

Characterizing navigation graphs for 360-degree videos.

Timo 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 reach new heights. There are numerous 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 based on the likelihood of switching from one view to another. This work aims to progress the field of VP through NGs by attempting to characterize the NGs and determine what NGs say about the content and vice versa. This resulted in interesting insights in NGs and VP like: ADD EXAMPLES OF RESULTS HERE

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 BW requirements of 360-degree is vital for the success of this medium. The previously mentioned improvements, better encoding, for example, cannot achieve the necessary reduction. 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. The part of the 360-degree video that is visible to the user is called their viewport. The technique of predicting the viewport is called Viewport Prediction (VP). There are various approaches to predict the viewport, from regression methods to neural networks [6], [7], [8], [9]. NGs seem to have the most possibilities as this is a novel approach to VP [10]. To allow for improvements and optimizations to VP through NGs, it is vital to understand the following three parts:

* What datasets to use
* How to generate the NGs
* What can be said about these NGs

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, three sub-questions will be answered during this research.

**Research Question 1:** *What datasets are usable 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?*

Section II introduces a brief version of NGs. After that, the methodology for answering the RQs and their results are discussed. Lastly, the main research question will be answered.

# 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. To be able to understand the theory behind NGs, some terms used in this section need explanation. Use Figure 1 as a visualization of the terms that are explained in this section.

Chart

Description automatically generated

**Figure 1.;** Segments, Tiles, Viewports, and Views adapted from [7]

A video consists of several segments and tiles. Segments are temporal sections of the video with a fixed duration that is usually between 1 and 15 seconds. Suppose seconds and a video is 60 seconds: there are = 4 segments of 15 seconds. Tiles are spatial sections of the video and form a grid, for example. Suppose the grid is indeed : there are equal sized tiles in the video. Combining the 4 segments (temporal) calculated earlier, there are unique tiles (spatial) in a 60 second video with and a 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 , or: the union of viewports within a segment with duration . A viewport is the set of visible tiles at time [7].

A CU NG is defined as

The vertices of the NG are defined as

where is the view, and a tuple of the segment index and a set of visible tiles . is the segment index, is the total number of segments in a video, is the set of visible tiles, and is the set of all possible combinations of tiles in a video.

The edges of the NG are defined as

that connects the vertices with the weight . are vertices that have an edge between them, is the set of all vertices of this graph, and is the weight function defined as which is the probability of transitioning from to [7].

An SU NG has only one difference when compared to a CU NG: instead of the vertex being a tuple of and , the SU vertex is defined as

# 3. Methodology

Figure 2 shows an overview of the steps taken in this research. Detailed descriptions are found below the figure.

Diagram

Description automatically generated

**Figure 2.;** Steps in this research

## 3.1. Research question 1

*What datasets are usable to generate NG?*

RQ1 considers only step 1 from Figure 2. The data used within this research was found in a survey paper by Xu et al. [11]. Out of the list of possible datasets, two seemed the most promising. Table 1 compares the two datasets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dataset | Subjects | Videos | Data Format | Notes |
| Wu et al. [12] | 48 | 18 | X/Y/Z and unit quaternion | Also includes age of subjects and prior experience with VR |
| 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]. This decision is made in order to minimize the time spent on generating the NGs and maximizing the time spend on actually analyzing them. The simplicity of VR-HM48 is due to the data format: longitude and latitude. As 360-degree video is often projected using equirectangular mapping, a 2D coordinate system is perfect. The centre of the screen is and ranges horizontally from to and vertically from to .

## 3.2. Research question 2

*How to generate NGs from large datasets?*

RQ2 considers all the steps from step 2 to step 8 from Figure 2. As Figure 2 shows, there are quite a few steps necessary to create an NG. 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 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**

In order to visualize the data on the video, X/Y coordinates are necessary. These are computed using the following conversion:

This makes visualization simpler and allows for visual confirmation of the statistics found at later stages.

**(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 II, equations 2 and 3.

**(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 II, equations 3 and 4.

**(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.

|  |  |  |  |
| --- | --- | --- | --- |
|  | 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 watching 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 watching behaviour | Dynamicity score |
| 3 | Does dynamicity stem from video or from 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

## 3.3. Research question 3

In order to analyze 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, there are various metrics that are of interest for characterizing NGs. Table 2 shows the characterizations and accompanying metric that will be used in this research.

The focus will be on the dynamicity score, a score that represents how dynamic a subject's watching behaviour is. One way to do this is by making use of the average degree of the NG. This is calculated as follows:

where is the sum of all in- and outdegrees of every vertex. Another way of determining the dynamicity score is by counting all visits to a tile, and order them with the highest tile count in the center, and increasingly less tile counts to the left and right of it. This results in a normal distribution of which the variance σ can be determined. The higher σ is, the more tiles with high/similar counts have been visited which could mean the watching behaviour is dynamic. To get a dynamicity score from this σ, the following function is used:

# 4. Results

**Here we want to elaborate on the actual results that will answer the RQs**

## 4.1. Research question 1

*What datasets are usable to generate NG?*

Any dataset that can output the set of visible tiles at a given moment is usable to generate NGs.

## 4.2. Research question 2

*How to generate NGs from large datasets?*

A SU NG can be seen in Figure \ref{fig:NG}. The nodes represent the set of visible tiles, and the edges are the transitions with their probabilities, where darker means more likely.

## 4.3. Research question 3

*How to characterize the generated NGs using statistics and visualization?*

**Here we will discuss the results of indegree/outdegree/entropy etc.**

**Here we will add the graphs produced by the characterization of the NGs**

# 5. Discussion

# 6. Conclusion

# 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. |