A Survey of Spherical Videos with an Emphasis on Spherical Projections

Brody Gimson
brodygimson@uvic.ca

University of Victoria
CSC421 A02 Fall 2023

Table of Contents

List of Figures and Tables iv

Abstract v

1.0 Introduction 1

2.0 Spherical Videos and their Standards 1

2.1 Types of Spherical Videos 1

2.1.1 Monoscopic Videos 1

2.1.2 Stereoscopic Videos 2

2.2 Stitching and Syncing 2

2.3 Standards 3

2.3.1 Omnidirectional Media Format (OMAF) 3

2.3.2 Spatial Media Format 4

3.0 Spherical Projections 5

3.1 Viewport Independent Projections 5

3.1.1 Equirectangular Projections 6

3.1.2 Cubemap Projections 7

3.2 Viewport Dependent Projections 9

3.2.1 Pyramid Projections 9

3.3 Summary of the Experiment 11

4.0 Other Formats and Future Work 12

4.1 VR180 Videos 12

4.2 6 Degrees of Freedom (6DoF) Filming 13

5.0 Conclusion 13

References 15

Cited References 15

General References 17

List of Figures and Tables

Figure 1: An equirectangular projection of Earth at night [18]. 6

Figure 2: Screenshot at 0:30 of the sample video in its default equirectangular projection. 7

Figure 3: 2-Dimensional diagram of a cubemap projection. 8

Figure 4: Screenshot at 0:30 of the sample video in a cubemap projection. 8

Figure 5: 2-Dimensional diagram of a pyramid projection. 10

Figure 6: Screenshot at 0:30 of the sample video in a truncated square pyramid projection. 11

Figure 7: Screenshot at 0:30 of the sample video projected to TSP then back to an equirectangular projection. 11

Table I: Summary of spherical projection experiment results 12

Abstract

Spherical videos are an immersive format with 3 degrees of freedom (3DoF) with standards such as MPEG's OMAF and Google's Spatial Media Format for their metadata. These videos are created from multiple cameras that cover 360-degrees in three axes where the footage from each camera is synced and stitched together into a sphere. The videos can be monoscopic and consist of one video channel or they can be stereoscopic and consist of two. Stereoscopic video channels are offset giving them depth at the cost of more complexity to create them, a headset to view them properly, higher bitrates to stream them, and more expensive hardware to capture them. Videos must be encoded into a flat format which requires a spherical projection to be used. These projections either encode everything at uniform quality, called viewport independent, or encode a section with more information than the rest, called viewport dependent. Two examples of viewport independent projections are equirectangular and cubemap projections, and a pyramid projection is an example of a viewport dependent one. To compare these different projections, the same equirectangular video was converted to cubemap and truncated square pyramid (TSP) projections. Cubemap was found to keep everything the same quality but reduced the file size and bitrate, and while TSP also reduced the file size and bitrate it removed information causing sections not to be the same quality. New formats are also emerging from traditional spherical video including VR180 which offers stereoscopic 3DoF videos that are easier for consumers to make and 6DoF which adds 3 more degrees of freedom and is still being actively researched.

1.0 Introduction

Spherical videos are gaining in popularity as virtual reality become more prevalent. They are videos that are projected into a sphere around the viewer, giving them 3 degrees of freedom (3DoF) to look around. These degrees include looking left and right, looking up and down, as well as tilting the head left and right. Platforms like YouTube often refer to spherical videos as “360 videos” due to the 360 degrees the user can move in any of those three axes [1]. These videos are often found in the entertainment industry, but there are other applications emerging for it, such as security [2]. As this format gains in popularity and use cases, the requirements for proper standards and considerations for working with them grows.

2.0 Spherical Videos and their Standards

2.1 Types of Spherical Videos

2.1.1 Monoscopic Videos

Monoscopic videos use one video channel and are primarily viewed without headsets but can be viewed with them [3]. These devices include phones using their accelerometers to look around and standard browsers using the mouse to look around through clicking and dragging. These videos require higher bitrates and resolutions than standard videos since the viewer will be looking at sections of the sphere at any point, and each of these sections need to be high enough quality.

Since there is one video channel, capturing these videos is more straightforward and inexpensive than the other type of video. Monoscopic videos can be captured with as little as two very wide-angle lenses, both covering 180 degrees of the sphere. This allows cameras that are incredibly small and not overly pricey. There are many consumer-grade monoscopic cameras, including GoPro’s MAX camera, which is small enough to be mounted on someone’s head [4].

2.1.2 Stereoscopic Videos

Spherical videos that are stereoscopic have two video channels, one for each eye [3]. They are slightly offset from each other to give depth to the video, like how our eyes work. To view these videos correctly a headset is needed. Without one, the channels can be viewed monoscopically or side by side, however the viewer will have difficulty checking if the offset is correct to give that feeling of depth. Since these videos have two channels, they require an even higher bitrate than monoscopic videos.

The hardware required to capture these types of videos often is not designed for standard consumers. They require a spherical apparatus that has cameras offset from each other and overlapping to cover 360 degrees. Often these high-grade camera’s do all the syncing and stitching for you as the task to do it manually is quite laborious. One such camera is Insta360’s Pro 2 which use 6 fisheye lenses arranged in a sphere and is aimed at professionals [5].

2.2 Stitching and Syncing

To create either type of spherical video, multiple cameras need to be used and their respective footage needs to be brought together into one video. This process requires syncing the different videos together in time and stitching them with each other to create the sphere [6]. Stitching is where most issues can occur, where poorly done stitching can be visible to the viewer as seams or other artifacts. With stitching, the more cameras that are used the more stitching is required. Once the spherical video is created, the software encodes it as one of the many spherical projections and adds the metadata needed for the software, or platform, we intend to play the video with. This process can be done by multiple different video editing suites, such as Adobe Premiere Pro [7]. If the tool suite doesn’t offer metadata injection, then tools like Google’s Spatial Media Injector can be used [8]. Some cameras can do everything themselves, or include software that does it, to varying levels of success.

With monoscopic videos since they can be captured with two cameras, they are the simplest to stitch together. Often it can easily be done manually, allowing for more fine-grained control of the stitching process. Stereoscopic videos have more complexity in that they have more footage to be stitched together due to having more cameras and the added requirement that the offset between channels is kept throughout different views in the video [6]. This adds a layer of complexity, which increases the need for software solutions that do this automatically for the video creator.

2.3 Standards

There are many platforms that support spherical videos now, so there are a few official and unofficial standards that exist. Two well-known examples are MPEG’s OMAF and Google’s Spatial Media format. There are multiple viewers that support both, including Nokia’s OMAF viewer [9].

2.3.1 Omnidirectional Media Format (OMAF)

To establish a global standard for immersive media, the Motion Picture Experts Group (MPEG) created the Omnidirectional Media Format (OMAF) which was designed to standardize spherical videos [10]. The standard works with ISOMBFF standard files and encodings, commonly called the MPEG-4 format, and simply outlines new 4-character-code (4CC) boxes that contain the metadata necessary for viewers to project the videos correctly.

The standard supports two projection formats, equirectangular and cubemap and has had two main versions [10]. The first version offered basic support for spherical videos and not much else. The second and current version expanded upon the first by offering overlay video support and different viewpoints in a video. These overlay videos are videos that can be put in front of the sphere or be seen wherever the user is looking like a HUD. The different viewpoints can be thought of with an example of having different seats in a stadium, allowing the viewer to change to them while watching. Version 2 also brings in some basic support for 6DoF videos, a new emerging format that is explored briefly in Section 4.2.

2.3.2 Spatial Media Format

Spatial Media is the format created by Google for 360 videos to be uploaded to YouTube [11]. Multiple tools support adding this metadata standard due to YouTube's popularity. The standard has sections for both the MPEG-4 and WebM formats, unlike OMAF which only supports MPEG-4 [12-13]. In this section we will focus on the MPEG-4 side of the standard across its two versions.

Version 1 of the standard came out before OMAF and utilized global metadata for the video in the form of XML files that would be linked with the video [12]. This global data included things such as the projection type of the video. Local metadata was stored alongside the videos tracks and would be localized to those tracks and included data such as GPS information. This first version only supported equirectangular projections due to their simplicity.

Version 2 made use of MPEG's 4CC boxes by creating a new box labeled "sv3d" [13]. This new version supported both equirectangular and cubemap projections. The use of these boxes also allowed videos to have both versions of the standard in case the viewer supported one but not the other, of course only working properly with both versions if the video was equirectangular.

3.0 Spherical Projections

There are many different projections out there, but only a few that are supported by standards. These projections can be grouped into two categories which are based on the concept of viewports. Viewports are sections of the sphere that the user is looking, or we want to be looking, at. These two categories are viewport independent and viewport dependent projections.

To understand these projection types further, some qualitative analysis was done on the same video but encoded in these different projections. The video used came from a sample aggregate dataset used for experiments with spherical videos [14]. The video labeled "2_Elephats.mp4" was chosen and is a 1080p spherical video of some elephants. This video was converted to other projections using the v360 filter that is part of FFmpeg [15]. FFplay was used to get more data on these converted videos. VLC was used to play these videos in their flat forms to qualitatively compare them [16].

3.1 Viewport Independent Projections

Viewport independent projections encode the video in uniform quality, so every viewport is of equal quality. This requires the video to be a very high bitrate and resolution, for example to have viewports viewed at full resolution of 1080x1200 on the Oculus Rift headset the video needs to be at a 6K resolution which requires a whopping 400 Mbps bitrate [17]. This bitrate is also needlessly high as most of that information isn't viewed at the same time since the user only sees a portion of the sphere in their field of view. This requires some consideration when streaming the content and has been an area of research to improve upon the format. The projection type selected can help with lowering this bitrate, but only to some degree. The projections of this type that were looked at were the equirectangular and cubemap projections.

3.1.1 Equirectangular Projections

These projections are quite common and were one of the first used in spherical videos. This projection is also commonly found in creating world maps, like the one in Figure 1. They are simple since they are a single face for the whole sphere, but this simplicity comes at a cost. The poles require more information to be allocated to them compared to the rest of the video, which needlessly adds to the bitrate and file size of the video. We can see this in Figure 1 where Antarctica is much bigger than it should be and takes up a huge portion of our map. This ends up being our main issue with equirectangular projections, but they still achieve this uniform quality, so they are supported by both OMAF and Spatial Media standards.

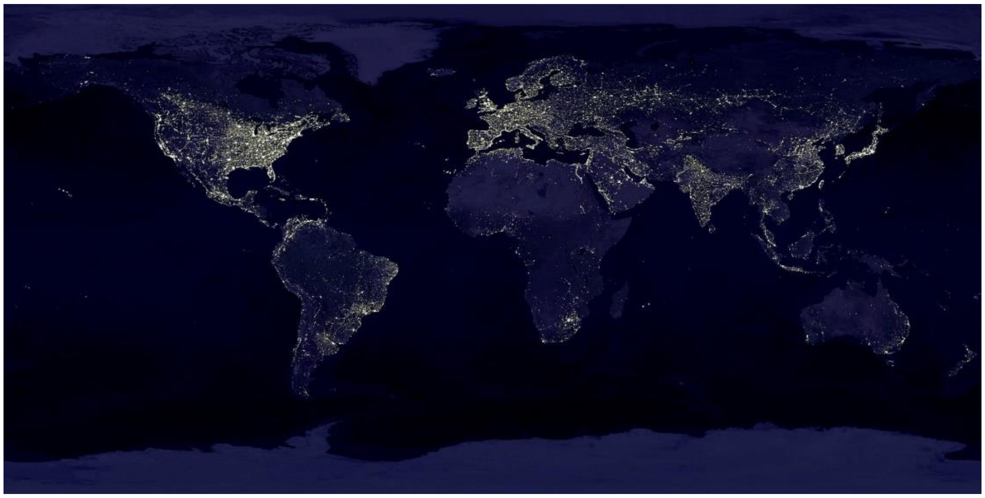

Figure 1: An equirectangular projection of Earth at night [18].

In the qualitative analysis experiment, the sample video already came in the equirectangular projection format, so no conversion was needed. Figure 2 is a screenshot at 0:30 of the video. The picture further illustrates this added info redundancy by looking at the clouds at the top and the water at the bottom of the image. The starting file size of the sample video was 33.1 MB and the bitrate while viewing was 4.3 Mbps.

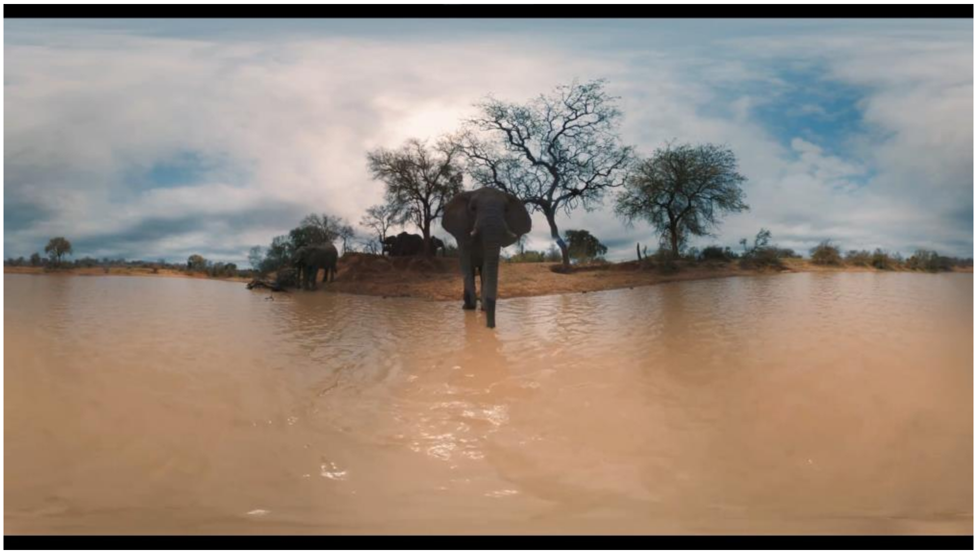

Figure 2: Screenshot at 0:30 of the sample video in its default equirectangular projection.

3.1.2 Cubemap Projections

Cubemaps map 90-degree sections of the sphere to the faces of a cube. Doing so requires a little more complexity when encoding these videos originally and decoding them to watch, but it is minuscule when compared to the reduction in bitrate and file size it offers. Unlike equirectangular, the only redundant info is around the edges of the cube faces, which is significantly less than equirectangular poles. Meta engineers found it could reduce the bitrate and file sizes by up to 25% compared to equirectangular while achieving the same quality [19]. This reduction is not only done by having less redundant information, but also due to motion vector estimation being more effective on the flat faces of the cubes then the distorted view of an equirectangular projection. Both OMAF and Spatial media support this projection type due to its uniform quality and reduction of bitrate.

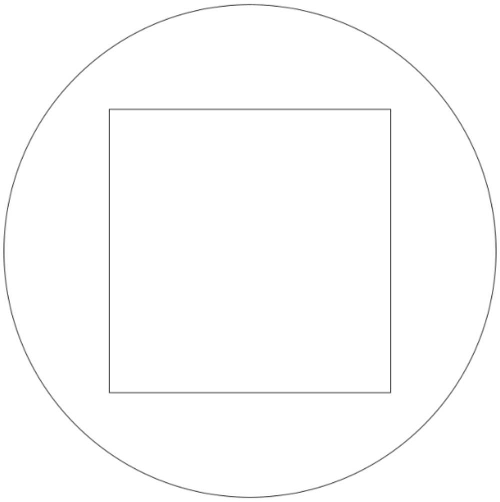

Figure 3: 2-Dimensional diagram of a cubemap projection.

In the experiment the cubemap projection did require a conversion and used the “c3x2” projection type in FFmpeg. The projection reduced the bitrate to 3.2 Mbps and the file size to 23.1 MB. Figure 4 shows a screenshot at 0:30 of the cubemap projection. In the screenshot you can see that compared to the equirectangular projection there is less redundant information, and the quality of any section is the same such as the main elephant in front of the viewer.

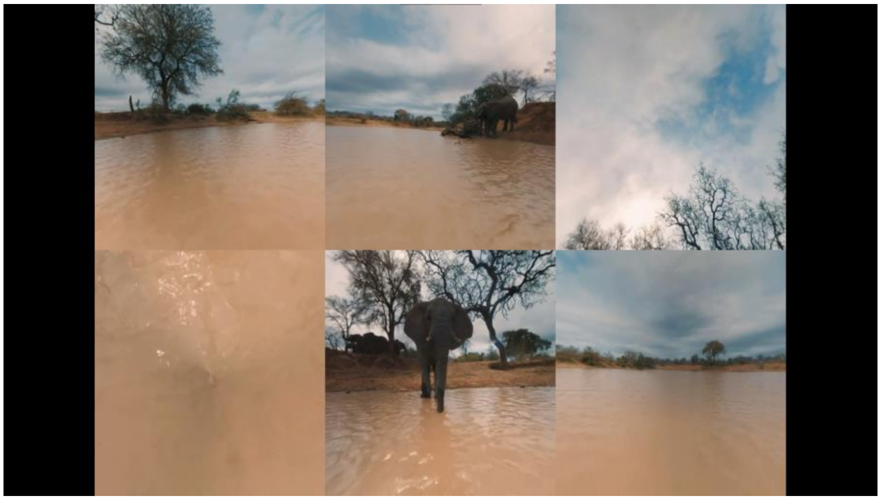

Figure 4: Screenshot at 0:30 of the sample video in a cubemap projection.

3.2 Viewport Dependent Projections

Viewport dependent projections encode a portion of the video with more information than the rest. Projection types of this nature significantly reduce bitrates due to this. The most common projection of this type is the pyramid projection.

3.2.1 Pyramid Projections

Pyramid projections work by projecting the viewport onto the bottom square face of the pyramid, and the rest of the video onto the triangle faces extending outside the sphere. This reduces the information not in the viewport but requires more GPU processing for decoding the video [17]. The quality degradation becomes more apparent the further the viewer looks away from the main viewport, reaching its max at 180-degrees behind them at the tip of the pyramid. The reduction in information significantly reduces file sizes and bitrates, up to 80% in the case for Meta's engineering team when they were testing it [19].

Pyramid projections can be used for a whole video if other sections of the video aren't important, for example in a video taken from within the cockpit of a plane looking out we may not care about showing the seat and the pilot since it is not likely to be where the viewer is looking. Research has also been done to use multiple videos for each viewport and change them while the user is looking around to lower bitrates but achieve uniform quality. Meta's engineering team investigated this using 30 different encoded viewports all at 5 different levels of quality to make a total of 150 videos [19]. The pyramid projection hasn't been incorporated into either OMAF or Spatial Media but may be considered in the future as research into its uses continue.

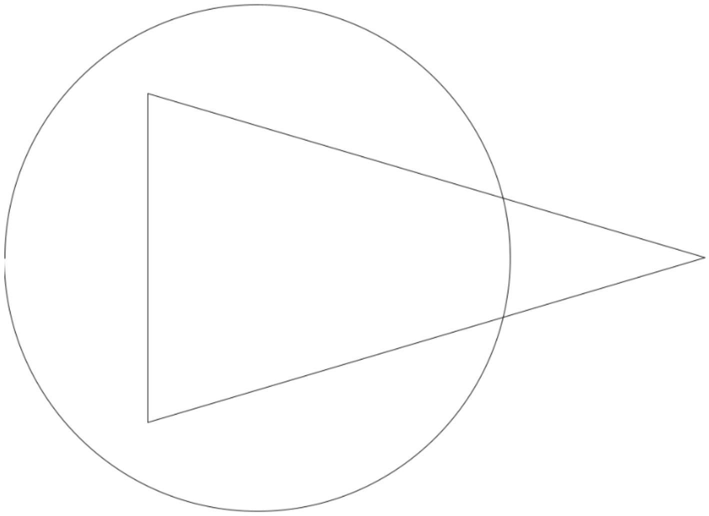

Figure 5: 2-Dimensional diagram of a pyramid projection.

In the experiment, the pyramid projection available was not the same as what Meta used, but its goal is similar. FFmpeg has a truncated square pyramid projection, abbreviated to “tsp” by the v360 filter, which still encodes one viewport with higher quality and projects the others onto the triangular pyramid sides but doesn’t reduce the information to the same extent. FFmpeg also kept the video at 1920x1080 which ended up upscaling the projection. To get the same quality the resolution could have been greatly reduced to about the size of two faces of the cubemap projection. Due to this upscaling the bitrate and file size ended up being higher than the equirectangular, at 4.7 Mbps and 35.0 MB respectively, which is likely due to there being more compact information in the video. Bringing the video back to equirectangular did lower the file size to 23.7 MB and bitrate to 3.2 Mbps, showing that information was in fact removed as expected. Figure 6 and Figure 7 show these two projections on the video at 0:30.

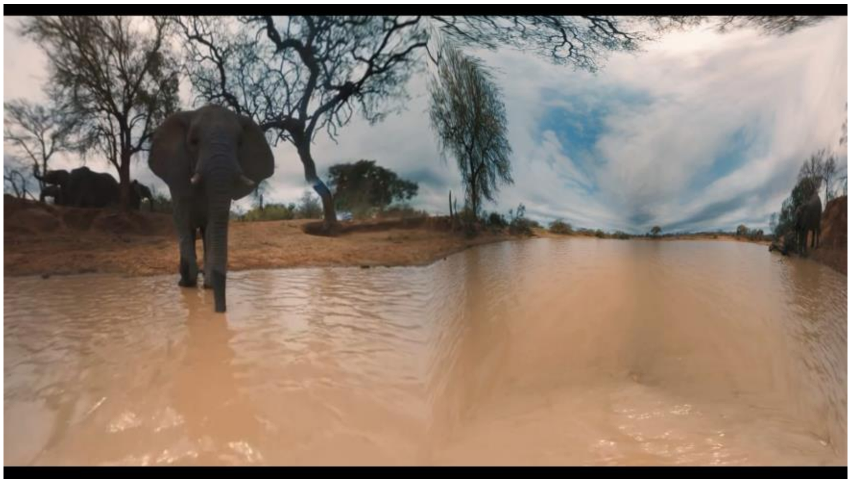

Figure 6: Screenshot at 0:30 of the sample video in a truncated square pyramid projection.

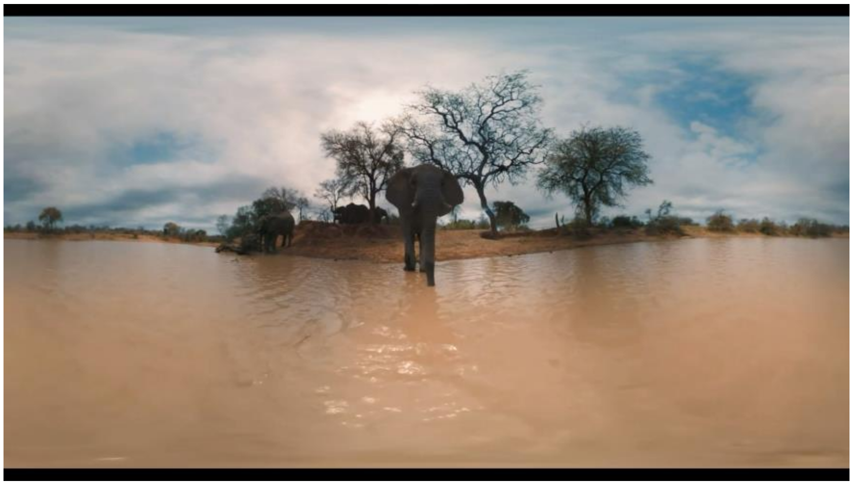

Figure 7: Screenshot at 0:30 of the sample video projected to TSP then back to an equirectangular projection.

3.3 Summary of the Experiment

The experiment offered more details into these different projections which was informative and insightful. It was clear why equirectangular was used originally due to its simplicity and it being well known outside of this application in map creation previously. The improvements one gets from using a cubemap projection was made clear with the reduced file size and bitrate while retaining the quality. The truncated square pyramid projection wasn’t quite the same as the projection used by meta but was a good example of a viewport dependent projection. The video used acted as a good example of why this projection type can be useful, since the elephant in the front viewport was more likely to draw the viewer's attention compared to the rest of the video. Although in the experiment most of the videos could not be videoed in their spherical format, viewing them flat still offered enough insight to survey their advantages and disadvantages. Table 1 shows a summary of the information that was outlined in the different subsections.

Table I: Summary of spherical projection experiment results

Projection Type	**Bitrate (Mbps)**	**File Size (MB)**
Equirectangular	4.3	33.1
Cubemap	3.2	23.1
Truncated Square Pyramid	4.7	35.0
TSP back to Equirectangular	3.2	23.7

4.0 Other Formats and Future Work

4.1 VR180 Videos

A newer format like spherical videos is the VR180 format. They have one standard, which is part of Google's Spatial Media standards [20]. These videos can be thought of as stereoscopic spherical videos that are only 180-degrees instead of the full 360. This allows them to have a lower bitrate than stereoscopic 360 videos and be easier to capture. Instead of requiring an apparatus with multiple overlapping cameras, these videos only require two cameras, one for each eye. This allows them to be inexpensive and accessible to consumers like monoscopic 360-degree cameras. Some cameras allow the user to even shoot both via the ability to fold and unfold the cameras, such as the Insta60 EVO [21]. This format is gaining popularity due to it being more accessible than full stereoscopic 360-degree videos while still giving that immersive depth.

4.2 6 Degrees of Freedom (6DoF) Filming

Spherical videos have 3 degrees of freedom (3DoF), which allows the viewer to look around them while the video is playing. If the user physically moves their head and not just tilt or pan it, then it can be disorienting since the view will not change. 6DoF videos would change this by allowing the user to look around like 3DoF and move their head up and down, move their head left and right, as well as move their head forward and back. These videos require expensive rigs to film them, needing even more cameras than a stereoscopic 360-degree video by having overlapping streams for depth and to compensate for the user moving their head in space. The more space you want to allow the user to move in, the larger the sphere of the apparatus needs to be, and the more cameras one needs to capture the video. Meta made such a device that has 16 wide angle cameras arranged in such a way that they overlap with each other [22]. This design may be on the market soon, but it may still be a while before platforms support this type of video. OMAF V2 already offers some basic support for these videos [10], but there may be more changes on the horizon as more work is done with 6DoF videos.

5.0 Conclusion

Spherical videos are continuing to become more popular as forms of media. Although their practical applications have not been fully explored, they are entertaining and allow for a unique way to capture and relive moments. The standards that have emerged are still being developed, and new improvements are being proposed often through new projections and axes of motion. Capturing these videos is becoming more accessible to consumers as cameras become smaller and software becomes better at working with them.

References

Cited References

[1] YouTube. “Virtual Reality - YouTube.” YouTube.com. https://www.youtube.com/@360/featured (accessed Oct. 7, 2023).

[2] Avigilon. “Panoramic & 360-Degree Cameras.” Avigilon.com. https://www.avigilon.com/security-cameras/panoramic-360 (accessed Dec. 10, 2023).

[3] M. Rowell. “VR Video Formats Explained.” 360Labs.net. https://360labs.net/blog/vr-video-formats-explained (accessed Dec. 10, 2023).

[4] GoPro. “GoPro MAX 360 Action Camera.” GoPro.com. https://gopro.com/en/ca/shop/cameras/max/CHDHZ-202-master.html (accessed Nov. 15, 2023).

[5] Insta360. “Insta360 Pro 2 - 360 VR Camera.” Insta360.com. https://www.insta360.com/product/insta360-pro2 (accessed Nov. 15, 2023).

[6] Meta. “Stitching 360 and 180 Video.” Creator.Oculus.com. https://creator.oculus.com/getting-started/getting-started-stitching/ (accessed Dec. 10, 2023).

[7] Adobe. "Edit 360/VR Video." Creativecloud.Adobe.com. https://creativecloud.adobe.com/en-CA/learn/premiere-pro/web/edit-360-vr-video (accessed Dec. 8, 20-23).

[8] *Spatial Media Metadata Injector*. (2023), Google. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/google/spatial-media/tree/master/spatialmedia

[9] *OMAF*. (v2.0.0), Nokia. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/nokiatech/omaf

[10] M. M. Hannuksela and Y. -K. Wang, "An Overview of Omnidirectional MediA Format (OMAF)," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1590-1606, Sept. 2021, doi: 10.1109/JPROC.2021.3063544.

[11] YouTube. "Upload 180- or 360-Degree Videos." Support.Google.com. https://support.google.com/youtube/answer/6178631 (accessed Dec. 10, 2023).

[12] *Spatial Media - Spherical Video RFC*. (2023), Google. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/google/spatial-media/blob/master/docs/spherical-video-rfc.md

[13] *Spatial Media - Spherical Video V2 RFC*. (2020), Google. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/google/spatial-media/blob/master/docs/spherical-video-v2-rfc.md

[14] *A-large-dataset-of-360-video-user-behaviour*. (2021), 360VidStr. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/360VidStr/A-large-dataset-of-360-video-user-behaviour/tree/main

[15] *FFmpeg*. (v6.1), FFmpeg. Accessed: Dec. 3, 2023. [Online]. Available: https://www.ffmpeg.org/

[16] *VLC Media Player*. (v3.0.20). VideoLAN. Accessed: Dec. 3, 2023. [Online]. Available: https://www.videolan.org/vlc/

[17] M. Zink, R. Sitaraman, and K. Nahrstedt, “Scalable 360° video stream delivery: Challenges, solutions, and opportunities,” *Proceedings of the IEEE*, vol. 107, no. 4, pp. 639–650, Apr. 2019. doi:10.1109/jproc.2019.2894817

[18] Pixabay. “Black Textile - Free Stock Photo.” Pexels.com. Accessed: Dec. 3, 2023. [Online.] Available: https://www.pexels.com/photo/black-textile-41949/

[19] E. Kuzyakov and D. Pio. "Next-generation video encoding techniques for 360 video and VR." Engineering at Meta. https://engineering.fb.com/2016/01/21/virtual-reality/next-generation-video-encoding-techniques-for-360-video-and-vr/ (accessed: Nov. 6, 2023).

[20] *Spatial Media - VR180 Video Format*. (2023), Google. Accessed: Dec. 3, 2023. [Source Code]. Available: https://github.com/google/spatial-media/blob/master/docs/vr180.md

[21] Insta360. “Insta360 EVO - Relive what you love.” Insta360.com. https://www.insta360.com/product/insta360-evo (accessed Dec. 3, 2023).

[22] A. P. Pozo et al., "An Integrated 6DoF Video Camera and System Design," *ACM Transactions on Graphics*, vol. 38, no. 6, pp. 1-16, Nov. 2019, doi: 10.1145/3355089.3356555.

General References

[1] A. Yaqoob, T. Bi, and G. -M. Muntean, "A Survey on Adaptive 360° Video Streaming: Solutions, Challenges and Opportunities," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2801-2838, Jul. 2020, doi: 10.1109/COMST.2020.3006999.