MOOC-DASH: A DASH System for Delivering High-Quality MOOCs Videos

Yi Wang, Wenjun Wu, Yihua Lou
National Laboratory of Software Development Environment
School of Computer Science and Engineering, Beihang University
Beijing, China
{satte, wwj, lhy}@nlsde.buaa.edu.cn

Abstract—Adaptive video streaming is very important for delivering high-quality video content of MOOCs (Massive Online Open Courses) to online learners because they often have Internet connections with different levels of bandwidth. Although DASH (Dynamic adaptive streaming over HTTP) is widely accepted as a viable streaming technology to implement scalable Internet streaming using HTTP transport, few research efforts have been made to investigate how to apply this relatively new technology on improving the quality of experience (QoE) of MOOC video streaming. This paper proposes a DASH scheme for MOOC video streaming (MOOC-DASH) to improve QoE of DASH-based MOOC. This scheme consists of content-aware ROI-based video encoding for MOOCs video and bitrate selection algorithm to provide a high-quality and smooth video streaming service to online learners. Experimental results demonstrate that it can effectively reduce the bandwidth of the video content and improve QoE of MOOC Streaming.

Keywords—MOOC-DASH, MOOC, QoE

I. INTRODUCTION

As the massive online open courses (MOOCs) is becoming popular, tens of millions of students across the world watch online video lectures through their web browsers every day. Given the geographic diversity of MOOC learners, they have different connection capacities to access high-quality MOOC videos. To deliver MOOC video streams to these receivers and provide quality-of-experience to them, it is essential to employ the DASH technique that is able to adaptively switch the video quality levels according to the network conditions of receivers.

A DASH system consisting of a DASH server and DASH players, is designed to deliver quality video streaming over HTTP connections. The DASH server running on a regular web server initiates a DASH process by splitting a very long video file, usually several hours, into a sequence of small file fragments with the interval of playtime up to 10 to 30 seconds. Each video segment is transcoded into multiple replicates with different quality levels and bitrates. On the client side, a DASH player downloads each segment separately and plays them as a stream. By estimating the available TCP throughput of the network connection, the video player decides the most suitable quality level of video segments to download. Thus, the DASH player can seamlessly adapt to changing network conditions and provide high quality playback without stalls or re-buffering events.

To apply the DASH technique on MOOC video streaming, we need to develop a content-aware DASH streaming solution customized for MOOC video content. Current design of most DASH systems focuses on general DASH streaming issues including bandwidth measurement and adaptive rate control algorithms. They attempt to improve the smoothness of DASH video streams and reduce the occurrences of rebuffering or playout stalls. But they don’t consider how to take advantage of the characteristics of video content to design a content-aware bitrate profile and optimize the bitrate regulation for better visual quality. But user-perceived visual quality is highly affected by scenarios in a video stream. Empirical studies show that different kinds of video content have varied requirements for bitrate levels to achieve the same acceptable QoE. In case of MOOC video lectures, many video clips are still images and animations of presentation slides as well as visual figures of instructors, which have far less motions than scenes of other kinds of videos such as movies and sports TV shows. Our subjective tests on MOOC video contents have confirmed that the lecture videos with only presentation slides and instructor can obtain the same level of visual quality as regular motion videos in a much lower bitrate level. This result indicates that we should develop a bitrate profile for each kind of MOOC content and design a content-aware DASH scheme to optimize the QoE and bandwidth usage.

Following this idea, we try to build a content-aware adaptive streaming system, called MOOC-DASH, for delivering MOOC video lectures. We introduce a video classification algorithm to detect the scene boundaries of a MOOC video sequence and label the scenes as four major types of MOOC videos. To analyze the bitrate-quality features of the MOOC video types, we run subjective tests to develop the bitrate profile for each type. Based on the result of the subject tests, we integrate the content-specific bitrate profile into our bitrate selection algorithm and improve its buffering scheme to reduce bit rate fluctuations and provide a consistent QoE for viewers.

The paper is organized as follows: Section 2 discusses related research work. Section 3 describes the video classification algorithm and ROI-based encoding scheme for MOOC-DASH systems. Section 4 introduces the bitrate selection algorithm that is customized for adaptive streaming MOOC videos. Section 5 presents the experimental results and performance analysis of the MOOC-DASH scheme. Section 6 gives the conclusion.

II. RELATED WORK

Many research efforts have been made to design adaptive schemes for HTTP video streaming. A recent research [1] divides rate adaptation methods into four steps: estimating
available network bandwidth, smoothing bandwidth estimation, quantizing bandwidth to determine proper video bitrate and further predict the best video quality suitable for current situation and finally scheduling the transmission of the next video segment.

A. Measurement of Available Bandwidth

When a DASH player needs to select the bitrate for next video segments, it should firstly estimate available network bandwidth by measuring TCP throughput. The condition of network varies from time to time, which makes it difficult to accurately predict the future network condition based on the past measurements. A recent measurement paper [2] reveals the difficulty of accurate bandwidth estimation on top of HTTP. This is because the true network status is hidden behind several layers below the application layer.

Bandwidth estimation in DASH schemes can be categorized into two kinds: proactive probing approach and reactive approach. QDASH [3] takes the first approach to send TCP probing packets to make RTT measurement for bandwidth estimation. The advantage of this approach enables DASH schemes more sensitive to the dynamics of connection bandwidth and to quickly detect the variation in bandwidth. But as the publication [4] points out, inaccurate estimation of bandwidth can create an undesirable feedback loop and further bias the estimation. As a result, players can suffer from poor video quality, unstable rates and unfairness across players. Moreover, the probing algorithm involves tuning parameters of the TCP layer, which increases the complexity of implementing DASH schemes. It often needs an intermediate network proxy that can facilitate RTT measurement for DASH players.

Some DASH schemes [5] estimate the available bandwidth by observing the downloading time of each video segments. They may calculate the average downloading speed, while a video trunk is downloaded, and take the average download speed as the estimation of the current network bandwidth. The client measures the segment fetch time, which covers a relatively long period of time, to determine if the bitrate of the current representation matches the available end-to-end bandwidth capacity. Compared to the proactive probing approach, it is easier to implement the reactive measurement in a web environment because it doesn’t involve any modification to the HTTP protocol stack and extra intermediate network service. Given the practical consideration, we choose to take the reactive approach to implement the bandwidth estimation in the MOOC-DASH system.

B. Quality-Aware Bitrate Adaption Algorithms

DASH techniques bring in variation in video bitrate and perceived quality. Liu et al. [6] studies the three factors affecting user-perceived quality of video watching for DASH: initial delay, stall, and fluctuation of video bitrate.

A few research papers propose to enhance the DASH adaptation logic by incorporating on-the-fly quality assessment of DASH video playback. QDASH conducted a subjective experiment to demonstrate that users prefer a gradual quality change between the best and the worst quality levels, instead of an abrupt switching. Based on that observation, they design a rule set about the quality level switch aiming to insert intermediate quality levels when bitrate of DASH video has to decrease. Other researchers [7] proposed a buffer-based rate selection algorithm based on the occupancy of the playback buffer to reduce the occurrences of re-buffers and provide a stable video rate.

All the above efforts have not investigated how content-based quality can impact the design of DASH scheme. When we define mapping between bitrate and subject video quality for bitrate adaption algorithm, few research has been made to check application-specific mapping. To our best knowledge, the only research on content-aware video adaption schemes is described in the paper [8]. But it is designed for streaming general video files and categorizes them into three types based their motion intensity. It cannot be applied to MOOC streaming since it doesn’t give any specific consideration about MOOC content.

III. MOOC VIDEO CLASSIFICATION AND ROI-BASED ENCODING

User-perceived quality depends upon many factors including the available bandwidth for video transmission, communication latency and network jitter as well as characteristics of video content. MOOC-DASH aims to design a content-aware adaptive streaming scheme for delivering high quality MOOC video content with optimal consumption of network access bandwidth. It takes advantage of motion and scenario features of MOOC video content and adopts a customized DASH encoding scheme considering user perception of MOOC videos.

A. QoE issue of MOOC Streaming

A MOOC-DASH scheme encodes a short-period video into multiple copies with different levels of video bitrate, each of which represents a quality level for the video content. Generally speaking, a higher bitrate can lead to a better video QoE for online viewers. After transcoding course video into several levels, it is essential for a MOOC player to achieve a balance between maximum QoE and video rates allowed by current network conditions. Instead of setting the bitrate for each quality level, most MOOC websites take a simple solution by directly configuring the resolution of MOOC video content and adopts a customized DASH encoding scheme considering user perception of MOOC videos.

B. QoE Evaluation of MOOC Video

The perception of video quality highly related to video content. To achieve the best possible QoE at the minimum download rate of video streams, we need to perform subjective evaluation of user perception of MOOC videos at different levels of bitrates. Based on the subjective measurement of the QoE of MOOC videos, we are able to
find the lowest bitrate for a MOOC video to present the same level of subjective quality for online viewers.

According to the study on MOOC video production styles [10], a MOOC course may consist of four major categories of videos including Slides-Presentation video, Instructor-Centered video, and combination of both instructor and slides, Motion video. Empirical study reveals that video learners have notably different QoE for each category of video clips with the same bitrate. The detailed description of each category is shown in Fig. 1.

1) Slides-Presentation Video (Abbr. Slides): Such a video clip is created through screen capturing to present PowerPoint slides. There are no human figures or animation characters in the video. Except for slides switching, no other motions can be detected between the consecutive video frames. Normally, a Slides-Presentation video stream can be delivered in a very low bitrate such as 40kbps to display text characters or graphic pictures of the slides in a good visual clarity. Our subjective quality experiments illustrate that there is no significant improvement in user-perception when Slides-Presentation video is transmitted in a higher bitrate.

2) Instructor-Centered Video (Instructor): This video production style puts the instructor of a MOOC course as the major focus in its video scene. Compared with the Slides-Presentation video streaming, the priority of delivering the style of video is to ensure the visual quality of the instructor in the video. In addition, as the figure of “talking head” usually does not have major motions in his body and the rapid change of scene is relatively rare, it only requires a relatively low bitrate for transmission.

3) Combination of Instructor and Slides Video (Instructor-Slides): This is the combination of PowerPoint slides and the Instructor video. Its visual layout scheme only leaves the majority of the display area for the slides and only a corn region for the instructor video. Given the compositional scene, this kind of video streaming normally demands a bitrate higher than that of the Slides-Presentation video style and lower than that of the Instructor video.

4) Motion video (Motion): A MOOC course sometimes needs to show genuine motion videos such as soccer games for sport courses. Apparently, this style of videos often contains more motions than the previous three styles, thus demanding a higher video bitrate for high quality streaming.

Through the experiments of subject quality assessment, we evaluate the QoE of video datasets from each category respectively to find the right bitrate levels for them. Our experiments adopted both Single Stimulus Method (SSM) and Double Stimulus Continuous Quality Scale (DSCQS) method [11]. As the functional relationship between the subject rating scores of videos and their bitrates normally has an increasing region that needs extensive assessments, we need to determine the upper bound and lower bound of the rate range. Following the DSCQS method, we asked observers to compare an original MOOC video and a group of impaired videos with a lower bitrates. If a video quality has the lowest bitrate and presents a similar subjective perception like the original video, its rate would be regarded as the upper bound of the video bitrate range for running the SSM tests. And we employed structural similarity index (SSIM) to determine the lower bound of the video bitrate range for running SSM tests.

In the subjective tests based on SSM, we asked volunteers to assess the quality of the MOOC videos with different bitrates and gave them scores from 1 point to 5 point. Each volunteer watches a single video and afterwards is required to score the video. In addition, when the system randomly assigns multiple video clips to the volunteers, it presents the video clips with the same quality to a volunteer at least twice. This is a built-in consistency-checking mechanism designed for judging whether the volunteer is scoring video clips seriously. If the system detects a remarkable difference between two scores given by a volunteer for the same video clip with the same quality, it can determine that he is grading the video carelessly and thus ignoring all the assessment scores from the volunteer. After having collected the results for different video bitrate levels within each category, we attempted to analyze the correlation between the quality of a MOOC video quality and its bitrate. Based the analysis, one can decide an optimal video quality profile for every category of MOOC video to achieve the best possible QoE using MOOC-DASH.

C. Classification of MOOC Videos

To support the content-aware bitrate allocation scheme, MOOC-DASH should be able to correctly classify the MOOC video to be delivered. As an actual MOOC lecture often consists of a sequence of scenes such as slide presentation, instructor and animation of courseware. It is necessary to divide such a MOOC video sequence into small segments based on their content scenes before classification in order to perform adaptive bitrate selection. The major phases of the algorithm are listed as follows:

1) Content Boundary Division

To divide a MOOC video into small segments by its scenes, the algorithm needs to detect the scene boundary between two segments. An easy way is to calculate color
segments. We introduce a four-step classification algorithm including motion area calculation, facial detection, instructor detection and structural similarity analysis. Fig. 2 illustrates the decision tree of this phase.

a) Calculate the largest motion area: This step aims to identify part of motion segments in a MOOC video stream by calculating the largest motion area. We assume that the Motion type video should contain the largest motion area than the other types. The algorithm checks the pixels between two adjacent video frames and marks all different pixels as \( P = \{ p_n \} \). Then, it dilates the marked pixels for several times and identifies the largest connected subset \( P_m \subset P \) as the motion area. If the size of the area goes beyond a threshold, the algorithm decides that this video segment belongs to the Motion type. To avoid incorrect classifications caused by the movements of instructors, the algorithm sets the threshold to 50%.

b) Facial detection: This step attempts to locate the facial regions in the video segment under processing. Assume that both the Instructor and Instructor-Slides type should contain the facial regions of the instructor in most video frames. To detect the instructor’s face in the video frames, the algorithm employs a Harr-based AdaBoost face detector. It is well known that such a detector may generate false classification in some circumstances. The algorithms needs to check the frequency of the instructor’s occurrences throughout the video frames of the segment and makes sure that the frequency is sufficiently high to increase the accuracy of the classification. Empirical value for the frequency threshold is one person every two frames.

c) Instructor detection: This step aims to distinguish the Instructor type of MOOC video from the Instructor-Slides type. The major distinct feature between both types lies on the proportion of the instructor region in the entire video frame. As the Instructor type intends to highlight the image of the instructor, its video frames definitely have more pixels for displaying the instructor. Thus, the one with larger size of instructor video should be classified as the Instructor type. Otherwise, it should be regarded as the Instructor-Slides type. Our experiments suggest the threshold value for the proportion of the instructor region as 10% of the entire video frame.

d) Structural similarity analysis: The last step is designed to separate the Slide type from the Motion type of MOOC videos that contains neither large motion regions nor human faces. In our observation, such a kind of MOOC videos can be easily misclassified as Slide videos. To address this problem, this step samples multiple frames in the segment and checks the changes of structural similarity (SSIM) among these frames. If the SSIM value remains the same, it indicates that the frames are highly similar and must belong to a slide image in a video lecture. Otherwise, it should be classified as regular motion video. Empirical value for the SSIM threshold is set to 0.9.

In the post-processing of the four-step classification, the algorithm sorts video sequences in a temporal order and merges adjacent segments of the same category into a new segment.

D. ROI-Based Encoder

The encoder is customized for Instructor-Slides video, where the instructor image merely occupies a corner region of the entire video frame and the rest of the frame is the slide image. As discussed above, the slide part demands a constant and low bitrate to keep its quality, whereas the instructor region needs a higher bitrate. Therefore, we introduce a ROI-based encoder to deal with this case. First, a face detection algorithm is employed to position the instructor area in each video frame. Then, the ROI-based encoder is utilized to process the two parts separately with different quantization parameters. The instructor region is encoded with a small quantization parameter to ensure high quality of the instructor. And, a higher value of quantization parameter can be set to encode the slide part of the video. Given the relatively small size of the instructor region in the entire video frame, the ROI encoder can create a high-quality video stream without incurring too many extra bitrate to the stream.

IV. BITRATE SELECTION ALGORITHM

The Bitrate Selection algorithm aims at providing a smooth content-aware bitrate setting and responsive regulation of streaming rate towards changes of the average network bandwidth. First, it is vital for the algorithm to deliver a stable QoE for MOOC viewers. To avoid QoE oscillations, the algorithm conservatively keeps the same bitrate level along the consecutive MOOC video segments belonging to the same video category if the current available bandwidth allows. But after the video scenes change indicated by the video type switch, the algorithm can take more opportunistic approach of selecting content-related bitrate levels.

Second, the algorithm needs to adjust the video buffer length at a reasonable level to avoid buffer stalls or overflows. Especially, buffer stalls can result in video streaming
interruptions, which seriously degrades QoE of MOOC viewers. By monitoring the current buffer status and network bandwidth, the algorithm infers the suitable bitrate level for the incoming video segment.

Last, the algorithm should be designed to be more responsive to the decrease of the available bandwidth. In the situation of drastic drop in network bandwidth, the bitrate level needs to be lowered in time to avoid packet losses in the HTTP transport. Table 1 enumerates the major variables and parameter used by the algorithm.

The selection algorithm is listed below. Let \( C = \{C_s\} \) as the sequence of video segments’ category. Segment\((s)\) has just finished downloading and is in the selection process right now. The preceding segment is denoted as Segment\((s-1)\) and the segment right after it is defined as Segment\((s+1)\) waiting downloading task.

The algorithm adopts a simple method of bandwidth estimation that measures the average downloading rate during a specific time interval. The calculation method is displayed in (1), where the size of the video segment \( n \) is donated as \( S_n \) and the elapsed downloading time as \( t_n \).

\[
B_n = \frac{S_n}{t_n}, \tag{1}
\]

Based on the bandwidth estimation, the algorithm attempts to curb the possible increase in the bandwidth estimation when Segment\((S)\) and Segment\((S+1)\) have the same MOOC video type.

\( B'_i \) is defined as the restricted value for the estimated network bandwidth, which can be computed in (2):

\[
B'_i = \begin{cases} 
B_i - \theta \max\{B_i - B_{i-1}, 0\} & \text{if } s_{s+1} = C_s \text{ and } i > 1 \\
B_i & \text{otherwise}. \tag{2}
\end{cases}
\]

To suppress the fluctuation in the estimation of the bandwidth, one can employ the exponential moving average method for bandwidth calculation. Using the adjusted bandwidth estimation, the algorithm computes the average bandwidth as \( \overline{B'_i} \) in (3):

\[
\overline{B'_i} = \left\{ \begin{array}{ll}
\frac{1}{p} \sum_{k=0}^{p-1} B'_{i-k} + \frac{1}{q} \sum_{k=p}^{q-1} B_{i-k} & i \geq p + q \\
\frac{1}{q} \sum_{k=0}^{q-1} B'_{i-k} & \text{otherwise}. \tag{3}
\end{array} \right.
\]

Given the average bandwidth, the algorithm can select the minimum bitrate level below \( \overline{B'_i} \). In order to restrict the selected bitrate level from causing possible buffer stalls, the algorithm needs to check the current video buffer to ensure the downloading process of the next segment can be finished before all the segments in the buffer play out. Therefore, we introduce a linear function \( \phi \) as a control factor to regulate the impact of the bitrate’s change on the buffer length:

\[
\phi = \min\left\{ \alpha \frac{f_i}{f_m} + \beta, 1 \right\}, \tag{4}
\]

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_s = {a_{s,l}} )</td>
<td>The List of the Available Bitrate Levels of Segment ( s )</td>
</tr>
<tr>
<td>( B = {B_i} )</td>
<td>Actual Bandwidth Sequence</td>
</tr>
<tr>
<td>( B' = {B'_i} )</td>
<td>Reduced Bandwidth Sequence</td>
</tr>
<tr>
<td>( C = {C_s} )</td>
<td>Segment Category Sequence</td>
</tr>
<tr>
<td>( \overline{B} = {\overline{B}_i} )</td>
<td>Weighted Average Bandwidth Sequence</td>
</tr>
<tr>
<td>( F = {f_i} )</td>
<td>Buffer Length Sequence</td>
</tr>
<tr>
<td>( f_m )</td>
<td>Maximum Buffer Length</td>
</tr>
<tr>
<td>( b = {b_j} )</td>
<td>The sequence of selected Video Bitrate Levels</td>
</tr>
<tr>
<td>( \theta, \tau, \varphi \in [0, 1] )</td>
<td>Constants</td>
</tr>
<tr>
<td>( \alpha, \beta \in R^+ )</td>
<td>( p &lt; q \in N^+ )</td>
</tr>
</tbody>
</table>

Let the available bitrate list of Segment\((s+1)\) as \( A_{s+1} = \{a_{s+1,l}\} \), (5) determines the bitrate level for Segment\((s+1)\) as \( b_{s+1} \):

\[
b_{s+1} = \min_{l \in \mathbb{N}^+} \{a_{s+1,l} \mid a_{s+1,l} \geq \varphi \overline{B}'_i, B'_i \in \overline{B}\}. \tag{5}
\]

The goal of (4) and (5) is to determine the suitable bitrate level so that the buffer length of the DASH player stays in the safe zone. When the current buffer length \( f_i \) drops down toward zero, the algorithm chooses the lower bitrate for the Segment\((s+1)\) so that the new video clip could be received and filled into the buffer sooner to avoid potential buffer underruns. And when the buffer length keeps increasing toward the \( f_m \), the algorithm should select the highest allowable bitrate level of the Segment\((s+1)\) to prevent the buffer from overflow. In practice, if the video buffer becomes full, the algorithm should pause the current downloading task and wait until the first segment in the buffer is played out.

V. EXPERIMENTS

A. Subjective Quality Assessment for Content-Specific Bitrate Profile

We conducted subjective quality experiments on all kinds of MOOC videos to develop a content-specific bitrate profile for each type. In the experiments, three 15-minute video clips in three course videos in a MOOC website were used as the test video dataset, each of which has the resolution of 640*480. The volunteers were asked to find the difference among the video with varied bitrates. Then following the five-point scoring system of the SSM, they rated four kinds of videos with different bitrates. Totally we received 270 evaluation results about both kinds of video with 10 different bitrate levels.

The result of the subjective measurements demonstrates that perceptions of MOOC viewers are not sensitive to the bitrate change in Slides-Presentation MOOC videos. When we set this type of MOOC video streaming to 60Kbps, it can present the same quality as the one with much higher bitrate. Therefore the DASH server only needs to provide a constant
video stream at the bitrate of 60Kbps to achieve a satisfying QoE for Slides-Presentation MOOC videos.

Subjective tests with the bitrate ranging from 100Kbps to 600Kbps have been conducted on the types of Motion Video and Instructor-Only videos. Fig. 3 illustrates the curve of quality score-bitrate of the Motion video and the Instructor-Centered video respectively. Apparently, Motion video type demands higher bitrate (up to 400-600Kbps for a good quality) than Instructor-centered video. And one can see two major points in the curve of Instructor-Centered videos: 80Kbps for the average quality score up to 4 and 180Kbps for the average score of 4.5.

Based on the empirical analysis of MOOC videos, we developed a video quality profile for adaptively delivering each category of MOOC videos. Assume we offer K levels of bitrates. Take three as example. In our opinion, 1-point is the lowest score representing unacceptable video quality. So, we split 2-5 point into three levels: 2-to-2.9-point, 3-to-3.9-point and 4-to-5-point. The bitrate of any level is set to the bitrate achieving the lowest score in this level. For instance, the bitrate of the best level is set to the highest bitrate with a score higher than 4 point and the bitrate of the worst level is set to the lowest bitrate with a score higher than 2 point. By this method, one can find an appropriate bitrate for each quality level, achieving the target feeling score without bitrate waste.

![Figure 3. MOS-bitrate curve (A) Motion (B) Instructor-Centered](image)

![Figure 4. The SSIM of the MOOC video streams under content-aware MOOC-DASH and OSMF](image)

![Figure 5. The bitrate adaption of both the content-aware MOOC-DASH player and OSMF player under a bandwidth-varying network link](image)

**B. Performance Test for MOOC-DASH Bitrate Selection Algorithm**

We tested the Bitrate Selection algorithm of MOOC-DASH in a simulated environment where the available network bandwidth for the DASH client fluctuates overtime. The experiments were designed to verify the performance of the content-aware bitrate selection algorithm in two circumstances: stable network condition and more dynamic network connection. In these experiments, we compared the performance of our algorithm with the quality adaption algorithm used by the Adobe’s Open Source Media Framework (OSMF) [5].

In the first experiment under a stable network condition, we configured the average of the bandwidth of the network connection to the MOOC video player to 600kps. The MOOC video clip consists of three parts: the first and last part is regular motion video, and the middle part is mostly Slides-presentation and Instructor video. In the MOOC-DASH scheme, each video segment is encoded by its content-specific bitrate profile, among which the bitrate of Slides-Presentation is assigned to the value of 50kbs, the bitrate of Instructor-Centered video is set to 200kbs and regular motion video to 600kbs. The OSMF player follows a unified bitrate profile for all the MOOC video segments without knowledge about the characteristics of visual perception for different kinds of MOOC video content.

We utilize structural similarity (SSIM) to compare video quality between MOOC-DASH and OSMF. Fig. 4 demonstrates the SSIM of both video streams under MOOC-DASH and OSMF. The beginning part and trailing part of the MOOC video have intense motions, which demands for more bitrate than the middle part whose MOOC content is mostly Slides presentation. As the OSMF algorithm treats these parts at the same level of bitrate to match the specified network bandwidth, it needs more average bitrate (582kbs) than that of MOOC-DASH (407kbs) to obtain the same level of video quality as the quality of the encoded video from MOOC-DASH.

The second experiment is conducted under a bandwidth-varying network link fluctuating between 100kbs to 500kbs. Fig. 5 illustrates the behavior of bitrate adaption performed by MOOC-DASH and OSMF algorithm in the experiment. The blue line represents the available bandwidth, the orange line and grey line denotes the bitrate computed by MOOC-DASH and OSMF algorithm respectively. Due to the
fluctuation of the network bandwidth, the OSMF algorithm frequently changes the level of bitrate between 100Kbps, 200Kbps, and 300K. Such a frequent switch among bitrate levels significantly impairs the quality-of-experience of MOOC viewers. In contrast, MOOC-DASH takes a more smooth strategy to respond to the change in the network bandwidth. During the two-minute time frame, MOOC-DASH only increases the bitrate twice and attempts to keep a stable level of the bitrate, thus providing a smooth quality for MOOC viewers even in a dynamic network environment.

VI. CONCLUSION

As worldwide online learners watch thousands of MOOC video lectures in daily basis, it is important to deliver MOOC video streams through the HTTP transport in a high-quality way. Given the heterogeneous network access of Internet users, the DASH technique plays a vital role in adapting MOOC video streaming to dynamic network connections. But so far few research efforts have been made to investigate how to adopt the DASH techniques for streaming MOOC video content. Most DASH systems are designed for general HTTP video streaming and haven’t give much consideration about content-aware video adaption for MOOC video delivery. This paper introduces a content-aware DASH scheme called MOOC-DASH to improve QoE of MOOC video viewers. This scheme consists of content-aware ROI-based video encoding for MOOCs video and bitrate adaption algorithm to provide a high-quality and smooth video streaming service. It defines four major categories of MOOC videos based on the scenes in a MOOC video course, each of which exhibits a unique property of bitrate and QoE. We develop a decision-tree based classification algorithm that computes multiple features in each scene including motion areas, facial image and structural similarity to classify these four types of MOOC videos. Furthermore, for each category of MOOC video, we build an optimal video bitrate profile to support the content-aware bitrate adaption algorithm. The algorithm can prioritize the usage of bandwidth resource for different kinds of MOOC video content and smoothly regulate the video bitrate to reduce the occurrences of stalls or re-buffering.

Subject video quality measurements on MOOC video clips have verified the assumption that MOOC viewers have different perceptions towards bitrate change in different types of MOOC content. And multiple bitrate adaption experiments have been conducted to compare the performance of our MOOC-DASH player and OSMF player. Experimental results demonstrate that MOOC-DASH can effectively improve the QoE of MOOC viewers by presenting better quality-level for MOOC videos with more motions and keeping a stable quality-level for MOOC video streams.

REFERENCES


ACKNOWLEDGEMENT

This work was supported in part by grant from State Key Laboratory of Software Development Environment (Funding No. SKLSDE-2013ZX-03) and National High-Tech R&D Program of China (Grant No. 2013AA01A210).