Improving User-Perceived Quality: Tile-based VR Adaptive Streaming using JND Model

Your N. Here
Your Institution

Second Name
Second Institution

Abstract

1 Introduction

In recent years, the world is becoming more virtual than we ever thought it would be. Many video service providers, such as YouTube, roll out Virtual Reality (VR) videos which provide immersive experience to users. While consuming VR videos, users can change their viewpoint, resulting in an interactive experience than consuming traditional videos with a fixed viewing direction. However, VR videos' high demand of resolution and bitrates hinder their wide spread over the Internet. How to provide high user-perceived quality with limited bandwidth becomes the biggest problem in VR video streaming.

Viewpoint-adaptive streaming is regarded as a promising way to solve the problem. It assumes that user-perceived quality of each part of a frame depends on its eccentricity to user viewpoint. The most common realization is tile-based streaming framework. In tile-based streaming, each temporal video segment is composed by several spatial tiles which can be independently encoded/decoded. User chooses high bit-rate for near-viewpoint tiles, and chooses low bit-rate for far-viewpoint tiles.

However, viewpoint-adaptive streaming is a very coarse approximation of user-perceived quality, since user-perceived quality is not only related to viewpoint-content eccentricity. Many prior works have proved that it is at least also related to luminance[], texture complexity[], temporal fluctuation[].

So we evaluate the potential improvement of considering these factors into VR video streaming. With consideration of luminance, texture complexity, temporal fluctuation, we build a Just-Noticeable-Distortion (JND) model, which is a promising model to evaluate user-perceived quality. Result shows that we can save x% bandwidth without decrease of user perceived quality. So there is big room improve VR video streaming performance by perceived quality computing.

However, it still can not solve all problems because perceived quality computing in VR video streaming is challenging in two aspects:

- *User experience of VR video display is quite different from that of traditional non-VR video display in many aspects*. For example, in a VR video display, user has 360-degree view, immersive experience, content Depth of Field (DoF) awareness and active viewpoint. So JND model for VR display may be different from non-VR video display. Moreover, there might be some new JND factors which are never considered in traditional non-VR video display.
- *Computing JND during VR video streaming is challenging.* Since server has information only about video contents, client-side bit-rate adaptation logic has information only about user viewpoint, neither of them can compute JND alone. How to make combination of these two informations and make bit-rate decision for each tile is a problem.

In this work, we address these challenges and present the design and implementation of a Tile-based VR Adaptive Streaming using JND Model (TAS-JND). TAS-JND is built on two key insights:

- *User-perceived quality in VR video display is very different from that in traditional non-VR video display.* On the one hand, according to our experiments, current well-studied JND factors may influence perceived quality differently in VR video and non-VR video. On the other hand, perceived quality in VR video is also highly related to several new factors: viewpoint moving speed, content Depth of Field and light / dark adaptation. These additional VR-only factors can further save us x% bandwidth without decreasing user perceived quality.
- *JND computing can be decoupled.* Although the relationship between different JND factors is complex, JND

computing can be decoupled into server-side and client-side. All JND factors about video content (luminance, texture complexity, inter-frame fluctuation, Depth-of-Field) can be considered by server through video precoding. After that, user can compute JND and select bit-rate of each tile only based on user behavior (viewpoint position, viewpoint moving speed, light / dark adaptation). Experiments shows that decoupled JND computing is a very good approximation of perfect JND computing.

Taken together, these insights enable us to engineer a Tile-based VR Adaptive Streaming using JND Model. Firstly, we explore factors which influence perceived quality in VR display. Then we build a JND model with consideration of luminance, texture complexity, viewpoint, viewpoint moving speed, content Depth of Field and light / dark adaptation. Finally, a novel tile-based rate adaptation logic is presented to select bit-rate for each tile based on this JND model.

We implemented a prototype of TAS-JND and and integrated it in a VR video streaming system. (briefly describe our evaluation)

Contributions and Roadmap:

- Identifying key factors which influence user-perceived quality in VR display and build a JND model based on them. (§2-3)
- Presenting a tile-based VR adaptive streaming using JND model. (§4-5)
- Real-world evaluation that demonstrates substantial performance improvement by TAS-JND (§6).

2 User-Perceived Quality in VR Video Display

In the field of psychology and biology, it is widely accepted that in a video display, user-perceived quality is very different from original video quality. It is mainly caused but human visual system [] and human brain signal processing system []. However, what exactly influence perceived quality and how are still open problems.

There are many prior works about user-perceived quality in video display. As proved by [], user-perceived quality is at least related to several factors, such as content lightness, contrast, distance to viewpoint,

However, VR video displays differ from traditional non-VR video display in many aspects, such as 360-degree view, immersive experience, content Depth of Field (DoF) awareness and active viewpoint. So traditional factors may influence perceived quality in VR display differently, and there might also be some VR-only factors which are never considered in traditional non-VR video display. So far there is no prior works about this, we aim to explore which factors may influence user-perceived quality in VR display.

Since user-perceived quality is a subjective thing, how to evaluate it becomes a problem. For example, it is very hard for a user to say the perceived quality of video A is 2 times better than that of video B, or it is just 1.7 times better. Fortunately, Just Noticeable Distortion (JND) is a well-studied model to evaluate perceived quality. (introduce JND in several sentence) It is widely used in video coding.

In section 2.1 we briefly introduce some JND factors which have been well-studied by prior works. In section 2.2 we present several VR-only JND factors along with our user studies.

2.1 Overview of JND factors in traditional video display

In traditional non-VR video display, there are 2 types of factors which influence JND: spatial JND factors and temporal JND factors.

Spatial JND factors includes luminance masking and texture masking.

Temporal JND factors mainly refers to inter-frame fluctuation.

2.2 VR-only JND factors

2.2.1 Eccentricity and visual threshold

Briefly introduce eccentricity. (a figure)

Eccentricity is widely considered in traditional JND model. However, in VR display, eccentricity influences JND differently in 3 aspect:

1. Viewpoint awareness. In traditional non-VR display, video provider can obtain a viewpoint heat map from historical video display. However, according to [], more than x% video frames have unless 2 different viewpoint clusters. In VR video display, the exact user viewpoint can be predicted in high accuracy.

2. Wider ranged eccentricity. At present, all eccentricity-JND models are foveal models. In non-VR JND model, user is often assume to watch video in a distance of 4-times screen height. So the screen covers only a small part of viewport. Eccentricity of video content is at most around 20 degree. However, in VR display, user has an immersive experience. The eccentricity can reach nearly 90 degree. It is necessary to build eccentricity-JND model in 20 degree to 90 degree.

3. Tile-screen mapping. Each tile is unevenly mapped onto HMD screen.

Our experiment.

2.2.2 Vergence-accommodation conflict and visual threshold

Briefly introduce vergence distance and accommodation distance. (A figure is needed)

In non-VR video displays, the screen is bioptic, where only a single display is presented that is viewed by both eyes, so contents have no Depth of Field (DoF) information, vergence distance and accommodation distance are consistence. However, in most VR video displays, the screen is stereoscopic, where the illusion of depth is created by delivering images rendered from different angles to each eye. In this situation, accommodation distance is fixed but vergence distance is different for different contents, thus causes vergence-accommodation conflict.

Some prior works point out vergence-accommodation conflict can cause decrease of visual acuity. So we set up an experiment to measure the relationship between vergence-accommodation conflict and visual threshold.

Our experiment.

2.2.3 Light/Dark adaptation and visual threshold

Briefly introduce light/dark adaptation.

In non-VR video displays, the environment illumination totally depends on the real environment (e.g. under the sunlight, or in a classroom with electric lamp, or in a dark room). However, VR video displays are very different. When user wears HMD, the environment illumination totally depends on the content itself. So when the illumination changes dramatically, eyes need a period of time to adapt the new illumination.

Some prior works point out in the process of light/dark adaptation, visual acuity decreases. So we set up an experiment to measure the relationship between light/dark adaptation and visual threshold.

Our experiment.

2.2.4 Viewpoint moving speed and visual threshold

One of the most highlighted feature of VR video is that users can freely move their viewpoints. According to our data analysis, more than x% viewpoints are moving faster than y deg/s. (figure of our data analysis)

Some prior works point out when users viewpoint is moving, visual acuity decreases dramatically. So we set up an experiment to measure the relationship between viewpoint moving speed and visual threshold.

Our experiment.

3 JND Model

In this section we present our JND Model base on above mentioned insights.

3.1 Classification of JND factors

In the last section, we present a brief overview of JND factors which are widely used in traditional video displays. Then we propose 4 new JND factors which only exist in VR video display. According to information needed to compute them, we classify all above JND factors into 2 classes:

Content-related JND:

User-related JND:

Fig. X shows the classification of all JND factors mentioned in this paper.

(figure)

3.2 JND Modeling

Although we have knowledge of how single JND factor influence the visual threshold, it is unknown that how different JND factors together influence the visual threshold. Since the relationship of them is very sophisticated, the most common approximation is that luminance and texture masking form a basic value of visual threshold, other JND factors are modeled as scale factors. The final visual threshold is defined as the product of basic visual threshold value and all scale factors. We also use this method to build our JND Model.

Fig. X shows the structure of JND Model. (a figure)

According to [], the basic value of visual threshold JND_0 is defined as:

According to [], the temporal scale factor F_temp is defined as:

According to Sec. 2.2.X, the Depth-of-Field scale factor F_{DoF} is defined as:

According to Sec. 2.2.X, the eccentricity scale factor F_{ecc} is defined as:

According to Sec. 2.2.X, the light / dark adaptation scale factor $F_{l/d}$ is defined as:

According to Sec. 2.2.X, the viewpoint moving speed scale factor F_{spd} is defined as:

4 Tile-based VR Adaptive Streaming using JND Model

At present, there are 2 mainstream streaming mode for VR videos: quality-driven streaming and bandwidth-driven streaming.

Quality-driven streaming:

Bandwidth-driven streaming:

In TAS-JND, we also design above 2 streaming mode. Now we present our methods of applying JND Model into these 2 streaming modes.

4.1 Quality-driven streaming

In this situation, perceived video quality is chosen by client. Adaptive Streaming logic needs to allocate bitrates of each tile each segment and minimize the bandwidth cost, while keeping user perceived quality unchanged.

Obviously, given the required perceived video quality, the optimal solution is choosing the bit-rate version which distortion is just under the visual threshold.

Rate Decision Algorithm 1 (quality-driven)

4.2 Bandwidth-driven streaming

In this situation, VR video is streaming in a constraint bandwidth. Adaptive Streaming logic needs to adjust the video bit-rate based on current bandwidth estimation, while trying to maximizing the perceived quality.

According to definition of JND model, it is known that when actual video distortion doesn't exceed visual threshold, user will not detect the difference. However, if the bandwidth is very limited and actual video distortion exceeds the visual threshold, how much distortion will be detected by user? [] gives a empirical function.

Detected Distortion function:

In order to maximize perceived quality, we need to minimize the detected distortion. So we have the following optimization problem.

optimization problem:

Rate Decision Algorithm 2 (bandwidth-driven)

5 System Design

Challenge 1: Incomplete information for computing JND in either client-side or server-side.

Challenge 2: Granularity of tile separation remains unknown.

In this section we first introduce our JND model decoupling method to solve the first challenge, and then we analyze the performance of different granularity of tile separation. Finally we present the system overview.

5.1 JND model decoupling

5.2 Granularity of Tile separation

Fine-grained tile separation can make JND computing accurate, thus we can allocate bit-rate correct. However, too fine-grained tile separation easily causes huge workload of video coding / decoding.

Coarse-grained tile separation is more friendly to video coding / decoding. But when contents with significantly different JND occur in one tile (a part of contents with low JND and a part of contents with high JND), we can only obtain a overall value of visual threshold and allocate a unique bit-rate. In this situation, surplus bit-rate is allocated to high-JND part and insufficient bit-rate is allocated to low-JND part.

In this section we present our experiments to make comparison between different-granularity tile separation.

We choose a video and make several tile-separated versions: 3*6 tiles, 6*12 tiles, 12*24 tiles and 24*48 tiles. We assume that in 24*48 version, JND computing is fully correct.

Figure X shows the function of tile separation granularity to notable distortion.

Figure X shows the function of tile separation granularity of overall bit-rate.

Based on above result, we choose x*y tiles version because it can obtain low notable distortion while keep overload of video coding / decoding acceptable.

5.3 System Overview

TAS-JND is build completely on client-side. There are 3 components: Viewpoint predicting module, JND computing module and bit-rate selection module, as shown in Fig. X.

(Figure)

In Viewpoint predicting module, we choose the method of ..., which can obtain ... accuracy.

Given the predicted viewpoint, JND computing module computes the scale factor of each tile.

After finishing JND computing, bit-rate selection module chooses proper bit-rate for each tile based on algorithm 1 or algorithm 2.

6 Performance Evaluation

To evaluate the performance of TAS-JND, we carry out extensive real-world Internet experiments with real VR video display.

6.1 Setup

Fig. X shows the network topology in the experiment, which consists of a client and a server.

In the experiments, we choose three videos in different types from the VR dataset. The information of these video is shown in Table 1. We set the duration of one video segment as 1 second. We adopt the 6 * 12 (rows * columns) tiling pattern for each video segment, thus the number of tiles is 72 (N = 72). To generate different quality videos, we use quantization parameter (QP) ranging from 22 to 42 in steps of five leading to five different bitrate versions.

In our experiment, 5 different streaming methods are chosen to make comparison:

MONO: This approach is monolithic streaming. The naive way by streaming the entire 360-degree scene in constant QP without exploiting and optimizing the quality for the users viewport.

JND: This approach applies a traditional non-VR JND model (with consideration of content luminance, contrast and viewpoint heat map) to video coding.

TAS-Viewport: Viewpoint prediction method is applied on client-side. For tiles in user's viewport, high quality is chosen. For other tiles, low quality is chosen.

TAS-JND-: Viewpoint prediction method is applied on client-side. For tiles in user's viewport, a traditional non-VR JND model is applied to chose their quality. For other tiles, low quality is chosen.

TAS-JND: Viewpoint prediction method is applied on client-side. For tiles in user's viewport, our proposed JND model is applied to chose their quality. For other tiles, low quality is chosen. This is the method presented in this paper.

6.2 Performance Comparison

6.2.1 Quality-driven streaming

In quality-driven streaming, perceived video quality is chosen by client. Adaptive Streaming logic needs to allocate bitrates of each tile each segment and minimize the bandwidth cost, while keeping user perceived quality unchanged.

Bandwidth comparison is shown in Fig. X.

Stalling comparison is shown in Fig. X.

In order to prove that different methods provide the same quality, a scoring system is designed based on real user.

6.2.2 Bandwidth-driven streaming

In the real-world video streaming, VR video is streaming in a constraint bandwidth. Adaptive Streaming logic needs to adjust the video bit-rate based on current bandwidth estimation, while trying to maximizing the perceived quality.

Quality comparison is also made by our scoring system on real user. The result is shown in Fig. X.

6.3 Insights from VR video display

In order to clarify in which situation our proposed TAS-JND has substantial improvement and in which situation its has marginal improvement, we choose one of the VR video display and plot the sequence diagram in Fig. X. The displayed content is a skiing video. We highlight 5 frames to analyze the performance of 5 methods.

Frame A:
Frame B:
Frame C:
Frame D:
Frame E:
MONO performs well when ... .
JND
TAS-Viewport
TAS-JND-
TAS-JND

7 Related works

7.1 VR video display

7.2 Perceived quality and JND

7.3 Adaptive Streaming