A

Dissertation On (Major Project-II)

"FOV PREDICTION FOR 360 VIDEO STREAMING IN VIRTUAL REALITY"

Submitted in Partial Fulfillment of the Requirement
For the Award of Degree of

Master of Technology

In

Software Technology

By

Abhinav Purwar
University Roll No. 2K15/SWT/502

Under the Esteemed Guidance of

Prof. Manoj Kumar
Department of Computer Science & Engineering

2015-2019(Jan)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
DELHI – 110042, INDIA

STUDENT UNDERTAKING

Delhi Technological University
(Government of Delhi NCR)
Bawana Road, Delhi- 110042

This is to certify that the thesis entitled**"FOV PREDICTION FOR 360 VIDEO STREAMING IN VIRTUAL REALITY"**done by me for the Major project-II for the achievement of**Master of Technology** Degree in **Software Technology** in the **Department of Computer Science & Engineering**, Delhi Technological University, Delhi is an authentic work carried out by me under the guidance of Prof. Manoj Kumar.

Signature:
Student Name
Abhinav Purwar
2K15/SWT/502

Above Statement given by Student is Correct.

Project Guide:

Prof. Manoj Kumar
Department of Computer Science & Engineering, DTU

ACKNOWLEDGEMENT

I would like to express sincere thanks and respect towards my guide**Prof. Manoj Kumar, Department of Computer Science & Engineering, Delhi Technological University Delhi.**

I consider myself very fortunate to get the opportunity for work with him and for the guidance I have received from him, while working on this project. Without his support and timely guidance, the completion of the project would have seemed a far. Special thanks for not only providing me necessary project information but also teaching the proper style and techniques of documentation and presentation.

Abhinav Purwar
M.Tech(SoftwareTechnology)
2K15/SWT/502

ABSTRACT

In the transmission and review of 360 degree videos on VR devices (HMDs) initiate numerous specialized difficulties. 360 degree video are omnidirectional spherical video and on an approximate 4~6 times high resolution then normal 2D videos of same quality. Since 360-degree video are round video so having omnidirectional perspective of the scene. However human having Field of View (FOV) approximately 90 ~114 degree so only 1/4th of the part is viewed at time. Moreover, VR devices require to respond within 11ms to head movements of user to show the next Field of View (FOV).

In this task I research the issue of foreseeing the Field-of-Views (FOV) of client's viewing 360° video utilizing VR gadgets ahead of time. Existing and mostly present solutions while watching the 360 video first find out the current orientation and FOV of user's and then request for high quality of video pertaining to FOV region and rest for low quality which induce bad user experience for some time called as latency to adapt the high quality.

I am developing solution to predict FOV in advance that simultaneouslyinclude content related historical data and sensor data to predict the viewer's Field of View(FOV) in advance. The sensor-related features include VR devices orientation sensor like magnetometer and accelerometer, while the historical data include the complete data of previous watching orientation of video by different users.

Based on historical data and sensor data, I shall train the system and validate design alternatives. Which will help in identify the better design to reduce the view latency and bandwidth required for 360 Video.

Advantages of this solution will be (i) Lower bandwidth usage (ii) Low latency and (iii) short running time.

TABLE OF CONTENTS

CHAPTER 5:

Results and Analysis

CHAPTER 6:

Conclusion

Chapter 1: Introduction

With the evolution of cheap, fast internet and smart devices mostly people starts sharing their experience and events of life with relatives,companions and publics over online social media whenever, anyplace. Among the content datashared in online social gatherings, video contents sharing is the most prominent and generally utilized contents. As of late, 360° videos are increasing greater ubiquity then flat videos, because they preserveImmersive experience, also allowsusers to better share theirexperience and life events.

When we take a gander at the development projection of the VR area and industry in the course of recent years, it appears to be just incredible things lie ahead and its future looks splendid. For one, only a couple of years back (in 2013), there were around 100 thousand active virtual reality users. This exapnded to 90 million dynamic users in 2017.

The whole VR industry is anticipated to be worth $5.2 billion of every 2018, and $45 billion by 2025 — up from a measly $90 million out of 2019.

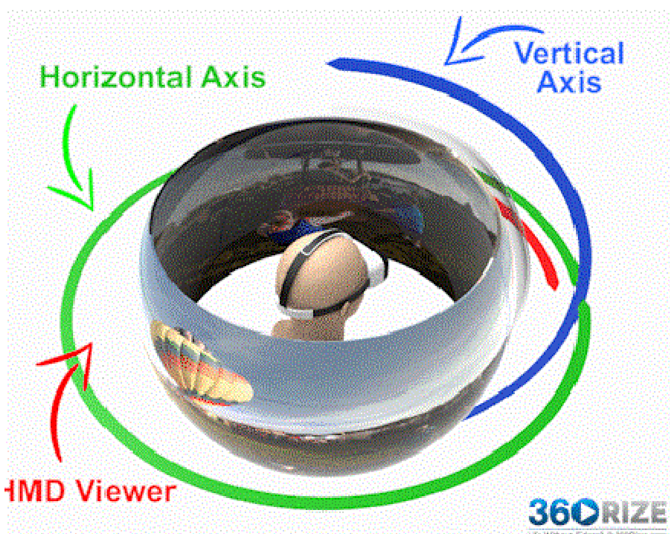


Fig.1 360 Video.

360° video is in extremely high resolution, with VR devices, each viewer would be able to see a small part of the wholevideo approx. $1/6^{th}$based on human stereoscopic vision which is 114 deg.Accordingly, sending the total 360° video in full resolutionmay prompt wastage of resources, for example processingpower,storage space and network bandwidth. Somostlyutilized approach to stream 360°videos to VR devices is to stream the present FOV of video in full resolution and rest part in low resolution. This solution is definitely better then above solution but still it waste the bandwidth ,resources because low resolution still not seen by user most of the time and only come into picture when user changes the FOV in that case user can see low resolution for some time until it adapts to high resolution which again degrade the user experience. So In this paper we come with better approach in which our system predicts the user's future FOV beforehand and cache it to buffer so that when user moves to new FOV we don't need to fetch or show the low resolution because we already predict and cached the new FOV in advance so we show them directly, thus this solution don't require to keep the complete 360 video in buffer like previous method. With this approach it saves the bandwidth and resources. Our new methodology makes structuring of a live 360° videostreaming system to VR devices quite complex and proficient, particularly because mostlyomnidirectional360 streaming methods currentlydeliver using tile based selective streams segment which last for few seconds only.

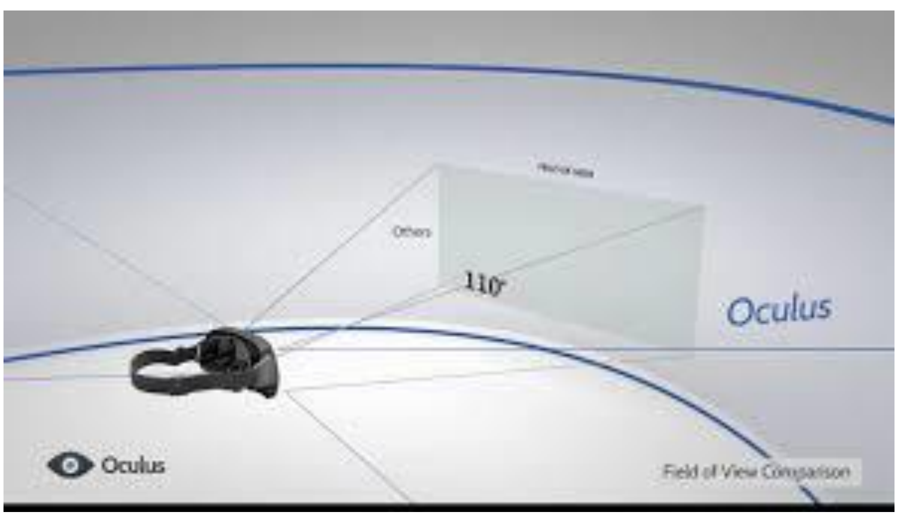


Fig 2: FOV in 360 video.

Albeit late research endeavor to anticipate the FOV ahead of time, they utilizes basic guess and presumption, which are less exact,and in this manner result in terrible user experience and wasted resources.

In this undertaking, we take care of the FOV expectation issue utilizing ML (machine learning) for higher precision.Procedure highlights of our FOV forecast systems is that we simultaneously consider both sensor and media contentrelated attribute. Media content attributeincorporateframe based gaze historical data and sensor'sattribute comefrom sensor present in VR devices which provides orientation and position of device at particular time.

Predicting FOV is not an easy task and thereafter showing the predicted FOV include FOV based transmission which again very complex part.

Chapter 2:Related Work

[1] Describes Adaptive 360 VR video streaming with immersive spherical video.
Explain about the 360 video over flat video and covers the adaptive streaming technique for transmission of 360 video also explains the problem exist in transmission of 360 video. Compare different techniques existed in market for 360 playback.

[2] Describe the FOV aware tile based adaptation in 360 virtual reality (VR) streaming. It also explains about the tiles based geometry in VR domain. Also include the tiling feature in MPEG DASH.

[3] Dash Streaming study and protocol understanding for applying it for VR streaming.
Dash streaming is adaptive streaming technology developed by MPEG. It allow high quality streaming video transmission over internet delivered by HTTP protocol. This is widely used protocol and also adapted by OMAF standard which is related to 360 video streaming playback.

[4] HEVC specification and understanding for tiling media.
Main purpose of HEVC standard to improve the compression quality and performance related to existing standard like h264 (AVC). This paper provide the knowledge on HEVC features and standard. Also provides the detail of tile mechanism in HEVC which is added to enable parallel processing in HEVC.
[5] HEVC overview
Provides the structure and architectural design of HEVC coding standard.
[6] Different Projection study for 360 video and their advantage and disadvantage.
Explain the major projection exist for 360 video which already covered in thesis with pros and cons of each projection.
[7] Fixation prediction, for predicting content related attribute and saliency map creation has been studied for static images.
This paper states the same problem as I covered in my project but using different techniques.

[8] Advance machine learning technologies with various supervised and non-supervised learning are accepted for better attributeprediction and extraction in fixation detection. Covers many types of machine learning techniques.

[9]Framework to propose to embrace the data of saliency guide and after that think about movements for dynamic saliency expectation.

[10] Perform obsession forecast in video utilizing highlights like movement vectors, thumbnail and navigation trajectory exploration.

HMDs and VR devices. Several VR devices are released by many companies and their applications has turned out to be progressively mainstream. Their execution has been estimated in the literature. Young et al. [12] execute subjective performance tests between two HMDs like searching sorting and provide opinionthat they do better for playing videos and supporting interactive tasks. Samaraweera et al. [13] study whether mobility impaired patients are less vulnerable to cyber sickness due to the latency. Chang et al. [13] given different structure to calculate the sensitivity and precision of HMDs. A few tradeoffs are seen on item HMDs in their investigation.

Chapter 3: Problem Statement

3.1 Description of Problem

When VR video is streamed over a network, the video data transfer capacity required to transmit VR video utilizing conventional strategies can be as high as 20-50 times the transmission capacity required by non-VR (also referred to as "flat" or "normal" video) videos, depending upon **chosen resolutions**. This is because unlike traditional single view cameras that only capture light from the front, 360-degree cameras capture light from all directions and usually in stereoscopic mode. This causes an explosion in the quantity of pixels captured and transmitted.

Fig-3: 360 degree video Capture

Existing 360 Video having very large size as compared to their 2D counterpart. Unfortunately a noteworthyissue is how to transmit such cumbersome 360 video in bandwidth constraint wireless gadgets such as mobile, VR devices in wireless network given their high bitrate requirements(50~80 Mbps).

Current 4G technology provide maximum 100 Mbps speed theoretically but in practical user get around 30~40 Mbps speed only in such network transferring high bitrate 360 video seems major issue.

Especially with the 4K 360 video resolution which is consider as practicalleast resolution standard for current VR devices, and 8K or higher being wanted, these new video standard are highly bandwidth intensive and hard to stream at desirable quality levels. Thus there must be a balance between the available resources and requirements of streaming on the VR devices.

One of the major challenges to achieving this balance is that we need to show the high quality to user without negative effect in user's viewing experience. However, VR is not supportive to currentcellular system where radio resources and data transmission is restricted. Likewise, downloading massive data continuouslyleads to cellular devices' power drain because cellular network is consider to beresource-hungry when in active use, the LTE radio networksrecords for at least half of the whole smartphone's energy utilization.

The presentation and conveyance of 360-degree videos on VR presents numerous specialized difficulties. 360 degree videos are bulky and ultra-high resolution round videos, which contain 360 degree view of the display**. While 360 video demands high bandwidth, at any time instant users are only viewing a small portion of the video according to the HMD field of view (FOV)**

3.2Hardware or Resource capability

To match the ultra-high quality of NON VR content, VR content require at least 12K resolution videos. To decode the 12K video device require 12K capable hardware decoder. Current devices having support of maximum 4K decoder so decoding of 12K content not possible with current available devices.

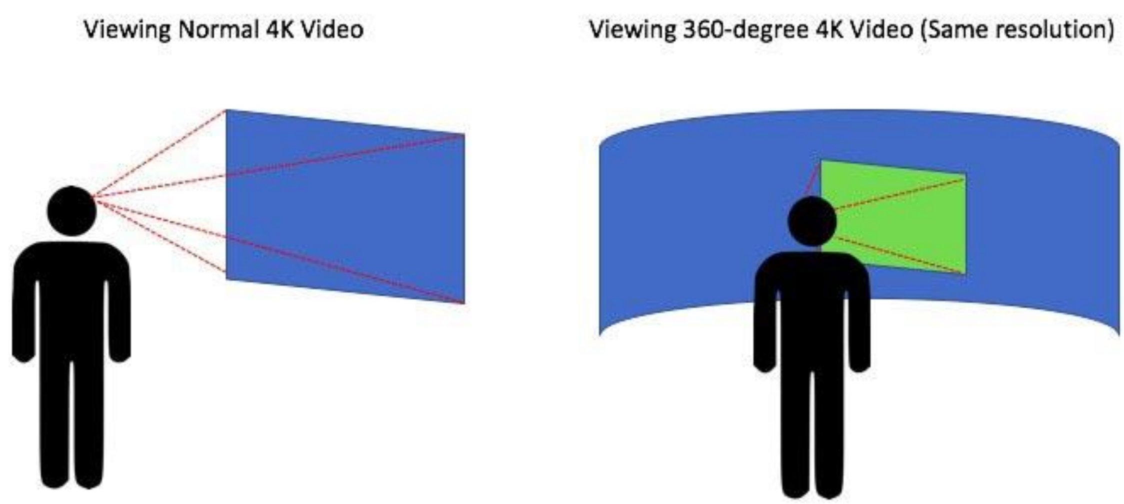


Fig-4: FOV for flat and 360 Video

4K of VR content equivalent to 1K of Non VR content which is much degraded quality in VR mode and having very bad user experience.

Currently some advance decoder (HEVC and VP9) coming with tiled based video concept. In which frames of video divided in multiple tiles and each can be played independently. Also they support the mixing of tile of different resolution and decode using single decoder.

In 360 video user can see up to 114 degree only as human stereoscopic vision have 110~120 degree field of view only. So to construct the 4K 360 frame with different resolution (12/8/4K) tiles is very tricky way which we adapt in our research and solution.

3.3Low latency adaptation

Mostly used VR solution having the concept of showing the high quality video in current FOV and rest part in Low quality.

While watching the 360 video in HMD,Ifauser moves his/her head outside of the current Field of View, then low resolution video is shown tothe user for sometimes, till the high resolution stream for new Field of view (FoV) from the server received to terminal and shown to the user. This approach havingtwo disadvantages:

(i) Changing between high and low resolutions leads to bad user experience

(ii) Transmittingcomplete 360 video streami.e. high quality at FoV and low quality for rest, still wastes precious resources. As Low quality video not seen by user at all.

Chapter 4: Research Methodology

4.1 Classification of work

We have divided our solution in various sub parts Application part, Server part and Prediction and Transmission part.

Application allow user to watch the video in 360 mode and provide way to watch anywhere in 360 view, also it capture the Field of View(FOV) of user while watching the video and transfer it to server for further processing.

Server part includes content creation for 360 video, transmission part & FOV data creation for each video in DB which further used for prediction.

Prediction and Transmission part includes training of system based on FOV data and to predict the future FOV while watching video. Transmission part includes how to transfer predicted part along with actual 360 frame. In below section we will cover every part in detail.

4.2 Application

360 degree video cannot be watch on normal video players. To show 360 degree video it require the video player's app having support for projection geometry. We need to first select the projection geometry in which video can be projected.

There are many types of projection geometry

- Cubical Projection
- Spherical Projection
- Pyramidal Projection
- Etc.

Cubical Projection

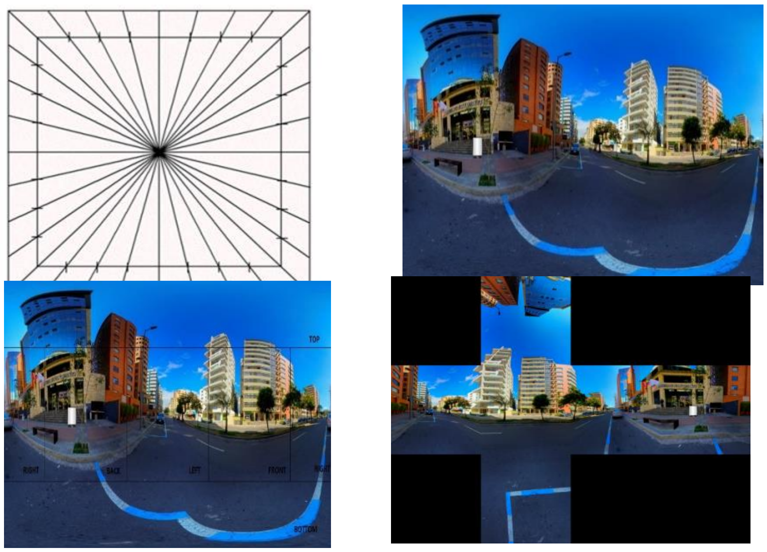


Fig-5: Cubical Projection

- *6 Faces i.e. Cubic Mapping*
- Usually some distortion observed while representing the environment as a cube because mapping from a sphere to cube is not perfect.
- Reflected vectors as shown in fig-5 are linearly interpolatedin cube mapping.
- All pixels are utilized as there is no overlapping since all the environment textures are square in shape and contain images of the environment.

Spherical Projection

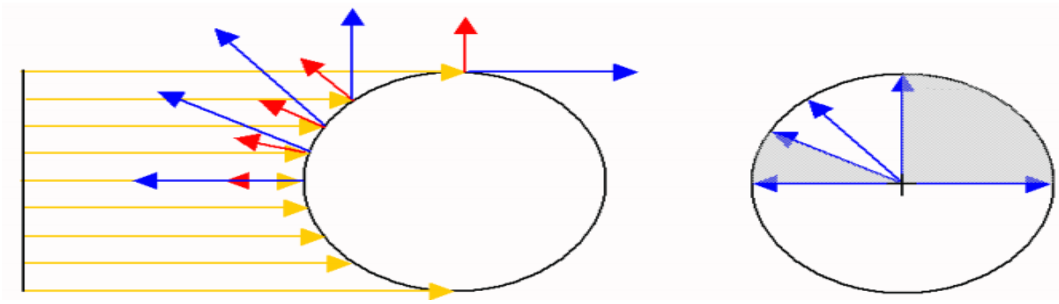


Fig-6: Sphere projection

- This is a latitude/longitude projection of the panoramic sphere.
- Standard for exchanging spherical panoramas between applications.
- Since the sphere map is a circle exist inside a square, so all of the map does not contains useful information. 21.46% of the map is wasted.
- Overrepresented at the poles; Singularity exists

Pyramid Projection

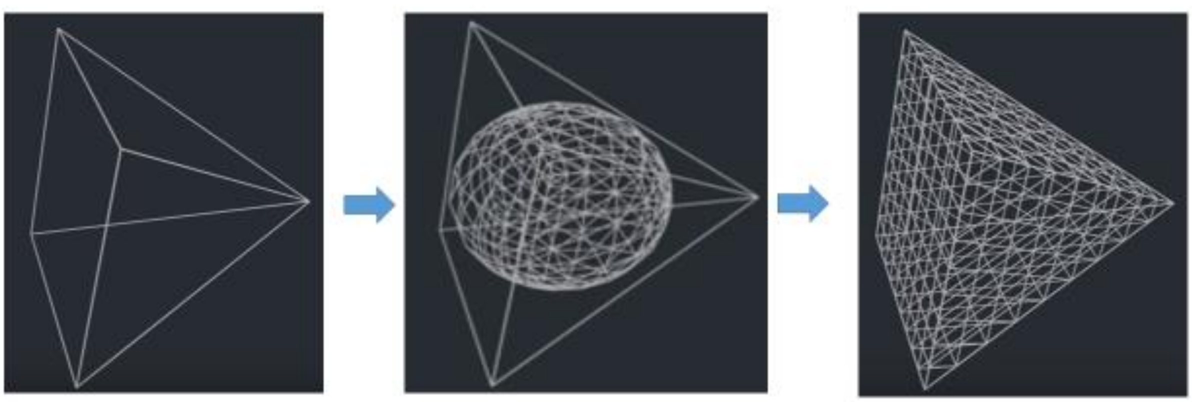


Fig-7: Pyramid Projection

- The spherical video is projected onto a pyramid layout from up to 30 central points to generate a set of video representations.
- Front face of the pyramid having the highest quality then the other faces.
- Distortion extremely high at the back and very low at the front.
- Based on user's viewport center, periodically select one of the representations in pyramid projection.

In our case we use **Spherical projection** to display the 360 degree video. We use Unity tool and Gear VR SDK for developing the application. GearVR SDK provide sensor API's which help in tracking the user's head movement. Unity tool is widely used for VR application which help in creating Screens and integrating many modules seamlessly.

Main work of Application to display the video in 360 degree view and allow user to watch video at any point in 360 mode. It collects the user orientation data from Gear VR device to predict the next look up position.

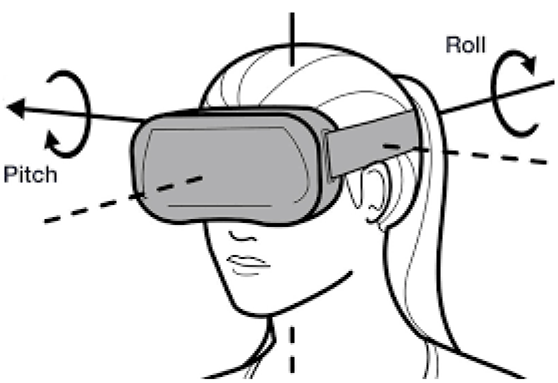


Fig-8: Rotation angle in VR.

When user start watching any video through GearVR using application. Application first setup connection with server where content actually exist and then open the database for particular video to maintain the new session of user for new FOV of that user. Application uses Oculus SDK for fetching the current gazing position of user while watching video for every frames of video and then store it in database of particular video .This data would be used later by Prediction module for training and predicting the FOV.

So application collect two data's first user movement from sensor and gaze point for each frame and then provide it to prediction module which uses both the data for predicting the FOV.

4.3 Server

Server part include the module related to content creation for 360 video, Server for content delivery, server for maintaining the FOV database for each 360 content.

Content Creation

In our project we modified the 360 content before delivery by dividing them in tiles of different quality. Division in tiles allow to construct the frame with different quality so in case of 360 degree video as user can not watch the whole video so we construct the frame with two qualities. 1) High quality tiles in FOV region 2) Low Quality tiles for all other region. Using this technique we saves transmission bandwidth as well resource at decoder side.

Image.

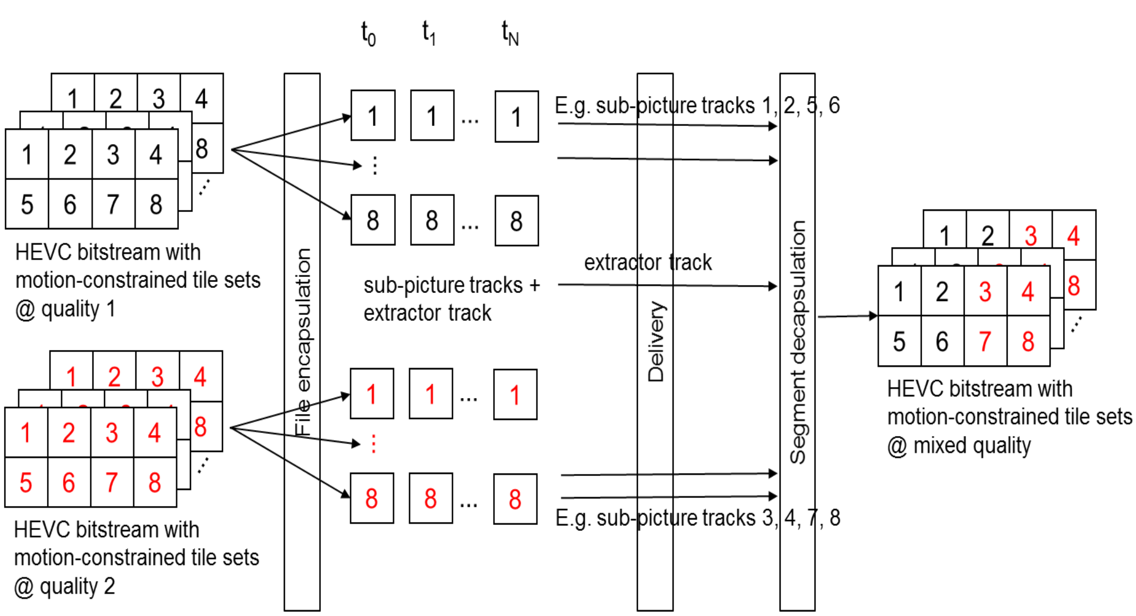


Fig-9: Content creation in 360 Video.

FOV database creation.

Client send the FOV for each user for every content looked by user, so there is PHP server which listen for request from client and capture all the information and save them in database with session wise for every frame of content. This data used later by prediction system to predict the most probable watching position in the frame by user.

Transmission of data.

Server created multiple connection with clients.

1) To send actual video content (high and Low quality)
2) Predicted data in high quality (2/3 second before the actual playback)
3) Fetching FOV data from client and adding value to database. Parallel.

Transmission is in HTTP protocol. We are using apache tomcat 2.0 and PHP 6.1 for this project.

4.4Database

We used Mysql database in this project for creating the dataset for prediction algorithm.

In this project we used very simple database architecture as we require to maintain the data for every content and FOV for user.

Please find the structure below.

ID	URL	NAME
1	http://suthep.mpd	SUTHEP
2	http://bridge.mpd	BRIDGE
3	http://heli.mpd	HELI

SUTHEP

ID	Session_ID	FOV	FOV_V	RESOLUTION
1	1	.3	.234	4K
2	1	.4	.345	4K

BRIDGE

ID	Session_ID	FOV	FOV_V	RESOLUTION
1	1	.33	.98	4K
2	1	.7	.88	4K

HELI

ID	Session_ID	FOV	FOV_V	RESOLUTION
1	1	.623	.76	4K
2	1	.456	.876	4K

LONDON

ID	Session_ID	FOV	FOV_V	RESOLUTION
1	1	.332	.234	4K
2	1	.412	.345	4K

Fig-10 :DB relation and structure

4.5Training and Prediction system

The gathered data in MySQL converted in to excel files for each table associated with each 360 degree video. Then this individual data corresponds to each excel send to prediction Module.

The gathered data was split into test and training data with 25% going to testing and 75%going to training. All the values were scaled between 0 and 1.

In our study following three models were applied:

1) **Random Forest** with 100 estimators with a minimum split of 2

Random Forest is an adaptable, simple to utilize machine learning calculation that produces, even without hyper-parameter tuning, an extraordinary outcome more often than not. It is likewise a standout amongst the most utilized calculations, since its straightforwardness and the way that it very well may be utilized for both characterization and relapse assignments.

Random Forest is a directed learning algorithm. Like we would as of now have the capacity to see from its name, it makes a forest and some way or another make it random. The “forest “it builds, is an outfit of Decision Trees, more often trained with the “bagging” technique. The common idea of the bagging techniqueis that a mix of learning models builds the general yield.

One major favorablenoteworthy pointof random forest is, that it is extremely wellutilized for both regression and classification problems, which outline the lion's share of present ML (machine learning) frameworks. I will discussabout this algorithm in classification technique, since classification is considered the main pillar of ML (machine learning) frameworks. Just take a look of below diagram of two trees to understand the random forest algorithm.

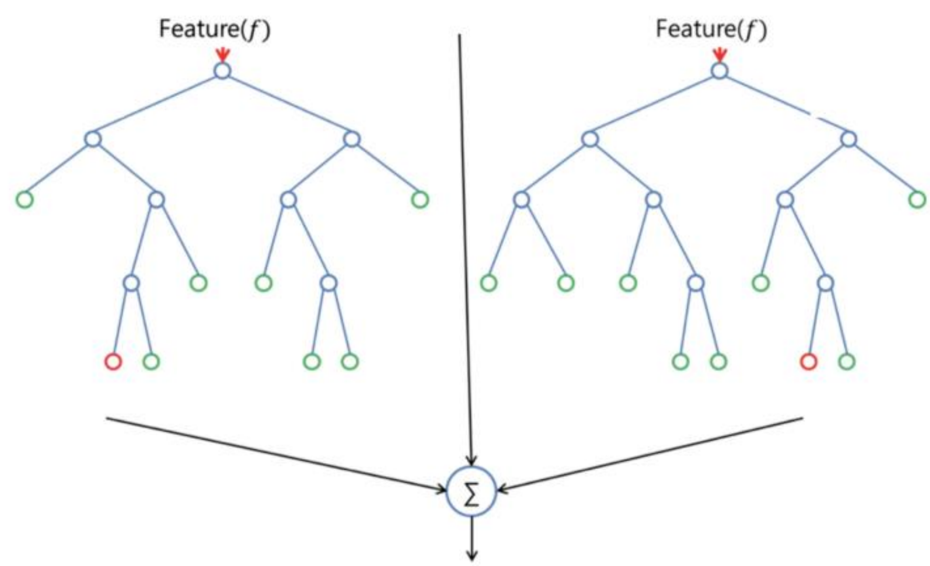


Fig-11: Random Forest

Random Forest has almost theequivalent hyper parameters as a bagging classifier or a decision tree. Luckily, we don't need to consolidate a decision tree with a bagging classifier and can just effectively utilize the classifier-class of Random Forest. Like I previously stated, with Random Forest, we can likewise managewith Regression tasks by utilizing the Random Forest regressor.

Random Forest adds extra haphazardness to the model, while developing the trees. Rather than scanning and finding the most critical feature while dividing a node, we looks for the best feature among all the features present in random subset. This outcome in a wide difference and diversity that results in a superior model.

As per the above explanation, in Random Forest, just an arbitrary subset of the features is taken into consideration by the algorithm for dividing a node. We can even make trees dynamically, by consider the dynamic thresholds for each feature rather than scanning the ideal thresholds (such astypical decision tree does).

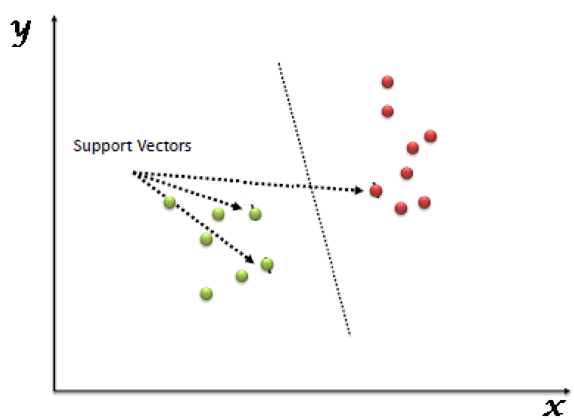
If we consider the performance parameter it is hard to beat Random Forestsin terms of performance. Of course we always try to find out the model that can do better than neural network, however this approach usually take more time in the development. What's more, in addition, they can deal with different feature properties, likecategorical,numerical and binary.

Overall, Random Forest is a fast, flexible and very simple method to use, although it also come with constraints and limitations.

2) **Random Forest** with 300 estimators with a minimum split of 2

3) **Support Vector Machine** with Radial basis function kernel with cubic polynomial as kernel function and 1/no of features as kernel coefficient.

Support Vector Machine" also known as SVM in short form is a part of supervised machine learning algorithm which is best suitable for both classification and regression challenges. However, it is often consider in problems related to classification. In SVM algorithm, we consider or plot every datum thing as a point in n-dimensional space with the estimation that each feature being the estimation of a specific coordinate. Then, we create classification by searching the hyper plane that separate the two



classes exceptionallywell(refer above diagram).

Fig 12: SVM

Support Vectors signifies the co-ordinates of human perception. SVM is a wilderness which best confine the two classes (line/hyperplane).

Upsides and downsides related with SVM

•Advantage:

- SVM works amazingly well with clear edge of partition ◦
- SVM is convincing in high dimensional spaces
- SVM is convincing in circumstances where number of estimations is more critical than the amount of tests.
- SVM uses a subset of getting ready concentrations in the decision limit (called support vectors), so it is in like manner memory viable.

• Disadvantage

- It doesn't perform well, when we have broad instructive gathering because the required planning time is higher
- It similarly doesn't perform uncommonly well, when the educational file has more hullabaloo for instance target classes are covering
- SVM doesn't clearly give probability measures, these are resolved using an expensive five-wrinkle cross-endorsement. It is associated SVC procedure for Python scikit-learn library.

The models were then tested on the testing data using Mean Squared Error as loss function

4.6Transmission & Hit of Predicted data

After completion of system training, Prediction module is able to predict the future frame will most likely be watched by user.

So our system maintain 2 caches, One is normal cache like existing system in which it contains the complete 360 data with high quality data at FOV region and low quality data for rest region of 360 video. Second cache contains 1 min of future data from current time. This future data fetch from the server on the basis of prediction module result. So if user's current FOV same as predicted FOV then 2nd cache will be utilized (hit) otherwise cache-1 will be used (miss-hit). Our prediction system accuracy is around 94% so most of the time hit occur.

Chapter 5: Results & Analysis

5.1 Data Collection and Preprocessing Details

In our project we simultaneously consider sensor and content related historical data to estimatethe viewer's FOV ahead of time. The sensor related features include VR device orientation sensors like magnetometer and accelerometer, while the historical data include the complete data of previous watching orientation of video by different users.

We called around 20 colleague to test and generate the data on 4 videos (Duck, Suthep, Helicopter and Bridge) over period of 2 months.

Content related:

The gathered data was split into test and training data with 25% going to testing and 75%going to training. All the values were scaled between 0 and 1.

1) Random Forest with 100 estimators with a minimum split of 2

2) Random Forest with 300 estimators with a minimum split of 2

3) Support Vector Machine with Radial basis function kernel with cubic polynomial as kernel function and 1/no of features as kernel coefficient.

Please see the result from all the three models.

Collected data based on time with respect to uv mapping in 360 video in ERP mapping.

UV mapping correspond to x and y pixel position within the image.

Please check below the result based on U and V mapping.

The models were then tested on the testing data using Mean Squared Error as loss function

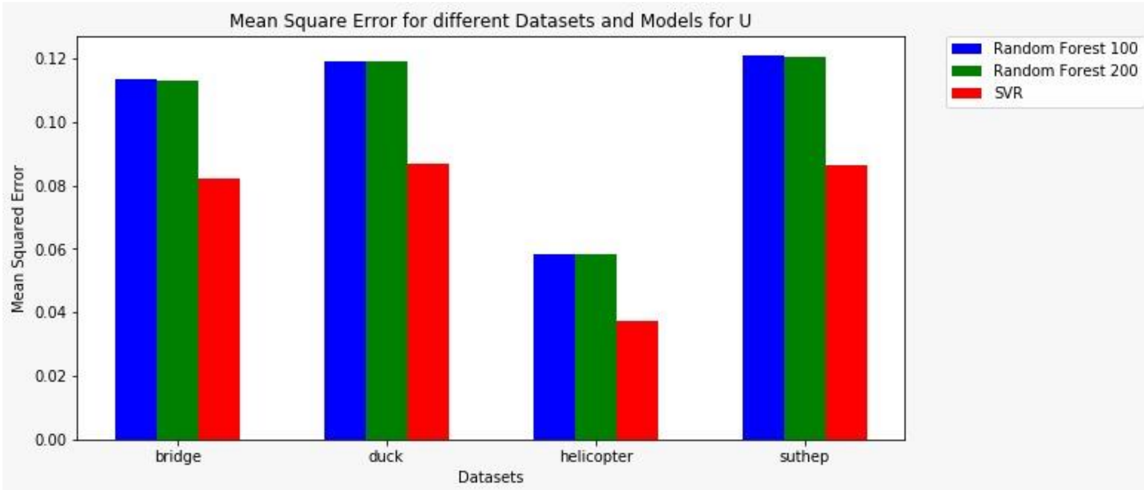


Fig-13 Result comparison for u

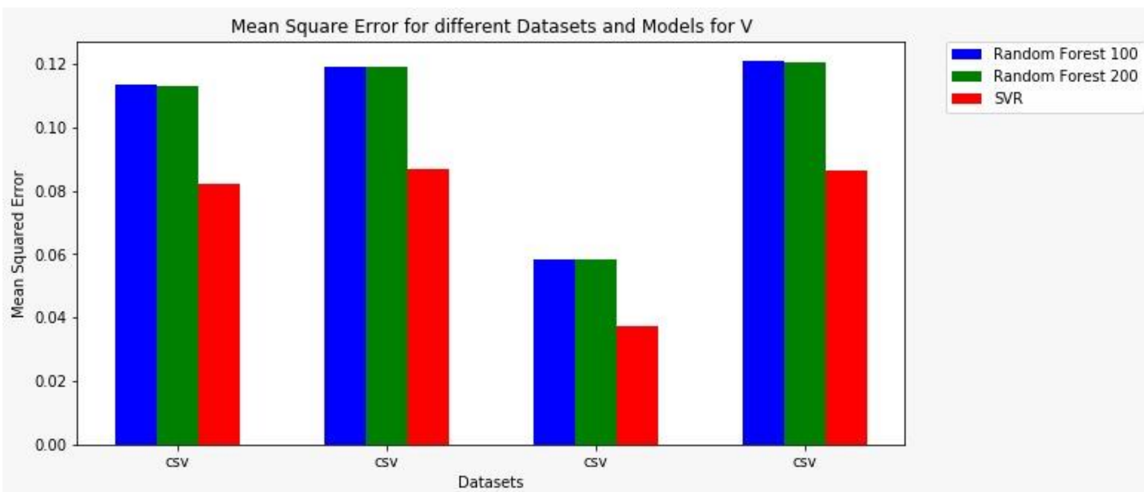


Fig-14: Result comparison for v

Sensor based Data

For Data collection we have used Unity software initially to simulate readings of VR which contains yaw, pitch, and roll along with time-stamps, the frequency of the readings is 60Hz.We'll be observing all these 60readings in a minute and try to predict the reading after 0.5 sec.

Approach:

So the idea is to make a best fit line out of these 60 readings, and using this best fit line to predict the reading 0.5 seconds later. (Linear Regression)

For better prediction I have given weights to readings in such a way that the readings later have more influence in predicting the output than the earlier readings. (Weighted Linear Regression).

Accuracy: We have defined accuracy in such a way that if the predicted angle is around 100 then it's a pass else it is a fail.

For the 1st model the total no. of readings are 412.

Count of readings which have a difference under the given angle	**10^0**	**15^0**	**20^0**	**30^0**	**40^0**
	162	240	326	381	411

So accuracy obtained is 39.41%.

5.2 Field of View Prediction and Results.

Based on the two feature sets (Content and Sensor) we have applied different model to predict the future FOV.

Content Feature Set – Predict the most probable UV mapping at particular time in each video frame. Using UV mapping we found the FOV area at particular time and request the same to server in advance.

Sensor Feature Set-Here when user start moving or changing the FOV in VR devices we calculate the next coming angular position in advance based on direction and speed of turning and calculate the exact data so using advanced data we know where FOV lands next.

Using both the feature data set we can make our system more **Robust and accurate** in Future.

Chapter 6: Conclusion

6.1 Conclusive Results

In our project, we collected 20 user's 4 video viewing data on VR HMD. Our models have improved gradually. The improvements have come based on careful and smart hyper parameter tuning, architecture selection and understanding of the data we are dealing with. The accuracy has grown from 80% to 96.5% if we include all the statistical parameters with appropriate hyper parameters.

After achieving such a good accuracy for Field of View prediction, we were able to show high quality data to user always without bad experience, which is obtained using SVM algorithm. SVM is the best known algorithm till now for topic modeling and has worked well for our use project. Once we get the user's FOV in advance we will be in position to reduce the latency of showing the high quality of data always.

At the end, if we achieve the accuracy approx. 99% then we don't require the complete 360 degree data at all. We only require the FOV data at time which is approximately $1/4^{th}$ of complete data so doing this we will able to save the $3/4^{th}$ of bandwidth along with best user experience.

References

[1] AfshinTaghaviNasrabadi , Anahita Mahzari, Joseph D. Beshay, Ravi Prakash 2017

IEEE Virtual Reality (VR) “Adaptive 360-degree videostreaming”

[2] Mohammad Hosseini, et al. "https://ieeexplore.ieee.org/document/8296778

[3] I. Sodagar. 2011. Œe mpeg-dash standard for multimedia streaming over the internet. IEEE MultiMedia 18,

[4] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, “Overview of the HighEfficiency Video Coding (HEVC) Standard,” Circuits and Systems forVideo Technology, IEEE Transactions on, vol. 22, no. 12, pp. 1649–1668,Dec 2012.

[5] L. Bian and H. Holtzman. Online friend recommendation through personality matching andcollaborative filtering. Proc. of UBI-COMM,pages 230-235, 2011.

[6] J. Kwon and S. Kim. ”Projection on 360 Video”, 10(11):116-120, 2010.

[7] X. Yu, A. Pan, L.-A. Tang, Z. Li, and J. Han. Geo-friends recommendation in gps-based cyber-physical social network. Proc. Of ASONAM, pages 361-368, 2011.

[8] Ching-Ling Fan1, Jean Lee1, Wen-Chih Lo1.Fixation Prediction in VR 360.

[9] Ha, Sojeong, and Seungjin Choi. "Convolutional neural networks for human activity recognition using multiple accelerometer and gyroscope sensors." *Neural Networks (IJCNN), 2016 International Joint Conference on*. IEEE, 2016.

[10] L. Gou, F. You, J. Guo, L.Wu, and X. L. Zhang. Sfviz: Interestbased. friends exploration and recommendation in social networks. Proc. of VINCI, page 15, 2011.

[11] Zeng, Ming, et al. "Convolutional neural networks for human activity recognition using mobile sensors." *Mobile Computing, Applications and Services (MobiCASE), 2014 6th International Conference on*. IEEE, 2014.

[12] Ronao, Charissa Ann, and Sung-Bae Cho. "Human activity recognition with smartphone sensors using deep learning neural networks." *Expert Systems with Applications* 59 (2016): 235-244.

[13] Li, J. A. Stankovic, M. A. Hanson, A. T. Barth, J. Lach, and G. Zhou. Accurate, Fast Fall Detection Using Gyroscopes and Accelerometer-Derived Posture Information. Proc. of BSN, pages 138-143, 2009.

[14] Wang, Zhibo, et al. "Friendbook: a semantic-based friend recommendation system for social networks." *IEEE transactions on mobile computing* 14.3 (2015): 538-551.J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson. EasyTracker: Automatic Transit Tracking, Mapping, and Arrival Time Prediction Using Smartphones. Proc. of SenSys, pages 68-81, 2011.

[15] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, “Implementation of a real-time human movement classified using a tri-axial accelerometer for ambulatory monitoring,” IEEE Trans. Inf. Technol. Biomed., vol. 10, no. 1, pp. 156–67, Jan. 2006.

[16] M. Mathie, B. Celler, N. Lovell, and A. Coster, “Classification of basic daily movements using a triaxial accelerometer,” Med. Biol. Eng. Comput., vol. 42, pp. 679–687, 2004.

[17] L. Bao and S. S. Intille, “Activity recognition from user-annotated acceleration data,” in Pervasive Computing, Berlin/Heidelberg, Germany: Springer-Verlag, 2004, pp. 158–175.

[18] N. Ravi,N.Dandekar, P. Mysore, and M. L. Littman, “Activity recognition from accelerometer data,” in Proc. 20th Nat. Conf. Artif. Intell., 2005, pp. 1541–1546.

[19] U. Maurer, A. Smailagic, D. Siewiorek, and M. Deisher, "Activity recognition and monitoring using multiple sensors on different body positions," and monitoring using multiple sensors on different body positions," pp. 113–116.

[20] A. M. Khan,Y.K. Lee, andT.-S. Kim, "Accelerometer signal-based human activity recognition using augmented autoregressive model coefficients and artificial neural nets," in Proc. 30th Annu. IEEE Int. Conf. Eng. Med. Biol. Soc., 2008, pp. 5172–5175.