QoE Aware Video Content Adaptation and Delivery

Pavan Kamaraju*, Pietro Lungaro† and Zary Segall†
*Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, USA
†Mobile Service Lab, Royal Institute of Technology (KTH), Kista, Sweden
Email: pavan4@umbc.edu, pietro@kth.se, segall@kth.se

***Abstract*—The explosion of traffic associated with video content poses significant challenges for mobile content provision. While, on the one hand, mobile video traffic surge is forecast-ed to require significant investments in bandwidth acquisition and infrastructure dimensioning and roll-out, on the other hand, users are not likely to be willing to pay significantly more than today. This increases the pressure to develop solutions capable of making the mobile provision of video more affordable without either affecting user experience or limiting usage.**

In this respect, this paper proposes a novel methodology for video content delivery which is based on a user video quality perception model. According to this scheme, the video quality of each scene in a movie is selected, from among a finite set of available qualities, with the purpose of reducing the overall bandwidth required to attain a given user experience level targeted by the system for each user and each video. This novel methodology also adopts a clustering approach to identify users with similar Quality of Experience (QoE) profiles and leverages this information for improving the accuracy of user perceived quality predictions. This approach has been validated through a crowd-sourced subjective test evaluation performed with real users using a novel method involving the Amazon Mechanical Turk platform. The results showed that the proposed method is capable of achieving a prediction accuracy in the order of ±0.5 MOS points. This approach can be effectively used to select the video qualities minimizing bandwidth costs while delivering predefined level of perceived quality to the end users.

I. Introduction

Global mobile data traffic is increasing at an unprecedented rate and video traffic alone currently constitutes about 50% of the total traffic. Video is predicted to grow up to 70% of the total traffic by 2021 [1] as user preferences are shifting towards more video based applications. Increasing screen sizes and resolutions on user devices, faster networks, increased data usage are feeding the growth. At the same time, users are not likely to be willing to pay significantly more than today, which poses a challenge to the operators for managing and maintaining a sustained demand. In response, mobile operators are attempting to charge over-the-top content providers like Netflix for smoother delivery of their traffic.

Wireless network capacity is a shared and expensive resource among the active users of a cell. Video content is expensive in terms of bandwidth and requires constant availability of resources for experiencing smooth playback of high quality content. However, this might pose a challenge to a mobile user because of his/her mobility and/or unavailability of capacity due to many active users in the cell. Also, new devices having higher resolutions and supporting [ultra] high definition video are entering the market and creating additional challenges in terms of dimensioning for the operators. All these factors are likely to increase the OPEX and CAPEX of the operators and decrease the gap between revenues and costs. Furthermore, user expectations for high quality video is constantly increasing [2] motivating the need for developing novel solutions that can sustain the growth of traffic.

978-1-5090-2185-7/16/$31.00

For many years, video content was delivered over UDP/RTP and was prone to disruptions over best-effort IP networks. Current video delivery techniques support adaptive streaming of video over HTTP using TCP. Firstly, this simplifies the design of streaming applications as video players can now adapt to throughput variations and secondly, it also simplifies video delivery over firewall and NATs. Typically, using dynamic adaptive streaming, over HTTP (DASH), video content is served in a best-effort manner by switching between qualities (bitrates) and delivering the maximum bitrate possible based on available bandwidth. However, this is prone to frequent fluctuations in quality and an unpredictable QoE and there is room for significant improvement in the rate-adaptation logic used by client players [3].

In this paper, we study video content delivery based on user perception.The proposed approach optimizes a video composition by selecting the quality (bitrate) of individual scenes based on a quality threshold. In doing so, we limit maximum bitrates of requested scenes and also optimize the overall bandwidth required. A clustering approach is adopted to identify and group users with similar viewing behavior for prediction of Quality of Experience (QoE). Using this method, we were able to (1) eliminate the noticeable video quality switches introduced by DASH (2) optimize the utilization of network bandwidth and (3) predict and deliver personalized video content based on user preference. We validated our approach using real users by performing subjective quality evaluations based on a novel crowd-sourced method involving Amazon Mechanical Turk platform.

A. Problem Statement

In this paper, we focus on the problem of optimizing the video quality composition of a video with the purpose of delivering predictable QoE to mobile users, while reducing overall content delivery bandwidth costs to achieve those QoE levels. In order to improve the tractability of this problem, three separate problem sub-areas have been considered. In the first one, the challenge is to devise a method to characterize the various scenes constituting a movie and to select among alter-

native video qualities those that can achieve a pre-determined quality level (quality factor).

Once movies are composed according to the aforementioned approach, the question is to understand how to map this constant quality factor into a user experience value for each of the users. How does QoE vary for different users, in function of this quality factor? Can users be grouped together and information from the groups effectively used to infer and predict quality perception for new users?

Finally, an important aspect of our investigation is to define and implement a scalable approach to perform testing with real users and assess the performances of our proposed solution.

B. Contributions

Our contributions can be summarized as follows:

1) A systematic methodology for modifying video content based on user perception, determined by VQM.
2) Validation using subjective quality testing with users using crowd sourced and personal interviews.
3) Methodology to predict QoE that utilizes collaborative filtering techniques to group users based on similarities in their viewing behavior.

The remainder of the paper is organized as follows: In Section II we describe the related work and in Section III we describe video perception and how it can be extracted as an objective metric. In Section IV we describe the methodology for video optimization based on user perception and the QoE prediction method. Section V describes the findings and finally, we conclude with Section VI.

II. Related Work

The investigations in [4], [5] studied the complexities of predicting QoE for internet video and proposed a machine learning based data-driven approach using user engagement metrics for predicting QoE. However, their dataset did not include actual QoE measurement from the users and instead they use indirect measures to infer this information. They also argued that performing user studies and validating perceptual scores given by users under a controlled setting may not translate into measures of user engagement in the wild. For this reason, in our study we propose a method to test in the wild through a mock up app for selecting the QoE information.

The investigations in [3], [6], [7] studied and evaluated current adaptive streaming players and shows that there is scope for significant improvement in the adaptation schemes for delivering content. In particular, the interaction between the rate-adaptation logic and TCP congestion control is not well understood and can cause undesired QoE fluctuations during playback. Our method can help video player designers to develop bitrate-adaptation schemes which are quality-aware as opposed to being based purely on network parameters. Using our video optimization method, we equalize the quality of the video on a per-scene basis and this helps the client player to deliver content for a specific quality target.

In [8], a method to estimate the quality of video streaming using scene characteristics was proposed while in [9] used cluster analysis was used to classify content into groups and predict video quality within a group. [10] studied and proposed a method for video quality prediction based on classification of video using spatial and temporal feature extraction. All the above investigations focus on classification or clustering based on content characteristics. However, results from our investigations revealed that even for the same content, the perceived quality varied widely. Our work focuses on clustering the users based on the similarities of their personal viewing behavior rather than content.

III. Video Perception and Quality Metrics

A. Perceptual Quality of Users

Perceptual video quality is commonly captured using Mean Opinion Score (MOS) [11], a five point scale used to rate absolute and relative quality of multimedia content. Absolute video quality refers to evaluating video quality without a reference video, while relative video quality represents quality degradation of impaired video when compared to a reference video. Video sequences are shown to a panel of users, whose opinion is recorded and averaged into MOS. This procedure is referred to as subjective evaluation video quality test. Subjective video quality tests are the most accurate method to measure the perceptual video quality. Based on the tests performed in the lab we observed that users behave differently for content encoded at the same quality and bitrate settings as illustrated in Fig. 1.[1]

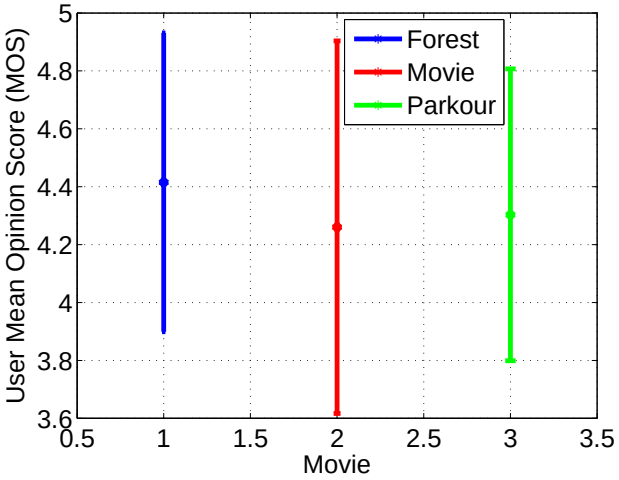


Fig. 1: Mean and standard deviation (σ) of MOS for 3 different movies encoded with same encoding settings (720p@5Mbps).

We also observed using user surveys that users had different preferences for viewing content. A subset of of users preferred certain video content in low quality, and thereby allowing them to view more content as they had restricted amount of data per month while another subset had strict quality requirements

[1]Please note that this experiment was performed using Absolute Category Rating (ACR) method. We asked users to judge the quality of experience to the best of their knowledge and not judge the content itself.

in terms of the video content. Hence, we can leverage this information to tailor content delivery and to maximize user experience.

B. Video Quality Metrics

Several metrics for video quality exist which include PSNR, SSIM and other variants of these algorithms and VQM was the only model that exceeded 0.9 threshold, given by Pearson correlation coefficient which resulted in standardization of this metric by ANSI and it is also included in ITU-T specifications [12]. Therefore, we chose this measure to compute the perceptual video quality of optimized videos in our work. VQM represents an automated video quality measurement system that is based on linear regression of technology independent parameters closely approximating how people perceive video quality. These parameters are extracted from spatio-temporal (S-T) regions of the video sequence.

VQM takes the source clip and the impaired (reference) clip as input and computes the score using a series of steps. The first step includes division of video into S-T regions and application of perceptual filters to compute the perceptual video quality. In the second step, features are extracted for S-T regions, while in the third step VQM score is calculated by thresholding values obtained from the extracted features. VQM scores have range from 0 to 1, with 0 being closest to the original video source. The "general model" of VQM software [13] was used to compute VQM score, since this model is optimized to achieve maximum correlation between the objective and subjective video quality scores. VQM score is computed for a 4 to 15 seconds long video clip, by temporally and spatially collapsing its behavior, and estimating the worst performance that can be achieved by processing this video.

IV. Video Optimization and QoE Prediction

A. Adaptation of Videos based on Perception

In our previous work [14], [15] movies were split into 15 seconds long clips and ran through VQM software to obtain VQM scores for each video resolution and also for each S-T region. The proposed video optimization works by identifying the appropriate resolution for each region, 6 frames of video, comparing this chunks VQM score in each down scaled resolution (starting with the lowest, 240p) with a VQM threshold, until finding the first score that is lower or equal than the given threshold. If the target score is not found, the highest resolution of the video chunk (720p) is kept. The consecutive video chunks with the same identified resolution are referred to as an optimized segment. With initial experiments, we found that 6 frames (0.15s for a 24fps video) was too short for the adaptation process and the chunk size was needed to be at least 1-2 seconds long.

To apply this optimization method to a more general use case, i.e., videos of longer duration, we modified the algorithm to a scene based procedure, illustrated in Algorithm 1. In this procedure, we use FFmpeg encoder to encode video in 4 different qualities (720p, 480p, 360p and 240p at YouTube recommended bitrate settings). Next, we extract individual scene represented as s_x, programatically using Shotdetect [16]. Note that $s_{240}(i)$ stores the ith scene in 240p resolution, while $\mathcal{S}_{240}$ represent the union of all consecutive scenes in 240p. Next, we use FFmpeg to cut individual scenes, and the VQM software is run on individual scenes [2] to obtain the scene's VQM score. Note that the VQM score of ith scene in 480p resolution is represented as $\nu_{480}(i)$ and ν_{480} is a vector that contains VQM scores for all consecutive scenes of a video with a total of N_s scenes. We define $\hat{\tau}$ as the quality factor threshold used to select a video quality based on the VQM score of each scene. Next, we ran the procedure to select all the scenes that have a VQM score that is lower than the selected threshold (this means higher MOS) into a final scene schedule, represented by $(S_{\hat{\tau}})$. Finally, we merge the respective video resolutions of the selected scenes in respective resolutions using FFmpeg to generate the optimized video.

Algorithm 1 Video Optimization

```
procedure V–OPTIMIZATION(s_x, ν_x, τ̂)
    for each i in [1, ..., N_s] do
        if ν_240(i) ≤ τ̂ then
            S_τ̂(i) = s_240(i);
        else if ν_360(i) ≤ τ̂ then
            S_τ̂(i) = s_360(i);
        else if ν_480(i) ≤ τ̂ then
            S_τ̂(i) = s_480(i);
        else
            S_τ̂(i) = s_720(i);
        end if
    end for
end procedure
```

Fig. 2 shows the optimization procedure with a selected threshold, $\hat{\tau}$ = 0.6, for a video of duration 102.8s (515 S-T regions). We modified the optimization procedure in [14] to be scene based, as we observed from our subjective quality experiments that transitions between qualities are more prominently visible when they occur within a scene.

While infinite values of $\hat{\tau} \in [0, 1]$ are feasible, only a discrete number of output schedules can be generated, depending on the vectors ν_{240}, ν_{360}, ν_{480}, ν_{720} of a specific movie. Thus, in practise we will only consider the smallest values of $\hat{\tau}$ that activate a specific schedule. These are labeled as $\hat{\tau}_m = \{\hat{\tau}_m^1, \hat{\tau}_m^2, ..., \hat{\tau}_m^l\}$, a set consisting of l discrete quality levels.

B. Experiment Setup

The section above describes a method to compose a video based on a quality factor $\hat{\tau}$. Using the described procedure, we can generate video content with different quality thresholds. However, we need to be able to identify which quality to deliver to a particular user. Specifically, we need to be able

[2]Please note that you are limited to a 15 second maximum and 4 second minimum length for computing VQM score. The average scene length is ≥ 4 secs. [17]. If the scene was found to be greater than 15 seconds, it was split in two and if it was less than 4 seconds, it was merged with the next scene in our method.

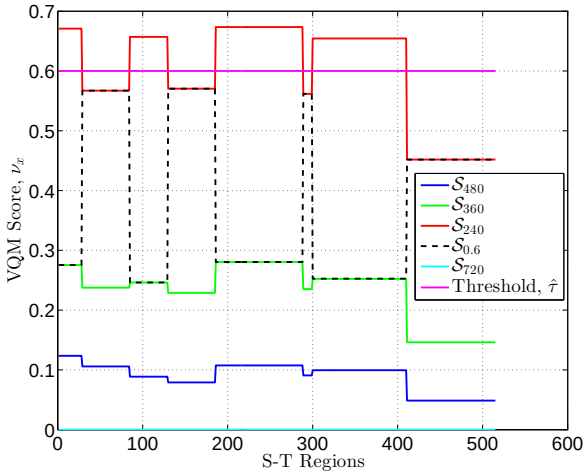


Fig. 2: The red line represents the union of all consecutive scenes in 240p, green represents 360p, blue represents 480p and cyan represents 720p. The values in the distribution are the VQM scores, ν_x of the respective resolutions. The black dotted line represents the final schedule of scenes for threshold, $\hat{\tau}$.

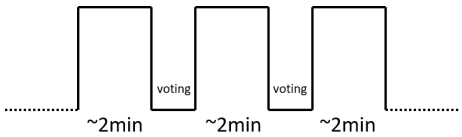


Fig. 3: ACR testing procedure

to select a quality threshold $\hat{\tau}_m^l$ to achieve a target QoE for a specific user. We designed the following experiment for studying the user behavior when they are exposed content with different types of quality factors and also to validate our hypothesis of grouping users based on viewing similarities for creating a personalized video delivery mechanism.

First, using the video optimization method, nine videos were created for use in subjective quality testing. Three different movies were selected and each movie was created in three different perceptual video qualities, indicated by quality threshold. The selected thresholds for optimization and the respective clip along with the associated file size in MB, $b_m(\hat{\tau}_m^l)$ are shown in Table I. High, medium, low indicate easily identifiable label for the specified thresholds.

Next, the optimized videos were used for subjective quality evaluation using Absolute Category Rating (ACR) method. ACR is a single stimulus category judgment method, where the test sequences are presented one at a time and are rated independently on a category scale [18]. The time pattern for the stimulus presentation is illustrated graphically in Fig. 3.

An Android app along with a backend was developed to present the videos and collect user grades using a crowd sourced approach. The order of playback was randomized and user grades were recorded for perceptual video quality.

	Threshold ($\hat{\tau}_m^l$), Size ($b_m(\hat{\tau}_m^l)$) [MB]		
Forest	0, 61.95	0.4, 11.87	0.6, 8.21
Movie	0, 66.01	0.1, 13.17	0.5, 3.26
Parkour	0, 64.90	0.3, 43.76	0.7, 13.73
	high	medium	low

TABLE I: Quality table[3]

The experimental setup consisted of initial lab-controlled tests and later released for online testing via crowd sourcing using the Amazon Mechanical Turk platform. Participating users were awarded a small monetary fee. The experiment comprised of displaying sequentially the nine videos (∼2 min long with no audio) that had been previously composed based on the proposed optimization procedure. After viewing each video, users were presented with a screen to enter a quality grade, for the video. Users were asked to use the following guidelines to grade a video : A grade can be any number between '1' and '5', with '1' being the worst grade and '5' the best. An *acceptable* quality for a user corresponded to a grade '3', while a *good* video quality was a '4'. Finally, a *very good* quality received a '5'. To provide more precise input for our research we strongly encouraged users to select grades including up to one digit after the comma, e.g., choosing '4.2' to indicate a grade slightly above *good*, or '3.5' to grade a video quality between *acceptable* and *good*. We used Samsung Galaxy S3 phones with a maximum supported resolution of 720p, for viewing the videos in the lab-controlled experiments, while the crowd sourced experiments were run remotely on users' own devices. The app [19] could be downloaded from the Android play store. All users were requested to register anonymously on our backend [20]. After the registration process, they could use the created credentials to login to the app and complete the experiments. The clips used for testing were prepared using three popular videos released on Vimeo with creative commons license and these are listed in [20]. Using the setup, a total of 33 users participated in the experiment, out of which 57% were male.

The Amazon Mechanical Turk platform offers a scalable and viable alternative for data collection to an otherwise time consuming task. The platform itself offers anonymous tracking of users for conducting long term experiments and also helps us create filters to ease the recruitment process. The filters can be used to weed out workers who do not match the required quality standards. Filters can be defined on (1) HIT approval rate percentage for workers, where HIT is defined as Human Intelligence Task on the platform (2) Total Approved HITs and (3) Location. We used 95% approval rate and at least 100 completed HITs as filters for our experiments and workers for our experiments were only from US and India although we did not use any filter on location. At the same time, the requesters have to maintain positive reputation among workers by providing feedback on their work and rewarding them in a timely manner. The experiment was expected to take about 23

[3]Note that threshold 0 selects the 720p resolution while a non-zero threshold optimizes the video based on Algorithm 1.

minutes and when the reward was set at $0.5 USD we received some negative feedback from the workers. Thus we increased the reward to $1.0 USD to compensate the workers fairly as it took a longer time than planned for some workers to download the content and complete the experiment. We also found that when we increased the reward fee to $1.5 USD, the time taken to complete the total task decreased. Hence, we believe that by carefully selecting the parameters for user surveys, which include filters and compensation/hr, the Amazon Mechanical Turk platform offers a scalable mechanism to collect data.

C. Methodology for Grouping Users

The recorded grades were analyzed and we describe the findings from the analysis below. Fig. 4 illustrates the MOS and the standard deviation in the distribution of grades in the entire user population for all the videos in Table. I. It can be observed that the standard deviation for all the video grades is high, indicating that there is high variability on the perceived QoE across users. For any given video quality, the standard deviation, σ, was observed to be in between a maximum and minimum of 0.51 and 0.92 respectively.

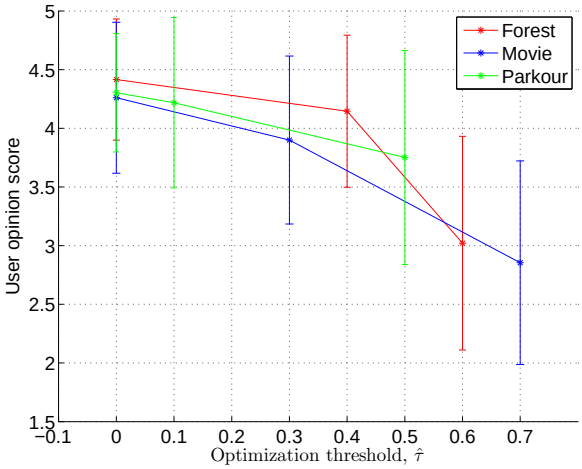


Fig. 4: Mean and error (σ) of MOS on entire population of users for 9 videos generated using the optimization method

By aggregating all the user experiences i.e., when MOS is computed over individual user grades, we compute the mean experience of the overall QoE. However, quality experienced by individual users may vary widely as observed. Fig. 5 illustrates the MOS and its deviation in the distribution, when a subset of users were grouped into two separate groups. Group 1 indicated by a dotted line contained 4 users and Group 2 indicated by the solid line contained 6 different users. By grouping the users based on similarities, there was a significant reduction in the maximum and minimum of standard deviation, σ, which lay between 0.08 to 0.40.

This observation inspired us to propose strategies to group users based on the similarities of their viewing behavior. Alternative methods were proposed and evaluated with respect to their accuracy in prediction. The accuracy was evaluated as

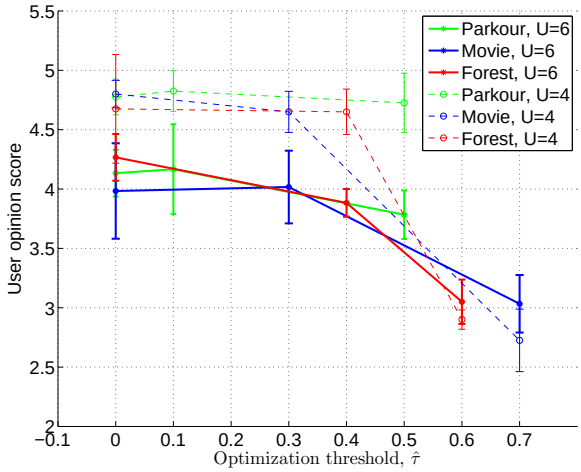


Fig. 5: Mean and error (σ) of MOS when users were grouped (dotted - 4 users, solid - 6 users)

the error in terms of difference between predicted and the actual user grade.

We propose clustering methods for collaborative filtering to predict the quality of video to be delivered to the user. The fundamental assumption behind collaborative filtering methods is that users' opinion on quality can be filtered, selected and aggregated in such a way as to provide reasonable prediction of active users' preference. Collaborative filtering has several methods to aggregate users with similar behavior for predicting, such as baseline prediction, User-User filtering, Item-Item filtering, probabilistic and hybrid filtering methods [21]. We propose different strategies to group a subset of users utilizing the similarities in history of viewing behavior and evaluate the performance of the methods against each other under different dimensions of prediction space.

To group the users, we first have to identify the space over which to apply the clustering methods. In our approach, each of the movies, composed with a specific quality threshold, represents one of the axis in a multi-dimensional space. The grade provided by a user for that movie represents a positional coordinate for that user on that axis. Thus the similarity between two users, in terms of video quality perception, is considered in function of the distance between these two users in the multi-dimensional space identified by a set of movie seen and graded by both users ("background movies"). Once a group of users with similar video quality perception is identified based on a space of videos watched by all users, the grades of the users in the group, on a movie seen only by a few of them, can be used to predict the expected quality perception of the remaining users, for that new movie. Even though in our investigation we have access to all grades, for all users on all the available movie qualities, when trying to predict users' QoE grades on a given movie, we perform the grouping only considering movies different than the one to be predicted.

Fig. 6 graphically illustrates the space dimensions considered in our grouping methods. In particular, it assumes that we want to predict a user grade for the video "Forest", composed with low threshold $\hat{\tau}$. To do so, we proceed by discarding all grades for that movie from the grouping space and we select different number of background movies to understand how the prediction accuracy changes in function of the number of movies considered for the grouping space. In our study, we consider the grades for the movies "Parkour" and "Movie" for performing grouping operations to predict the quality associated to the video "Forest", while "Forest" and "Movie" to group users to predict the grades of "Parkour" and finally the grades from "Forest" and "Parkour" to group users for predicting the grades of the movie "Movie". Each of the quality for each movie represents an individual axis thus in total our considered space is nine-dimensional. In real system, the dimensionality will be significantly higher, however we anticipate that only a subset of relevant movies can be used to perform the grouping operations.

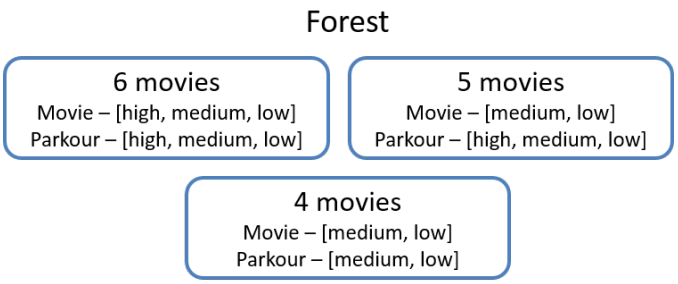


Fig. 6: Six, five and four dimensional background movie spaces used for grouping users to predict the QoE of the movie Forest, composed with low threshold.

Five different grouping strategies have been considered:

Strategy 1 (S_1): In this strategy, we used $k-means$ to first cluster the users into groups based on the chosen user grade space. We vary k between 1 and the total number of users, for different seeds and stop the process when we first encounter an individual group size with less than a critical mass of 4 users ($\hat{k}$ groups). The penultimate round of grouping ($\hat{k}-1$) is considered as the grouping outcome for this strategy. This procedure ensures that all the users are grouped and also every individual group contains at least 4 users. The maximum of all distances between each user grade and its cluster head is defined as the distance threshold, θ.

Strategy 2 (S_2): This strategy is the baseline predictor of collaborative filtering techniques, where we clustered all the users into a single group. The predicted grade, for a user on a movie, is computed using the average grade of all the users, excluding the the active user, for the same movie.

The next three strategies are centered on the active user we are predicting the quality to be delivered. This means that the groups may vary depending on the content type.

Strategy 3 (S_3): In this strategy, we used $k-means$ to first cluster the users into groups based on the user grades, varying k between 1 and the total number of users, for different seeds, similar to S_1. However, once the groups were formed, for a particular user, the group formation that had the smallest threshold, θ, with a minimum number of 4 users including that user was selected for clustering.

While, the aforementioned three strategies used $k-means$ clustering as the basis for grouping users, the next set of two strategies use the nearest neighbors of a specific user as the basis for group formation.

Strategy 4 (S_4): In this strategy, three closest users in terms of opinion grade to the active user were selected and clustered into a group for prediction. Note that the number of users to group can be varied, we chose three closest users to keep the minimum group size consistent across our investigations (at least 4 in a group).

Strategy 5 (S_5): In this strategy, instead of closest users in terms of distance, we used a radius threshold, ρ, on user grades to filter and cluster users into a group. For a particular user, all the users who fell below the selected radius threshold were clustered into a group. Additionally, we also required the group size to contain a minimum of 4 users.

Formally, each of the aforementioned grouping scheme will result in the definition of a cluster C_i, for each user i in the system, containing the IDs of the N_i users in his group. Once users are grouped, this information is used to select the quality of a movie to be delivered, to a specific user, to achieve a given QoE target. The methodology for prediction and selection of quality factor for a specific user is described in Section IV-D.

Fig. 7 illustrates the median error in prediction for different strategies when all 33 users are grouped and all the movies are individually predicted. Firstly, we observe that as we increase the number of movies, the error decreases. This is a strong indicator that by accessing higher dimensional background movie spaces, the accuracy in prediction increases. Moreover, it is clear that the baseline predictor, S_2 (population based approach), has the highest median error.

To group all the 33 users, we had to consider a large distance threshold, θ = 4.8 or radius threshold, ρ = 2.25. By increasing the thresholds, more users were grouped in total as we can tolerate larger distances between user grades and as a result the error in prediction increases. Instead, by considering smaller thresholds fewer users are group together but these are closer, thus leading to better prediction accuracy. Smaller values of θ might lead to have some users left out from the groups. In practise these could be serve using alternative methods, such as using baseline prediction strategies. However, in this paper, the users that are not grouped are excluded from the system.

We observed that S_5 gave the best performance when compared with varying radius versus number of users grouped for serving. We study and show below how the error varies as a function of radius threshold, ρ, to show that by grouping fewer users we can reduce the error in prediction.

Fig. 8 illustrates the median error of predicted MOS in the increasing order of the radius threshold, ρ, where points {*A,B,C,D,E,F,G,H,I*} correspond to $\rho = \{0.25, 0.50, 0.75, 1.00, 1.25, 1.50, 2.0, 2.25, 2.50\}$,

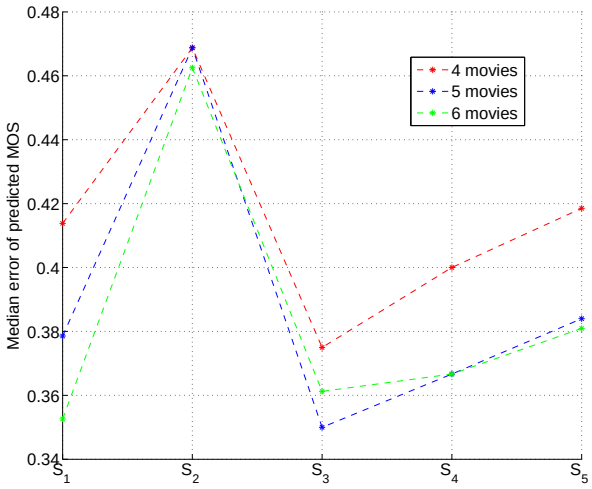


Fig. 7: Median error in prediction for different strategies over all movies (all users grouped)

plotted against the number of users in groups to serve from a total of 33 users. We can observe that for a small radius threshold, fewer users are grouped and as we increase the threshold, the number of users grouped increases. Note that when fewer users than the total number of users are grouped, only their grades are considered when computing the error. In an ideal case scenario, we would like to be in the top left corner where the number of users grouped are high and the error in prediction is low.

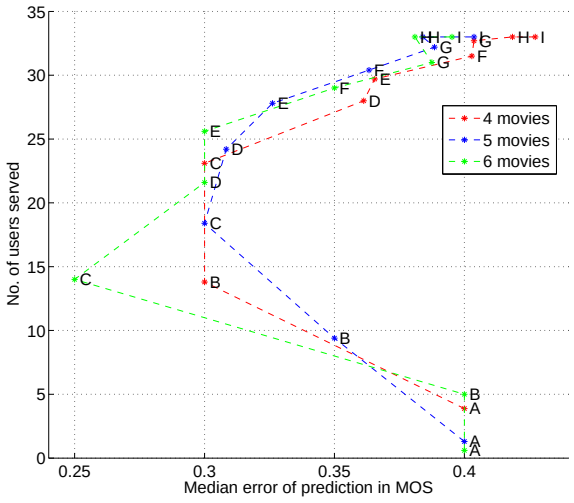


Fig. 8: Median error in prediction for S_5 with varying ρ (point [A-I] indicate ρ between [0.25-2.5])

At the same time we can also observe that as we increase the number of movies for prediction, the error in prediction goes down. The green curve in the Fig. 8 outperforms the red and blue curves for the most part, but as the threshold increases they converge together. We believe that adding more movies, more users grades will lead to a better performance of the prediction strategy. Hence, the radius threshold, ρ and the number of movies used for prediction remain the most important factors influencing the performances of the QoE prediction strategy.

D. Prediction of QoE and Content Delivery

Once groups have been formed using a chosen grouping strategy, we compute the predicted grade for all the quality levels of a movie and the video quality to be delivered to a specific user using Eqn.1.

$$P_m^i(\hat{\tau}_m^l) = \frac{1}{(N_i - 1)} \sum_{k \in C_i, k \neq i} G_m^k(\hat{\tau}_m^l)$$

$$\mathcal{T}_m^i(Q_e) = \begin{cases} 0, & \text{if } \max\limits_{\hat{\tau}_m^l}\{P_m^i(\hat{\tau}_m^l)\} < Q_e \\ \operatorname{argmin}\limits_{\hat{\tau}_m^l}\{P_m^i(\hat{\tau}_m^l) \geq Q_e\}, & \text{otherwise} \end{cases} \quad (1)$$

There $P_m^i(\hat{\tau}_m^l)$ represents the predicted grade of user i of a movie m with threshold $\hat{\tau}_m^l$ and $G_m^k(\hat{\tau}_m^l)$ represents the grade of a user k who belongs to group or cluster C_i for the same movie m composed with the same threshold $\hat{\tau}_m^l$. Q_e represents the target QoE value to be delivered, while $\mathcal{T}_m^i(Q_e)$ indicates the quality level threshold that needs to be used to compose a movie m, for provision to user i to achieve a QoE equal to Q_e. For example, let us consider a case in which a QoE target Q_e = 4 is attempted for a user i and that there are three available video qualities $\hat{\tau}_m$ = {0, 0.1, 0.3}, each with corresponding predicted grades P_m^i = {4.8, 4.1, 3}. In this case, the target quality selected according to Eqn. 1 corresponds to $\mathcal{T}_m^i(Q_e = 4) = \hat{\tau}_m^2 = 0.1$.

V. Results

To evaluate the performance of serving users based on prediction, we considered the following scenario: all the 33 users in the system are served a movie, which is available in three different qualities. The quality served for the movie to each of the users is selected to achieve a specific target Q_e, using the approach described in Eqn.1. Grouping strategies S_2 and S_5 with ρ = 1.0 are considered and compared with an ideal case scenario for the "Parkour" movie. We chose Strategy S_2 to represent a population based approach which is a baseline prediction strategy and Strategy S_5 represents a personalized approach where ρ was selected such that at least 50% of the users were delivered with a predicted grade higher or equal to the user grade. The ideal case represents a hypothetical scenario with full information (or perfect prediction), where the real user grades are used instead of their predicted values. This can be seen as an upper bound on performances.

We evaluate the impact of delivering video content for prediction for a target QoE (Q_e) utilizing the following performance metrics: (1) Satisfaction ($\Delta_{QOE}^{+}(Q_e)$), (2) Dissatisfaction ($\Delta_{QOE}^{-}(Q_e)$) and (3) bandwidth costs to serve the users ($B(Q_e)$). These are defined according to Eqn.2:

$$\Delta^i_{QOE}(Q_e) = G^i_m(\mathcal{T}^i_m(Q_e)) - P^i_m(\mathcal{T}^i_m(Q_e)),$$
$$\Delta^+_{QOE}(Q_e) = \sum_{i|\Delta^i_{QOE}\geq 0} \Delta^i_{QOE},$$
$$\Delta^-_{QOE}(Q_e) = \sum_{i|\Delta^i_{QOE}< 0} \Delta^i_{QOE}, \quad (2)$$
$$B(Q_e) = \sum_i b_m(\mathcal{T}^i_m(Q_e)),$$

There $P^i_m(\mathcal{T}^i_m(Q_e))$ is the predicted grade, computed for a chosen prediction strategy for a movie m and a quality target Q_e on user i, $G^i_m(\mathcal{T}^i_m(Q_e))$ is the recorded grade for the same movie. $V_m(\mathcal{T}^i_m(Q_e)$ is the size of video that has been selected to be delivered which is personalized to the same user.

Fig. 9a shows the number of satisfied users for a given experience target (Q_e) ranging between 3.0 and 5.0 and the associated cost for delivery of the selected movie.The black curve represents the results of S_2, which considers an average over entire population. We can clearly see three activation levels, where users are served among the three available quality levels based on MOS over the entire population. The green curve, instead, depicts a personalized approach where users are grouped based on their viewing behavior and a personalized quality can be selected based on their past viewing behavior (S_5). Finally, the magenta curve denotes the ideal case where the actual user grades are used. The numbers on the curves denote different experience targets, Q_e.

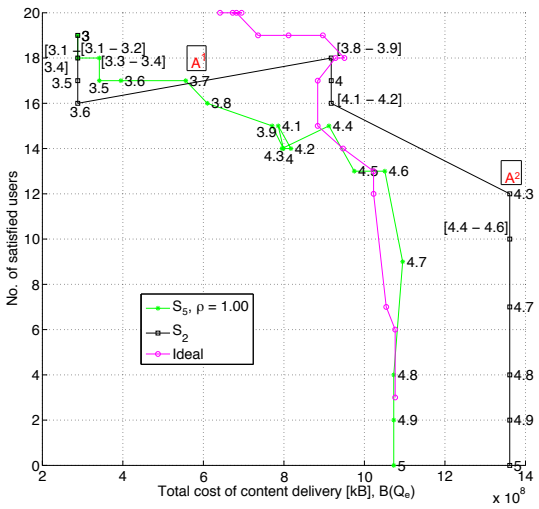
Consider the points A^1, for $Q_e = 3.7$, on the personalized scheme (S_5) curve and A^2, for $Q_e = 4.3$, on the population-based scheme (S_2) curve in Fig. 9a. It can be observed that we can deliver content to a higher number of satisfied users using a personalized approach by grouping the users based on similarities. Also, the cost of delivering the content using personalized approach is approximately 2.5 times less than the population based approach . The total satisfaction and dissatisfaction among users can be seen in the Fig. 9b.

We can also observe for the aforementioned operational points that, even though we deliver high quality content to more number of users using the population based approach, fewer users are satisfied when compared with the personalized approach. This is due to the fact that higher grades were recorded for medium-quality content than high-quality (720p) content for a subset of users.

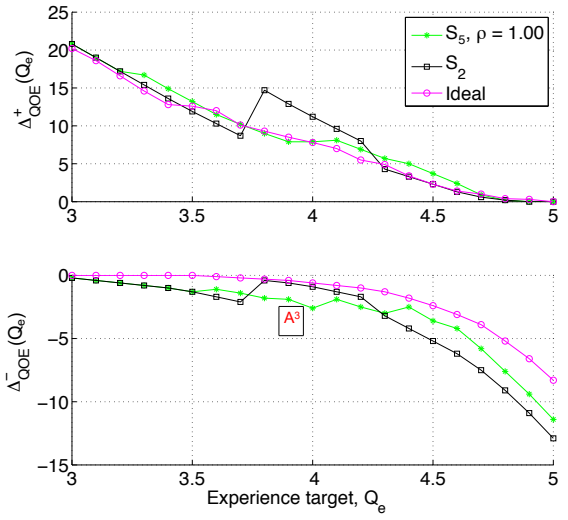
Finally, by delivering high-quality (720p) content even when a user is satisfied with a lower threshold, does not necessarily increase his/her QoE. Considering Q_e = 3.9, point A^3 in Fig. 9b, the dissatisfaction level per user was 0.1, but with 8% total bandwidth reduction when compared with a population based approach. Hence, by delivering personalized quality, we consume significantly less bandwidth while attaining relatively lower levels of dissatisfaction from users.

VI. Conclusion and Future Work

In this paper, we presented a systematic methodology for optimization of video content delivery. The proposed approach



(a) Number of users satisfied and associated bandwidth costs when serving the Parkour movie attempting to achieve different Q_e.



(b) $\Delta^+_{QOE}(Q_e)$ and $\Delta^-_{QOE}(Q_e)$ for different Q_e (Parkour movie)

Fig. 9: Performance evaluation of QoE aware content delivery for Parkour movie

first identifies the VQM scores on all the scenes of all available quality versions of a movie. Then, an optimization procedure is used to composed the video for achieving a chosen quality target for a specific user. This procedure adopts a novel clustering strategy to group users based on similarities in their viewing behavior. This information is shown to be effective for selecting the video quality that is personalized to individual user appreciation. Utilizing the proposed strategies, we were able to show that the median error in prediction was in the order of ±0.5 MOS points. Finally, this approach has been validated through a crowd sourced subjective test evaluation

performed with real users using a novel experimental methodology involving the Amazon Mechanical Turk platform.

We observed that VQM score is not portable across different types of video content inherently, especially for content with low bitrate. We believe the reason for this specific behavior lies in the fact that VQM is a full reference (FR) metric and specific imperfections in the source video are qualitatively judged against the reference video. For example, we observed that if a specific scene was intentionally blurry or out of focus, even at high quality users perceived this as low quality. However, the proposed schemes are not limited to use this VQM. Any metric that holds across videos and has a high correlation with MOS could be easily adopted into our optimization procedure. We plan to extend our framework to include the NR metric, which considers the sharpness of individual frames and that can be applied across different videos [22]

Our approach comes with the potential costs associated with recording and storing users' grades and performing user clustering and QoE prediction operations. However some of these costs could be mitigated using a scalable method for collecting users' QoE utilizing a crowd sourced approach. QoE recording can be easily included after playback, considering existing feedback mechanisms e.g., QoE feedback after a Skype call.

Since our experimental tests only considered three quality levels for a video, additional investigations are needed to evaluate how our approach scales with increasing number of threshold levels on a larger set of movies. Moreover, also user grouping operations need to be extended to include cases in which not all user grades are available for all the quality levels of a movie.

REFERENCES

[1] Ericsson mobility report. [Online]. Available: http://www.ericsson.com/res/docs/2015/mobility-report/ericsson-mobility-report-nov-2015.pdf

[2] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 362–373. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018478

[3] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http," in *Proceedings of the Second Annual ACM Conference on Multimedia Systems*, ser. MMSys '11. New York, NY, USA: ACM, 2011, pp. 157–168. [Online]. Available: http://doi.acm.org/10.1145/1943552.1943574

[4] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "Developing a predictive model of quality of experience for internet video," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, ser. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 339–350. [Online]. Available: http://doi.acm.org/10.1145/2486001.2486025

[5] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "A quest for an internet video quality-of-experience metric," in *Proceedings of the 11th ACM Workshop on Hot Topics in Networks*, ser. HotNets-XI. New York, NY, USA: ACM, 2012, pp. 97–102. [Online]. Available: http://doi.acm.org/10.1145/2390231.2390248

[6] S. Akhshabi, L. Anantakrishnan, A. C. Begen, and C. Dovrolis, "What happens when http adaptive streaming players compete for bandwidth?" in *Proceedings of the 22Nd International Workshop on Network and Operating System Support for Digital Audio and Video*, ser. NOSSDAV '12. New York, NY, USA: ACM, 2012, pp. 9–14. [Online]. Available: http://doi.acm.org/10.1145/2229087.2229092

[7] L. De Cicco and S. Mascolo, "An experimental investigation of the akamai adaptive video streaming," in *Proceedings of the 6th International Conference on HCI in Work and Learning, Life and Leisure: Workgroup Human-computer Interaction and Usability Engineering*, ser. USAB'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 447–464. [Online]. Available: http://dl.acm.org/citation.cfm?id=1947789.1947828

[8] M. Ries, C. Crespi, O. Nemethova, and M. Rupp, "Content based video quality estimation for h.264/avc video streaming," in *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, March 2007, pp. 2668–2673.

[9] A. Khan, L. Sun, and E. Ifeachor, "Content clustering based video quality prediction model for mpeg4 video streaming over wireless networks," in *Communications, 2009. ICC '09. IEEE International Conference on*, June 2009, pp. 1–5.

[10] Y.-x. Liu, R. Kurceren, and U. Budhia, "Video classification for video quality prediction," *Journal of Zhejiang University SCIENCE A*, pp. 919–926, 2006. [Online]. Available: http://dx.doi.org/10.1631/jzus.2006.A0919

[11] ITU-R, "Methodology for the subjective assessment of the quality of television pictures," ITU-R Recommendation BT.500-13, Jan. 2012.

[12] M. Pinson and S. Wolf, "A New Standardized Method for Objectively Measuring Video Quality," *IEEE Transactions on Broadcasting*, vol. 50, pp. 312–322, Sep. 2004.

[13] National Telecommunications & Information Administration (NTIA) Institute for Telecommunication Sciences (ITS). VQM software. www.its.bldrdoc.gov/resources/video-quality-research/software.aspx.

[14] A. Devlic, P. Kamaraju, P. Lungaro, Z. Segall, and K. Tollmar, "Qoe-aware optimization for video delivery and storage," in *2015 IEEE 16th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, 2015, pp. 1–10.

[15] A. Devlic, P. Kamaraju, P. Lungaro, Z. Segall, and K. Tollmar, "Qoe-aware optimization for video delivery and storage," in *Quality of Service (IWQoS), 2014 IEEE 22nd International Symposium of*, 2015.

[16] Shotdetect. [Online]. Available: http://johmathe.name/shotdetect.html

[17] Cine. Movie Measurement and Study Tool Database. http://www.cinemetrics.lv/.

[18] P.910. [Online]. Available: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-P.910-200804-I!!PDF-E&type=items

[19] Video quality testing app. [Online]. Available: https://play.google.com/store/apps/details?id=com.kth.mslab.video

[20] Video quality testing website. [Online]. Available: http://pavan4.duckdns.org:5000

[21] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Foundations and Trends in HumanComputer Interaction*, vol. 4, no. 2, pp. 81–173, 2010. [Online]. Available: http://dx.doi.org/10.1561/1100000009

[22] A. Leclaire and L. Moisan, "No-reference image quality assessment and blind deblurring with sharpness metrics exploiting fourier phase information," *Journal of Mathematical Imaging and Vision*, vol. 52, no. 1, pp. 145–172, 2015. [Online]. Available: http://dx.doi.org/10.1007/s10851-015-0560-5