UNIVERSITY OF TWENTE.

Academic Research Skills
Conference Proceedings

internal
edition

Characterizing navigation graphs for 360-degree videos.

Timo H. van der Kuil, Romas Zubavicius, Suzan Bayhan, Mahboobeh Zangiabady

Pre-master track: CS and IST

Submission date: January 22, 2021

Viewport Prediction (VP) is the technique of predicting where the viewer of a 360-degree video will look at in the next one to three seconds. VP is the cornerstone of reducing Bandwidth (BW) usage of 360-degree video, whilst at the same time increasing the Quality of Experience (QoE) by allowing higher resolution streaming. This is vital to allow 360-degree video to increase its popularity. There are many ways to do VP, but after a literature survey, Navigation Graphs (NGs) look to be the most promising technique. Based on the well-proven graph theory, NGs predict where the viewer will look at using the likelihood of transitioning from one view to another. This work aims to progress the field of VP through NGs by characterizing the NGs and find what NGs say about the content of the videos. We propose a dynamicity score that characterizes the NGs. Classifying how dynamic the viewing behaviour is, helps us to show that video content has a significant impact on viewing behaviour.

360-Degree video; Quality of Experience; Viewport Prediction; Navigation Graphs

1. Introduction

High Definition (HD) videos have been on streaming platforms like YouTube for more than a decade [1] with 4K video support following shortly after [2]. The methods used to achieve streaming of 4K videos are quite straightforward: finding better compression techniques, keeping track of buffer health, and developing improved streaming protocols.

360-degree videos, however, are relatively new and face different challenges than regular high-resolution video. One of the main challenges is the sheer size of a 360-degree video: a file size increase of 4-6x compared to conventional video [3], [4]. This is because 360-degree video needs to cover a spherical area, surrounding the viewer with high-resolution video. Streaming 4-6x more data is the major issue, especially when one considers the fact that regular 4K video requires an internet speed of around 25Mbps [5].

Reducing Bandwidth (BW) requirements of 360-degree is vital for the success of this medium. Since it is not possible to achieve this reduction with conventional methods, other methods of reducing BW requirements are being researched. Many of these innovative approaches are trying to predict what the user will look at in the next 1 to 3 seconds. BW usage can be reduced by only streaming the parts of the video that the viewer will look at in the near future. There are various approaches to predict the viewport, from regression methods to neural networks [6]-[9]. Navigation Graphs (NGs) seem to have the most possibilities as this is a novel approach to Viewport Prediction (VP) [10].

The Main Question (MQ) of this research follows from those three parts:

Main Research Question: *What are the characteristics of navigation graphs generated from publicly available datasets and what can they say about the original 360-degree videos?*

Furthermore, this research will answer the following three sub-questions.

Research Question 1: *What are the requirements for a dataset to be used to generate NGs?*

Research Question 2: *How to generate NGs from large datasets?*

Research Question 3: *How to characterize the generated NGs using statistics and visualization?*

Some noticeable results that this paper shows is that there are not many requirements for datasets, that generating NGs scales well when using large datasets, and that dynamicity of NGs can characterize NGs well.

Section 2 introduces the purpose and definition of NGs. After that, the methodology used to answer the RQs and their results are discussed. Lastly, the main research question will be answered by using the results found by answering the three RQs.

2. Background of Navigation Graphs

An NG is a weighted, directed graph that represents a view transition model of someone watching a 360-degree video. This is done for either Single User (SU) or Cross User (CU) VP. We will use Figure 1 as a visualization of the terms that this section explains.

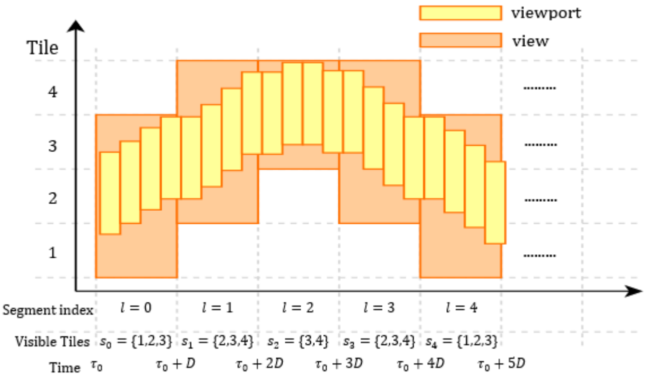

Figure 1.; Visualization of 4 segments, adapted from Park et al. *[7]*

A video consists of several segments and tiles. Segments are temporal sections of the video with a fixed duration D that is usually between 1 and 15 seconds. Suppose $D = 15$ seconds and a video is 60 seconds long: there are $\frac{60}{15} = 4$ segments of 15 seconds.

Tiles are spatial sections of the video and form a grid, $4x4$ for example. Suppose the grid is indeed $4x4$: there are $4 * 4 = 16$

equal-sized tiles in the video. Combining the 4 segments (temporal) calculated earlier, there are $4 * 16 = 64$ unique tiles (spatial) in a 60-second video with $D = 15$ and a $4x4$ tiling. Because NGs represent a view transition model, views need to be defined as well. A view is the set of visible tiles (visible to the user) within a segment with duration D, or: the union of viewports within a segment with duration D. A viewport is the set of visible tiles at time t [7].

The CU NG G is defined as $G = (V, E)$. The set of vertices V of the NG G are defined as:

$$V = \{v | v = (l, s), l \in \{1,2, \ldots, L\}\ and\ s \in S\}, \qquad (1)$$

where v is the view, and a tuple of the segment index l and a set of visible tiles s, l is the segment index, L is the total number of segments in a video, s is the set of visible tiles, and S is the set of all combinations of tiles in a video.

The set of edges E of the NG G are defined as:

$$E = \{(v_i, v_j) | v_i, v_j \in V, w(v_i, v_j) = p(v_j | v_i), i, j \in 1, \ldots, N\}, \qquad (2)$$

that connects the vertices (v_i, v_j) with the weight $w(v_i, v_j)$. (v_i, v_j) are vertices that have an edge between them, V is the set of all vertices of this graph, and $w(v_i, v_j)$ is the weight function defined as $p(v_j|v_i)$ which is the probability of transitioning from v_i to v_j. N is the number of vertices in V [7].

The weight function for the edges is defined as:

$$w(v_i, v_j) = \frac{number\ of\ transitions\ from\ v_i to\ v_j}{number\ of\ visits\ to\ v_i}$$

The SU NG G has a similar definition as the CU NG discussed earlier. The set of edges has the same definition as Equation 2 and the SU vertex is defined as:

$$V = \{v | v \in S\} \qquad (3)$$

3. Methodology

The following subsections provide detailed descriptions of the steps that are necessary for this research as shown in Figure 2.

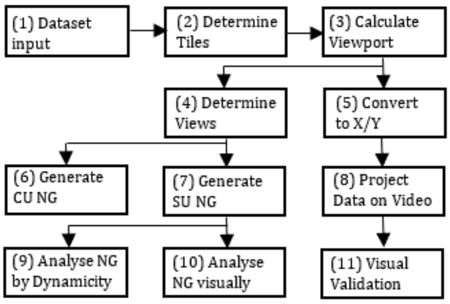

Figure 2.; *Steps in this research*

3.1. RQ 1: What are the requirements for a dataset to be used to generate NGs?

RQ1 considers only step 1 from Figure 2. We use the data from Xu et al. [11] in our characterization. Out of the list of available datasets, two are picked to compare in Table 1.

Dataset	Subjects	Videos	Data Format
Wu et al. [12]	48	18	X/Y/Z and unit quaternion
Xu et al. [13]	40	48	Longitude and latitude

Table 1.; *Comparison of two datasets*

Since the main purpose of this research is the characterization of NGs, we chose the more simplistic dataset, namely VR-HM48 by Xu et al. [13] to minimize the time spent on generating the NGs and maximizing the time spent on characterizing them. The simplicity of VR-HM48 is due to the data format: longitude and latitude. As a 360-degree video is often projected using equirectangular mapping, a 2D coordinate system is perfect.

3.2. RQ2: How to generate NGs from large datasets?

RQ2 considers all the steps from step 2 to step 8 from Figure 2. The steps will be discussed in order:

(2) Determine Tiles

Tiling can vary from video to video and platform to platform, so this tool has the ability to change the tiling easily. For the majority of tests, $4x4$ tiling is used. This step divides the video into the given tiling.

(3) Calculate Viewport

This step calculates the viewport that the subject is looking at. This viewport is based on the hardware used to record the dataset, the HTC Vive. Using the Field of View (FoV) of this device, the viewport can be extrapolated.

(4) Determine Views

Determining the view is relatively straightforward. Take the top left corner of the viewport and check in what tile this is. Then, check the bottom right corner of the viewport and make a set of these two tiles. These are unique for this viewport, as only these two tiles can draw that specific viewport.

(5) Convert to X/Y

To visualize the data on the video, X/Y coordinates are necessary.

(6) Generate CU NG

When all relevant data is present, the CU NG can be generated. This generation follows the definition of the vertices and edges that can be found in Section 2, Equations 1 and 2.

(7) Generate SU NG

When all relevant data is present, the SU NG can be generated. This generation follows the definition of the vertices and edges that can be found in Section 2, Equations 2 and 3.

(8) Project Data on Video

This step is only used for visualizing the dataset on the video. The tiles are drawn in white lines over the video, and the viewport is shown as a green rectangle. The rectangle moves over the video and visualizes what the subject is looking at.

3.3. RQ 3: How to characterize the generated NGs using statistics and visualization?

To analyse the NGs that are generated using the aforementioned dataset a set of statistics and metrics will be used. This turns out to be much harder to do for directed graphs. Where for an undirected graph one can find metrics such as the shortest-path, commute time, and diffusion distances, these metrics are not specifically adapted for directed graphs and Markov chains [14]. Nonetheless, various metrics are of interest for characterizing NGs. Table 2 shows the characterizations and the accompanying metric that will be used in this research.

The main metric is the dynamicity score, a score that represents how dynamic a subject's viewing behaviour is. One way to calculate this is by making use of the average degree of the NG. We calculate this using $Dynamicity_{v1} = \frac{TotalDegree}{TotalVertices}$, where $TotalDegree$ is the sum of all in- and outdegrees of every vertex.

Another way of calculating the dynamicity score is by using σ, the standard deviation of a distribution. First, we count all visits to a tile by a subject. Then, we order the counts of tiles to form a normal distribution shape with the tile index on the x-axis and tile count on the y-axis. The higher σ is, the more tiles with high or similar counts have been visited by the subject, which could mean the viewing behaviour is dynamic. We get a dynamicity score from this σ by using $Dynamicity_{v2} = \sigma * \frac{\text{TotalVertices}}{\text{TotalTiles}}$.

In the following section we discuss a comparison we conducted between the two dynamicity scores.

	Characterization	Description	Methods
1	Dynamicity of subjects	By scoring how often a subject moves their head, the dynamicity of the subject can be determined and compared across various videos. **Goal:** find whether a subject has predictable viewing behaviour.	Dynamicity score
2	Dynamicity of videos	By scoring how often subjects move their head in a video, the dynamicity of the video can be determined and compared. **Goal:** find whether content causes dynamic viewing behaviour	Dynamicity score
3	Does dynamicity stem from video or subject	How do the results of 1 and 2 compare? **Goal:** find whether there is a connection between content, viewing behaviour, and subjects	Comparison of 1 and 2

Table 2.; *Characterization methods*

4. Results

This section discusses the results found for the three RQs and use those to answer the MQ.

Table 3 shows the requirements necessary to generate an NG from a dataset. If a dataset conforms to all requirements, then this dataset can be used to generate NGs. These requirements follow from the definition of NGs, discussed in Section 2, Equations 1-3.

Requirement	Reason
Video files or a way to obtain the used video files	The duration and resolution of the video are necessary, but the videos could also be used as visual validation of the results.
Head movements	The head movements need to be saved in a format that allows conversion to a view
Head Mounted Display (HMD) that was used	The used hardware is important since the FoV varies for types of HMDs

Table 3.; *Requirements for head movement datasets*

A SU NG can be seen in Figure 3. The vertices represent the set of visible tiles, and the edges are the transitions between these sets of visible tiles. The labels that are on the vertices and edges are the probabilities of either staying at that vertex or transitioning over an edge. The labels on the edges are closer to the origin vertex of that transition. Whenever an edge has an arrow in both directions, the label that is closest to the vertex is the probability of originating at that vertex and transitioning to the connected vertex.

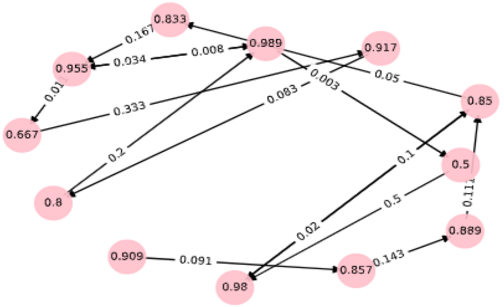

Figure 3.; *SU NG for Subject_1 and video 'Help.mp4' with a duration of 31 seconds, a 4x4 tiling, and a segment length $D = 1$ second*

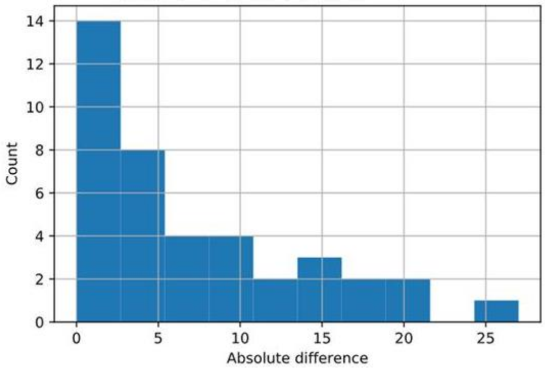

Figure 4.; *Comparison between dynamicity v1 and v2*

Since the CU NG is much bigger than the SU NG shown in Figure 3, the visualization of such a graph would be of no use. CU NGs in this research have around 350 vertices and many more edges. That is why metrics are used in the next section to still be able to characterize the NGs.

Section 3.3 and Table 2 explain what metrics will be used and how they have been chosen. Moreover, Section 4.2 explains why metrics are necessary for CU NGs, and introduce the dynamicity score and two ways of calculating it.

We compared both scores by measuring the absolute mean difference between the ranks (how the subject placed relative to other subjects according to dynamicity score). A score of 1 for $dynamicity_{v1}$, means that subject was completely static during the video, without ever moving to a different viewport, while a score of > 2 denotes some dynamicity, with a maximum observed score of 4.5. A Higher dynamicity score says, that viewing behaviour was dynamic. The same applies to $dynamicity_{v2}$, while in this case, the score of 0 means the subject was static, and the maximum observed score was 167, allowing for more range to explain the behaviour more precisely. Figure 4 shows the similarity of the two scores, from which we can conclude that $Dynamicity_{v2}$is the best way to determine dynamicity.

Figure 5 shows the boxplot that compares the means of all 40 subjects per video. The boxplot whiskers which are quite wide (the lower and upper extreme) show that for some videos, e.g., Star Wars, the subjects do not have similar viewing behaviours, while videos with narrow whiskers, can explain the viewing behaviour based on the video content. This largely depends on the content and the subject that is watching the content.

To get an approximation of how dynamic the video might be, the mean score of all subjects for that specific video can indicate some viewing behaviour. However, for similar dynamicity scores for each video, the subjects do not have the same viewing behaviour. This was checked by a pairwise comparison between the rankings of subjects per video. These comparisons were done across the three most dynamic and three least dynamic videos according to their mean dynamicity scores. The results showed that each subject had a different viewing behaviour relative to other subjects.

5. Discussion

During this research, we experienced some setbacks, but mostly these setbacks consisted of research that has not been carried out yet, or research that we did not have time enough to carry out ourselves. An important setback is that there was not enough time to try out more datasets. E.g., the dataset by Wu et al. [12] contains extra information about the subject's age and previous VR experience. These setbacks also show that there is a lot of research left to be done in the area of NGs and VP for 360-degree video.

A good direction to go from this research is to improve on bias correction, which should adjust the scores based on the video length. Fortunately, the chosen dataset videos were not very long, the shortest video being 20 seconds, and the longest 45 seconds. This meant that the longest video did not have the highest mean of dynamicity score. Moreover, introducing dynamicity score for the video content itself, and correlating it with the dynamicity score of the subject, may show interesting results and it may help to categorise the dynamicity scores. Further research should prove how this proposed $dynamicity_{v2}$ scoring is applicable and whether it is a valid way of scoring dynamicity.

Overall, this research progresses the field of VP through NG and opens a new area to continue research on how NGs could be used to conclude about viewers and the videos they watch. Besides that, we have published all the code that is necessary to generate SU and CU NGs.

6. Conclusion

360-Degree video has a bright future, but due to the BW constraints of current streaming solutions, popularity of the media format has stagnated. By characterizing NGs, we progress the field of VP, which in turn allows for a reduction in BW usage. We found that viewing behaviour can tell us something about the content of the video itself. Moreover, the content can also predict to some level how subjects are going to watch a video. Our dynamicity score is at the centre of these conclusions. Future work could include reviewing the dynamicity score and determining how this characterization can reduce BW.

References

[1] B. Biggs, *YouTube Blog : 1080p HD Is Coming to YouTube,* YouTube Official Blog, 2009.

[2] K. Wilms, *4K live streaming: Live has never looked so good,* YouTube Official Blog, 2016.

[3] Y. Bao, H. Wu, T. Zhang, A. A. Ramli and X. Liu, "Shooting a moving target: Motion-prediction-based transmission for 360-degree videos," *Proceedings - 2016 IEEE International Conference on Big Data,* pp. 1161-1170, 2016.

[4] M. Yu, H. Lakshman and B. Girod, "A framework to evaluate omnidirectional video coding schemes," *Proceedings of the 2015 IEEE International Symposium on Mixed and Augmented Reality,* pp. 31-36, 2015.

[5] Netflix, *Internet Connection Speed Recommendations,* Netflix.

[6] F. Qian, B. Han, Q. Xiao and V. Gopalakrishnan, "Flare: Practical viewport-adaptive 360-degree video streaming for mobile devices," *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM,* pp. 99-114, 2018.

[7] P. Jounsup and K. Nahrstedt, "Navigation graph for tiled media streaming," *Proceedings of the 27th ACM International Conference on Multimedia,* pp. 447-455, 2019.

[8] A. Nguyen, Z. Yan and K. Nahrstedt, "Your attention is unique: Detecting 360-degree video saliency in head-mounted display for head movement prediction," *Proceedings of the 2018 ACM Multimedia Conference,* pp. 1190-1198, 2018.

[9] Y. Zhang, P. Zhao, K. Bian, Y. Liu, L. Song and X. Li, "DRL360: 360-degree Video Streaming with Deep Reinforcement Learning," *Proceedings - IEEE INFOCOM,* Vols. 2019-April, pp. 1252-1260, 2019.

[10] T. van der Kuil and R. Zubavicius, *State of the Art of Viewport Prediction,* 2020.

[11] M. Xu, C. Li, S. Zhang and P. L. Callet, "State-of-the-Art in 360° Video/Image Processing: Perception, Assessment and Compression," *IEEE Journal of Selected Topics in Signal Processing,* vol. 14, pp. 5-26, 2020.

[12] C. Wu, Z. Tan, Z. Wang and S. Yang, "A Dataset for Exploring User Behaviors in VR Spherical Video Streaming," *Proceedings of the 8th ACM on Multimedia Systems Conference,* pp. 193-198, 2017.

[13] M. Xu, C. Li, Y. Liu, X. Deng and J. Lu, "A subjective visual quality assessment method of panoramic videos," *2017 IEEE International Conference on Multimedia and Expo (ICME),* pp. 517-522, 2017.

[14] Z. M. Boyd, N. Fraiman, J. L. Marzuola, P. J. Mucha, B. Osting and J. Weare, *A metric on directed graphs and Markov chains based on hitting probabilities,* 2020.

[15] B. Ribeiro, P. Wang, F. Murai and D. Towsley, "Sampling directed graphs with random walks," *Proceedings - IEEE INFOCOM,* pp. 1692-1700, 2012.

[16] D. Sashika, *Measuring Graph Similarity Using Neighbor Matching,* 2014.

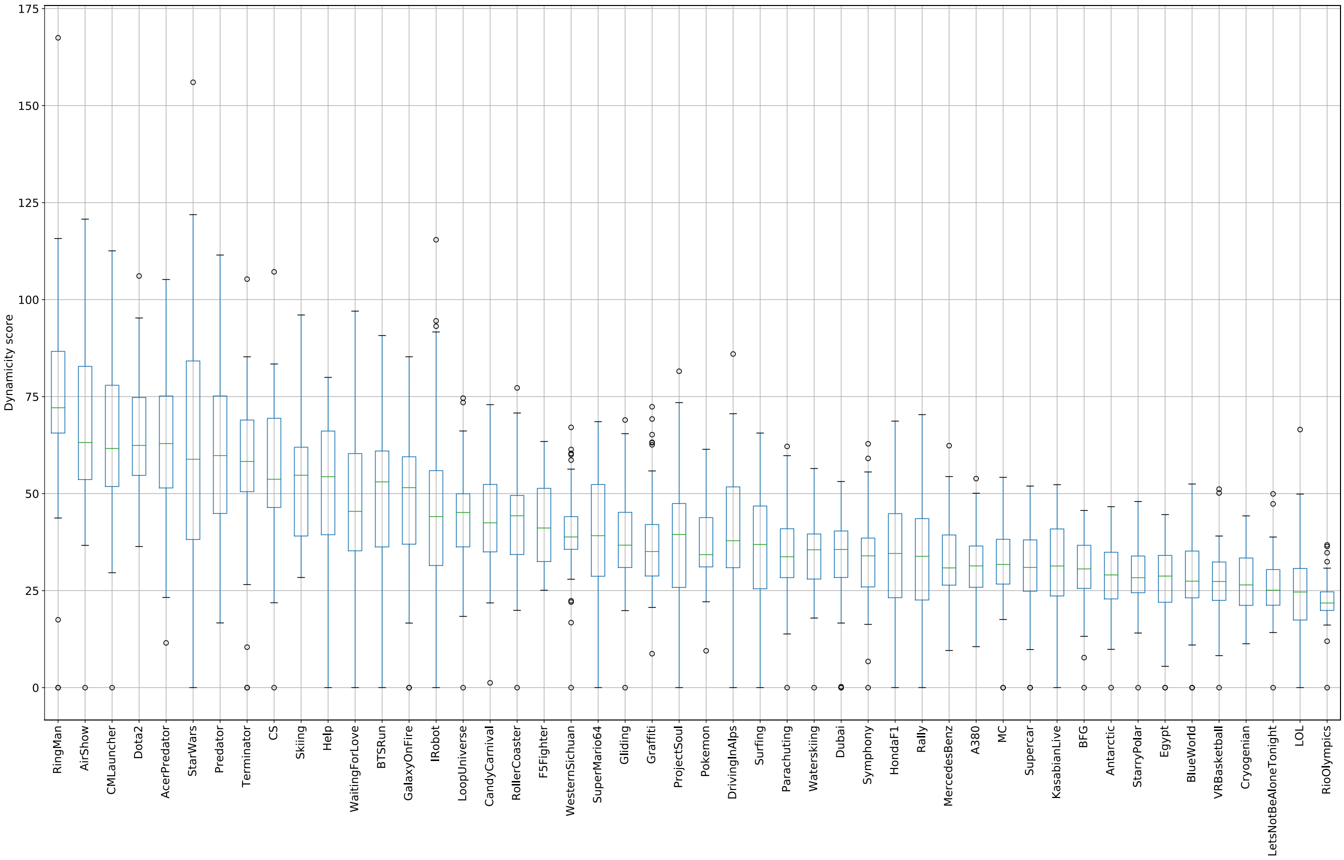

Figure 5.; *Boxplot that compares the dynamicity scores per video*