NOVEL STRATEGIES TO IMPROVE THE EFFECTIVENESS OF FIELD OF VIEW – AWARE EDGE CACHING FOR ADAPTIVE 360° VIDEO STREAMING

by

Novonil Das

APPROVED BY SUPERVISORY COMMITTEE

____

Ravi Prakash, Chair

____

Neeraj Mittal

____

Kamil Sarac

To my father and mother

NOVEL STRATEGIES TO IMPROVE THE EFFECTIVENESS OF FIELD OF VIEW – AWARE EDGE CACHING FOR ADAPTIVE 360° VIDEO STREAMING

by

NOVONIL DAS, B.TECH

THESIS

Presented to the Faculty of

The University of Texas at Dallas

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN

COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT DALLAS

August 2021

ACKNOWLEDGMENTS

I am overwhelmed in all humbleness and gratefulness to acknowledge my work to all those who have helped me to transition my ideas into something fruitful and concrete. I would like to express my special thanks of gratitude to my guide, motivator, and mentor, Dr. Ravi Prakash, without whose guidance and encouragement, this would have been far from achievable. I am also thankful to him for giving me the golden opportunity to do research and contribute to this wonderful project - Novel Strategies to Improve the Effectiveness of Field of View - Aware Edge Caching for Adaptive 360° Video Streaming. Nothing could have been satisfactorily completed without the support, guidance, and blessings of my parents. I am also thankful to all my friends who were always supportive and encouraging. I am also grateful to Dr. Kamil Sarac and Dr. Neeraj Mittal for being a part of my thesis defense committee.

July 2021

NOVEL STRATEGIES TO IMPROVE THE EFFECTIVENESS OF FIELD OF VIEW – AWARE EDGE CACHING FOR ADAPTIVE 360° VIDEO STREAMING

Novonil Das, MS
The University of Texas at Dallas, 2021

Supervising Professor: Ravi Prakash, Chair

Virtual Reality (VR) and 360° Video Streaming have attained a lot of popularity recently. Streaming 360° video to Head Mounted Displays (HMDs) over the internet is extremely demanding owing to its huge size, desirability to be viewed at higher resolutions, high bandwidth, and low latency requirements. However, viewers can view only a small portion of a scene in the video at a time, since viewers are limited by the Field of View (FoV) of the HMD. A few solutions use adaptive 360° video streaming by streaming high resolution video of only the part on the video in the viewers FoV, and low-resolution video for the part of the video that is not in the viewers FoV. FoV Adaptive 360° video streaming has been instrumental in decreasing the bandwidth requirements, but network latency is another factor that adversely affects the streaming of 360° videos from distant content servers. To overcome this, some solutions use caching of popular content at the mobile edge cloud server close to the end user. This caching policy helps reduce latency in the network and alleviate network bandwidth demands by decreasing the number of future requests that must be sent to the content server, thus reducing the load on the server. But most of these strategies use generic heat maps to determine popular content in videos among users.

A viewers' FoV is a depiction of that viewers' area of interest in the video at any point in time - with the center of the FoV being of utmost importance grabbing the viewers' attention and the peripheries of the FoV of relatively lesser importance, importance decrease as we move from the center to the periphery in the FoV. Anything outside the FoV of the viewer is of no importance since the user chose not to see that part in the video. The importance of a part in the video portrays its popularity among users – the more the importance, the more the popularity. The popularity of different parts of the video based on the past viewers' viewing history determines how significant each part of the video is to be cached at the edge servers in the above-mentioned caching policy. More the popularity, the more the probability of the content to be cached. However, the use of traditional heat maps to determine the popularity of video content gives equal importance to the entire FoV and fails to cater to the requirement of declining importance given to different parts of the FoV as we move farther away from the center of the FoV. In this thesis, we show how the use of such heatmaps gives wrong impression of the popularity of video contents -contents that are not so popular appear to be popular. This false notion created by heat maps renders them useful only in highly constricted cases. We demonstrate the relevance of some indices used in election analysis to overcome this limitation in heat maps and discuss where they would work best. We also introduce the concept of Vote decay in the popularity of contents in the FoV to remove misinterpretations of content importance so that we can improve caching decisions and future FoV predictions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

SL No.	Full Form	Abbreviation
1	Virtual Reality	VR
2	Field of View	FoV
3	Head Mounted Display	HMD
4	Quality of Experience	QoE
5	Content Delivery Network	CDN
6	Radio Access Network	RAN
7	Dynamic Adaptive Streaming over HTTP	DASH
8	Density-Based Spatial Clustering of applications with Noise	DBSCAN

CHAPTER 1
INTRODUCTION

Popularity of Virtual Reality (VR) has been growing over the years. This has been made possible by technologies such as 360° video streaming. 360° videos, also called spherical or immersive videos are video recordings where an omnidirectional camera or collection of cameras is used to record a view in every direction simultaneously. This allows us a view of the entire surroundings around the camera. These videos are used not just in gaming and entertainment but also in education, medicine, and sports. The viewers are placed at the center of the spherical 360° video and are allowed to look in any direction at will. This gives them an experience of being a part of the environment they are looking. These videos at times allow users to interact with them. Traditional fixed-frame videos, limit the view of the user to wherever the camera is pointing at, whereas 360° videos offer a full 360° x 180° panoramic field of view. The users can look up, down, and all around and can explore a scene from any point of view they choose.

Due to omnidirectional content of 360° videos, they are much larger in size. Streaming 360° video over the internet is extremely demanding due to its large size, low latency, and high bandwidth requirements. To add onto the challenges the videos need to be streamed at higher resolutions since the images in the videos are close to the viewer's eyes.

Displaying the video at low resolution will expose the artefacts and negatively impact the Quality of Experience (QoE) of the viewers. With this increasing usage of Virtual Reality in various fields, factors such as communication network requirement, Field of View (FoV) prediction and edge caching of popular contents become important considerations.

A few solutions in VR use adaptive 360° video streaming which decreases the bandwidth consumption by leveraging the fact that users at any point of time cannot watch the entire 360° video. They can watch only a part of the video at any point of time since their field of view is limited by the viewport of the Head Mounted Display (HMD). Field of View (FoV) is the extent of observable environment in the video at any given time. This adaptive streaming technique reduces bandwidth consumption by transferring only the contents in the users FoV at high resolution to the user's HMD, while sending the remaining contents of the video to the HMD at low resolutions. Some solutions also suggest not sending the remaining contents to the client at all to reduce the bandwidth consumption further. Although this streaming technique has been effective in decreasing bandwidth requirements, network latency still makes streaming 360° video from distant content servers demanding.

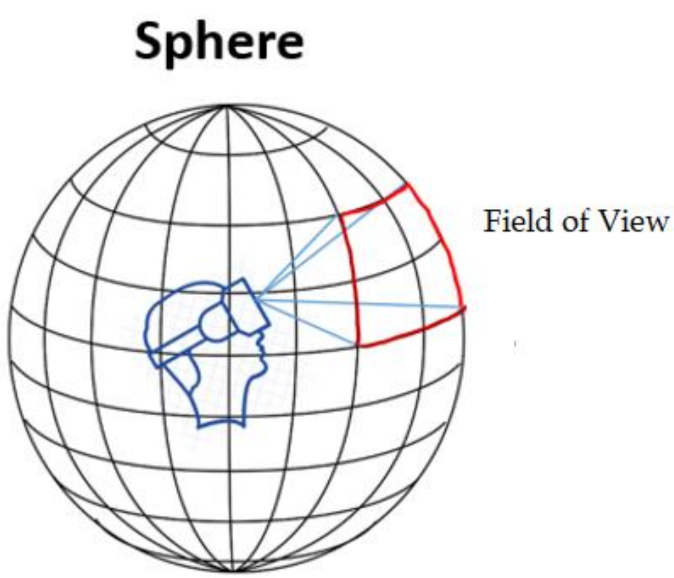

Figure 1: Viewer, 360° Video and FoV of the viewer [33]

Network latency and bandwidth consumption has been significantly decreased by deploying proxy servers with caching abilities between users and content servers. This reduces the load on the content server by decreasing the number of future requests that must be sent all the way to remote content servers and makes the implementation scalable. This concept leverages the caching service used in Content Delivery Networks (CDNs) architecture to improve viewer's quality of experience (QoE).

CDNs have been instrumental in decreasing the network latency and bandwidth consumption. However, with the ever-increasing demand on mobile data traffic, efforts have been made to bring content closer to the viewer by deploying mobile cloud servers at the edge of the Radio Access Network (RAN) on base stations, to enhance the performance of the mobile network and maintain good QoE. This strategy allows to address some of the client's requests for contents in the video without contacting CDNs. This results in reduced load of CDNs, quick response time and reduced backhaul link consumption in the RAN. Caching at the edge of RAN also allows spatial and temporal decisions in updating cache's content in near real-time. Figure 2 shows the architecture of edge caching.

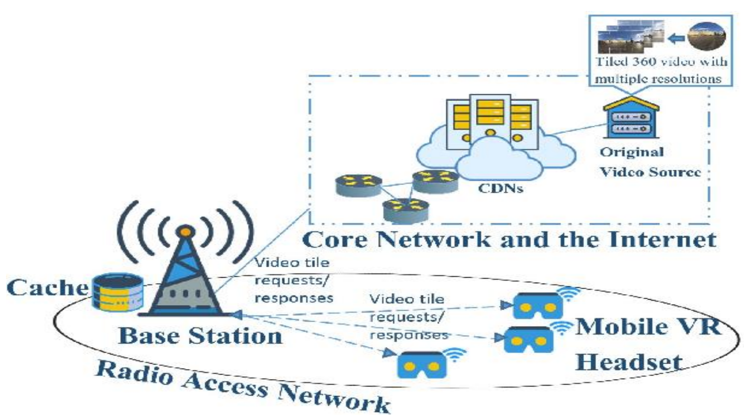

Figure 2: Architecture of edge caching [32]

A 360° video is made up of segments of a few second duration. Each segment is made up of frames, and each frame is divided into tiles. The brute force solution to caching tile based 360° video is to cache all contents at all resolutions for the whole video. But this method would require ever increasing cache storage to accommodate all videos or for a fixed cache storage, be capable of caching only a limited number of 360° videos. One other approach is to cache all the tiles in only high resolution and apply transcoding method to generate them at other resolutions. But this would impose processing requirements on the cache servers. FoV - Aware Edge Caching for adaptive 360° video streaming comes to the rescue to overcome this problem. In this method, whether a tile must be cached or not depends on how popular the tile is. The more popular a tile has been in the past, the more likely it is to be viewed in the future, and hence a good candidate to be cached. This is achieved with the use of heat maps where tiles in a video initially start with a popularity vote of 0, whenever a tile appears in the FoV of a viewer the tile gets a vote of 1 otherwise it gets a vote of 0. The cumulative votes of every tile at any point of time denotes how popular a tile is. Based on the popularity, tiles are cached and evicted to make space for other tiles when the cache capacity is exceeded.

With increase in 360° video contents, caching decisions must be highly efficient. We should be able to cater to the QoE of the viewer and not over utilize the cache. The FoV of a user in a segment of the video is the part of the segment the user is interested in. Since the user ignores the part outside the FoV in that segment in the video, his votes go to only those tiles in the user's FoV. None of the user's vote go to the tiles outside the FoV. Also, the center of the FoV is the area which the viewer is most interested in, and the peripheries are the areas that the user has relatively lower interest in. The strategy of assigning equal importance to all tiles in the FoV in case of a heat map can result in erroneous representation of the importance of certain tiles. We analyze the shortcomings of the prevailing heat map-based approach in more details and present a solution to rectify this problem in the subsequent chapters.

CHAPTER 2
SYSTEM ARCHITECTURE

2.1 Network Model

The network has two parts – the wired backbone and the wireless access network. A wired network connection is one where cables are involved in establishing a connection to other devices on the network or to the Internet. Ethernet cables are used to transfer data from one device to another or over the Internet. Wireless networks operate on radio frequency or microwave signals. The radio signals allow wireless enabled devices to communicate with the Internet or with one another without having to be connected to the network via an Ethernet cable. Wired networks are generally faster, more reliable, and stable as compared to Wireless networks, whereas wireless networks offer more mobility and less maintenance requirement.

We are considering a network model where we have 360° videos stored in original video source servers at one end. Traditionally these servers are connected over a wired network. Users trying to view video contents on their HMDs, generally over a wireless network, are at the other end. In between the source servers and the users, CDNs are deployed. There are cache-enabled base stations between CDNs and users, deployed in proximity of clients watching these videos. The base stations have limited cache capacity and we need to make the most of the cache storage. Since cache storage is limited, whenever the cache capacity is exceeded, we need to make space using a proper cache replacement policy. Two scenarios might arise when it comes to fetching data from the cache - cache hit and cache miss. If the requested content is available in the cache, it immediately sends the content. Otherwise, it downloads the requested content first from the remote content delivery networks and then sends it to the client. Therefore, a cache miss increases the bandwidth consumption and download latency experienced by the client. Hence, increasing cache hit would be beneficial both for the network and the clients. So, we need effective caching strategy to ensure optimal usage of the cache.

2.2 Video Model

Dynamic Adaptive Streaming over HTTP (DASH) is at the heart of the content transfer protocol used in 360° video streaming. In this protocol a video is temporally divided into few seconds long segments, each segment is made up of several frames which correspond to a sphere centered at the viewer. This spherical frame is mapped onto a two-dimensional space - equirectangular representation of the frames as shown in Figure 3. This is like the mapping of the globe to a rectangular world map. A frame consists of tiles. Tiles can be encoded in different video resolutions and have varying sizes: encoding a tile in higher resolution means greater size in bytes.

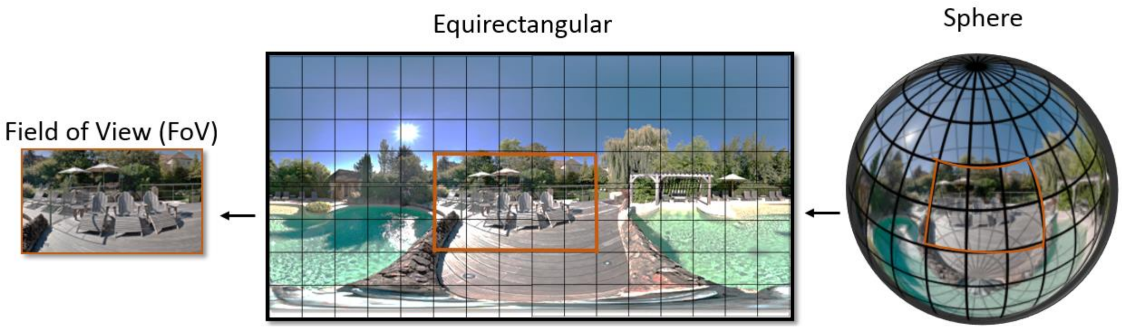

Figure 3: Equirectangular Projection of Frames in a 360° Video and Field of View [32]

Each frame in a 360° video is made up of M x N array of tiles as shown in Figure 4**.** Here M is the number of vertical tiles on each column and N is the number of horizontal tiles on each row forming the equirectangular representation of the frame. A viewer cannot watch the entire frame at any given point in time. They watch only the part of the frame that is in the viewer's FoV. The entire figure depicts a frame, and the shaded portion represents a FoV in the frame. A viewers' FoV is broken into m x n array of tiles. Here, m is the number of vertical tiles on each column, and n is the number of horizontal tiles on each row forming the FoV.

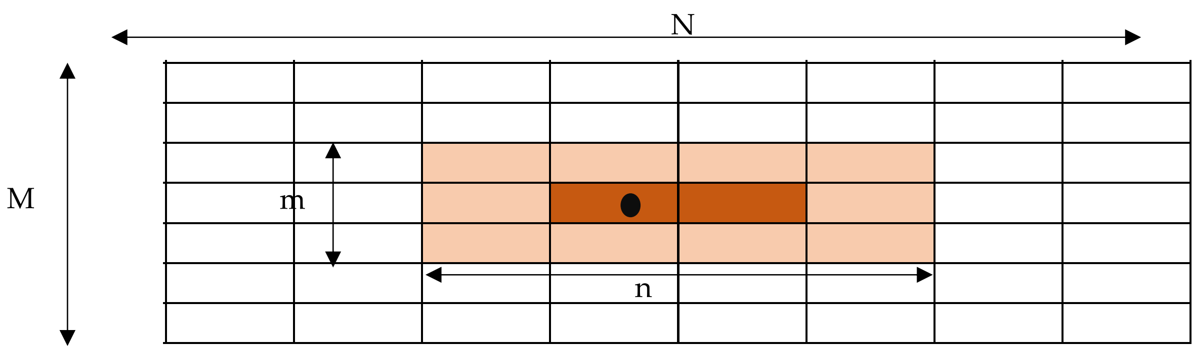

Figure 4: Tiling of each frame and the FoV

Each FoV in a frame is made up of r concentric rectangular ring of tiles around the center of the FoV as we see in Figure 5. We call this value r the radius of the FoV. The radius r of the FoV is equal to the value - Min $\left(\left\lceil\frac{m}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)$. These rings cover all the m x n tiles in the FoV. The innermost central ring in the FoV has $\mathrm{m} - 2 * (\mathrm{r} - 1)$ vertical tiles on each column and $\mathrm{n} - 2 * (\mathrm{r} - 1)$ horizontal tiles on each row. All other outer rings form concentric rectangular rings around the central ring. These outer rings have a width of 1 tile around its immediate inner ring.

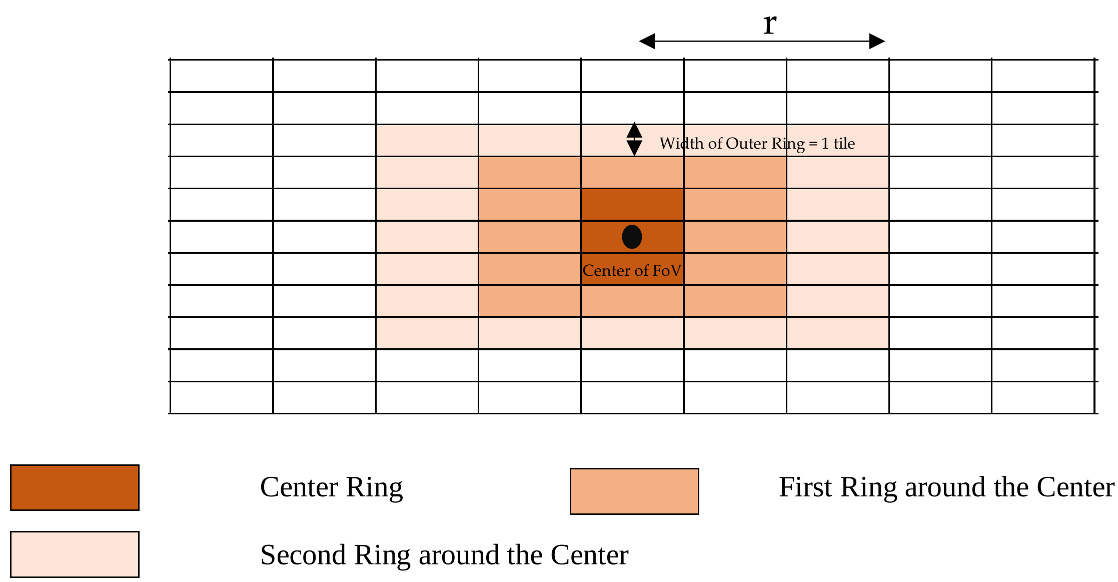

Figure 5: Illustration of Center and Rings of a sample FoV

Past viewer's FoVs in each frame in a 360° video could be grouped into x different clusters [17]. So, each frame in the video could have x different FoV clusters based on the past viewers viewing pattern. This value of x would differ across different frames. x would also change in the same frame with time. Each cluster could be made up of overlapping or non-overlapping FoVs of different users.

2.3 Voting Models

Since clients watch only a part of the 360° video at a time, their FoV shows which part of the video they are interested in. Anything outside the FoV, is what the viewer chooses to ignore which tells that the user is comparatively less interested in the other regions on the video frame. Even in an FOV there are tiles arranged in the form of concentric rectangular rings, with tiles on the ring at the center being the tiles of highest importance since the users focus on the center of the FOV. Tiles on the subsequent rings are of lower importance, farther the ring from the center the lesser the importance. This is analogous to multi-party election system where there are multiple candidates fighting an election and there are voters who cast a preference list of votes for those candidates. The candidates in the election here are the tiles in a frame and the voters are the viewers watching the video, their preference list of votes contain the tiles in all the rings that constitutes the FoV. If they are interested in a particular candidate, they cast their vote for those candidates with their eyes in the form of a preference list - by watching all the candidate tiles in that part of the video. Using this preference list, we consider the different voting methods. In all these methods, initially every tile starts with a clean slate and all of them has a vote of 0 associated with them. As viewers start watching videos, tiles on each frame get votes from the viewer if that tile is in the FoV. The cumulative vote associated with a tile on a ring $r_{k'}$ at any point in time is given by $Vr_{k'}$. With time this value may be different for different tiles on the ring.

CHAPTER 3

RELATED WORK

Users watch a limited subset of tiles most of the time. However, they do not watch the same group of tiles [30]. Viewers can be divided into several clusters based on their region of interest (FoV) in the frame, with each region corresponding to a subset of tiles appearing in the users' FoV. In the past Density-Based Spatial Clustering of applications with Noise (DBSCAN) [30] has been used to cluster previous viewers' FoV in each frame and calculate the probability of tiles in each cluster being viewed. This probability is the same as the proportion of viewers mapped to that cluster. The maximum number of such clusters in a frame can be assessed using the Laakso-Taagepera Effective Number of Parties [13][28].

Heat maps have been traditionally used to make caching decisions of tiles in a 360° video in the edge cache server. Heat maps [15][17][32][33] use the concept of elections to cast votes in favor of tiles in the FoV of a user. We also use the works of Borda [3][5][10] and Dowdall [10] and apply it to our setup to see if they can overcome the shortcomings of the traditional heat map. We also introduce the idea of vote decay value to overcome the limitations of various voting methods.

3.1 Traditional Heatmap

In traditional heatmap approach every time a tile appears in the FoV of the viewer the tile is given a vote of 1. At any point of time the cumulative vote of each tile denotes the popularity of that tile in terms of all the past users' viewing histories. Ideally, popularity determines a tile's importance in terms of being cached - more the popularity more is the importance to be cached. Since, each tile gets a vote of 1 as and when the tile appears in the FoV of the user, there is no distinction between the tiles at the center of the FoV and the ones at the periphery of the FoV, each tile in the FoV is equally important.

3.2 Borda Count

The Borda count [3][5][10] is used in elections in Slovenia to come up with a societal preference from a collection of individual preferences. Borda count gives highest weight to the first-choice candidates, and the wight reduces at a constant rate as we go down the ranked list of preferred candidates. In this method, the most preferred candidate of a voter among k candidates fighting the election gets k votes, the second most preferred candidate gets k– 1 votes, and so on from that voter. The least liked candidate gets 1 vote from the voter. Applying this to the setup of r rings of tiles in the FoV, the tiles on the center ring are the most preferred candidates. The center ring is the part on the video frame the user is most attracted to. In this method the tiles on the center ring get the highest number of votes r, and the tiles on the outermost ring get the lowest number of votes 1 each. The number of votes offered to tiles on each ring decreases as we go away from the center in arithmetic progression: $r, r-1, r-2, \dots, 2, 1$. In this method, the difference in weight between any two consecutively ranked candidates is the same giving no preference to the better ranked candidates, that is the second-choice candidate is at a same disadvantage from the first-choice candidate as is a third-choice candidate from the second-choice candidate.

3.3 Dowdall Rule

The Dowdall rule [10] is used in elections in the island of Nauru. Here the most preferred candidate is given a vote of 1, the second most preferred candidate is given a vote of $1/2$, the third is given a vote of $1/3$, and so on. So, when this method is applied to the setup of r rings of tiles in the FoV, votes on the tiles on each ring changes in a harmonic progression: $1, 1/2, 1/3, \ldots, 1/r$ for each ring around the center from the center itself to the periphery. So, in this method, the decline in the weight is not steady across rings as in Borda Count. The decline in weight is the fastest close to the center and tapers off as we move farther from the center. Afterall, the decline of $1/2$ in the number of votes between the tiles in the two innermost rings is greater than the decline of $1/6$ in the number of votes between the tiles on the second and third ring. Thus, a first-place candidate is given a much higher preference than the second-place candidate when compared to the preference given to the second-place candidate over the third-place candidate.

3.4 Vote Decay Method

In this method we use a decay value q to reduce the count of votes assigned to tiles as we move outward from the center ring to the peripheral ring. This vote assigned is in the form of a geometric progression where the common ratio is $1/q$. Thus, the count of votes assigned to tiles on each ring as we move away from the center looks like this: $1, 1/q, 1/q^2, \ldots, 1/q^{(r-1)}$ as we move from the center to the peripheral rings. In this method the decline in the votes is even more drastic than the normal Dowdall rule, thus increasing the difference in preference of the first-choice candidate over the second choice and so on. We can change the decay value q as and when required independently without changing any other parameters to increase/decrease the weight of votes given to a candidate over the next preferred candidate. When we change the decay factor q with time, all the votes that were cast by the past viewers need to be recalculated with the changed value of q.

CHAPTER 4

PROPOSED METHODS AND RESULTS

As per the system architecture in Chapter 2, each frame in a segment has $M\ x\ N$ tiles, the video server maintains information of the votes associated with each tile in a frame, based on all previous viewers' viewing history. These votes are assigned to each tile based on the voting protocol used - traditional heat map, Borda, Dowdall rule, or Vote Decay based voting method. As discussed earlier, votes assigned to a tile is determined by two factors: (i) how frequently a tile appears in a viewer's FoV and (ii) the proximity of the tile to the center of the FoV. To address viewers' Quality of Experience, it is desirable that we cache the most popular contents at high resolution and less popular content at a slightly lower resolution on the edge cache server, unpopular contents can be chosen not to be cached at all. We group all these FoVs into different clusters based on past viewers' region of interest in each frame and calculate the proportion of viewers viewing a particular cluster. We can cap the number of FoVs or clusters to be considered for caching using the Laakso-Taagepera Effective Number of Parties [13][28]. In this approach, the maximum number of effective clusters is determined by the formula $N = 1/\sum_{\substack{i=1 \\ n}} p_i^2$ where N is the effective number of clusters, p_i is the proportion of users viewing that cluster and n is the total number of clusters in the frame. We would take one extra ring around the periphery of the FoV as a margin ring to ensure that partial overlaps between the FoV and the tiles on each frame in a video segment is taken care of as in Figure 6.

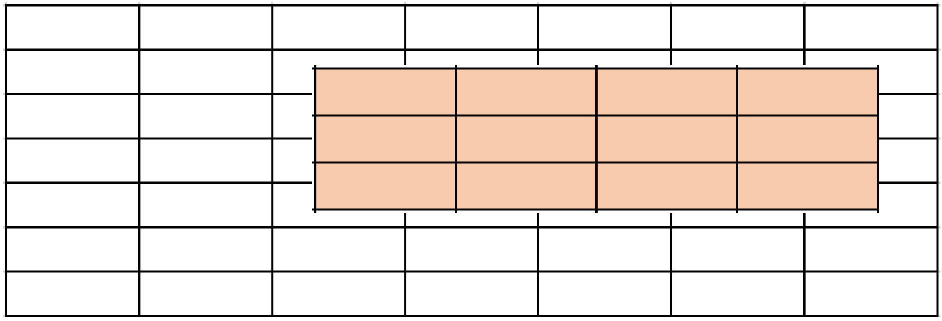

Figure 6: Illustration of FoV not perfectly aligned with the tiles

Correctness Condition

It is desirable to cache tiles with higher vote count at higher resolution compared to tiles with lower vote count. This will ensure good QoE and efficient usage of cache space. Based on this premise and since the tiles on the center ring of a FoV is more popular than the outer rings, we should ensure that a tile on an inner ring (close to the center of an FOV) has a cumulative vote count higher than the cumulative vote count of an adjacent tile on a comparatively outer ring in the FoV. A tile is said to be adjacent to another tile if and only if one of either the row or column of both the tiles is the same.

$$Vr_{j'} \geq Vr_{k'}$$

where $r_{j'}$ and $r_{k'}$ are two rings in an FoV i, $r_{j'}$ is the inner ring closer to the center ring and $r_{k'}$ is the outer ring farther away from the center ring as compared to $r_{j'}$, that is $1 \leq r_{j'} < r_{k'} \leq r$. This is shown in Figure 7. This condition must hold for the solution to be correct. We check under what conditions each method satisfies this correctness criteria, and if it is possible to achieve such a criterion to determine which method works best to determine importance of contents.

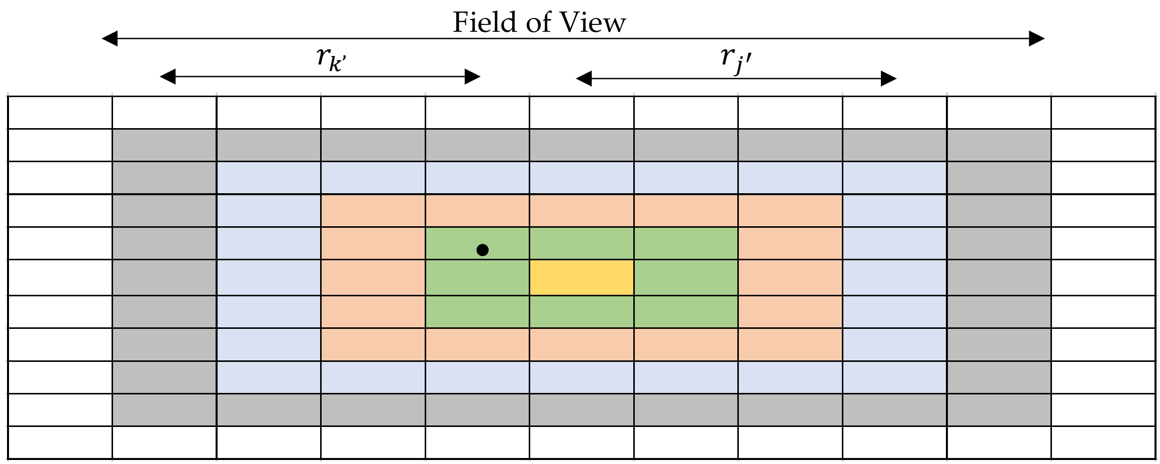

Figure 7: Depiction of center of the FoV, inner and outer rings around the center in FoV

4.1 Traditional Heatmap

Based on the traditional heat map approach, initially each $M\ x\ N$ tile in every frame of the 360° video has a vote of 0 associated with them. Each time a tile is in a viewers' FoV it gets a vote of 1, all other tiles not in the FoV get a vote of 0. The resultant heat map depicts how frequently each tile in a frame has appeared in the past viewers' FoV. This frequency depicts how popular a tile is - higher the frequency, higher the popularity and vice versa. The more popular a tile is, the more important it is to be considered for caching. As discussed earlier, the center ring of FoV in a frame of a video is more important than the peripheral ring on the FoV since that center ring is where the viewer is interested the most. So, in terms of viewers Quality of Experience it would be acceptable to have the tiles on the central ring at a higher resolution and the tiles on the peripheral rings at a slightly lower resolution, but the other way round would not be acceptable. But a heat map makes no difference between a tile on the center ring and a tile on the peripheral ring and casts equal votes to all the tiles in the FoV. Thus, in terms of achieving QoE and effective usage of edge cache storage, heat maps do not do a great job. Also, it is possible that FoVs i and j in a frame intersects with each other, that is the distance between the centers of FoV i and j is less than $2r$. Then there will be tiles that will be common in both these FoV, $i \cap j$ is non-empty, and these tiles will get votes from both the FoVs. When the distance (d) between the center of the FoVs i and j is $r < d \leq 2r$ then the votes of tiles on the outer peripheral rings would be greater than the votes of the tiles on the inner central rings, which violates the correctness condition. Thus, heat maps do not provide an efficient means of using cache storage and providing good QoE.

Lemma

For the solution using heat maps to be correct, the correctness condition $\mathrm{Vr}_{\mathrm{j}'} \geq \mathrm{Vr}_{\mathrm{k}'}$ must hold. A tile on an inner ring $r_{j'}$ (close to the center of an FOV) should have a cumulative vote count ($Vr_{j'}$) greater than the cumulative vote count ($Vr_{k'}$) of an adjacent tile on a comparatively outer ring $r_{k'}$ in the FoV.

Proof

If there are x FoVs in a frame of a video and the proportion of viewers looking at FoV i be p_i, then, $\sum_{i=1}^{x} p_i = 1$. This value p_i is also the probability of the FoV i to be viewed by a user. Then, using the traditional heat map method all tiles in each FoV i would get a vote of 1 every time it is in the FoV. So, each tile in the FoV would have a normalized vote count of $p_i * 1 = p_i$.

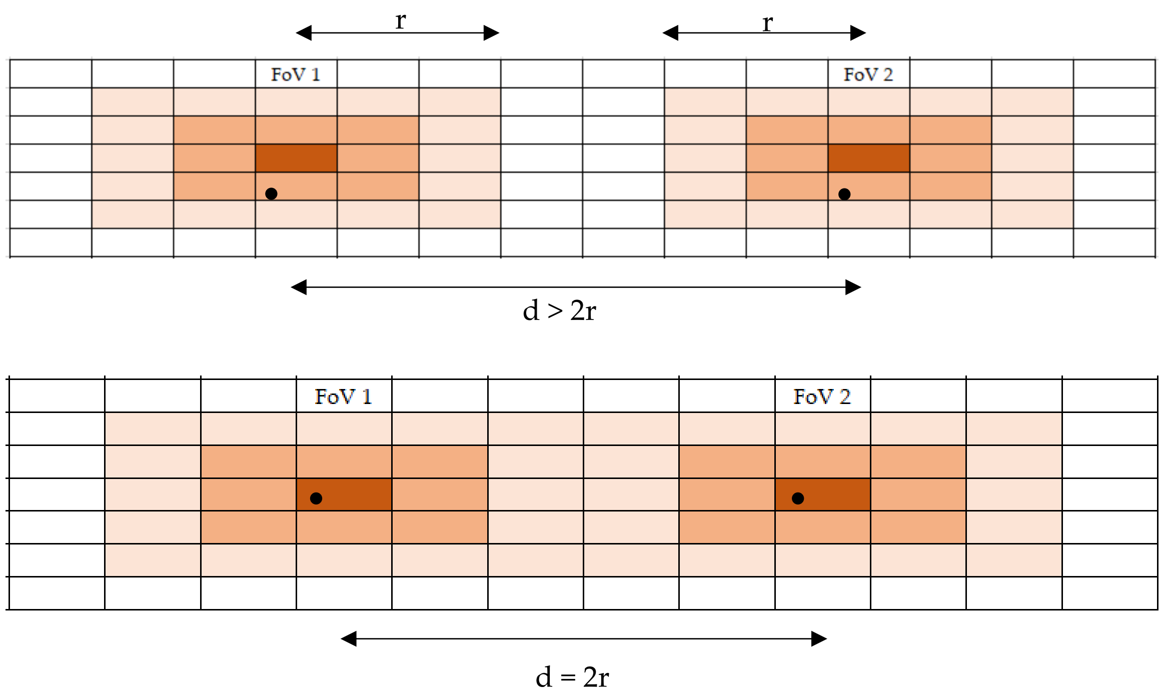

Figure 8: Two non-overlapping FoVs with d > 2r and d = 2r

If there is no intersection among FoVs that is the distance between the center of all FoVs d is greater than or equal to $2r$, then the vote count of each tile in every ring on FoV i would be p_i. Thus, all rings in an FoV – inner or outer have the same number of votes $Vr_{j'} = Vr_{k'}$ and the correctness condition holds.

Let y of these x FoVs ($2 \leq y \leq x$) in the frame intersect with each other and form a cluster such that the distance d between the center of the FoVs is greater than r and less than $2r$, $r < d < 2r$. We consider an FoV j in those y overlapping FoVs and all other $y - 1$ FoVs in the cluster separately. Each tile on the FoV j would have a normalized vote of p_j. Because of the overlap from the other $y - 1$ FoVs, the tiles on the overlapping section would get an additional vote of $\sum_{i \in \mathrm{y}, \mathrm{i} \neq \mathrm{j}} p_i$ from the other $y - 1$ FoVs. So cumulative normalized vote on tiles on the overlapping section $Vr_{k'} = \sum_{i \in \mathrm{y}, \mathrm{i} \neq \mathrm{j}} p_\mathrm{i} + p_\mathrm{j} = \sum_{i \in \mathrm{y}} p_\mathrm{i}$. Tiles on the immediate inner ring r_j would have a vote of $Vr_{j'} = p_j$. Since $2 \leq y$ there is at least another FoV in addition to FoV j. The sum of the proportion of viewers watching both these FoVs is greater than the proportion of viewers watching just FoV j. That is $\sum_{i \in \mathrm{y}} p_\mathrm{i} = \sum_{i \in y, i \neq j} p_\mathrm{i} + p_\mathrm{j} > p_\mathrm{j}$, $Vr_{k'} > Vr_{j'}$ that is a tile on an outer ring would have a vote count greater than a tile on an inner ring. Thus, the correctness condition $Vr_{j'} \geq Vr_{k'}$ does not hold. In heat map-based approach the correctness condition does not hold if there is an overlap between FoVs.

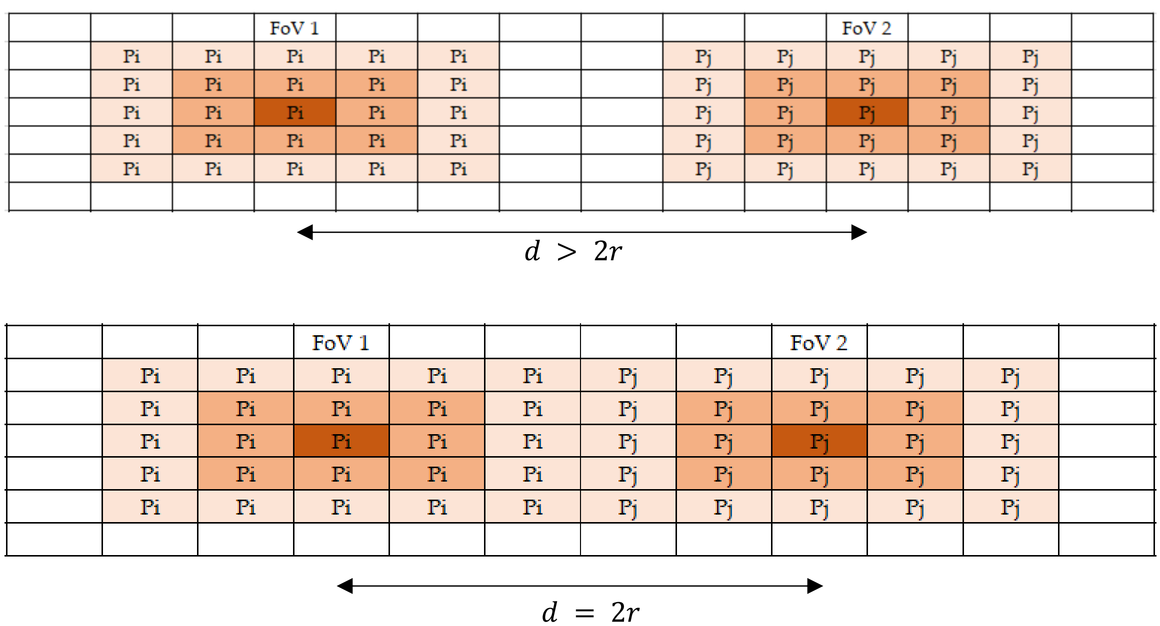

Figure 9: Votes of tiles of non-intersecting FoVs using Borda Count (d > 2r and d = 2r)

Let us also look at the example in Figure 11. There are 2 FoVs of radius r - FoV_1 and FoV_2. The distance between the center of the FoVs d is $2r - 1$, that is they overlap on the outermost ring. Let the proportion of users viewing FoV_1 be p_i and the proportion of users viewing FoV_2 be p_j. Now, let us evaluate only FoV_1 - the normalized vote count of each tile on every ring in FoV_1 is equal to p_i. Due to the overlap from FoV_2, FoV_2 contribute a vote of p_j to the overlapping sections of FoV_1. On the outer intersecting ring the tiles will have a cumulative normalized vote equal to the $p_i + p_j$ as shown in the figure. Now this vote is greater than the vote p_i on the immediate inner ring of FoV_1 and p_j of FoV_2. Thus, the correctness condition does not hold.

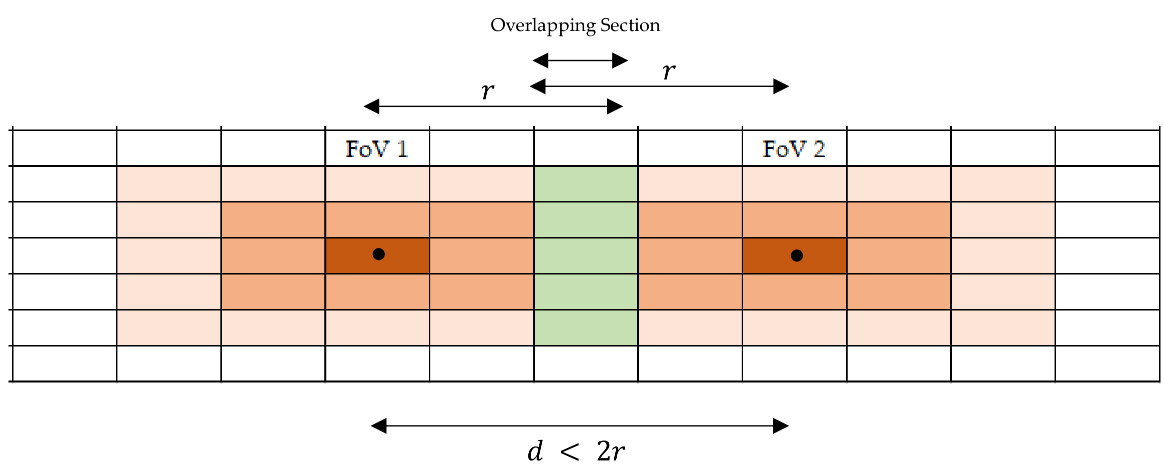

Figure 10: Two FoVs overlapping with each other (d < 2r)

Heat maps give a lot of importance to the intersection regions of FoVs, so heatmaps should be used in scenarios where tiles on the intersecting regions of FoVs are of utmost priority to be cached to improve viewers' QoE. Heatmaps are not efficient means of voting if the tiles on the center ring are more important candidates to be cached to improve viewers' QoE.

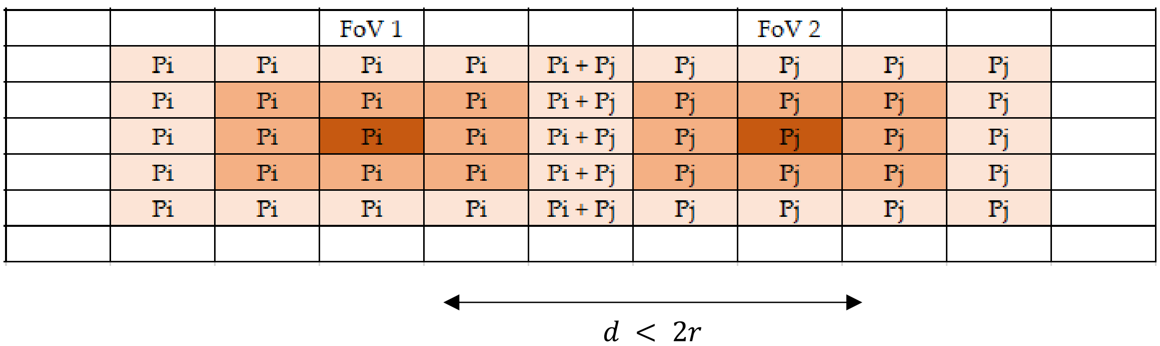

Figure 11: Votes of tiles of intersecting FoVs using Heat maps

4.2 Voting using Borda Count

In the voting using Borda Count, initially all $M \ x \ N$ tiles in every frame have a vote of 0 associated with them. Each time a tile is in the FoV, it gets a vote. All tiles on the center ring get r votes, all tiles on the immediate next outer ring get $r-1$ votes, and so on, the tiles on the outermost peripheral ring get 1 vote. This method has both the factors mentioned earlier needed to ensure good QoE and effective usage of caching storage.

Lemma

For the solution using Borda Count to be correct, the correctness condition $\mathrm{Vr}_{j'} \geq \mathrm{Vr}_{k'}$ must hold. A tile on an inner ring $r_{j'}$ (close to the center of an FOV) should have a cumulative vote count ($Vr_{j'}$) greater than the cumulative vote count ($Vr_{k'}$) of an adjacent tile on a comparatively outer ring $r_{k'}$ in the FoV.

Proof

If there are x FoVs in a frame in the video and the proportion of viewers looking at FoV i be p_i, i.e., $\sum_{i=1}^{x} p_i$ = 1 then using the method of voting with Borda Count, each FoV i will have vote counts on tiles on every ring around the center in an arithmetic progression decreasing as we go further away from the center. Tiles on the center ring will have r votes each, tiles on the first ring around the center will have $r-1$ votes, decreasing all the way to the outermost ring where each tile will get just one vote.

Actual Votes on tiles on each ring in FoV i:

$$r, r-1, r-2, \ldots, 2, 1$$

Normalized Votes on tiles on each ring in FoV i:

$$p_i * r, p_i * (r-1), p_i * (r-2), \ldots, p_i * 2, p_i * 1$$

			FoV 1							FoV 2			
	1 * P1	1 * P1	1 * P1	1 * P1	1 * P1			1 * P2	1 * P2	1 * P2	1 * P2	1 * P2	
	1 * P1	2 * P1	2 * P1	2 * P1	1 * P1			1 * P2	2 * P2	2 * P2	2 * P2	1 * P2	
	1 * P1	2 * P1	3 * P1	2 * P1	1 * P1			1 * P2	2 * P2	3 * P2	2 * P2	1 * P2	
	1 * P1	2 * P1	2 * P1	2 * P1	1 * P1			1 * P2	2 * P2	2 * P2	2 * P2	1 * P2	
	1 * P1	1 * P1	1 * P1	1 * P1	1 * P1			1 * P2	1 * P2	1 * P2	1 * P2	1 * P2	

Figure 12: Votes of tiles of non-intersecting FoVs using Borda Count

Without any overlap among FoVs the correctness condition holds true based on the voting criteria. This is because $\mathrm{Vr}_{\mathrm{j}'}$, the cumulative normalized vote of a tile on an inner ring $r_{j'}$ is always greater than $\mathrm{Vr}_{\mathrm{k}'}$, the cumulative normalized vote of a tile on an outer ring $r_{k'}$.

			FoV 1				FoV 2			
	1 * P1	1 * P1	1 * P1	1 * P1	P1 + P2	1 * P2	1 * P2	1 * P2	1 * P2	
	1 * P1	2 * P1	2 * P1	2 * P1	P1 + P2	2 * P2	2 * P2	2 * P2	1 * P2	
	1 * P1	2 * P1	3 * P1	2 * P1	P1 + P2	2 * P2	3 * P2	2 * P2	1 * P2	
	1 * P1	2 * P1	2 * P1	2 * P1	P1 + P2	2 * P2	2 * P2	2 * P2	1 * P2	
	1 * P1	1 * P1	1 * P1	1 * P1	P1 + P2	1 * P2	1 * P2	1 * P2	1 * P2	

Figure 13: Votes of tiles of intersecting FoVs using Borda Count

If there are y FoVs among those x FoVs in the frame that overlap with each other and form a cluster, $2 \leq y \leq x$. Let an FoV j in those intersecting FoVs have an overlap with all other $y - 1$ FoVs and that overlap is up to the k^{th} ring from the center, that is the distance between the center of the FoV j and all other intersecting FoVs is $r + k$. The tiles on the overlapping section of FoV j would receive votes from (i). The FoV j itself for being a part of it and (ii). The vote from the outermost ring of the other $y - 1$ FoVs overlapping with it. The vote the tiles on the $k + 1^{th}$ ring will receive for being a part of FoV j is $p_j * (r - k - 1)$, and the vote cast by the outermost ring of the other $y - 1$ FoVs is $\sum_{i \in y,\ i \neq j} p_i * 1$. The vote on the immediate inner ring of this overlapping section will have a vote of $p_j * (r - k)$ for being a part of FoV j, and no other votes since there is no overlap on that ring.

The maximum cumulative vote count in the overlapping section is –

$$\mathrm{Vr}_{k'} = p_j * (r - k - 1) + \sum_{i \in y,\ i \neq j} p_i * 1, \text{ where } j \in y$$

Cumulative Sum of votes of tiles on the immediate inner ring

$$\mathrm{Vr}_{j'} = p_j * (r-k), \text{ where } j \in y$$

For the correctness condition to hold –

$$Vr_{j'} \geq \mathrm{Vr}_{k'}$$

Solving the inequality,

$$p_j * (r - k) \geq p_j * (r - k - 1) + \sum_{i \in y,\ i \neq j} p_i * 1$$

$$p_j \geq \sum_{i \in y,\ i \neq j} p_i$$

So, the proportion of viewers watching each FoV that overlap in the cluster must be greater than or equal to the proportion of viewers watching the other videos in the cluster. This is only possible when $y = 2$, that is the cluster is made up of just 2 overlapping FoVs and both the FoVs have equal probabilities of getting viewed. So, the correctness condition stands good if any overlap of FoVs in the frame is constituted by maximum 2 FoVs i and j, and probabilities of these FoVs being viewed are equal, $p_i = p_j$. Otherwise, the correctness condition is not satisfied.

4.3 Voting using Dowdall Rule

In the voting method using Dowdall Rule, initially all $M\ x\ N$ tiles in every frame in the video has a vote of 0 associated with them. Each time a tile is in the FoV, it gets a vote. All tiles on the center ring get 1 vote each, all tiles on the immediate next outer ring get a vote of $1/2$, and the tiles on the next outer ring get a vote of $1/3$ and so on, the tiles on the outermost peripheral ring gets a vote of $1/r$. This method has both the factors mentioned earlier needed to ensure good QoE and effective usage of caching storage.

Lemma

For the solution using Dowdall rue to be correct, the correctness condition $\mathrm{Vr}_{j'} \geq \mathrm{Vr}_{k'}$ must hold. A tile on an inner ring $r_{j'}$ (close to the center of an FOV) should have a cumulative vote count ($Vr_{j'}$) greater than the cumulative vote count ($Vr_{k'}$) of an adjacent tile on a comparatively outer ring $r_{k'}$ in the FoV.

Proof

If there are x FoVs in a frame in the video and the proportion of viewers looking at FoV i be p_i, i.e., $\sum_{i=1}^{x} p_i = 1$ then using the method of voting with Dowdall rule, each FoV i will have vote counts on tiles on every ring around the center in a harmonic progression decreasing as we go further away from the center. Tiles on the center ring will get a vote of 1 each, tiles on the first ring around the center will get a vote count of 1/2, decreasing all the way to the outermost ring where each tile will get $1/r$ vote.

Actual Votes on tiles on each ring in FoV i:

$$1, 1/2, 1/3, \ldots, 1/(r-1), 1/r$$

Normalized Votes of tiles on each ring in FoV i:

$$p_i * 1, p_i * 1/2, p_i * 1/3, \ldots, p_i * 1/(r-1), p_i * 1/r$$

			FoV 1							FoV 2			
	P1/3	P1/3	P1/3	P1/3	P1/3			P2/3	P2/3	P2/3	P2/3	P2/3	
	P1/3	P1/2	P1/2	P1/2	P1/3			P2/3	P2/2	P2/2	P2/2	P2/3	
	P1/3	P1/2	P1	P1/2	P1/3			P2/3	P2/2	P2	P2/2	P2/3	
	P1/3	P1/2	P1/2	P1/2	P1/3			P2/3	P2/2	P2/2	P2/2	P2/3	
	P1/3	P1/3	P1/3	P1/3	P1/3			P2/3	P2/3	P2/3	P2/3	P2/3	

Figure 14: Votes of tiles of non-intersecting FoVs using the Dowdall rule

			FoV 1				FoV 2			
	P1/3	P1/3	P1/3	P1/3	(P1 + P2)/3	P2/3	P2/3	P2/3	P2/3	
	P1/3	P1/2	P1/2	P1/2	(P1 + P2)/3	P2/2	P2/2	P2/2	P2/3	
	P1/3	P1/2	P1	P1/2	(P1 + P2)/3	P2/2	P2	P2/2	P2/3	
	P1/3	P1/2	P1/2	P1/2	(P1 + P2)/3	P2/2	P2/2	P2/2	P2/3	
	P1/3	P1/3	P1/3	P1/3	(P1 + P2)/3	P2/3	P2/3	P2/3	P2/3	

Figure 15: Votes of tiles of intersecting FoVs using the Dowdall rule

Without any overlap among FoVs the correctness condition holds true based on the voting criteria. This is because $\mathrm{Vr}_{j'}$, the cumulative normalized vote of a tile on an inner ring $r_{j'}$ is always greater than $\mathrm{Vr}_{k'}$, the cumulative normalized vote of a tile on an outer ring $r_{k'}$.

If there are y FoVs of those x FoVs in the frame that overlap with each other and form a cluster, $2 \leq y \leq x$. Let an FoV j in those intersecting FoVs have an overlap with all other $y-1$ FoVs in such a manner that there is no overlap is up to the k^{th} ring from the center. The overlap starts from the $k+1^{th}$ ring of FoV j where the peripheral ring of all the other $y-1$ FoVs overlap. The distance between the center of the FoV j and all other intersecting FoVs is $r+k$. The tiles on the overlapping section of FoV j would receive votes from (i). The FoV j itself for being a part of it and (ii). The vote from the outermost ring of the other $y - 1$ FoVs overlapping with it. The vote the tiles on the $k+1^{th}$ ring will receive for being a part of FoV j is $p_j * 1/(k+1)$, and the vote cast by the outermost ring of the other $y-1$ FoVs is $\sum_{i \in y, i \neq j} (p_i * 1/r)$. The vote on the immediate inner ring of this overlapping section will have a vote of $p_j * 1/k$ for being a part of FoV j, and no other votes since there is no overlap on that ring.

The maximum cumulative vote count in the overlapping section is –

$$\mathrm{Vr}_{\mathrm{k}'} = p_j * 1/(k+1) + \sum_{i \in y, i \neq j} (p_i * 1/r), \text{ where } j \in y$$

Cumulative Sum of votes of tiles on the immediate inner ring

$$\mathrm{Vr}_{\mathrm{j}'} = p_j * 1/k, \text{ where } j \in y$$

For the correctness condition to hold –

$$Vr_{j'} \geq \mathrm{Vr}_{\mathrm{k}'}$$

$$p_j * 1/k \geq \sum_{i \in y, i \neq j} (p_i * 1/r) + p_j * 1/(k+1), \text{ where } j \in y$$

$$p_j\big(1/k - 1/(k+1)\big) \geq \sum_{i \in y, i \neq j} p_i * 1/r$$

$$p_j * 1/(k * (k + 1)) \geq \sum_{i \in y, i \neq j} p_i * 1/r$$

$$r/(k * (k + 1)) \geq \sum_{i \in y, i \neq j} p_i/p_j$$

$$r/(k * (k + 1)) \geq 1/p_j, \text{ since } \sum_{i \in y, i \neq j} p_i < 1$$

$$r \geq k * (k + 1) * 1/p_j$$

Solving for r, we get

$$r \geq k * (k + 1) * Max(1/p_j)$$

So, the correctness condition holds good if $\boldsymbol{r \geq k * (k + 1) * Max(1/p_j)}$ that is if we can modify the radius of the FoV dynamically, we will be able to ensure that the tiles on the center ring always get higher vote than the tiles on the peripheral ring, and thereby work towards providing good viewer QoE.

4.4 Vote Decay Method

In the Vote Decay method-based Voting, initially all $M\ x\ N$ tiles in every frame in the video has a vote of 0 associated with them. Each time a tile is in the FoV, it gets a vote. All tiles on the center ring get a 1 vote, all tiles on the immediate next outer ring get a vote of $1/q$, and the next outer ring get a vote of $1/q^2$ and so on, the tiles on the outermost peripheral ring get a vote of $1/q^{(r-1)}$. This method has both the factors mentioned earlier needed to ensure good QoE and effective usage of caching storage.

Lemma

For the solution using Vote Decay method to be correct, the correctness condition $\mathrm{Vr}_{\mathrm{j}'} \geq \mathrm{Vr}_{\mathrm{k}'}$ must hold. A tile on an inner ring $r_{j'}$ (close to the center of an FOV) should have a cumulative vote count ($Vr_{j'}$) greater than the cumulative vote count ($Vr_{k'}$) of an adjacent tile on a comparatively outer ring $r_{k'}$ in the FoV.

Proof

If there are x FoVs in a frame in the video and the proportion of viewers looking at FoV i be p_i, i.e., $\sum_1^x \ p_i$ =1, then using Vote Decay based Voting, each FoV i will have vote counts on tiles on every ring around the center of the FoV in geometric progression with a common ratio of q, decreasing as we go further away from the center. Tiles on the center ring will get 1 vote each, tiles on one ring around the center will get a vote count of $1/q$, decreasing all the way to the outermost ring where each tile will get $1/q^{(r-1)}$ vote. This common ratio q is the decay factor, the factor by which the count of votes decay as we move one ring away from the center of the FoV towards the periphery.

Actual Votes on tiles on each ring in FoV i:

$$1, 1/q, 1/q^2, \ldots, 1/q^{(r-2)}, 1/q^{(r-1)}$$

Normalized Votes on tiles on each ring in FoV i:

$$p_i * 1, p_i * 1/q, p_i * 1/q^2, \ldots, p_i * 1/q^{(r-2)}, p_i * 1/q^{(r-1)}$$

			FoV 1							FoV 2			
	P1/q^2	P1/q^2	P1/q^2	P1/q^2	P1/q^2			P2/q^2	P2/q^2	P2/q^2	P2/q^2	P2/q^2	
	P1/q^2	P1/q	P1/q	P1/q	P1/q^2			P2/q^2	P2/q	P2/q	P2/q	P2/q^2	
	P1/q^2	P1/q	P1	P1/q	P1/q^2			P2/q^2	P2/q	P2	P2/q	P2/q^2	
	P1/q^2	P1/q	P1/q	P1/q	P1/q^2			P2/q^2	P2/q	P2/q	P2/q	P2/q^2	
	P1/q^2	P1/q^2	P1/q^2	P1/q^2	P1/q^2			P2/q^2	P2/q^2	P2/q^2	P2/q^2	P2/q^2	

Figure 16: Votes of tiles of non-intersecting FoVs using the Vote Decay Method

			FoV 1				FoV 2			
	P1/q^2	P1/q^2	P1/q^2	P1/q^2	(P1 + P2)/q^2	P2/q^2	P2/q^2	P2/q^2	P2/q^2	
	P1/q^2	P1/q	P1/q	P1/q	(P1 + P2)/q^2	P2/q	P2/q	P2/q	P2/q^2	
	P1/q^2	P1/q	P1	P1/q	(P1 + P2)/q^2	P2/q	P2	P2/q	P2/q^2	
	P1/q^2	P1/q	P1/q	P1/q	(P1 + P2)/q^2	P2/q	P2/q	P2/q	P2/q^2	
	P1/q^2	P1/q^2	P1/q^2	P1/q^2	(P1 + P2)/q^2	P2/q^2	P2/q^2	P2/q^2	P2/q^2	

Figure 17: Votes of tiles of intersecting FoVs using the Vote Decay Method

Without any overlap among FoVs the correctness condition holds true based on the voting criteria. This is because $\mathrm{Vr}_{j'}$, the cumulative normalized vote of a tile on an inner ring $r_{j'}$ is always greater than $\mathrm{Vr}_{k'}$, the cumulative normalized vote of a tile on an outer ring $r_{k'}$.

If there are y FoVs of those x FoVs in the frame that overlap with each other and form a cluster, $2 \leq y \leq x$. Let an FoV j in those intersecting FoVs have an overlap with all other $y - 1$ FoVs in such a manner that there is no overlap till the k^{th} ring of the FoV j from the center, but on the next ring of FoV j, the peripheral rings of all other $y - 1$ FoVs intersect. That is the distance between the center of the FoV j and all other FoVs that intersect with it is $r + k$. The tiles on the overlapping section of FoV j would receive votes from (i). The FoV j itself for being a part of it and (ii). The vote from the outermost ring of the other $y - 1$ FoVs overlapping with it. The vote the tiles on the $k + 1^{th}$ ring will receive for being a part of FoV j is $p_j * 1/q^k$, and the vote cast by the outermost ring of the other $y - 1$ FoVs is $\sum_{i \in y, i \neq j} \left(\mathrm{p_i} * 1/\mathrm{q}^{(\mathrm{r}-1)}\right)$. The vote on the immediate inner ring of this overlapping section will have a vote of $p_j * 1/q^{(k-1)}$ for being a part of FoV j, and no other votes since there is no overlap on that ring.

In this scenario, the Cumulative Sum of votes of tiles on the immediate inner ring:

$$\mathrm{Vr_{j'}} = p_j * 1/q^{(k-1)}$$

This cumulative vote count of tiles on the overlapping section is

$$\mathrm{Vr_{k'}} = p_j * 1/q^k + \sum_{i \in y, i \neq j} p_i * 1/q^{(r-1)}$$

This cumulative vote count of tiles on the overlapping section that is the tiles on the outer ring should be less than or equal to the vote of tiles on the immediate inner ring.

$$Vr_{j'} \geq \mathrm{Vr_{k'}}$$

$$p_j * 1/q^k + \sum_{i \in y, i \neq j} p_i * 1/q^{(r-1)} \leq p_j * 1/q^{(k-1)}$$

Solving the inequation,

$$\sum_{i \in y, i \neq j} p_i * 1/q^{(r-1)} \leq p_j * 1/q^{(k-1)} - p_j * 1/q^k$$

$$\sum_{i \in y, i \neq j} p_i * 1/q^{(r-1)} \leq \left(p_j * q - p_j\right)/q^k$$

$$\sum_{i \in y, i \neq j} p_i * 1/q^{(r-1)} \leq p_j * (q - 1)/q^k$$

$$\sum_{i \in y, i \neq j} p_i * 1/q^{(r-k-1)} \leq p_j * (q-1)$$

$$q^{(r-k-1)} * (q-1) \geq \sum_{i \in y, i \neq j} p_i/p_j$$

Now since $1/p_j \geq p_i/p_j$, if we prove that for some value of q, $q^{(r-k-1)} * (q-1) \geq 1/p_j$ we can say that the correctness condition holds true. On solving for q, we see that if $\boldsymbol{q \geq Max(1/p_j)}$, for all $j \in y$ then the correctness condition is satisfied.

CHAPTER 5

CONCLUSION

From the above results we conclude that traditional heat maps would work best in cases where the tiles in the intersection are more important candidates to cache in the mobile edge cloud than the tiles at the center of the FoVs. Traditional heat maps are not ideal to ensure QoE in cases where the center of the FoV is considered more popular than peripheries of the FoV no matter how many FoVs overlap. Voting using the Borda Count satisfies the correctness condition only if there is a maximum of two FoVs overlapping with each other in a cluster, and the proportion of users viewing those two FoVs are the same. If these conditions are not met Voting using Borda Count fails to address the QoE and effective usage of cache space. Voting using the Dowdall rule can cater to both these factors if we are able to modify the radius r of the FoV dynamically and maintain $\boldsymbol{r \geq k * (k+1) * Max(1/p_j)}$. The Vote Decay based voting is best in terms of providing good QoE and edge cache space utilization since we just need to modify the decay factor q such that $\boldsymbol{q \geq Max(1/p_j)}$ to meet the correctness condition. As and when proportion of viewers viewing the least popular cluster that has been cached changes with time the value of q would also change and all votes past or present needs to be recalculated with the updated value. This would ensure that the solution is correct.

BIBLIOGRAPHY

[1] Yixuan Ban, Lan Xie, Zhimin Xu, Xinggong Zhang, Zongming Guo, and Yue Wang. 2018. Cub360: Exploiting cross-users behaviors for viewport prediction in 360-degree video adaptive streaming. In 2018 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 1–6.

[2] Xavier Corbillon, Gwendal Simon, Alisa Devlic, and Jacob Chakareski. 2017. Viewport-adaptive navigable 360-degree video delivery. In 2017 IEEE international conference on communications (ICC). IEEE, 1–7.

[3] Peter Emerson. 2013. The original Borda count and partial voting. Social Choice and Welfare 40, 2 (2013), 353–358.

[4] Ching-Ling Fan, Jean Lee, Wen-Chih Lo, Chun-Ying Huang, Kuan-Ta Chen, and Cheng-Hsin Hsu. 2017. Fixation prediction for 360-degree video streaming in head-mounted virtual reality. In Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and Video. 67–72.

[5] Jon Fraenkel and Bernard Grofman. 2014. The Borda Count and its real-world alternatives: Comparing scoring rules in Nauru and Slovenia. Australian Journal of Political Science 49, 2 (2014), 186–205.

[6] Vamsidhar Reddy Gaddam, Michael Riegler, Ragnhild Eg, Carsten Griwodz, and Pål Halvorsen. 2016. Tiling in interactive panoramic video: Approaches and evaluation. IEEE Transactions on Multimedia 18, 9 (2016), 1819–1831.

[7] Michael Gallagher. 1992. Comparing proportional representation electoral systems: Quotas, thresholds, paradoxes and majorities. British Journal of Political Science (1992), 469–496.

[8] Grigorii V Golosov. 2010. The effective number of parties: A new approach. Party politics 16, 2 (2010), 171–192.

[9] Mario Graf, Christian Timmerer, and Christopher Mueller. 2017. Towards bandwidth efficient adaptive streaming of omnidirectional video over http: Design, implementation, and evaluation. In Proceedings of the 8th ACM on Multimedia Systems Conference. 261–271.

[10] Bernard Grofman, Scott L Feld, and Jon Fraenkel. 2017. Finding the Threshold of Exclusion for all single seat and multi-seat scoring rules: Illustrated by results for the Borda and Dowdall rules. Mathematical Social Sciences 85 (2017), 52–56.

[11] Mohammad Hosseini and Viswanathan Swaminathan. 2016. Adaptive 360-degree VR video streaming based on MPEG-DASH SRD. In 2016 IEEE International Symposium on Multimedia (ISM). IEEE, 407–408.

[12] Xueshi Hou, Sujit Dey, Jianzhong Zhang, and Madhukar Budagavi. 2018. Predictive View Generation to Enable Mobile 360-degree and VR Experiences. In Proceedings of the 2018 Morning Workshop on Virtual Reality and Augmented Reality Network. ACM, 20–26.

[13] Markku Laakso and Rein Taagepera. 1979. "Effective" number of parties: a measure with application to West Europe. Comparative political studies 12,1(1979), 3–27.

[14] Chenge Li, Weixi Zhang, Yong Liu, and Yao Wang. 2019. Very Long Term Field of View Prediction for 360-degree Video Streaming. In 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). IEEE, 297–302.

[15] Afshin Taghavi Nasrabadi, Anahita Mahzari, Joseph D Beshay, and Ravi Prakash. 2017. Adaptive 360-degree video streaming using scalable video coding. In Proceedings of the 25th ACM international conference on Multimedia. 1689–1697.

[16] Afshin Taghavi Nasrabadi, Aliehsan Samiei, Anahita Mahzari, Ryan P McMahan, Ravi Prakash, Mylène CQ Farias, and Marcelo M Carvalho. 2019. A taxonomy and dataset for 360° videos. In Proceedings of the 10th ACM Multimedia Systems Conference. 273–278.

[17] Afshin Taghavi Nasrabadi, Aliehsan Samiei, and Ravi Prakash. 2020. Viewport prediction for 360-degree videos: a clustering approach. In Proceedings of the 30th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video. 34–39.

[18] Anh Nguyen, Zhisheng Yan, and Klara Nahrstedt. 2018. Your attention is unique: Detecting 360-degree video saliency in head-mounted display for head movement prediction. In Proceedings of the 26th ACM international conference on Multimedia. 1190–1198.

[19] Jounsup Park and Klara Nahrstedt. 2019. Navigation graph for tiled media streaming. In Proceedings of the 27th ACM International Conference on Multimedia. 447–455.

[20] Jounsup Park, Mingyuan Wu, Kuan-Ying Lee, Bo Chen, Klara Nahrstedt, Michael Zink, and Ramesh Sitaraman. 2020. SEAWARE: Semantic Aware View Prediction System for 360-degree Video Streaming. In 2020 IEEE International Symposium on Multimedia (ISM). IEEE, 57–64.

[21] Hasti A Pedersen and Sujit Dey. 2015. Enhancing mobile video capacity and quality using rate adaptation, RAN caching and processing. IEEE/ACM Transactions on Networking 24, 2 (2015), 996–1010.

[22] Stefano Petrangeli, Viswanathan Swaminathan, Mohammad Hosseini, and Filip De Turck. 2017. An HTTP/2-based adaptive streaming framework for 360-degree virtual reality videos. In Proceedings of the 25th ACM international conference on Multimedia. 306–314.

[23] Ravi Prakash and Ehsan Nourbakhsh. 2011. Gradient-based Aggregation in Forest of Sensors (GrAFS). In 2011 International Conference on Parallel Processing. IEEE, 120–129.

[24] Feng Qian, Lusheng Ji, Bo Han, and Vijay Gopalakrishnan. 2016. Optimizing 360-degree video delivery over cellular networks. In Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges. 1–6.

[25] Stephen A Rhoades. 1993. The Herfindahl-Hirschman index. Fed. Res. Bull. 79 (1993), 188.

[26] Patrice Rondao Alface, Maarten Aerts, Donny Tytgat, Sammy Lievens, Christoph Stevens, Nico Verzijp, and Jean-Francois Macq. 2017. 16k cinematic VR streaming. In Proceedings of the 25th ACM international conference on Multimedia. 1105–1112.

[27] Jean-Jacques Rousseau. 1964. The Social Contract (1762).

[28] Rein Taagepera and Bernard Grofman. 1981. Effective size and number of components. Sociological Methods & Research 10, 1 (1981), 63–81.

[29] Chenglei Wu, Zhihao Tan, Zhi Wang, and Shiqiang Yang. 2017. A dataset for exploring user behaviors in VR spherical video streaming. In Proceedings of the 8th ACM on Multimedia Systems Conference. 193–198.

[30] Lan Xie, Xinggong Zhang, and Zongming Guo. 2018. CLS: A Cross-user Learning based System for Improving QoE in 360-degree Video Adaptive Streaming. In 2018 ACM Multimedia Conference on Multimedia Conference. ACM, 564–572.

[31] Mai Xu, Yuhang Song, Jianyi Wang, MingLang Qiao, Liangyu Huo, and Zulin Wang. 2018. Predicting head movement in panoramic video: A deep reinforcement learning approach. IEEE transactions on pattern analysis and machine intelligence 41, 11 (2018), 2693–2708.

[32] Anahita Mahzari, Afshin Taghavi Nasrabadi, Aliehsan Samiei and Ravi Prakash. 2018. FoV-Aware Edge Caching for Adaptive 360◦ Video Streaming. In MM’18: 2018 ACM Multimedia Conference. ACM, 173-181.

[33] Aliehsan Samiei, Ravi Prakash 2021 - Viewers Vote with Their Eyes: Novel Techniques to Improve 360-Degree Video Field-of-View Prediction and Edge Caching. ACM 2021

BIOGRAPHICAL SKETCH

Novonil Das was born in Kolkata, West Bengal, India. He completed his schoolwork at South End School in Kolkata in 2009. Novonil received a Bachelor of Technology degree from the West Bengal University of Technology in Kolkata in 2013 with a major in Electronics and Communication Engineering. He was employed as a Full Stack Developer at Odessa Inc. for the next six years where he was working in the capacity of a Module Lead. In August 2019, he entered the computer science graduate program at The University of Texas at Dallas.

CURRICULUM VITAE

NOVONIL DAS

EDUCATION

Master of Science in Computer Science Aug 2021
UNIVERSITY OF TEXAS AT DALLAS, USA **GPA 3.93/4.0**

Bachelor of Technology in Electronics & Communication Engineering June 2013
WEST BENGAL UNIVERSITY OF TECHNOLOGY, INDIA **GPA 8.16/10.00**

RESEARCH

Field Of View Aware Edge Caching for Adaptive 360° Video Streaming

Reduce bandwidth requirement, and network latency and load on the content servers by edge caching popular tiles in 360° videos, based on how frequently they appear in user's FOV from past user's viewing histories, using voting protocols to build heat maps and form clusters for caching.

WORK EXPERIENCE

Software Developer Intern, CenturyLink Inc., US Jun 2020 – Aug 2020

- Developed new and enhanced existing modules in a MEAN stack. Followed, optimized existing development processes.
- Created an API endpoint to replace combining responses from multiple APIs, improving efficiency of a module by 40%.
- Debugged and solved existing issues by collaborating with team members. Participated in peer design and code reviews.

Module Lead, Odessa Inc., India Jul 2017 – Jun 2019

- Researched and designed a data warehouse, developed performance tuned ETLs using SSIS to load data onto it. Created efficient procedures in SQL to create dynamic/static visualization reports on Tableau, SSRS.
- Automated SQL script generation using C# while building the warehouse, allowing developers to save 20% of their time.
- Drove a project to successful go-live by leading a team of 7 members, setting examples by taking up responsibilities for the team, planning deliverables, taking stock of situations, removing bottlenecks, working cohesively in and across teams.
- Led and mentored a team of 4 developers in another project simultaneously, successfully delivering releases to the client.
- Prioritized work, gathered requirements by coordinating with clients, trained them, provided workarounds, fixed issues.

Senior Software Engineer & Software Engineer, Odessa Inc., India Jun 2013 – Jun 2017

- Developed critical modules in the web application, built efficient reports, designed jobs, created UI, coded client-side scripts, wrote business logic, troubleshot bugs, analyzed large data volumes, provided code fixes, deployed deliverables.
- Implemented and delivered diverse client requirements by traveling onshore to the client location.