10.2352/EI.2022.34.9.IQSP-394
© 2022, Society for Imaging Science and Technology

Designing a User-Centric Framework for Perceptually-Efficient Streaming of 360-degree Edited Videos

***Lucas S. Althoff*[1]*, Henrique D. Garcia*[2]*, Dario D. R. Morais*[2]*, Sana Alamgeer*[2]*, Myllena A. Prado*[2]*, Gabriel C. Araujo*[2]*, Ravi Prakash*[3]*, Marcelo M. Carvalho*[2]*, Mylène C. Q. Farias*[2]**

[1] ***Department of Computer Science, University of Brasilia, Brazil.***

[2] ***Department of Electrical Engineering, University of Brasilia, Brazil.***

[3] ***Department of Computer Science, University of Texas at Dallas, USA.***

Abstract

This paper presents a set of developments for the design of a user-centric framework for perceptually efficient streaming of 360-degree edited videos. First, we introduce a software for annotation of regions of interest (RoI) in 360-degree videos with the goal of creating datasets for the training of machine learning-based visual attention computational models and for performance evaluation studies. Based on this software, we designed a subjective experiment to evaluate its usability and to create a dataset of annotated 360-degree videos. Then, to showcase an application of this dataset, we present a preliminary comparative analysis between saliency maps generated by a well-known visual attention computational model and RoI maps created from the dataset. Finally, we describe our efforts to investigate 360-degree video editing techniques that can improve the user's quality of experience (QoE). In particular, we show some preliminary results on a subjective experiment designed to evaluate perceived QoE, expressed in terms of "comfort" and "presence," for 360-degree videos that were statically edited (that is, off-line editing). The particular editing technique focuses on RoI alignment in the same scene (i.e., intra-scene alignment of distinct RoIs) for purposes of reducing the users' head movements. We look at both instantaneous (i.e., "snap-change") and gradual rotation editing, and we present the mean opinion scores (MOS) for the different cases, in different video categories.

1. Introduction

Recent advances in digital video coding and the proliferation of sensors embedded in devices such as cell phones and head-mounted displays (HMDs) have allowed the development of technologies and applications of virtual reality (VR). Watching 360-degree videos on such devices has become quite popular due to the feeling of "immersion" experienced by users, as they can explore the scenes in all possible directions. However, because 360-degree videos demand significantly higher bandwidth than traditional 2D videos, their streaming over the current Internet infrastructure has become a great challenge, since the average bandwidth of residential connections worldwide is far below the minimum transmission rate required for 360-degree videos. In fact, today only about 50 countries have residential Internet connections that support the minimum average bandwidth required to stream 2K video resolution [1], which means that immersive multimedia applications are still inaccessible to most Internet users around the world [2]. Consequently, in recent years, the design of *adaptive bit rate* (ABR) algorithms for 360-degree video streaming has received considerable attention because it is considered a key technology enabler for immersive applications over the Internet [3].

To date, most efforts in designing ABR algorithms have focused on dealing with the time-varying and limited nature of end-to-end Internet bandwidth through careful request of video frame qualities (translated into bit rates) compatible with the instantaneous bandwidth of a connection. Moreover, because no user is able to visualize the whole sphere at any given time instant, a number of techniques have been proposed to prioritize the request of portions of the sphere (i.e., regions of a video frame) the user is most likely to watch for a certain time period into the future over the viewport of an HMD. In this way, not only can bandwidth be saved, but also the client's buffer is not unnecessarily filled with data that never get to be watched. To support this task, the concept of partitioning a video file into "chunks" of fixed time duration, and each video frame into a set of "tiles," has been instrumental, since each chunk (associated to a set of consecutive tiles) can be encoded independently and stored on servers for later retrieval via DASH (Dynamic Adaptive Streaming via HTTP) [4, 5, 6]. As a result, a significant body of research has been done on short-term prediction, not only of the available bandwidth of a connection, but also of the user's viewport directions, among other features, that collectively provide a rich set of information that the ABR algorithm can exploit to deliver the best possible viewing experience to the end user [7, 8, 9].

While research on the issues mentioned above is certainly key for streaming 360-degree videos at the highest possible quality, most current approaches seem to assume that 360-degree videos are a single, continuous shot recorded by an omnidirectional camera. It turns out that, similar to 2D videos, the creation of 360-degree video content is expected to evolve and become a sequence of shots separated by cuts, as dictated by content producers and film directors, as a tool to guide the viewers' attention along the course of a story. However, editing 360-degree video content with the goal of creating a coherent narrative is more challenging, as viewers have partial "control of the camera" and the freedom to explore the scene. Due to this, some filmmakers have argued that the (story telling) editing of 360-degree content should be guided by estimates of which areas of the content are more salient or perceptually important to the user [10, 11, 12]. By doing so, the user is more likely to follow the intended narrative without missing important plots, at different parts of the sphere, across successive cuts along the time. Furthermore, if editing is done skillfully, it can help reduce any discomfort or "cybersickness" symptoms, while improving (or at least not decreasing) the feeling of "presence" or immersion.

Finding areas of a video frame that are perceptually important to a user requires understanding how visual attention works. Visual attention is a feature of the human visual system (HVS) that aims to reduce the complexity of scene analysis. It can be divided into *bottom-up* and *top-down* mechanisms that, combined, define which areas of the scene are considered relevant and therefore should be attended. The analysis of how humans perceive scenes is generally studied through subjective saliency maps, which are usually considered the ground truth of HVS. But due to the difficulty of using such maps in practical applications, several computational models of visual attention have already been proposed, which are generally classified as *bottom-up*, *top-down*, or *hybrid* [13, 14, 15]. However, despite being a very active area of research, the design of visual attention computational models for 360-degree videos is still incipient, with most of the available models focus on estimating the saliency in the viewport area alone [16]. Although such efforts are important, there is still a need for computational models that identify "regions of interest" (RoI) in the whole 360-degree video, since their identification can help general content editing. For example, depending on the content, there can be multiple RoIs in a scene, and each of them can lead the viewer to different "paths" along the underlying story. This knowledge can be creatively exploited by content creators, or it can be used to improve narrative control over the viewers across successive scene shots.

In the latter case, if content edition takes into account the estimated RoIs across the whole sequence of scenes, then a rich set of information can be stored at content servers to be later fetched by the ABR algorithm, via DASH, for optimal decision-making regarding the request of video tiles. For instance, metadata such as the coordinates and number of RoIs in the video (as chosen by a movie director), along with time stamps of the scene cuts, could help the ABR algorithm to request, in advance, the video tiles that should be viewed by the user in consecutive scene cuts according to the intended narrative of the content creator. To accomplish that, ideally, the user's RoI on a given shot should be aligned to the next (intended) RoI in the next scene cut. Such an action would not only maintain, with high probability, the user's attention on the intended region of the sphere, but it would also decrease head movements and, consequently, excessive discomfort or "cybersickness" [17]. To accomplish that, the user's history of viewport coordinates would generally be needed for short-term prediction of future head movements, and such predictions could guide the request of future video tiles, considering the aforementioned metadata.

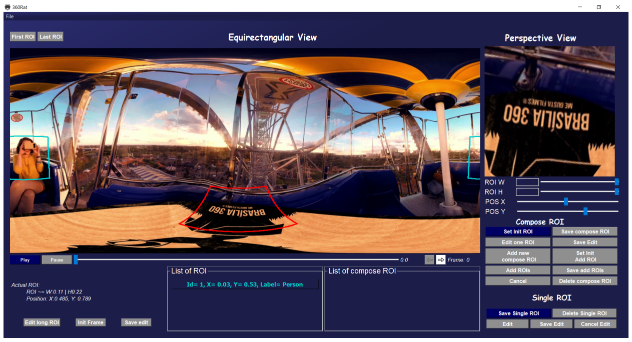
Considering the issues and challenges just described, we present in this paper some of our current efforts in developing a user-centric framework for perceptually efficient streaming of 360-degree edited videos. The basis of this framework is the development of a hybrid bottom-up and top-down visual attention computational model that can estimate RoIs in the complete 360-degree video. In particular, we target the identification of semantic context in video scenes that generally work as main drivers of attention, such as people, animals, vehicles, etc. Then, the estimated RoIs are to be used in the development of different strategies of video editing, particularly focusing on the alignment of RoIs, as described previously. The performance of each of these editing strategies will be evaluated with respect to their impact on overall quality of experience (QoE) of the end user. For that, a number of subjective experiments are being planned and carried out with the goal of understanding the user response to different editing strategies, considering the different dimensions that form the overall QoE when viewing 360-degree videos, such as "comfort," "presence," and "cybersickness", to name a few. Finally, the information derived from the proposed video editing strategies will serve as input to a new edition-aware ABR algorithm that aims to exploit the editing information and other metadata transmitted via DASH for the decision-making process, so that the highest possible QoE can be delivered to end-users under bandwidth-constrained connections.

In this paper, in particular, we focus on presenting some preliminary results in the development of our framework. First, we introduce the 360RAT software to recognize RoIs in 360-degree videos, described in Section 2. This software was developed with the main goal of creating datasets with annotated RoIs in different 360-degree videos. Such datasets can be used in the training of machine learning-based visual attention computational models or in performance evaluation studies. With this purpose in mind, we describe in Section 3 a subjective experiment in RoI annotation designed to create a dataset based on a set of 360-degree videos, and to evaluate the usability of our annotation tool. Based on this dataset, we showcase in Section 4 a preliminary comparison analysis between saliency maps, as generated by the Cube Padding computational model [18], and RoI maps created from the dataset. Finally, we present our efforts to investigate editing techniques in Section 5. In particular, we present preliminary results on a subjective experiment designed to evaluate QoE, expressed in terms of "comfort" and "presence," when users watch 360-degree videos that were statically edited (that is, when editing is done offline). The editing technique focuses on the alignment of RoIs in the same scene (that is, the RoIs are not in different scene cuts). We look at instantaneous (i.e., "snap-change" [17, 19]) and gradual rotation editing, and we present the mean opinion scores (MOS) obtained for the different cases, in different video categories.

2. Annotation Tool for Regions of Interest

Given that viewers have the freedom to explore a 360-degree video in all possible directions, content producers and directors agree that it is critical to establish a suite of techniques that allow viewers to explore a scene without missing the intended storyline [10, 20, 17]. Therefore, in some way, viewers need to be guided through the content by directing their attention to a specific *regions of interest* (RoI) on the sphere, as the underlying story unfolds over a succession of shots and cuts. Consequently, careful alignment of the shots and sophisticated video processing techniques will be needed because content manipulation can degrade QoE, reduce presence, cause cybersickness or decrease comfort [19, 21]. Hence, to produce an enjoyable immersive experience, filmmakers and content producers must understand how viewers watch 360-degree videos and interact with the content (i.e., how viewers choose their own RoI on a scene), so they can manipulate the scene to attract the viewers' gaze.

To help filmmakers and content producers in the task of



```
Actual ROI:
    ROI ~= W:0.11 | H0.22
    Position: X:0.485, Y: 0.789
```

```
Id= 1, X= 0.03, Y= 0.53, Label= Person
```

Figure 1: 360RAT visual interface.

defining RoIs in scenes or understanding viewer RoIs on a given content, we have developed the software *360RAT*. The goal of 360RAT is to make it easy to perform frame-by-frame annotations on 360-degree videos by allowing users to *i*) mark multiple RoIs in each frame of a video and *ii*) semantically classify the annotated RoI. As such, 360RAT can also help create annotated video datasets for the study of different aspects of 360-degree videos through machine learning algorithms. 360RAT is implemented in Python, using OpenCV, PyQT5, and the IDE Visual Studio Code. The software can be downloaded from https://github.com/MyllenaAPrado/360RAT and is available under a General Public License.

Figure 1 shows the main 360RAT interface. After loading a 360-degree video, the first frame of the video is shown in the main *Equirectangular View* window. A slider below this window allows the annotator to play the video or examine each frame individually. The annotator can then begin the process of choosing RoIs for video frames by positioning the slider in the first frame of the video and selecting an initial RoI by clicking the button *Set Init RoI*. Then a new screen appears asking the annotator to assign a "semantic class" to the selected RoI. For this, we adopted Microsoft's coco classes [22]. On the right-hand side of the interface, there are tools to visualize and adjust the selected RoI. In particular, the *Perspective View* field displays the Field-of-View (FoV) of the selected RoI, which helps the annotator to better visualize the selected area. The sliders found below the *Perspective View* allow a finer adjustment of the RoI size and position.

Since a given RoI may encompass multiple frames, 360RAT allows automatic annotation of an RoI over a set of N consecutive frames (e.g., a car moving down a road). For this annotation, the user must first select an RoI in the initial frame and then navigate through the following frames (using the slider below the *Equirectangular View* window) to choose a future frame N where the same RoI is visible. The software then performs a linear interpolation between the central coordinates of these two RoIs to compute the coordinates of the $N-2$ RoIs in the intermediate frames. However, since the adopted method for defining the coordinates of the intermediate RoIs does not perform well under movement discontinuities, the annotator must make sure that the first and last RoIs are relatively close in space and that the movement does not contain any discontinuities (e.g., start at the right-hand side of one frame and continue to appear on the left-hand side of another frame). Then the annotator can save the set of consecutive RoIs by clicking on the *Save Composite RoI* button. Finally, in addition to these features, 360RAT performs traditional operations such as *Delete*, *Edit*, and *List* saved RoIs. All information about the RoIs, for the complete video, is saved in a comma-separated values (CSV) format.

3. Subjective Annotation Experiment

An annotation experiment was carried out to evaluate the use of 360RAT software. In this experiment, participants annotated a benchmark dataset of uncut 360-degree videos of 60 seconds using 360RAT software. Eleven videos containing moving objects and a clear storyline were chosen. Of the 11 chosen videos, eight were taken from the University of Texas at Dallas dataset [23], one from the V-Sense Director's Cut dataset [24], and two were acquired from a Brazilian VR producer (CaixoteBR) [1]. The experiment was divided into two sessions, indicated as "group 1" and "group 2" in Table 1, which contains a list of the chosen videos, their original dataset or source, spatial resolution, frame rate, and time interval of the original content used in the experiment. The annotation experiment had the goal of testing 360RAT and creating a dataset of annotated 360-degree videos. Nine participants took part in the experiment, and their ages ranged from 23 to 50 years, with 2 being female and 7 being male. They were all researchers in the area of Computer Science and Electrical Engineering with different levels of familiarity with 360-degree videos. After a training phase, participants were asked to use 360RAT to annotate RoIs in the chosen set of videos according to the following rules:

- Only one RoI per frame could be selected to represent the most important area (or object) in the frame, considering the video storyline;
- All video frames should contain exactly one RoI;
- The size of the RoI should include the chosen object/area in the best possible way. By default, the maximum RoI size was set to around 1/4 of the viewport, but this size could be adjusted using the sliders on the right side of the interface;
- When selecting an RoI, the participant should assign an appropriate semantic class to that RoI.

Figure 2 shows a sample of the data collected for the video "Closet Set Tour." Based on this dataset, we created RoI maps to investigate their relationship to a computational saliency model, as described next.

4. Comparative Analysis of a Saliency Computational Model

To showcase one possible application for the use of 360RAT software, this section presents a comparison analysis between RoI maps, created from the annotation experiment using 360RAT, with saliency maps computed according to a computational visual saliency model. The goal of this comparison is to quantify the similarity between the RoIs annotated by the experiment participants and the salient areas of the videos, as predicted by the chosen computational saliency model. To perform this comparison, we built RoI maps from the annotated videos by executing the following steps:

1. First, we created difference maps between the reference

[1] http://caixotexr.com/projects/brasilia-360/

Table: Properties of the dataset of 360° videos used in the experiment.

Group	Video Name	Dataset	Resolution	Frame Rate	Interval (60s)
1	Ben-Hur Chariot Race	UTD [23]	4320 × 2160	24	00:00 – 01:00
1	Closet Set Tour	UTD [23]	4320 × 2160	29	00:07 – 01:07
1	FPV Race Drone Car Chase	UTD [23]	4320 × 2160	29	02:11 – 03:11
1	New York City Drive	UTD [23]	4320 × 2160	30	00:12 – 01:12
1	UTD Campus walk	UTD [23]	4320 × 2160	29	00:00 – 01:00
1	Wingsuit over Dubai	UTD [23]	4320 × 2160	29	00:00 – 01:00
2	Dubstep Dance	UTD [23]	4320 × 2160	29	00:00 – 00:30
2	Blue Angels Jets	UTD [23]	4320 × 2160	29	01:00 – 01:30
2	Partnership India	V-Sense [24]	4320× 2160	30	01:41 – 02:11
2	Amizade	Brasília 360°	4320 × 2160	30	00:42 – 01:12
2	Park	Brasília 360°	4320 × 2160	30	00:00 – 00:30

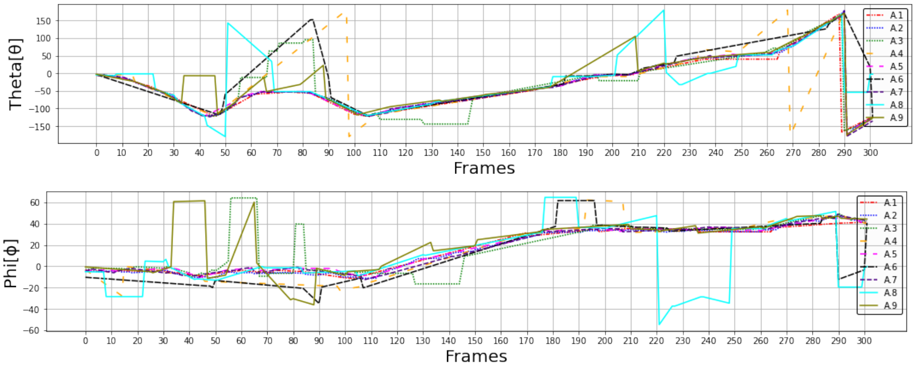


Figure 2: Yaw (θ) and pitch (Φ) values of annotated RoI for the video "Closet Set Tour" by each experiment participant A_i, ($1 \leq i \leq 9$).

frames and the annotated ones, which consist of frames where only the RoI has non-zero values;

2. Then, we created binary maps from these difference maps, with pixel values inside the RoI given a value '1', and pixel values outside the RoI given a value '0';
3. Lastly, RoI maps were created by applying a Gaussian filter ($\sigma = 50$) on these binary maps.

To compute the saliency maps, we use the Cube Padding saliency computational model [18], which is designed specifically for 360° videos. Cube Padding is a state-of-the-art model that is based on a convolutional neural network (CNN) architecture. It extracts spatial and temporal features and feeds them into a CNN and a Long Short-Term Memory (LSTM) architecture. For this analysis, we use the architecture "as is," pre-trained with the original authors' dataset. Figure 3 shows examples of RoI maps and predicted saliency maps for two videos. To compare RoI maps with saliency maps, we used three performance metrics [25]: Judd Area under the curve (AUC_Judd), cross-correlation (CC), and similarity (SIM). Metric values closer to '1' indicate a closer agreement of the maps. Table 2 shows the results of our evaluation. Notice that the annotated RoIs are in good agreement with the saliency maps generated by the Cube Padding saliency model. This result is interesting and indicates that there is a relationship between importance and saliency for 360 videos.

5. Subjective Experiment on User Experience for a Set of Static Video Editing Techniques

To lay the ground for the development of editing techniques that will work integrated with our ABR algorithm, in this section, we present some preliminary results of a subjective experiment that we designed to understand the user response, measured in terms of "presence" and "comfort," to a specific set of *static* video editing strategies. As introduced by Dambra *et al.* [19], the idea of static editing reflects the fact that the editing is made offline at video creation time, that is, any changes to a given edited video would require the creation of another video file, as opposed to *dynamic* editing where it happens at runtime. In particular, we are interested in the editing that aligns RoIs to achieve the so-called "match on attention" [10], as a way to control the user's attention on the underlying story and diminish head movements motivated by searches for other RoIs. Unlike Dambra *et al.* [19], however, we first consider *intra-scene* RoI alignments, that is, when editing occurs in the same scene and not across scene cuts. This is the case, for instance, when there are multiple RoIs in the same shot, and a director wants to drive the user's attention to a specific RoI within the same scene, departing from another one.

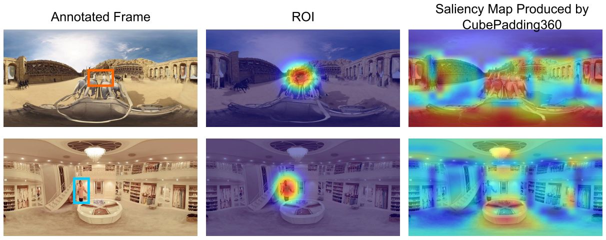


Figure 3: Comparison between RoI maps and saliency maps produced by the CubPadding360 model for "Ben-Hur Chariot Race" and "Closet Set Tour" videos.

Table 2: AUC_Judd, CC, and SIM values between saliency maps predicted via Cube Padding and RoIs maps created from the subjective experiment for videos in both Groups 1 and 2.

Group	Metrics		
	AUC_Judd	CC	SIM
1	0.8258	0.5203	0.8411
2	0.8603	0.6009	0.8412

In our experiments, two specific types of editing were considered: "instantenous" (that is, the so-called snap changes [26]) and "gradual" editing. Instantaneous editing aims to align a given RoI with another different RoI in the following video frame. In gradual editing, a *yaw* rotation of the scene is performed gradually, at constant speed, to bring the intended new RoI to a given direction aligned with a previous RoI. To mitigate cybersickness, we introduce the effect of "fade-in fade-out," as proposed by Farmani *et al.* [27]. From each original video, we created one video with instantaneous editing and four videos with gradual editing, each with a specific angular speed of camera rotation: 10 degree/s, 20 degree/s, 40 degree/s, and 60 degree/s, respectively. All videos were 30 seconds long, and the videos were selected in such a way that they all had the beginning of the gradual editing at the 14th second, with the fading effect lasting 1 second. Static editing was manually implemented using Adobe Premiere.

As far as video content is concerned, we selected videos based on the nature of the observed camera movement. This is because camera motion can affect viewer attention and transform the motion of moving objects, as discussed by Nasrabadi *et al.* [23]. Therefore, we focus on the acceleration of the camera movement in the scene, and we defined three video categories: "fixed," if the camera is static; "steady," when no significant acceleration of camera motion occurs in most of the video; and "dynamic," if otherwise. Two videos fitting each of these three categories were selected, and we did not limit the number of moving objects in the scene. In the experiment, each subject evaluated a total of 36 videos: six original videos, each associated with the five static editing techniques described above. To avoid bias in attention, audio was removed from all videos.

To carry out the subjective experiment, we developed an evaluation platform called Mono360[2] that consists of a web application that presents a scoring interface for videos displayed in an embedded 360-degree video player. Since it is a web application, it can be easily installed on any HMD. The interface was designed to be flexible enough to be tailored to new experiment designs. Figure 4 shows a snapshot of the subjective evaluation interface, together with a diagram containing the key information collected by the Mono360 application. All subjective experiments followed the ITU P.919 recommendation [28], and participants reported their comfort and presence scores using a 5-level rating scale. Additionally, we collect information on head movement and demographic data. The experiment was carried out on two HMD devices in two parts. First, we used the Oculus Rift HMD with 40 participants (60% female, 0% nonbinary, 55% first-time VR technology user, 35.6 years old on average (14.0 standard deviation)). Then we used the Oculus Quest HMD with 23 participants (43% female, 0% nonbinary, 60% first-time user of VR technology, 29.6 years old on average (7.0 standard deviation)).

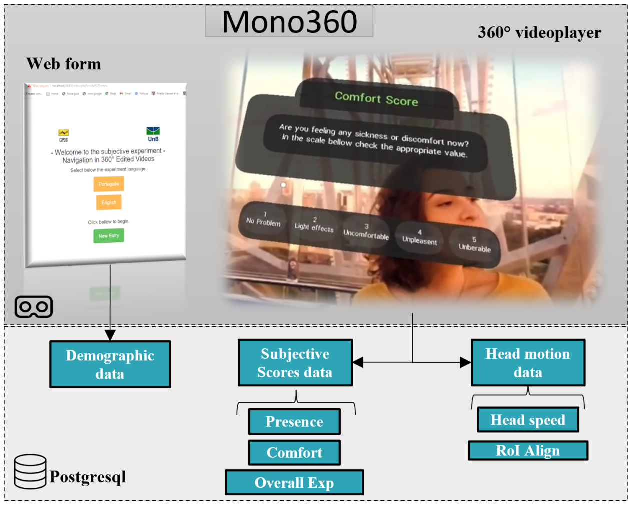


Figure 4: Example of the Mono360 interface for subjective evaluation. In the bottom: the type of data gathered from users.

Figure 5 presents the mean opinion score (MOS) values for each of the six videos, averaged over all types of editing. We notice that comfort and presence appear to be slightly inversely related: the higher the comfort, the lower the presence (and vice-versa). Despite the small number of video samples, this detected behavior seems to add to the balance of evidence that favors the existence of a negative correlation (or inverse relationship) between presence and comfort, as discussed by Wheech *et al.* [29] when treating cybersickness as a "constellation" of discomfort symptoms.

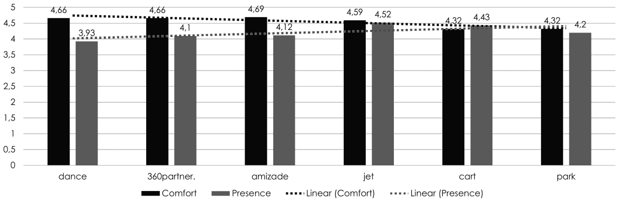


Figure 5: Mean Opinion Score (MOS) values for "presence" and "comfort" for each video content and over all editing types.

Figure 6 contains the MOS values for presence and comfort for each type of editing, grouped according to each video category: fixed, steady, and dynamic camera motion. In the case of comfort, the steady and fixed categories present a slightly higher MOS value than the dynamic one. This is a reasonable result to expect because the dynamic movement of the camera makes it difficult for the user to focus on any RoI or on any aspect of the scene. Moreover, in most categories, there is a slight decrease in comfort as the angular speed of gradual editing increases. Considering that the editing was done offline, the higher the angular speed, the higher the chances that the user misses an intended RoI and feels disoriented, making it difficult to maintain focus on any aspect of the scene. In the case of presence, the MOS values do not change much across all editing types in each video cate-

[2] https://gitlab.com/gpds-unb/mono360

gory, which indicates that the editing type does not have much impact on the viewers' experience of presence. It is interesting to note that the MOS values for the fixed category is lower than the MOS values for the steady and dynamic categories, indicating that the user feels less immersed in the content when there is no camera motion. However, since we only used two videos in each category, this result may be affected by the video content itself. More studies are needed to confirm this result. As previously mentioned, these are preliminary results of a major investigation which aims to deliver a user-centric framework for perceptually efficient streaming of 360-degree edited videos.

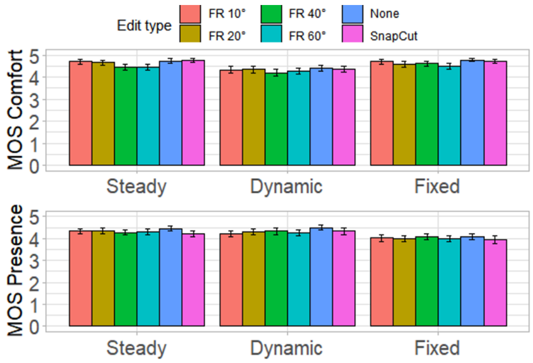


Figure 6: MOS values for each editing type and video category.

7. Conclusions

This paper presented a set of developments towards the design of a user-centric framework for perceptually efficient streaming of 360-degree edited videos. First, we introduced a tool for annotation of regions of interest (RoI) in 360-degree videos that allows the creation of datasets for the training of machine learning visual attention computational models and for the use in performance evaluation studies. Based on this tool, a subjective experiment was designed to evaluate the usability of the software and to create a dataset of annotated 360-degree videos. This dataset was then used to create RoI maps that were compared to saliency maps generated by the Cube Padding visual attention computational model. The saliency maps were found to agree well with the RoI maps, indicating that there is a relationship between the importance given by users to specific points in the scenes and the saliency regions detected by the model. Such findings encourage further investigation of the automatic detection of RoI in 360-degree videos for use in advanced video editing techniques. Finally, we presented preliminary results on a subjective experiment designed to evaluate the users' QoE in terms of comfort and presence when they are exposed to a specific set of static editing techniques (i.e., off-line editing) applied to different video categories regarding camera motion. Specifically, we investigated intrascene RoI alignment with the goal of reducing the user's head movements. We considered both instantaneous (i.e. "snap-changes") and gradual rotation editing, under different angular speeds. Preliminary results indicate that comfort and presence appear to be slightly inversely related: the higher the comfort, the lower the presence (and vice versa). Regarding the editing type, the users' comfort is reduced as the angular speed of gradual rotation increases, and the users' experience on presence seems to be not affected by the editing type.

Acknowledgments

This work was supported in part by Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) - Processo nº 2018/23086-1, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001, and by the Decanato de Pesquisa e Inovação, University of Brasília, Brazil.

References

[1] Ookla, "Speedtest global index," (Acessed in February 21, 2022). [Online]. Available: https://www.speedtest.net/global-index

[2] Cisco, "Cisco annual report 2018-2023," 2018, (Acessed in February 12, 2021). [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[3] C.-L. Fan, W.-C. Lo, Y.-T. Pai, and C.-H. Hsu, "A survey on 360° video streaming: Acquisition, transmission, and display," *ACM Comput. Surv.*, vol. 52, no. 4, aug 2019. [Online]. Available: https://doi.org/10.1145/3329119

[4] C. Ozcinar, J. Cabrera, and A. Smolic, "Visual attention-aware omnidirectional video streaming using optimal tiles for virtual reality," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 1, pp. 217–230, 2019.

[5] J. Fu, X. Chen, Z. Zhang, S. Wu, and Z. Chen, "360SRL: A sequential reinforcement learning approach for ABR tile-based 360 video streaming," in *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, 2019, pp. 290–295.

[6] D. V. Nguyen, H. T. Tran, and T. C. Thang, "An evaluation of tile selection methods for viewport-adaptive streaming of 360-degree video," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 16, no. 1, pp. 1–24, 2020.

[7] Y. Ban, L. Xie, Z. Xu, X. Zhang, Z. Guo, and Y. Wang, "CUB360: Exploiting cross-users behaviors for viewport prediction in 360 video adaptive streaming," in *Proc. IEEE International Conference on Multimedia and Expo (ICME)*, 2018, pp. 1–6.

[8] A. T. Nasrabadi, A. Samiei, and R. Prakash, "Viewport prediction for 360 videos: a clustering approach," in *Proceedings of the 30th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2020, pp. 34–39.

[9] F.-Y. Chao, C. Ozcinar, and A. Smolic, "Transformer-based long-term viewport prediction in 360° video: Scanpath is all you need," in *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2021, pp. 1–6.

[10] J. Brillhart, "In the blink of a mind – prologue," 2016, (Acessed in February 15, 2022). [Online]. Available: https://medium.com/the-language-of-vr/in-the-blink-of-a-mindprologue-7864c0474a29#.v0gfq5v0x

[11] M. Gödde, F. Gabler, D. Siegmund, and A. Braun, "Cinematic narration in VR–rethinking film conventions for 360 degrees," in *International Conference on Virtual, Aug-*

mented and Mixed Reality. Springer, 2018, pp. 184–201.

[12] S. Weaving, "Evoke, don't show: Narration in cinematic virtual reality and the making of entangled," *Virtual Creativity*, vol. 11, no. 1, pp. 147–162, 2021.

[13] L. Itti and C. Koch, "Computational modelling of visual attention," *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, Mar 2001.

[14] A. Borji, D. N. Sihite, and L. Itti, "What/where to look next? modeling top-down visual attention in complex interactive environments," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 5, pp. 523–538, 2014.

[15] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 569–582, 2015.

[16] M. Qiao, M. Xu, Z. Wang, and A. Borji, "Viewport-dependent saliency prediction in 360° video," *IEEE Transactions on Multimedia*, vol. 23, pp. 748–760, 2021.

[17] L. Sassatelli, A.-M. Pinna-Déry, M. Winckler, S. Dambra, G. Samela, R. Pighetti, and R. Aparicio-Pardo, "Snap-changes: a dynamic editing strategy for directing viewer's attention in streaming virtual reality videos," in *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, 2018, pp. 1–5.

[18] H.-T. Cheng, C.-H. Chao, J.-D. Dong, H.-K. Wen, T.-L. Liu, and M. Sun, "Cube padding for weakly-supervised saliency prediction in 360° videos," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2018.00154

[19] S. Dambra, G. Samela, L. Sassatelli, R. Pighetti, R. Aparicio-Pardo, and A.-M. Pinna-Déry, "Film editing: New levers to improve VR streaming," in *Proc. of the 9th ACM Multimedia Systems Conference*, 2018, pp. 27–39.

[20] C. Brown, G. Bhutra, M. Suhail, Q. Xu, and E. D. Ragan, "Coordinating attention and cooperation in multi-user virtual reality narratives," in *2017 IEEE Virtual Reality (VR)*, 2017, pp. 377–378.

[21] A. Serrano, V. Sitzmann, J. Ruiz-Borau, G. Wetzstein, D. Gutierrez, and B. Masia, "Movie editing and cognitive event segmentation in virtual reality video," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–12, 2017.

[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[23] A. T. Nasrabadi, A. Samiei, A. Mahzari, R. P. McMahan, R. Prakash, M. C. Farias, and M. M. Carvalho, "A taxonomy and dataset for 360° videos," in *Proceedings of the 10th ACM Multimedia Systems Conference*, 2019, pp. 273–278.

[24] S. Knorr, C. Ozcinar, C. O. Fearghail, and A. Smolic, "Director's cut: a combined dataset for visual attention analysis in cinematic VR content," in *Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*, 2018, pp. 1–10.

[25] M. Kummerer, T. S. A. Wallis, and M. Bethge, "Saliency benchmarking made easy: Separating models, maps and metrics," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[26] Facebook, "VR201: Lessons from the frontlines," Facebook Developers Conference, 2017.

[27] Y. Farmani and R. J. Teather, "Evaluating discrete viewpoint control to reduce cybersickness in virtual reality," *Virtual Reality*, pp. 1–20, 2020.

[28] ITU-T, "Subjective test methodologies for 360º degree video on head-mounted displays," 2020, Recommendation ITU-T P.919.

[29] S. Weech, S. Kenny, and M. Barnett-Cowan, "Presence and cybersickness in virtual reality are negatively related: A review," *Frontiers in Psychology*, vol. 10, 2019.