a prototypical system for editing and playback of 6DoF videos (6DIVE). While there are implementations of 6DoF video viewers available[6] (Fleisher, 2020), there are, to the best of our knowledge, no applications for editing these videos with real-time feedback. With our system, we intend 1) to demonstrate the *feasibility* of immersive 6DoF editing, by demonstrating real-time performance and 2) to analyze the usability by non-expert users. Additionally, we are interested in whether an immersive VR editing experience is an improvement on 2D video editing tools, and if such improvements are task related. We used a similar study design like the one used by Nguyen et al. (2017).

4.1 Design

Our study was designed to measure the workload, usability, and preference differences between editing 6DoF content on a monocular, desktop-based interface and a 3D immersive HMD based interface. The study also serves as a chance to get qualitative feedback from casual and expert editors, on the immersive approach in general and our system specifically.

It is important to note that within our study, we focus on the directly comparable aspects between the desktop-based interface and the immersive interface. For this purpose, we decided to exclusively measure the editing interaction without taking the exporting step into account as this would create a disadvantage for the "Desktop" condition. While in the VR condition the user can directly view the results in an immersive environment, for the "Desktop" condition this would require additional switching between the desktop application and the VR headset and would create an unfair comparison. The additional overhead related to exporting the video from the desktop application and importing it into the VR viewer would unfairly favor the VR condition. This is, in particular, the case for shorter tasks as the ones used in our experiment. Within our study, we are most interested in exploring editing methods themselves in a desktop tool and a VR environment as Nguyen et al. (2017) already investigated the benefits of immersive editing of 360 videos compared to a standard workflow used by editors.

While within Nguyen et al.'s Vremiere study participants were asked to compare their experience to their own workflows editing VR content, we employ a direct comparison between implementations of a desktop and VR editing interface. This comparison places some limitations on the design of our editing widgets. In order to maintain a direct comparison we use the lowest common denominator of input, pointing, and selecting, either with a mouse on the desktop or the 6DoF controllers for the HMD. While custom widgets for VR and desktop would be useful, they would impose an additional learning burden on the participants. This direct comparison design allows us to quantify the workload, efficiency, and preferences for three specific tasks in editing. We use a within-subject design and limited the editing operations to arranging, cutting, and rotating to reduce the complexity.

4.2 Apparatus

We implemented the application as described in **Section 3** for our experiment. The VR condition is implemented using WebXR and

[6]http://pseudoscience.pictures

was used on an Oculus Quest with two 6DoF controllers. The desktop condition was used on a 16" Laptop with both a trackpad and mouse available to the user (MacBook Pro, with 3,072 × 1920 resolution). The desktop editing application is implemented as a browser-based interface running in Google Chrome. Users were sitting for the desktop application, and for the VR application, they could either stand or sit in a swiveling office chair.

4.3 Procedure

In the experiment, the participants started with filling out a demographic questionnaire along with a before SSQ questionnaire. We then introduced the system on either the desktop interface or the VR interface (randomized order) and explain how the basic navigation controls, the undo/redo buttons, and the controls for adjusting the scale of the timeline work. We repeated this process for the other mode of the interface. During this process, we ensured that the headset fits well and that the image is clear and sharp. We defined three tasks within our experiments including arranging video clips, cutting video clips, as well as adjusting the rotation of VR video clips. For each task, we measured workload and completion times.

4.3.1 Task 1—Arranging

The arranging task begins with three videos, located on a timeline. We explained that the task is to re-order these videos to a specification and that two videos can be swapped by selecting both videos and pressing the swap button. We repeated the task using a randomized order for VR and desktop. Both specifications require at least two video swaps to achieve the desired result. After the task was completed for each interface, a NASA-TLX (task load index) questionnaire was filled in by the participant (Hart and Staveland, 1988).

4.3.2 Task 2—Cutting

The cutting task begins with one long video presented in the timeline. The participants were told that the in and out buttons can be used to set the start or endpoints of the video to the current playhead position. Participants were given a specification to cut a section from the beginning of the video and asked to complete the task on both the VR interface and the desktop interface (in randomized order). We measured task load after each task is finished.

4.3.3 Task 3—Rotation

The rotation task begins with a timeline including four videos. Participants were given the task, for two of the videos (in randomized order) to decide what they perceive as the main subject of the scene, and rotate that to the center of the user interface widgets. Participants were told that they can rotate the current video using the "Clockwise" and "Counter-Clockwise" buttons or can rotate multiple videos at once by selecting them. The task was again repeated for both the VR and desktop interfaces, though with different videos for each. Again, we measured task load after each task is finished.

4.3.4 User Experience Questionnaire

After the three tasks were completed for both interfaces, we presented the User Experience Questionnaire. This is a psychometric evaluation of the user experience of an application. A questionnaire was filled in for both, the VR and the desktop interfaces.

Afterward, the user filled out another SSQ, to measure any change in simulator sickness after completing the tasks. We then provided a questionnaire that asks, on a seven-point Likert scale, for each task whether the participants preferred the VR or desktop interface. We also asked about the comfort of the application, and an open answer question about any functionality the participants missed while completing the tasks.

4.4 Ethical Approval

This study has been approved by the University of Otago's Human Ethics Committee (D20/270). In particular, in terms of risks to the participants, we had to consider how to safely perform a study using an HMD under COVID-19 Alert Level 2 (New Zealand) restrictions. We included a COVID-19 screening form, along with social distancing, hand sanitizer, and anti-bacterial wipes, and a disposable mask system for the HMD[7]. These additional steps increased the time taken to run the user study but was a reasonable solution given the circumstances.

4.5 Participants

Due to increased time effort related to the COVID-19 restrictions, we had to restrict the numbers of participants. We recruited eight participants, (aged 21-65, median 25, 2F, 6M) mostly from staff and students at the university. We especially looked for participants with experience in video editing. All participants had experience in video editing using a variety of software (Final Cut Pro = 2, iMovie = 2, Premiere Pro = 3, Movie Maker = 2, Avid = 1, DaVinci Resolve = 1, Other = 2). Four of the participants reported having experience with VR video.

4.6 Hypotheses

For this study, we were mainly interested in the aspects of how much the VR and the desktop interfaces differ from each other for different tasks, regarding workload, task completion time, and user preference. To investigate these aspects, we postulated three hypotheses.

- H1: The workload will be different between the VR and desktop interfaces, particularly for the rotation task.
- H2: The completion time of the tasks in the VR interface will not be significantly slower, nor faster than the desktop version.
- H3: Users will prefer the VR interface.

H2 is based on the assumption that we assumed the VR interface would be a more suitable interface for completing editing tasks, particularly rotation. Due to the short nature of the tasks, and the participants' unfamiliarity with VR would not necessarily see this reflected in absolute efficiency metrics like completion time.

4.7 Results

All statistical analysis was performed using a significance level $\alpha = 0.05$.

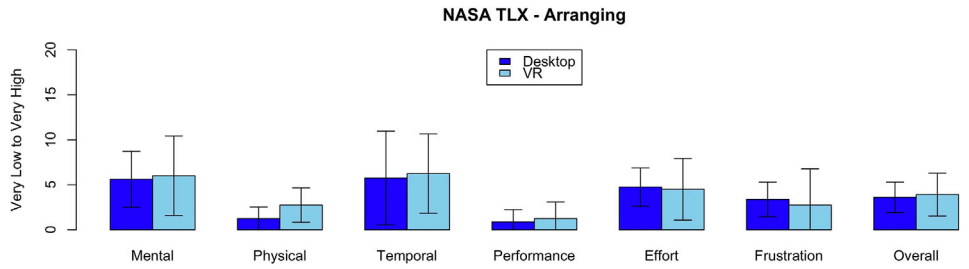

FIGURE 8 | Task 1: TLX values for the arranging task comparing the Desktop and the VR condition.

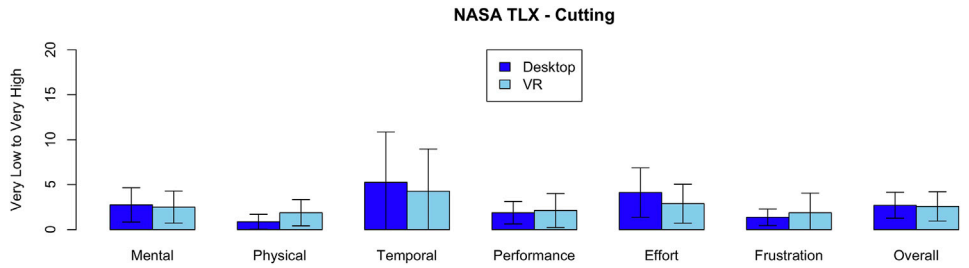

FIGURE 9 | Task 2: TLX values for the cutting task comparing the Desktop and the VR condition.

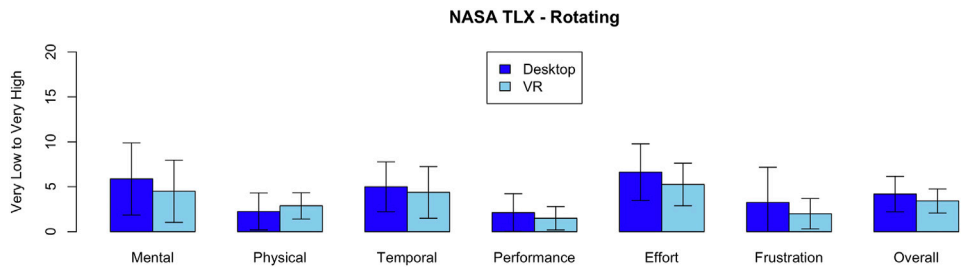

FIGURE 10 | Task 3: TLX values for the rotation task comparing the Desktop and the VR condition.

4.7.1 TLX

We used the NASA-TLX (Hart and Staveland, 1988) to measure workload. Participants were asked six questions (on a 20 point scale). The questions included "How mentally demanding was the task?", "How physically demanding was the task?", "How hurried or rushed was the pace of the task", "How successful were you in accomplishing what you were asked to do?", "How hard did you have to work to accomplish your level of performance", "How insecure, discouraged, irritated, stressed, and annoyed were you?"

The average scores for all participants are given in **Figures 8–10**. NASA-TLX is sometimes used with a pair-wise weighting questionnaire, but this increases the time taken to complete the questionnaire and so we use the Raw-TLX scores, which are also commonly used (Hart, 2006).

We analyzed the data, first by testing for normal distribution using the Shapiro-Wilk Normality Test. For normally distributed data (Shapiro-Wilk Normality Test p-value > 0.05), we used a paired student's t-test, otherwise, we used the Wilcoxon's signed-rank test for paired data.

While it seems that the overall task load varies between tasks and seems to be lower for the rotation task in the VR condition, we could not find any statistically significant differences for the overall scores for any of the tasks (tasks 1 ("Arranging", normally distributed): Desktop mean = 3.60, std = 1.70, VR mean = 3.92 std = 2.38, p = 0.6225, Cohen's d: 0.182 (small effect size), task2 ("Cutting", normally distributed): Desktop: mean = 2.71, std = 1.44, VR: mean = 2.58, std = 1.63, p = 0.809, Cohen's d = 0.0887 (small), task 3 ("Rotating", not normally distributed): Desktop: mean = 4.19 std = 1.97, VR: mean = 3.42 std = 1.34, p = 0.16091, Wilcoxon effect size r = 0.520 (large)) or any pairs of the TLX scores (omitted due to space reasons). This could be due to the small number of participants or the similarity of the two interfaces.

TABLE 1 | Results of the User Experience Questionnaire for Desktop and VR.

Attribute	Desktop						VR Interface					
	Mean	STD	N	Confidence	CI		Mean	STD	N	Confidence	CI	
Attractiveness	0.48	0.52	8	0.36	0.12	0.84	1.31	0.55	8	0.38	0.93	1.69
Perspicuity	1.66	0.55	8	0.38	1.28	2.04	1.66	0.79	8	0.55	1.11	2.20
Efficiency	0.63	0.78	8	0.54	0.09	1.16	1.22	0.80	8	0.55	0.67	1.77
Dependability	0.91	0.98	8	0.68	0.23	1.59	0.81	0.80	8	0.55	0.26	1.37
Stimulation	0.53	0.49	8	0.34	0.19	0.87	1.41	0.42	8	0.29	1.11	1.70
Novelty	−0.28	1.06	8	0.74	−1.02	0.46	1.75	0.67	8	0.46	1.29	2.21

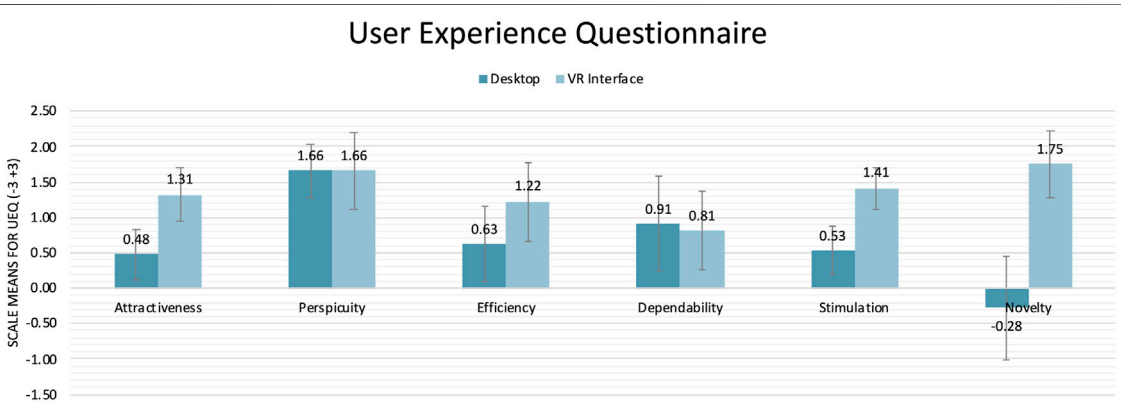

FIGURE 11 | Responses to the User Experience Questionnaire (Hart (2006)). Differences in attractiveness (p-value = 0.0072), stimulation (p-value = 0.0019) and novelty (p-value = 0.0007) are found to be significant different using a paired t-test.

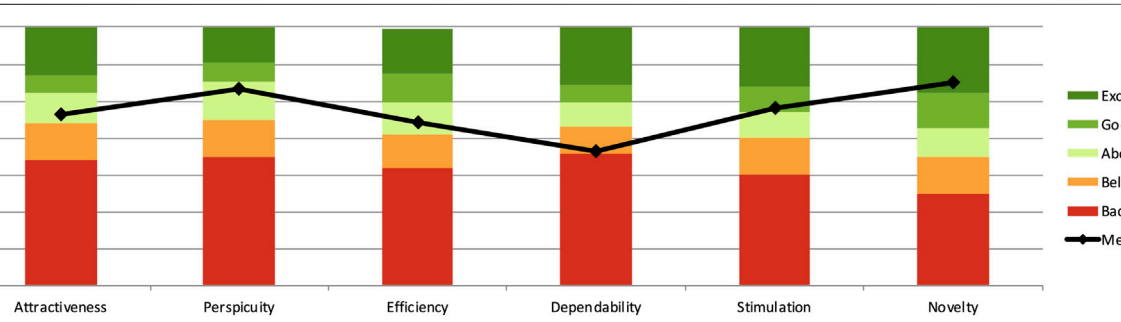

FIGURE 12 | User Experience Questionnaire results of the VR video editor set in relation to existing values from a benchmark data set [Containing feedback from 20190 persons from 452 studies concerning different products (business software, web pages, web shops, social networks), (Hart (2006))].

4.7.2 User Experience Questionnaire

We used the User Experience Questionnaire (UEQ) to measure usability aspects such as efficiency, perspicuity, and dependability, as well as user experience aspects, originality, and stimulation (Schrepp et al., 2014). The scores are based on 26 questions which the participants answer on a scale between −3 and 3. The order of the questionnaires for the VR or the desktop condition was randomized. We found that the users found the VR interface to be significantly more attractive (Desktop: mean = 0.48, VR: mean = 1.31, $p = 0.0072$), stimulating (Desktop: mean = 0.53, VR: mean = 1.31, p-value = 0.0019) and novel (Desktop: mean = −0.28, VR: mean = 1.75, $p = 0.0007$). We could not find any significant differences for perspicuity (Desktop mean = 1.66, VR mean = 1.66, $p = 1.0$), efficiency (Desktop: mean = 0.633, VR mean = 1.22, $p = 0.09$) and dependability (Desktop mean = 0.91, VR mean = 0.81, $p = 0.837$, compare with **Table 1** and **Figure 11**).

We also evaluated the interface by comparing it with benchmark data for the UEQ (**Figure 12**, Schrepp et al., 2017). The VR interface scores "Good" for attractiveness, "Good" for perspicuity, "Above Average" for efficiency, "Below Average" for dependability, "Good" for stimulation, and "Excellent" for novelty. Dependability is the only metric where the VR interface scores poorly, and this is perhaps a reflection of the prototypical nature of the system. We

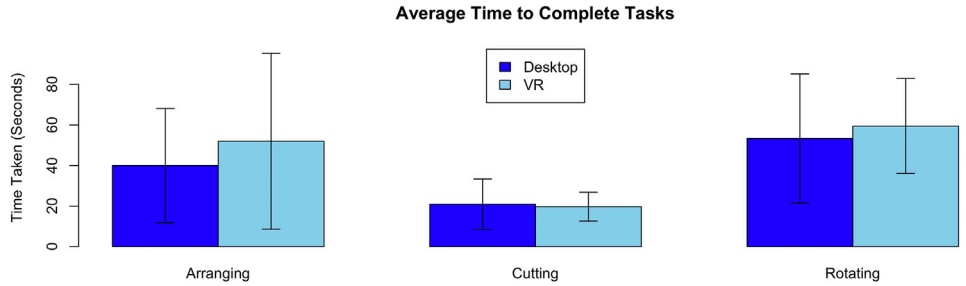

FIGURE 13 | The task completion time for each of the three tasks. With $n = 8$ we found no significant differences in times between the VR and desktop interfaces for the three tasks.

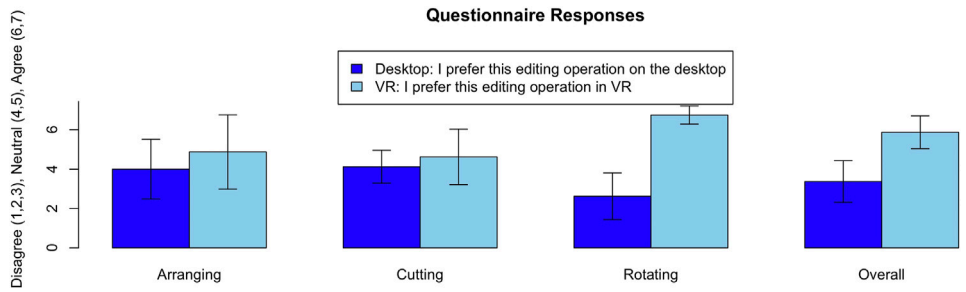

FIGURE 14 | Mean responses to the final questionnaire for each of the three tasks.

must also note that the dataset for these benchmarks was primarily business applications so it is not clear how user expectations would change for a video-editing application.

4.7.3 Comfort and Cybersickness

No participant reported motion sickness during the study, there were no dropouts, and we found no significant differences in scores between the before and after simulator sickness questionnaire (mean SSQ score before 5.61, after 5.1425, paired t-test p-value = 0.89). We also asked the participants how strongly they agreed or disagreed with the two statements "Editing in VR was comfortable" and "Editing on the Desktop was comfortable". For the Desktop the mean was 4.88 (std = 1.73) and in VR the mean = 5.12 (std = 0.99) on a 7-point Likert scale. We found no significant difference (p-value = 0.756, Cohen's d = 0.114 (small effect size)) between the two.

Note that the participants were not using the interfaces for as long as average video editing usage times. Thus comfort for long-term usage is something to explore in future work. In addition, it is important to note that the SSQ has been administered after the UEQ, thus SSQ scores might be influenced by this order and potential short-term discomfort might have already subsided after filling the UEQ.

4.7.4 Task Completion Time

We measured the task completion time for each task (**Figure 13**). The time is measured from the moment when the participant said they were ready to begin until they were satisfied they had completed the task. Participants were asked to judge themselves whether they had completed the tasks to the specification.

A Shapiro-Wilks test indicated that the timing data for the arranging task (Desktop mean = 40.0 s, std = 28.1, VR mean = 51.9 s, std = 43.3) was normally distributed (Shapiro-Wilks, p-value = 0.003964). Thus, we used the non-parametric Wilcoxon signed-rank test to test for differences. We found the task completion time to not be significantly different (p-value = 0.64, Wilcoxon effect size r = 0.198 (small)). The time for the cutting task (Desktop mean = 20.9 s, std = 12.4, VR mean = 19.7 s, std = 7.11) was found to be normally distributed (Shapiro-Wilks p-value 0.878). Thus, we used a paired student's t-test which indicated no significant difference between the VR and desktop condition (p-value = 0.8369, Cohen's d = 0.0755 (small effect size)). The timing data for the rotation task (Desktop mean = 53.3 s, std = 31.8, VR mean = 59.5 s, std = 23.4) was found to be normally distributed (Shapiro-Wilks p-value = 0.65). Thus, we used a paired T-Test which suggested no statistically significant difference in the time to complete the task between the VR and Desktop conditions (p-value = 0.555, Cohen's d = 0.2192 (small)).

4.7.5 Questionnaire

A final questionnaire was used to evaluate the users' preferences between the two conditions, for each task and overall (**Figures 14**, **15**). We used a paired 7-Point Likert scale[7] questionnaire design.

[7]Disagree (1,2,3), Neutral (4,5), Agree (6,7)

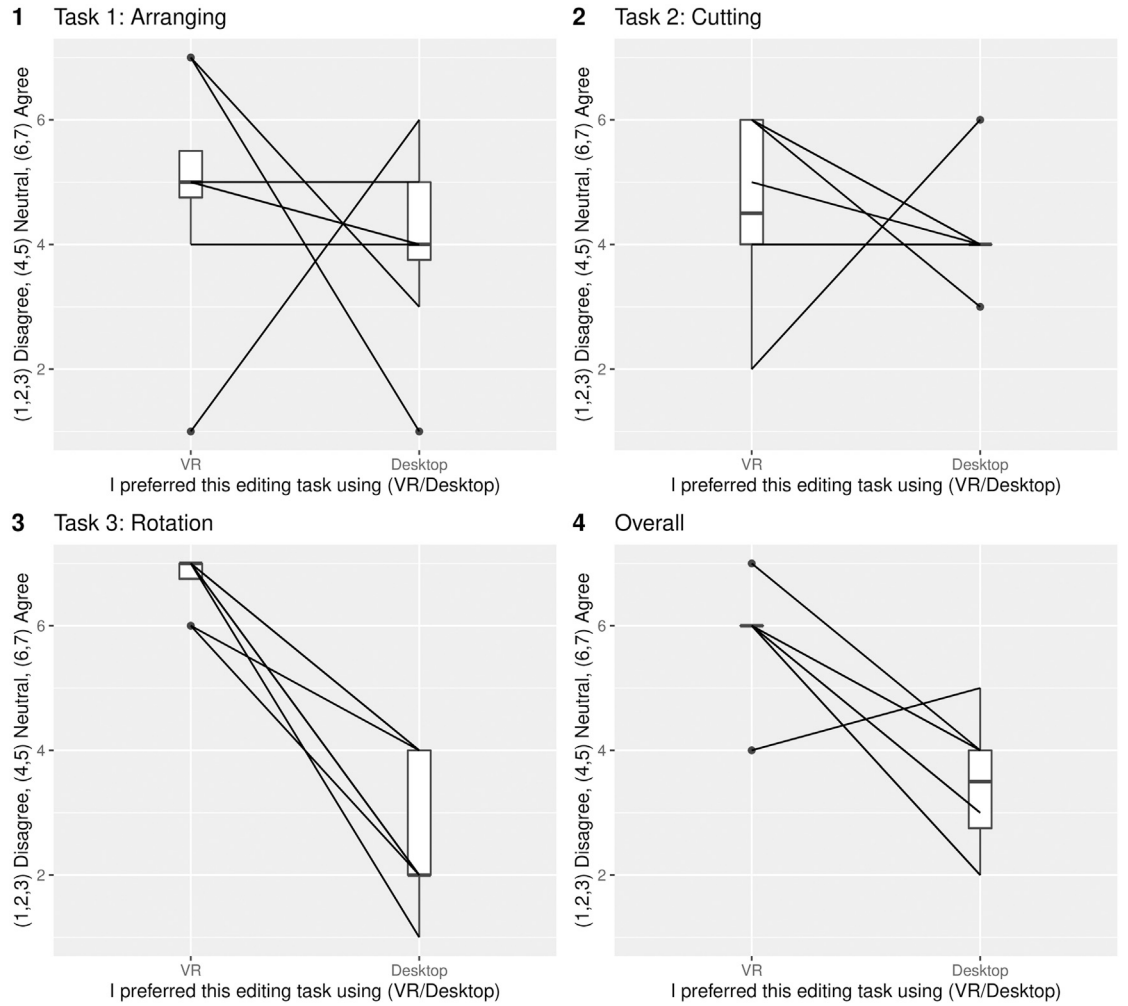

FIGURE 15 | Paired responses to the final questionnaire for the VR and Desktop conditions. Most participants agree with their responses. However, there are a few exceptions such as in the arranging task. Here, one participant strongly preferred VR and another participant preferred the Desktop option.

Even we did not measure significant differences in workload between the immersive and non-immersive interfaces, our interface design should be user-centered. Thus, we were interested in the preferences of the video editors for the different tasks. Again, we used the Shapiro-Wilks normality test and found all the responses were normally distributed for all but the overall questionnaire (p-value = 0.043). Thus, we used a Wilcoxon signed-rank test with continuity correction for the overall score and a paired student's t-test for the task-specific preferences. We found higher scores for the VR interface on all questions. However, we only found a significant difference for the rotating task and overall question.

For the arranging task, the Likert scores showed a slight preference for the VR condition (Desktop mean = 4.00, std = 1.51, VR mean = 4.88 std = 1.89) However, we did not measure a significant difference (paired student's t-test p-value = 0.46, Cohen's d: 0.271 (small effect size)). For the cutting task the Likert scores showed again a slight preference for the VR interface (Desktop mean = 4.12, std = 0.835, VR mean = 4.62, std = 1.41, p-value = 0.52, Cohen's d = 0.234 (small effect size). The rotation showed the biggest difference [Desktop mean = 2.62, std = 1.19, VR mean = 6.75, std = 0.463, p-value = 5.715e-05, Cohen's d = 3.041 (indicating a large effect size)]. We suspect that this is the case for two main reasons. Rotating using the desktop interface requires a conscious effort to look around, while in the immersive interface the user can intuitively look around. Secondly, the rotation task required participants to make editing decisions based on the specific details of the scene and therefore this task required more exploration.

For the overall questionnaire, participants answered Desktop: mean 3.88, std = 1.06, VR: mean 5.88, std = 0.835. We found these responses to not be normally distributed (Shapiro-Wilks p-value = 0.0425) and so used the Wilcoxon signed-rank test with continuity correction which found the differences to be significant with p-value = 0.0199, Wilcoxon effect size r = 0.848 (indicating a large effect size). The paired questionnaire responses are displayed in **Figure 15**.

4.7.6 Open Questions

The responses to the question "What features that were not included would be most important for a VR editing application" included requests for more fine-grained navigation and cutting controls (3/8 participants), more drag and drop interactions (2/8), easier scrubbing[8] (3/8) as well as requests for the user interface to follow the user (2/8). One participant also requested titling tools and speed manipulation tools (e.g. for slow-motion video).

5 DISCUSSION

When we examine our hypotheses, with the analyzed data it becomes clear that we have to reject H1: "The workload will be different between VR and Desktop interfaces". We could not find significant differences in any of the TLX measurements. While we would have expected to see a difference, particularly in the rotation task, perhaps our widgets are not optimized as much as possible for the immersive environment, given they are also designed to work within the desktop interface. In addition, it could be that there are differences, but they may have been too small to measure with eight participants.

We found an indication for H2: "The VR interface will not be significantly slower, nor faster than the Desktop version" as we could not find a significant difference in completion time between the VR and Desktop interfaces for any of the three tasks. It is important to mention here that the measured timing does not include any preview times for the desktop condition. So, it could be that once editors are exporting their video to VR from the desktop application, they would use more time by going forwards and backward. However, the overall editing time seems to be similar.

We also found evidence for H3: "Users will prefer the VR interface". The responses to the user experience questionnaire showed higher attractiveness, stimulation, and novelty ratings for the VR interface. Our preference questionnaire also showed a higher preference for the VR interface for the rotation task, and overall. It is also important to note that all participants were able to complete all the tasks, with only minor clarifications, despite participants mostly being new to VR.

Given that both the workload and the task completion time are similar for both the desktop and VR applications, we believe this is a good indication for the potential of immersive editing. Editing 6DoF video in VR has the advantage that the editor is able to directly perceive depth and motion aspects of the video that are relevant to storytelling. For example, editors will be able to tell if an object is too close to the viewer for comfort, or that the motion of the camera is uncomfortable in VR. Editing 6DoF video on the desktop has the perceived advantages of more precise input devices and familiarity, but we have shown (for a small group of users) that editing using a VR interface can be as effective as a desktop interface. We believe with more familiarity with VR and refinement of interaction techniques this will become more evident.

[8]Quickly navigating through videos to see the content.

In order to create immersive VR storytelling experiences that provide the viewer with a sense of presence, it is important to provide tools that support editors of such content. These tools must be readily available, efficient, and comfortable. We believe our work, is a step towards this.

6 CONCLUSION AND FUTURE WORK

In this paper, we presented 6DIVE, a 6 degrees-of-freedom immersive video editor. 6DIVE allows users to edit 6DoF videos directly in an immersive VR environment. We implemented an interface for arranging, cutting, rotating, spatial editing, and transitioning. We analyzed the benefits of this approach within a user study and collected data about workload, user experience, the effects on simulator sickness, performance measures as well as user preferences and compare this with a desktop-based implementation of a VR video editor. The results were promising but also showed that there are many directions for future work in this area. Immersive editing of 6DoF content is under-explored and there are still many possible areas of interest. Our initial study demonstrated the potential for immersive editing and indicates that immersive editing has the potential to outperform 6DoF video editing using a traditional desktop interface. While our work provides an initial exploration and indications for future work, a more comprehensive evaluation will be required.

In addition to these initial results, our prototype and feedback from participants showed there are many possible ways how to interact in VR that are not available in traditional desktop interfaces. Additional interaction methods that arise from immersive video editing include gaze-based interaction where the user could use gaze to set the orientation of videos. We could browse content spatially by laying it out in a physical space, such as using the user's hand to display a tool palette.

In this work, we have mainly focused on the temporal aspects of editing 6DoF video, but volumetric filmmaking also allows the editor to control spatial aspects of the experience. While we implemented some aspects of spatial editing, there are many aspects to be explored in this area. We also have done some initial testing of transitions. However, the aspects of how transitions of 6DoF video work and influence users are underexplored. There are recent approaches to synthesize high-quality 6DoF video in real-time from 360 ODS footage on high-end but still consumer-grade hardware (Attal et al., 2020). Integrating an approach like this as a viewer component in the 6DIVE system would allow higher fidelity video and would improve on some of the artifacts present in the displaced geometry approach. In addition, many other aspects need further exploration, such as the placement of UI elements as well as the representation of the timeline.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by University of Otago's Human Ethics Committee (D20/270). The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

This work is based on the work of RG as part of his honours thesis and his research project at the University of Otago. All implementations were done by RG as well as the study has been conducted by RG. The work has been supervised by TL and SZ. SZ was the primary supervisor. Both provided guidance on system design and implementation and edited the final version of this paper. SZ also provided guidance on study design, ethics application, and revisions.

FUNDING

We gratefully acknowledge the support of the New Zealand Marsden Council through Grant UOO1724.

REFERENCES

Anderson, R., Gallup, D., Barron, J. T., Kontkanen, J., Snavely, N., Hernández, C., et al. (2016). Jump: Virtual Reality Video. *ACM Trans. Graph.* 35, 1–13. doi:10.1145/2980179.2980257

Attal, B., Ling, S., Gokaslan, A., Richardt, C., and Tompkin, J. (2020). MatryODShka: Real-Time 6dof Video View Synthesis Using Multi-Sphere Images. Proceedings of the 16th European Conference on Computer Vision (ECCV), 24-08-2020 (Springer, 441–459. doi:10.1007/978-3-030-58452-8_26

Baker, L., Mills, S., Zollmann, S., and Ventura, J. (2020). "Casualstereo: Casual Capture of Stereo Panoramas with Spherical Structure-From-Motion," in 2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), 782–790. doi:10.1109/VR46266.2020.00102

Brown, M., and Lowe, D. G. (2007). Automatic Panoramic Image Stitching Using Invariant Features. *Int. J. Comput. Vis.* 74, 59–73. doi:10.1007/s11263-006-0002-3

Broxton, M., Busch, J., Dourgarian, J., DuVall, M., Erickson, D., Evangelakos, D., et al. (2020). DeepView Immersive Light Field Video. *ACM Trans. Graphics (Proc. SIGGRAPH)* 39, 1–86. doi:10.1145/3388536.3407878

Elmezeny, A., Edenhofer, N., and Wimmer, J. (2018). Immersive Storytelling in 360-degree Videos: An Analysis of Interplay between Narrative and Technical Immersion. *Jvwr* 11, 1–13. doi:10.4101/jvwr.v11i1.7298

Fleisher, O. (2020). Three.sixdof. [Dataset]SixDOFAvailable at: https://github.com/juniorxsound/THREE (Accessed May 5, 2021).

Gladstone, J., and Samartzidis, T. (2019). Pseudoscience Stereo2depth. [Dataset] Available at: https://github.com/n1ckfg/pseudoscience_stereo2depth (Accessed May 5, 2021).

Grubert, J., Witzani, L., Ofek, E., Pahud, M., Kranz, M., and Kristensson, P. O. (2018). "Text Entry in Immersive Head-Mounted Display-Based Virtual Reality Using Standard Keyboards," in 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), 159–166. doi:10.1109/VR.2018.8446059

Hart, S. G. (2006). Nasa-task Load index (Nasa-tlx); 20 Years Later. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.* 50, 904–908. doi:10.1177/154193120605000909

Hart, S. G., and Staveland, L. E. (1988). "Development of Nasa-Tlx (Task Load index): Results of Empirical and Theoretical Research," in *Human Mental Workload (North-Holland), Advances in Psychology*. Editors PA Hancock and N Meshkati (Elsevier), 52, 139–183. doi:10.1016/S0166-4115(08)62386-9

Ishiguro, H., Yamamoto, M., and Tsuji, S. (1992). Omni-directional Stereo. *IEEE Trans. Pattern Anal. Machine Intell.* 14, 257–262. doi:10.1109/34.121792

Langlotz, T., Zingerle, M., Grasset, R., Kaufmann, H., and Reitmayr, G. (2012). "AR Record&Replay," in Proceedings of the 24th Australian Computer-Human Interaction Conference (New York, NY, USA: ACM), 318–326. OzCHI '12. doi:10.1145/2414536.2414588

Maimone, A., and Fuchs, H. (2011). "Encumbrance-free Telepresence System with Real-Time 3D Capture and Display Using Commodity Depth Cameras," in 2011 10th IEEE International Symposium on Mixed and Augmented Reality, 137–146. doi:10.1109/ISMAR.2011.6092379

Nebeling, M., and Speicher, M. (2018). "The Trouble with Augmented Reality/virtual Reality Authoring Tools," in 2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), 333–337. doi:10.1109/ISMAR-Adjunct.2018.00098

Nguyen, C., DiVerdi, S., Hertzmann, A., and Liu, F. (2018). "Depth Conflict Reduction for Stereo Vr Video Interfaces," in Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (New York, NY, USA: Association for Computing Machinery), 1–9. CHI '18. doi:10.1145/3173574.3173638

Nguyen, C., DiVerdi, S., Hertzmann, A., and Liu, F. (2017). "Vremiere," in Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (New York, NY, USA: Association for Computing Machinery), 5428–5438. CHI '17. doi:10.1145/3025453.3025675

Pavel, A., Hartmann, B., and Agrawala, M. (2017). "Shot Orientation Controls for Interactive Cinematography with 360 Video," in Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology, 289–297. doi:10.1145/3126594.3126636

Radianti, J., Majchrzak, T. A., Fromm, J., and Wohlgenannt, I. (2020). A Systematic Review of Immersive Virtual Reality Applications for Higher Education: Design Elements, Lessons Learned, and Research Agenda. *Comput. Edu.* 147, 103778–103829. doi:10.1016/j.compedu.2019.103778

Regenbrecht, H., Park, J.-W., Ott, C., Mills, S., Cook, M., and Langlotz, T. (2019). Preaching Voxels: An Alternative Approach to Mixed Reality. *Front. ICT* 6, 1–7. doi:10.3389/fict.2019.00007

Rematas, K., Kemelmacher-Shlizerman, I., Curless, B., and Seitz, S. (2018). "Soccer on Your Tabletop," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4738–4747. doi:10.1109/cvpr.2018.00498

Reyna, J. (2018). "The Potential of 360-degree Videos for Teaching, Learning and Research," in 12th International Technology, Education and Development Conference (INTED) (IATED-INT ASSOC TECHNOLOGY EDUCATION & DEVELOPMENT).

Richardt, C., Hedman, P., Overbeck, R. S., Cabral, B., Konrad, R., and Sullivan, S. (2019). "Capture4VR," in *ACM SIGGRAPH 2019 Courses* (New York, NY, USA: Association for Computing Machinery), 1–319. SIGGRAPH '19. doi:10.1145/3305366.3328028

Richardt, C. (2020). *Omnidirectional Stereo*. Cham: Springer International Publishing, 1–4. doi:10.1007/978-3-030-03243-2_808-1

Richardt, C., Pritch, Y., Zimmer, H., and Sorkine-Hornung, A. (2013). "Megastereo: Constructing High-Resolution Stereo Panoramas," in 2013 IEEE Conference on Computer Vision and Pattern Recognition, 1256–1263. doi:10.1109/CVPR.2013.166

Schrepp, M., Hinderks, A., and Thomaschewski, J. (2014). "Applying the User Experience Questionnaire (Ueq) in Different Evaluation Scenarios," in Conference: International Conference of Design, User Experience, and Usability, 383–392. doi:10.1007/978-3-319-07668-3_37

Schrepp, M., Hinderks, A., and Thomaschewski, J. (2017). Construction of a Benchmark for the User Experience Questionnaire (Ueq). *Ijimai* 4, 40–44. doi:10.9781/ijimai.2017.445

Schroers, C., Bazin, J.-C., and Sorkine-Hornung, A. (2018). An Omnistereoscopic Video Pipeline for Capture and Display of Real-World Vr. *ACM Trans. Graph.* 37, 1–13. doi:10.1145/3225150

Schwarz, S., Hannuksela, M. M., Fakour-Sevom, V., and Sheikhi-Pour, N. (2018). "2d Video Coding of Volumetric Video Data," in 2018 Picture Coding Symposium (PCS), 61–65. doi:10.1109/PCS.2018.8456265

Serrano, A., Kim, I., Chen, Z., DiVerdi, S., Gutierrez, D., Hertzmann, A., et al. (2019). Motion Parallax for 360° RGBD Video. *IEEE Trans. Vis. Comput. Graphics* 25, 1817–1827. doi:10.1109/TVCG.2019.2898757

Sheikh, A., Brown, A., Evans, M., and Watson, Z. (2016). Directing Attention in 360-degree Video. IBC 2016 Conference, 9–29. doi:10.1049/ibc.2016.0029

Szeliski, R. (2007). Image Alignment and Stitching: A Tutorial. *FNT Comput. Graphics Vis.* 2, 1–104. doi:10.1561/0600000009

Tan, J., Cheung, G., and Ma, R. (2018). 360-degree Virtual-Reality Cameras for the Masses. *IEEE MultiMedia* 25, 87–94. doi:10.1109/MMUL.2018.011921238

Vert, S., and Andone, D. (2019). Virtual Reality Authoring Tools for Educators. *ITM Web Conf.* 29, 03008–3017. doi:10.1051/itmconf/20192903008

Zollmann, S., Dickson, A., and Ventura, J. (2020). "CasualVRVideos: Vr Videos from Casual Stationary Videos," in 26th ACM Symposium on Virtual Reality Software and Technology (New York, NY, USA: Association for Computing Machinery), 1–3. VRST '20. doi:10.1145/3385956.3422119

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.