IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 6 March 2024, accepted 19 March 2024, date of publication 27 March 2024, date of current version 2 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3382155

RESEARCH ARTICLE

QoE Aware Video Streaming Scheme Utilizing GRU-Based Bandwidth Prediction and Adaptive Bitrate Selection for Heterogeneous Mobile Networks

TIEN VU HUU [1], VAN SU PHAM [2], THAO NGUYEN THI HUONG [2], AND HAI-CHAU LE [3], (Member, IEEE)

[1]Faculty of Multimedia, Posts and Telecommunications Institute of Technology, Hanoi 10000, Vietnam
[2]Faculty of Electronics Engineering, Posts and Telecommunications Institute of Technology, Hanoi 10000, Vietnam
[3]Department of Data Engineering, Posts and Telecommunications Institute of Technology, Hanoi 10000, Vietnam

Corresponding author: Hai-Chau Le (chaulh@ptit.edu.vn)

This work was supported by the Research Collaboration Project between PTIT and Naver Corporation under Grant 02-PTIT-NAVER-2022.

ABSTRACT Nowadays, video streaming has become a popular form of multimedia for communication and entertainment. The rapid traffic explosion spurred by the development of emerging video-centric services such as e-science, virtual reality, and video conference has caused network congestion and the degradation in quality of experience (QoE), especially in heterogeneous mobile networks. To cope with that, the development of QoE-efficient video streaming solutions, i.e., HTTP adaptive video streaming, is critical. In this work, we investigate HTTP adaptive video streaming solutions that are capable of improving QoE for heterogeneous mobile networks in which the network conditions including bandwidth are significantly varied. We propose an effective QoE-aware adaptive bitrate video streaming scheme that integrates a bandwidth prediction based on Gated Recurrent Unit (GRU) neural networks with an adaptive bitrate selection strategy to wisely determine the suitable quality level for each video chunk. Thanks to the accurate bandwidth estimation and the adaptation of each video chunk bitrate to the network conditions, QoE metrics have been enhanced. Numerical experiments have been deployed to verify the performance of our proposed solution in comparison with that of the notable conventional methods. The attained simulation results demonstrate that the developed solution is significantly more effective than the conventional methods. The proposed method obtains a performance increment, in terms of QoE, of up to 19.4% compared to the conventional ones.

INDEX TERMS Video streaming, adaptive bitrate, HTTP adaptive streaming, quality of experience.

I. INTRODUCTION

Recently, the dominant digital content traffic over the Internet and mobile networks has been video-on-demand and video streaming, which is estimated to account for 70% percent of all mobile data traffic and is forecast to grow more in the near future [1], [2]. 5G and beyond mobile networks are expected to be capable of providing heterogeneous bandwidths and various service-level agreements. At the same time, users' demand becomes pickier in terms of video quality and availability of video services. Due to the abundance of video sources, users will easily stop consuming the video if its quality is not as expected [3]. Therefore, to satisfy users' expectations, content providers continuously enhance and deliver higher-quality video. However, there has been a trade-off between high-quality video expectations and video

The associate editor coordinating the review of this manuscript and approving it for publication was Alessandro Floris.

transmission over the network. High-quality video means a large amount of data to transmit, leading to longer delays and a higher probability of failure due to degraded network conditions [4], [5], [6].

To facilitate the video traffic explosion, the concept of HTTP adaptive streaming (HAS) protocol was introduced by the company Move Networks in 2007 as a delivery method for video streaming [7]. At the HAS server side, a raw video sequence is encoded at different bitrates and resolutions and then the encoded video sequence is divided into segments (video chunks). Bitrate and resolution are two of the most important factors that can affect the quality of video streaming, and they can be set in different combinations, so-called quality levels, to yield different qualities of video streaming. On the client side, video players estimate the network bandwidth and use the GET method to request video chunks with bitrates or quality levels suitable to network bandwidth to create a seamless playout.

To standardize HAS, the Moving Picture Experts Group (MPEG) and the 3rd Generation Partnership Project (3GPP) have introduced a standard called Dynamic Adaptive Streaming over HTTP (DASH) [8], [9]. Thanks to DASH, video transmission can exploit the available HTTP servers and Content Delivery Network (CDN) frameworks. Moreover, the DASH standard allows the client player to freely choose the optimal mechanism to meet its own requirement of the user's quality of experience (QoE) rather than using predetermined control rules. There has been broad industry support for the standard such as Microsoft Smooth Streaming [10], Apple Live HTTP Streaming [11], and Adobe HTTP Dynamic Streaming [12]. Two big content providers, YouTube and Netflix, have relied on DASH; YouTube has employed DASH as its default playing method and Netflix has been the largest DASH content provider [13], [14].

The core of DASH is an adaptive bitrate (ABR) algorithm that is an efficient solution for video delivery over the networks in order to meet the users' expectations [15], [16]. In DASH systems, a pool of video chunks with various bitrates and/or quality levels is available on the server side for dynamic and flexible delivery in different network conditions. The ABR algorithm plays a key role in determining suitable bitrates/quality levels of the requested video chunks based on observation factors such as network throughput, playback buffer occupancy, and video freezing ratio to enhance users' QoE [17], [18]. Most existing ABR algorithms use predetermined control rules for selecting the chunks. The rules, which are constructed based on some network and player conditions, can be grouped into three classes. The first one is the throughput-based class, in which information on downloaded video chunks is used to predict the network bandwidth, and based on the prediction of network bandwidth, the client player requests the most suitable video bitrate [19], [20]. The second is the buffer-based class in which buffer occupancy–bitrate functions are constructed for the player to select suitable bitrates for video chunks [21]. The third one, the so-called the hybrid class, combines both information on buffer occupancy and network bandwidth and then applies the optimization process to make bitrate adaptation rules [2], [15], [22]. Unfortunately, there exist some limitations in those methods. For example, buffer-based techniques cannot adapt well to the changes in network conditions whereas throughput-based techniques are more aggressive and only work well when the network is stable. On the other hand, hybrid techniques must compromise the performance to make the computation load possible for real-time bitrate adaptation.

In mobile networks, many disturbed factors such as variation in radio link, multipath fading, interference, and noise cause network link instability and continuous changes in network bandwidth [23]. Therefore, maintaining the optimal video bitrate, i.e., the optimal QoE, is a huge challenge. If the client player requests chunks with a bitrate too high, it likely faces video freezing and re-buffering when the network bandwidth fluctuates. In contrast, lowering the demand bitrate can decrease the probability of the aforementioned problems, but it will highly lead to a low or even unacceptable video quality. Since the network condition information is helpful for making rate adaptation policy, it is essential if the information can be foreseen in the short and/or long term. Many researchers attempted to estimate/predict network bandwidth for switching up or down the rate adaptation [15], [24], [25]. It is shown that with the aid of the foreseen network bandwidth knowledge the process of quality-chosen decision outperforms the conventional one [26].

Moreover, Thang et al. [25] introduced an aggressive bandwidth prediction while Tian and Liu [24] and Jiang et al. [15] also developed a conservative bandwidth estimation. However, these methods seem to be inefficient when applied in mobile networks. Google's Exoplayer, known as one of the most popular video streaming players, employs the window-sliding percentile for estimating the instantaneous bandwidth [27]. To improve the accuracy of the bandwidth prediction, machine-learning-based bandwidth prediction has been considered [28], [29].

In this work, we study QoE-aware HTTP adaptive video streaming problems in heterogeneous mobile networks in which network bandwidth is significantly and rapidly varied. We exploit the short-term and long-term memory abilities of a Gated Recurrent Unit (GRU) network to increase the accuracy of bandwidth prediction, and then, develop a modified rate adaptation policy, that is based on the accurately predicted bandwidth, aiming to enhance users' QoE. In our developed solution, a new customized GRU network is firstly proposed for the prediction of network bandwidth in both the short term and long term more accurately. The history of network bandwidth is taken into account with two perspectives namely variation range and variation trend when applied as the inputs of GRU. Secondly, an adaptive bitrate strategy is introduced to determine the best quality level of video chunks based on the predicted network bandwidth. The performance of the proposed solution will be evaluated by using numerical simulations and compared to that of notable conventional works.

The remainder of this paper is organized as follows. Firstly, Section II provides the research background and related works. Then, Section III describes our proposed QoE-aware adaptive bitrate video streaming solution to improve QoE for HAS mobile systems. Afterward, Section IV implements the experimental simulations, and presents the obtained performance evaluation results. Finally, Section V summarizes the work and discussion.

II. BACKGROUND AND RELATED WORKS

A. HTTP ADAPTIVE STREAMING

Figure 1 shows a typical HTTP adaptive streaming architecture [8], [9], [15]. In the HAS system, video content is encoded in different bitrates or quality levels, and is divided into smaller video chunks, e.g., one to ten seconds in length. Principally, each quality level is determined by its appropriate average bitrate and video resolution. These segments also can be independently decoded. In fact, at the beginning of a new session, the HAS client application downloads the manifest that contains the description of the segments and their available quality levels. With the consideration of current network and device states like bandwidth and buffer availability, the HAS client's adaptive bitrate strategy chooses the quality level of the next video chunk. The goal of the quality level adaptation is to improve the QoE metrics which rely on specific parameters including the number of times the video stalled, the average quality level, and the quality switching frequency [17], [18].

Compared to traditional real-time protocols, the main advantage of HAS is to enable the adjustment of video quality along with the available bandwidth to avoid video stalling. Therefore, HAS smooths video streaming with the best-effort approach over the network. Moreover, in HAS, video streams are transferred via HTTP and can exploit existing HTTP infrastructure, including HTTP servers, HTTP proxies, and CDN nodes, while passing through firewalls easily. That is the reason video-oriented service providers like Apple, Netflix, Microsoft, and Google have deployed massively HAS models. Most HAS solutions utilize a similar system architecture with DASH. Besides DASH, another popular HTTP-based adaptive bitrate streaming media protocol introduced by Apple in 2009 is HTTP Live Streaming (HLS). HLS, known to be the most popular streaming format, is similar to MPEG-DASH. HLS works by splitting the video stream into a sequence of small HTTP-based file downloads, each download loading one short block of an overall potentially unbounded transport stream. Media streams are also encoded at different bitrates and the available streams list is transferred to the client by using the expanded M3U playlist. Despite many advantages, DASH and HLS protocols still have to deal with some inefficiencies to cope with network variability and enhance users' QoE, especially in mobile networks and for live video streaming.

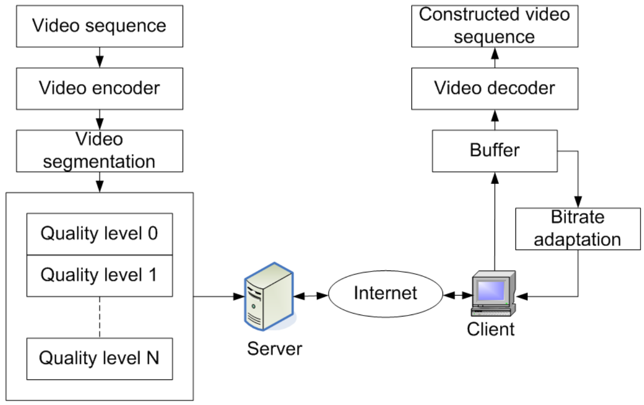

FIGURE 1. HTTP adaptive streaming architecture.

B. BANDWIDTH ESTIMATION FOR VIDEO STREAMING

Bandwidth estimation/prediction is an important module of the HAS systems. This module helps the system to choose a reasonable video rate and consequently avoid buffer exhaustion in video players. Up to now, there are many methods have been proposed for bandwidth estimation [25], [28], [29], [30], [31]. In these methods, the network bandwidth is estimated by transmitting probe packets and measuring the time it takes to receive a response packet. However, these methods are difficult to be deployed widely due to the modification requirements in the devices and standard protocols.

For more feasible, the bandwidth estimation usually is implemented at the application layer on the server side or client side. In [32] and [33], a machine learning model using a deep learning network is used to estimate the bandwidth on the server side. The bandwidth estimation at the server has the advantage that complicated algorithms can be implemented due to the high processing capacity of the server. However, in the estimation process, the server requires feedback information such as buffer state from the client. It may cause a delay in selecting the bitrate of the video chunk for bandwidth adaptation, especially in radio network environments with constantly changing bandwidth. Consequently, it may cause buffer overflow or exhaustion on the client side. To overcome this limitation, the bitrate selection algorithm may be moved to the client side [27], [29]. In [27], Exoplayer exploits the sliding percentile algorithm (denoted as SP) for the bandwidth prediction. The SP algorithm is based on k previous sample values of bandwidth and selects the bandwidth that ensures the sum of the weights calculated by multiplying a sliding percentile value by k equals or is greater than the required weight. Although the SP algorithm is simple to implement, it is a statistic-based estimation method and consequently, the accuracy of SP is not high. On the other hand, in [29], the authors developed

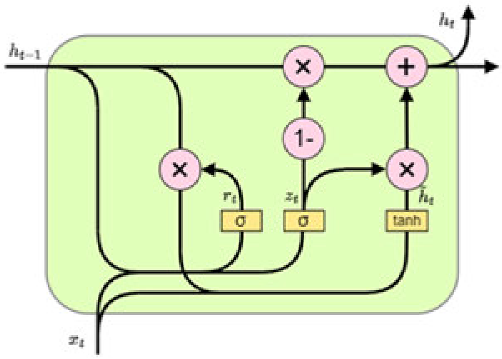

FIGURE 2. GRU model architecture [34].

an LSTM neural network solution to predict bandwidth. LSTM, shortened from Long Short-Term Memory, is a type of recurrent neural network (RNN) architecture specifically designed to address the vanishing gradient problem in traditional RNNs. Its unique structure allows it to retain information over long sequences, making it well-suited for tasks involving time series prediction. The advantage can even be further extended with Gated Recurrent Unit, an advancement of LSTM [34]. GRU has fewer gates and parameters while offering faster and more efficient performance than LSTM. Hence, in this work, we target an accurate bandwidth prediction method using a GRU neural network on the client side.

Figure 2 illustrates the working principle of a GRU model. A gated recurrent unit network was introduced to incorporate gating mechanisms that regulate the flow of information within the network and allow update and reset of hidden states selection [34]. In the realm of predicting h_t at time t^{th} based on the previously known values $x_{t-k},\ x_{t-k+1},\ \ldots,\ x_{t-1},\ x_t$ and hidden state h_{t-1}. Here, h_t is a linear interpolation and can be calculated as follows:

$$h_t = (1 - z_t)\, c \odot h_{t-1} + z_t \odot \widetilde{h}_t \tag{1}$$

The candidate activation $\widetilde{h}_t$ is calculated by (2) to reduce the effect of the previously hidden state h_{t-1}.

$$\widetilde{h}_t = tanh\ (x_t w_{xh} + (r_t \odot h_{t-1})\, w_{hh} + b_h) \tag{2}$$

When r_t close to 0, (2) makes effectively the unit act as if the first value of an input sequence is read, permitting it to ignore the previously computed hidden state. The update gate z_t and reset gate r_t are computed similarly as follows.

$$z_t = \sigma(x_t w_{xz} + h_{t-1} w_{hz} + b_z) \tag{3}$$
$$r_t = \sigma(x_t w_{xr} + h_{t-1} w_{hr} + b_r) \tag{4}$$

In (2), (3) and (4), b_h, b_r, b_z, w_{hh}, w_{hr}, w_{hz}, w_{xh}, w_{xr}, and w_{xz} are the networks' parameters that are needed to train, while $\odot$ in (2) is Hadamard product, also known as element-wise product, and σ in (3) and (4) is the sigmoid function.

As opposed to r_t, z_t decides how much the unit updates its activation or how much the previously known information is used to gain the next prediction. The GRU mechanism affects the whole state each time, instead of choosing which particular state is exposed.

C. QoE ASSESSMENT MODEL

Currently, many QoE evaluation models have been introduced and standardized [30], [35], [36], [37], [38]. The core challenge of an ABR strategy is to maximize the perceived QoE while coping with the system requirements, i.e., video smoothness and buffer size limitation, by dynamically adapting to the fluctuation of network conditions. Consequently, the applied QoE assessment model plays a key role in designing ABR algorithms. Recent advances in state-of-the-art DASH-based technologies show a noticeable transition from traditional QoE measurement that is based on video quality (e.g., Peak Signal-to-Noise Ratio) and user experience (e.g., subjective mean opinion scores) to more complex quality metrics (e.g., rebuffering time, video bitrate, startup delays) [16].

Particularly, a simple and efficient ABR strategy that was based on the client's buffer information was introduced early in [21]. Although this algorithm is light and helps reduce video freezing frequency, similar to other buffer-based ABR solutions, it only considers the buffer occupancy to determine the video bitrate. Unfortunately, the QoE model is normally more complicated and its metrics may include other important parameters, i.e., bandwidth. As a result, the QoE efficiency of buffer-based algorithms is limited. On the other hand, ExoPlayer, recently the most popular video streaming player developed by Google, utilizes a throughput-based ABR strategy for choosing the bitrate for the next video chunks [27]. Exoplayer's ABR algorithm selects 1) the highest bitrate for the next chunk when the highest bitrate is less than or equal to the estimated bandwidth modified by a pre-determined factor α, or 2) the proper bitrate in the set of 8 preset bitrate values if the predicted bandwidth becomes greater than the previous bandwidth while the buffer level remains low or if the predicted bandwidth is smaller than the previous one while the buffer level remains high. The flexible bitrate determination strategy helps to improve ABR performance however, it may still struggle with less accurate bandwidth prediction. Hence, QoE assessment methods adaptively taking into account the effects of system parameters including the bandwidth of the network, bitrate changing, and buffer occupancy have been introduced and standardized in [30], [35], [36], [37], [38], [39], and [40].

To date, two main QoE estimation methods, which are Mean Opinion Score (MOS) and utility score, are widely applied. The first method, MOS, is employed for quality-of-service monitoring while the second one, utility score, is usually applied in adaptive bitrate selection algorithms. The formulas in these two methods are different, but the input parameters are similar, including video and audio bitrates, resolution, framerate, and information related to rebuffering. MOS is determined by the MOS assessment model through the quality estimation process by a subjective method. Since MOS best reflects the image quality perceived by the user, many MOS evaluation methods have been investigated and proposed. These methods are different in input parameters, accuracy, and computational costs. In the HAS system, the viewer has no original video to compare to the received one. Therefore, a no-reference model can be used to estimate the MOS. Non-referential models are classified into models including metadata-based models [39], [40] and pixel-based models [38], [40].

On the other hand, the utility score is estimated by the sum of three parameters: bitrate, quality level change, and rebuffering [41]. The formula for calculating the utility score for a set of M consecutive video chunks is given as follows.

$$Utility = \sum_{n=1}^{M} Q(R_n) - \mu T - \lambda \sum_{n=1}^{M-1} |Q(R_{n+1}) - Q(R_n)| \tag{5}$$

where $Q(R_n)$ is the function measuring user perceived quality according to bitrate R_n and R_n is the bitrate of the n^{th} chunk. In [41], $Q(R_n)$ is set to R_n. $\triangle Q(R_n) = |Q(R_{n+1}) - Q(R_n)|$ is the penalty after each change in quality level. μ and λ are the weights for the penalty for the rebufferings and variability of quality, $\mu = 3000$ and $\lambda = 1$. T is the total rebuffering time. In the calculation of the utility score, the first element is the bitrate utility, and the second and the third elements are the penalties of rebuffering and bitrate change respectively. There are many variations of this QoE calculation model in ABR algorithms with useful bitrate calculations and weights for penalties [37], [42]. In this work, a method using utility score is proposed to improve QoE for video streaming systems using the DASH protocol. A detail of the proposed method is introduced in the following section.

III. PROPOSED METHOD

A. FRAMEWORK OF THE PROPOSED QoE-AWARE ADAPTIVE BITRATE VIDEO STREAMING SCHEME

The framework of the proposed method is described in Figure 3. On the video player side, a GRU model is utilized to predict the current bandwidth of the network. The output of the GRU model is then used as input to a video chunk selector accompanied by the current buffer occupancy. Based on input values, the video chunk selector selects the most suitable quality level for the next video chunk to achieve the maximum QoE metric. After that, a request including the chunk index n and corresponding quality level q of the chunk is sent to the video server.

B. PROPOSED GRU MODEL FOR BANDWIDTH ESTIMATION

In this section, we propose a bandwidth prediction algorithm based on the GRU model to estimate the network's available bandwidth for the next video chunk and assess its performance versus the ground-truth (real) data of mobile network bandwidths [43], [44]. The underlying problem for adaptive bitrate prediction generally requires high accuracy, a lightweight model, and a short inference time. Our proposed model architecture is illustrated in Figure 4. To generalize and enhance the contextual understanding of the model, we take a length-k sequence as an input including 3 features: *transferred data* measured in bits (the amount of data that the client transmits to the server in a measurement), *elapsed time* (the time it takes from transfer start to transfer end), and *bitrate* (transferred data divided by elapsed time). These features are fetched into a Bidirectional GRU (BiGRU) followed by a GRU layer and a Fully Connected Layer (FCL). While BiGRU simultaneously processes the input sequence in both directions (forward and backward), enabling capturing an overview of the data flow, GRU and FCL make convergence faster by their simple structure and prediction based on their previous known information.

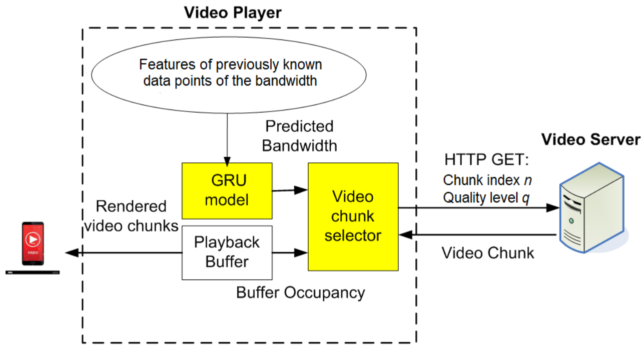

FIGURE 3. The framework of the proposed method.

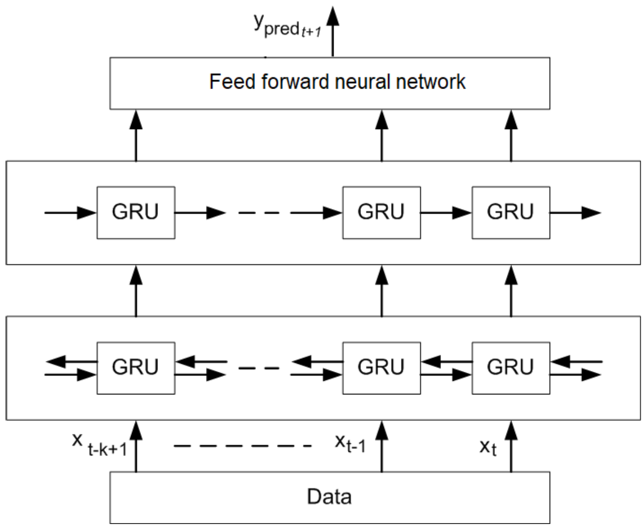

FIGURE 4. The architecture of the proposed bandwidth prediction model.

As mentioned above, our proposed model takes sequence data with length-k as input and estimates the following bandwidth value $y_{pred_{t+1}}$. Firstly, we convert our data measurements into megabits (Mb), second (s), and megabits per second (Mbps) respectively, in order to confine their fluctuations within a smaller range. Instead of normalizing

TABLE 1. Feature selection for the proposed model.

Exp	FEATURE SET	MAE	MSE
1	*elapsed time, transferred data, bitrate*	0.49 ± 0.07	0.45 ± 0.15
2	*elapsed time, bitrate*	0.57 ± 0.11	0.74 ± 0.39
3	*transferred data, bitrate*	0.72 ± 0.11	1.09 ± 0.2
4	*elapsed time, transferred data*	0.71 ± 0.02	1.43 ± 0.18

the input data, this approach helps us preserve the trends and accurate original values of the data.

Though bitrate can be calculated from the other two features, the experiment shows that three features still help the model achieve better results. The simulated results of this experiment are given in Table 1. In particular, four experiments (denoted as *Exp*s) are implemented and the accuracy is calculated by two typical metrics that are mean squared error (MSE) and mean absolute error (MAE). For the other 3 experiments, one feature is removed and the loss value is observed to see how it affects the model's performance. *Exp*. (2) demonstrates that the transferred data does not contribute significantly because when it is removed from the feature set, the appropriate MAE and MSE are the lowest among those of 3 other experiments with 2 features. However, a full feature set of three still gives the best result with an MAE equal to 0.49, a standard deviation equal to 0.07, an MSE equal to 0.45, and a standard deviation equal to 0.15. Figure 5 demonstrates the accuracy of the proposed model in the case of the average bandwidth of 3000 Kbps. It implies that the GRU-based method predicts the network bandwidth accurately in a real-time manner; the estimated bandwidth properly fits the ground truth curve of the real mobile network bandwidth and a little over or underestimated prediction only occurs with the peaks of sudden bandwidth drops having a negligible impact on video streaming.

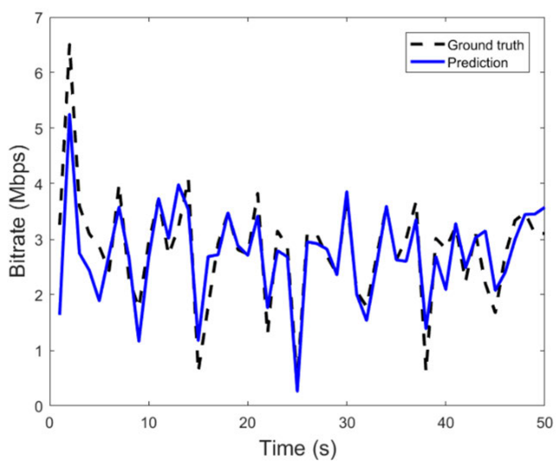

FIGURE 5. The accuracy of the proposed GRU-based bandwidth prediction model.

C. PROPOSED VIDEO CHUNK SELECTION ALGORITHM

Motivated by the algorithm in [41], we develop a novel adaptive bitrate selection algorithm (nABR) to figure out the most suitable quality level of the next video chunk. In [41], the optimal bitrate is selected to maximize QoE based on the current buffer state and the estimated bandwidth. However, the bitrate selection adopts a lookup table with available bitrate corresponding to a specific bandwidth and buffer state. Thus, the bitrate depends on the available bandwidth range of the look-up table. In our proposed method, the optimal quality level, a combination of bitrate and resolution, is computed online considering the attained QoE and the buffer state. In addition, in the QoE function, the weights of rebuffering and quality variation are considered to adapt to the buffer state. In particular, if the available buffer is increasing, it means that bandwidth is in a good state, and the role of rebuffering time is less influential than the variability of quality. Consequently, the weight of the rebuffering time is decreased while the weight of quality variation is increased. Otherwise, the weight of rebuffering time is increased and the weight of quality variation is decreased. Pseudo codes of the proposed algorithm are given as follows.

Algorithm 1 Novel Adaptive Bitrate Algorithm (*nABR*)

Input: R^*: Bitrate of the previously downloaded chunk
R_i: Bitrate of the video chunk corresponding to the quality level i^{th}
L: Length of the video chunk
$B_{current}$: The current buffer occupancy
B_{prev}: The buffer occupancy for the previous video chunk
C: The network bandwidth predicted by the GRU model
μ: Rebuffering time
λ: Quality switching frequency penalty
Output: *quality_level*: The optimized quality level of the next video chunk

$\mu := 4.3$
$\lambda := 1$
$QoE_{max} := 0$
$quality\_level := 0$
if $(B_{current} > B_{prev})$ **then**
  $\mu = \frac{\mu}{2}$
  $\lambda = \lambda \times 2$
else
  $\mu = \mu \times 2$
  $\lambda = \frac{\lambda}{2}$
end if
for $i = 0$ **to** $quality\_level_{max}$ **do**

$$QoE_{R_i} = R_i - \mu \times \left(\frac{R_i \times L}{C} - B_{current}\right) - \lambda \times |R_i - R^*| \quad (6)$$

  if $(QoE_{R_i} > QoE_{max})$ and $(R_i \leqslant C)$ **then**
    $QoE_{max} = QoE_{R_i}$
    $quality\_level = i$
  end if
end for
return *quality_level*

In (6), R_i is the bitrate utility concerning the quality level i^{th} which is considered for the current video chunk while R^* is the bitrate of the previous chunk. $\frac{R_i \times L}{C} - B_{current}$ and $|R_i - R^*|$ are terms which are considered respectively as the penalty of rebuffering and smoothness. The index i corresponds to quality level i^{th} of available video sequences at the server side. The target of the loop in the algorithm is to seek an optimal bitrate R_i among available quality levels of a video sequence to achieve the highest score of the QoE metric. The output of the algorithm is the quality level of the video chunk that the player expects to download from the server.

TABLE 2. The quality level of video sequences.

QUALITY LEVEL	RESOLUTION	BITRATE (*bits per second*)	
		3D animation	The other videos
0	1280 × 720	3370400	2962000
1	992 × 560	2270400	2056000
2	768 × 432	1170400	1427000
3	592 × 332	1170400	991000
4	448 × 252	1170400	688000
5	368 × 208	1170400	477000
6	284 × 168	1170400	331000
7	224 × 128	1170400	230000

IV. PERFORMANCE EVALUATION

A. EXPERIMENT SETUP

In this experiment, a testbed of an HTTP-based video streaming system is built on the Nginx web server [45]. Nginx is an open-source web server that uses an asynchronous, event-driven approach where requests are handled in a single thread. On the client side, ExoPlayer [27] is used to deploy the developed QoS-aware video streaming scheme incorporating the GRU-based bandwidth prediction algorithm (denoted as *GRU*) with the *nABR* algorithm, named *GRU_nABR*, and the comparable traditional video streaming methods [21], [27], [41]. Moreover, the dataset of practical mobile network bandwidths is collected from [43] and [44]. The dataset includes trace files recording the bandwidths of mobile networks in Belgium and the United States. Based on the bandwidth dataset, to evaluate a wide range of scenarios, Mahimahi tools [46] are employed to emulate mobile network environments that have various uplink and downlink bandwidths connecting the Nginx web server and ExoPlayer. On the other hand, the tested mobile network is also assumed to support three heterogeneous bandwidth video services including low speed (1000 Kbps), medium speed (1500 Kbps), and high speed (3000 Kbps).

In addition, the parameters used in the evaluation are *average QoE* and QoE component factors including *buffer occupation*, *average quality level*, and *quality switching frequency*. Buffer occupation is the data volume (measured in seconds) available in the buffer for playing out. This metric reflects the accuracy of the model to predict bandwidth and select the optimized quality level of video chunks to download. The higher the buffer occupation is, the lower the freezing ability of the video is. The quality switching frequency refers to the number of changes in quality level during the media playback. The average quality level is the average quality level of downloaded video chunks during the media playback. Quality level is assumed to range from 0 to 7 and the smaller the quality level is, the better video quality concerning bitrate and solution is obtained. The average QoE, as computed in (6), relies on the QoE component factors and reflects the overall quality experienced by the end user.

Furthermore, numerical simulations have been conducted for three different network bandwidth scenarios including low-speed (1000 Kbps), medium-speed (1500 Kbps), and high-speed (3000 Kbps). Each network bandwidth scenario was evaluated with 25 bandwidth samples randomly selected from the processed bandwidth dataset while each bandwidth sample was run with five pre-selected different video sequences covering music video, news video, action movie, 2D animation, and 3D animation video. The length of each video sequence is in range from 2 to 5 minutes. These video sequences are encoded with 8 quality levels corresponding to 8 resolutions. The details of the quality levels assigned to video sequences used in the simulation are summarised in Table 2. To evaluate the effectiveness, each method is tested with all five video sequences in each case of bandwidth. The final results are the ensemble averages of the network bandwidth scenarios.

Performance of the proposed QoE-aware video streaming scheme incorporating the novel dynamic bitrate adaptation strategy, *nABR*, with the developed GRU-based bandwidth prediction algorithm, *GRU*, is evaluated and compared to that of the notable traditional video streaming methods. The buffer-based scheme [21] only considers the buffer availability to determine the video bitrate while the throughput-based method [27], Exoplayer, includes Sliding Percentile algorithm (named *SP*) to estimate the network bandwidth and a simple adaptive bitrate algorithm, denoted as *eABR*. The efficient hybrid-based method given in [41] consists of a conventional ABR algorithm (named *cABR*) and an LSTM-based bandwidth prediction method (denoted as *LSTM*). Firstly, the *GRU* efficiency is estimated in comparison with that of *SP* and *LSTM* algorithms to verify the bandwidth prediction enhancement of our developed method. Performances of the appropriate ABR algorithms (*eABR*, *cABR*, and *nABR*) are then assessed concerning three examined bandwidth prediction approaches including *SP*, *LSTM*, and *GRU*. Finally, a performance comparison of our proposed method and the three remarkable throughput-based, buffer-based, and hybrid-based methods is conducted to validate the effectiveness of our proposed QoE-aware video streaming scheme.

B. BANDWIDTH PREDICTION EVALUATION

To evaluate the accuracy of bandwidth prediction methods, the proposed prediction method (*GRU*) is compared to the sliding percentile algorithm (*SP*) and the LSTM-based method (*LSTM*). The numerical experiments have been

TABLE 3. Comparison of the comparable bandwidth prediction algorithms.

BANDWIDTH (Kbps)	ALGORITHM	MAE	MSE
3000	GRU	$\mathbf{0.49 \pm 0.07}$	$\mathbf{0.45 \pm 0.15}$
	$LSTM$	0.52 ± 0.07	0.75 ± 0.17
	SP	0.96 ± 0.07	1.5 ± 0.19
1500	GRU	$\mathbf{0.44 \pm 0.05}$	$\mathbf{0.42 \pm 0.25}$
	$LSTM$	0.52 ± 0.09	0.52 ± 0.09
	SP	0.61 ± 0.04	0.93 ± 0.35
1000	GRU	$\mathbf{0.2 \pm 0.02}$	$\mathbf{0.09 \pm 0.02}$
	$LSTM$	0.23 ± 0.06	0.42 ± 0.3
	SP	0.37 ± 0.05	0.22 ± 0.1

done with various bandwidth services including low-speed, medium-speed, and high-speed bandwidths and the obtained results are summarized in Table 3. The results show that in all cases of bandwidths, the proposed model, GRU, always achieves the highest accuracy while SP, the simplest algorithm, attains the worst performance. Especially, in the case of high bandwidth (3000 Kbps), the GRU model gains accuracy higher than SP 49% in terms of MAE and 70% in terms of MSE. In the case of low bandwidth (1000 Kbps), the MSE of the GRU model is the smallest. This means that the variance of the proposed model is very small compared to the other models. It is worth mentioning that, similar to SP, GRU has a faster inference time compared to that of LSTM, however, the difference is almost negligible.

C. QoE FACTORS EVALUATION

Firstly, the performance of the simplest ABR algorithm, the Exoplayer's original adaptive bitrate strategy (*eABR*), has been evaluated concerning various bandwidth prediction techniques including SP, $LSTM$, and GRU. The appropriate adaptive bitrate variants are named *SP_eABR*, *LSTM_eABR*, and *GRU_eABR* respectively. Table 4 shows the effectiveness of the considered adaptive bitrate variants in terms of the parameters including buffer occupation, average quality level, quality switching frequency, and average QoE. The results demonstrate that *GRU_eABR* always offers the best performance while SP_eABR is the worst, in terms of the average QoE. The reason is that with the same ABR algorithm, the overall performance strongly relies on the efficiency of the bandwidth prediction; applying a more accurate algorithm helps improve the performance.

Regarding the component parameters of QoE, the *GRU_eABR* also provides the best values in most cases except low-speed bandwidth. In the case of bandwidth 1000 Kbps, *GRU_eABR* is better than those of SP and LSTM in terms of the average quality level while the quality switching frequency is higher. The reason is that in low bandwidth conditions, the GRU-applied method needs to change the quality level more rapidly than the other methods to adjust buffer occupation. Consequently, its buffer occupation and average quality level have been improved and these lead to an enhancement of the average QoE. In the case of bandwidth 1500 Kbps and 3000 Kbps,

TABLE 4. QoE influent factors comparison of four methods using ABR algorithm in exoplayer (***eABR***).

QoE FACTOR	SP_eABR	$LSTM\_eABR$	GRU_eABR
1000 Kbps			
Buffer occupation	3.1	3.0	**3.2**
Average quality level	7.0	**6.6**	**6.6**
Quality switching frequency	**0**	2	2
Average QoE	1080.4	1205.8	**1313.3**
1500 Kbps			
Buffer occupation	33.03	23.13	**31.36**
Average quality level	5.0	**4.4**	**4.4**
Quality switching frequency	14	9	**8**
Average QoE	13554	10234.4	**13829.2**
3000 Kbps			
Buffer occupation	44.09	46.55	**51.37**
Average quality level	2.58	2.54	**2.42**
Quality switching frequency	12	12	**10**
Average QoE	19708.08	20122.69	**22773.08**

thanks to higher bandwidth prediction accuracy, *GRU_eABR* achieves better results than the others. In particular, the buffer occupation of *GRU_eABR* is higher than that of *SP_eABR* and *LSTM_eABR* up to 16.6%, and 35.6% respectively. *GRU_eABR* also offers up to 12% and 4.72% better quality level (smaller value of average quality level) than *SP_eABR* and *LSTM_eABR* severally, while its quality switching frequency is respectively 42.9% and 16.7% lower than the comparative ones. Consequently, the average QoE of *GRU_eABR* gains 15.6%, and 35.1% increment compared to *SP_eABR* and *LSTM_eABR*.

On the other hand, Figures 6, 7 and 8 demonstrate the obtained buffer occupation, quality levels, and QoE of the three compared methods with respect to running time when streaming a 3D animation with the bandwidth of 3000 Kbps. Graphs of buffer occupation and QoE as illustrated in Figures 6 and 8 show that they are accumulated and varied with the variation of bandwidth, and as a result, estimating bandwidth more precisely provides better performance. It is recognized that applying our developed bandwidth prediction

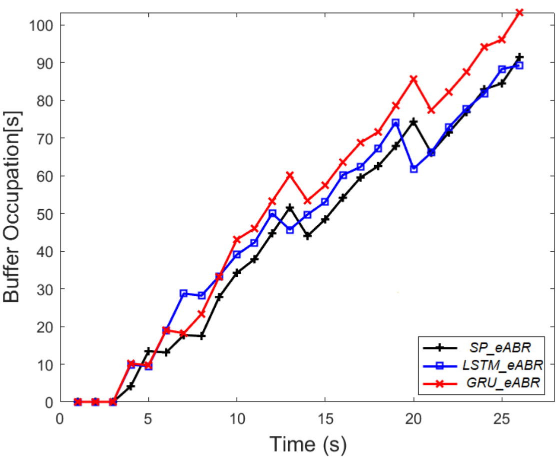

FIGURE 6. Buffer occupation of three methods in case of bandwidth 3000 Kbps.

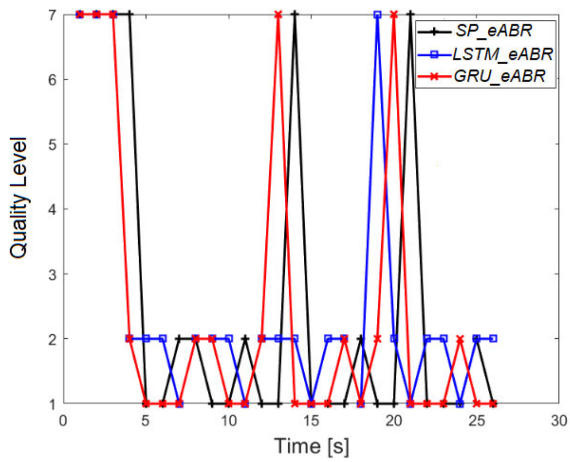

FIGURE 7. The quality level of the three methods in the case of bandwidth is 3000 Kbps.

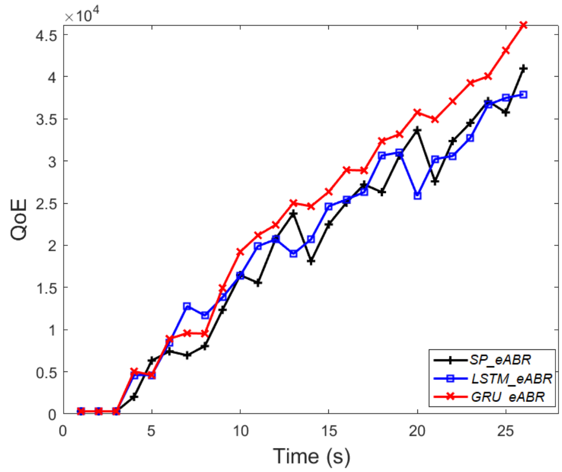

FIGURE 8. QoE values three methods in case of bandwidth 3000 Kbps.

TABLE 5. QoE influent factors comparison of three methods using *cABR*.

QoE FACTOR	*SP_cABR*	*LSTM_cABR*	*GRU_cABR*
1000 Kbps			
Buffer occupation	22.0	27.5	**27.9**
Average quality level	7.0	6.6	**6.0**
Quality switching frequency	**0**	2	2
Average QoE	9214.6	11825.0	**12067.5**
1500 Kbps			
Buffer occupation	63.82	59.16	**64.36**
Average quality level	4.8	6.0	**4.6**
Quality switching frequency	13	**6**	9
Average QoE	26790.4	25662.8	**27486**
3000 Kbps			
Buffer occupation	74.92	82.38	**82.72**
Average quality level	2.50	2.42	**2.31**
Quality switching frequency	15	**7**	10
Average QoE	32760.38	35004.62	**35466.54**

TABLE 6. QoE influent factors comparison of three methods using *nABR*.

QoEFACTOR	*SP_nABR*	*LSTM_nABR*	*GRU_nABR*
1000 Kbps			
Buffer occupation	21.6	**23.6**	23.1
Average quality level	7.0	6.6	**6.1**
Quality switching frequency	**0**	2	4
Average QoE	1799077	2075001	**2140582**
1500 Kbps			
Buffer occupation	58.66	63.85	**64.63**
Average quality level	4.72	4.52	**4.2**
Quality switching frequency	12	**7**	16
Average QoE	3660353	4069831	**4371758**
3000 Kbps			
Buffer occupation	74.64	80.73	**89.00**
Average quality level	2.69	2.27	**2.23**
Quality switching frequency	15	15	**7**
Average QoE	3774506	3968922	**3987345**

surpasses the other approaches most time, except in the initial streaming stage. Figure 7 shows changing quality levels adaptively during the video streaming process to adjust the bitrate and resolution of video chunks for improving the QoE.

In addition, a performance comparison of the adaptive bitrate variants combining *cABR* and one of three different bandwidth prediction strategies (*SP*, *LSTM*, and *GRU*), so-called *SP_cABR*, *LSTM_cABR*, and *GRU_cABR* orderly, are summarized in Table 5. The numerical results also verify that the method with our developed bandwidth prediction, *GRU_cABR*, achieves the best QoE over the compared adaptive bitrate methods, *SP_cABR* and *LSTM_cABR*. It means that with the same *cABR* algorithm, higher accurate bandwidth prediction such as *GRU* provides better performance, in terms of QoE. However, regarding the QoE component factors, it is worth noting that the buffer occupation and the quality switching frequency of *GRU_cABR* are always the best (greatest buffer occupation value and smallest average quality level value) whereas, in some cases, the required quality switching frequency is high, comparing to those of *SP_cABR* and *LSTM_cABR*. It implies that even the quality switching frequency needs to be kept as small as possible to ensure video smoothness, an increment of the quality switching frequency may help to switch more often to higher video quality levels (bitrate and/or resolution) and as a result, the overall QoE is enhanced.

Moreover, the efficiency of both our proposed bandwidth prediction, *GRU*, and adaptive bitrate, *nABR*, strategies is further emphasized with the comparison results obtained among the combinations of *nABR* and the corresponding bandwidth prediction approaches, named *SP_nABR*, *LSTM_nABR*, and *GRU_nABR* consecutively, shown in Table 6. It is confirmed that our proposed solution, *GRU_nABR*, integrating *nABR* with *GRU* outperforms the comparable methods and offers the highest QoE. Particularly, in the case of bandwidth 1500 Kbps, the average QoE of *GRU_nABR* is up to 19.4% and 7.4% respectively higher than that of *SP_nABR* and *LSTM_nABR*. In addition, the average quality level of *GRU_nABR* is also higher than *SP_nABR* and *LSTM_nABR* up to 21.74% and 12% orderly.

Finally, to clarify thoroughly the effectiveness of our proposed QoE-aware video streaming scheme, its performance, in terms of the average QoE, is compared to that of three

TABLE 7. QoE performance comparison.

BANDWIDTH	Ref. [21] (Buffer-based)	Ref. [27] (Exoplayer) (Throughput-based)	Ref. [41] (Hybrid-based)	Proposed method
1000 Kbps	980	1080.4	11825.0	**2140582**
1500 Kbps	14274	13554	25662.8	**4371758**
3000 Kbps	20079.60	19708.08	35004.62	**3987345**

remarkable methods, mentioned in previous sections, representative of three typical adaptive bitrate strategy classes; the worldwide adopted throughput-based approach (Exoplayer) [27], the notable buffer-based method [21], and the comparatively efficient hybrid-based one [41]. The obtained simulation results are summarized in Table 7. It demonstrates that thanks to the integration of the QoE-aware dynamic adaptive bitrate and the highly accurate bandwidth prediction algorithm, our proposed solution significantly outperforms other comparing solutions. The other hybrid-based method comes after because of less effective ABR and bandwidth prediction algorithms, however, it still attains relatively higher QoE on average than both the throughput-based and the buffer-based approaches. On the other side, the throughput-based method provides the worst performance. It is only slightly better than the buffer-based method in the case of low-speed bandwidths when estimating the network bandwidth plays a more important role in improving the QoE than controlling the buffer.

V. CONCLUSION

In this paper, we have proposed an efficient video streaming strategy that incorporates a GRU-based bandwidth estimation and an adaptive bitrate selection algorithm to enhance the QoE of adaptive bitrate video streaming mobile systems. In our approach, the short-term and long-term memory abilities of the GRU neural network are exploited to provide the accurate predicted bandwidth information for determining wisely and adaptively the most suitable bitrate of video chunks based on the conditions of mobile devices and networks. The performance of the proposed video streaming solution has been verified and compared to that of typical conventional works by using numerical simulations. The obtained results imply that our developed scheme outperforms the comparable ones and offers the best quality of experience. It can achieve up to 19.4% QoE increase on average.

REFERENCES

[1] Ericsson. (2022). *Mobile Data Traffic Outlook*. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report/dataforecasts/mobile-traffic-forecast

[2] A. H. Zahran and C. J. Sreenan, "ARBITER: Adaptive rate-based intelligent HTTP streaming algorithm," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Seattle, WA, USA, Jul. 2016, pp. 1–6, doi: 10.1109/ICMEW.2016.7574709.

[3] F. Dobrian, A. Awan, D. Joseph, A. Ganjam, J. Zhan, V. Sekar, I. Stoica, and H. Zhang, "Understanding the impact of video quality on user engagement," *Commun. ACM*, vol. 56, no. 3, pp. 91–99, Mar. 2013, doi: 10.1145/2428556.2428577.

[4] G. Lv, Q. Wu, Q. Tan, W. Wang, Z. Li, and G. Xie, "Accurate throughput prediction for improving QoE in mobile adaptive streaming," *IEEE Trans. Mobile Comput.*, early access, Sep. 11, 2023, doi: 10.1109/TMC.2023.3313592.

[5] S. Laine and I. Hakala, "Network capacity estimators predicting QoE in HTTP adaptive streaming," *IEEE Access*, vol. 10, pp. 9817–9829, 2022, doi: 10.1109/ACCESS.2022.3145185.

[6] H. B. Thanh, S. N. Quang, T. V. Huu, and X. HoangVan, "Learning adaptive motion search for fast versatile video coding in visual surveillance systems," *IET Image Process.*, vol. 18, no. 4, pp. 981–995, Mar. 2024, doi: 10.1049/ipr2.13001.

[7] H. Yang, X. Chen, Z. Yang, X. Zhu, and Y. Chen, "Opportunities and challenges of HTTP adaptive streaming," *Int. J. Future Gener. Commun. Netw.*, vol. 7, no. 6, pp. 165–180, 2014, doi: 10.14257/ijfgcn.2014.7.6.16

[8] I. Sodagar, "The MPEG-DASH standard for multimedia streaming over the internet," *IEEE Multimedia Mag.*, vol. 18, no. 4, pp. 62–67, Apr. 2011, doi: 10.1109/MMUL.2011.71.

[9] *Dynamic Adaptive Streaming Over HTTP (DASH) Part 1: Media Presentation Description and Segment Formats*, Standard ISO/IEC 23009-12019, 2019.

[10] A. Zambelli, "IIS smooth streaming technical overview," Microsoft Corp., White Paper 3.40, Mar. 2009.

[11] Apple. *HTTP Live Streaming Overview*. Accessed: Sep. 15, 2023. [Online]. Available: https://developer.apple.com/documentation/http_live_streaming

[12] Adobe. *HTTP Dynamic Streaming*. Accessed: Sep. 15, 2023. [Online]. Available: http://www.adobe.com/cn/products/hds-dynamic streaming.html

[13] J. Roettgers. (2017). *Don't Touch That Dial: How YouTube is Bringing Adaptive Streaming to Mobile, TVs*. [Online]. Available: http://www.gigaom.com/2013/03/13/youtubeadaptive-streaming-mobile-tv

[14] V. K. Adhikari, Y. Guo, F. Hao, M. Varvello, V. Hilt, M. Steiner, and Z.-L. Zhang, "Unreeling netflix: Understanding and improving multi-CDN movie delivery," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1620–1628.

[15] J. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in HTTP-based adaptive video streaming with festive," *IEEE/ACM Trans. Netw.*, vol. 22, no. 1, pp. 326–340, Feb. 2014, doi: 10.1109/TNET.2013.2291681.

[16] J. Kua, G. Armitage, and P. Branch, "A survey of rate adaptation techniques for dynamic adaptive streaming over HTTP," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1842–1866, 3rd Quart., 2017, doi: 10.1109/COMST.2017.2685630.

[17] X. Zuo, J. Yang, M. Wang, and Y. Cui, "Adaptive bitrate with user-level QoE preference for video streaming," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, May 2022, pp. 1279–1288.

[18] D. Lorenzi, "QoE- and energy-aware content consumption for HTTP adaptive streaming," in *Proc. 14th Conf. ACM Multimedia Syst.*, Jun. 2023, pp. 348–352.

[19] C.-H. Lin, C.-K. Shieh, W.-S. Hwang, and D.-Y. Liu, "Throughput-based mapping algorithm for video streaming over IEEE 802.11e WLAN," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–14, Oct. 2021, doi: 10.1155/2021/8852095.

[20] S. Keshvadi and Y. Sharma, "Enhancing video streaming quality of service with software-defined networking and network slicing: A scalable AV1 approach," in *Proc. 19th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Jun. 2023, pp. 255–260, doi: 10.1109/WiMob58348.2023.10187887.

[21] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," *Comput. Commun. Rev.*, vol. 44, no. 4, pp. 187–198, 2015, doi: 10.1145/2619239.2626296.

[22] Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. C. Begen, and D. Oran, "Probe and adapt: Rate adaptation for HTTP video streaming at scale," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 4, pp. 719–733, Apr. 2014, doi: 10.1109/JSAC.2014.140405.

[23] G. L. Stüber, *Principles of Mobile Communication*, vol. 9781461403. Boston, MA, USA: Kluwer, 2012, pp. 1–830, doi: 10.1007/978-1-4614-0364-7.

[24] G. Tian and Y. Liu, "Towards agile and smooth video adaptation in HTTP adaptive streaming," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 2386–2399, Aug. 2016.

[25] T. C. Thang, Q.-D. Ho, J. W. Kang, and A. T. Pham, "Adaptive streaming of audiovisual content using MPEG DASH," *IEEE Trans. Consum. Electron.*, vol. 58, no. 1, pp. 78–85, Feb. 2012, doi: 10.1109/TCE.2012.6170058.

[26] S. Kanhere, "On-move 2016 keynote: Optimizing HTTP-based adaptive streaming in vehicular environment using Markov decision process," in *Proc. IEEE 41st Conf. Local Comput. Netw. Workshops (LCN Workshops)*, Dubai, United Arab Emirates, Nov. 2016, p. 26, doi: 10.1109/LCN.2016.019.

[27] Exoplayer. *Exoplayer's Development Document*. Accessed: Sep. 20, 2023. [Online]. Available: https://exoplayer.dev/

[28] H. Du, Q. Zheng, W. Zhang, and X. Gao, "A bandwidth variation pattern-differentiated rate adaptation for HTTP adaptive streaming over an LTE cellular network," *IEEE Access*, vol. 6, pp. 9554–9569, 2018, doi: 10.1109/ACCESS.2017.2788057.

[29] A. Biernacki, "Improving streaming video with deep learning-based network throughput prediction," *Appl. Sci.*, vol. 12, no. 20, p. 10274, Oct. 2022, doi: 10.3390/app122010274.

[30] K. Yamagishi and T. Hayashi, "Parametric quality-estimation model for adaptive-bitrate-streaming services," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1545–1557, Jul. 2017, doi: 10.1109/TMM.2017.2669859.

[31] K. Yamagishi, "Audio-visual quality estimation device, method for estimating audio-visual quality, and program," U.S. Patent 10 869 072, Dec. 15, 2020.

[32] H. Mao, R. Netravali, and M. Alizadeh, "Neural adaptive video streaming with pensieve," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2017, pp. 197–210, doi: 10.1145/3098822.3098843.

[33] A. Lekharu, K. Y. Moulii, A. Sur, and A. Sarkar, "Deep learning based prediction model for adaptive video streaming," in *Proc. Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2020, pp. 152–159, doi: 10.1109/COMSNETS48256.2020.9027383.

[34] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1724–1734, doi: 10.3115/v1/D14-1179.

[35] T. Kimura, T. Kimura, A. Matsumoto, and K. Yamagishi, "Balancing quality of experience and traffic volume in adaptive bitrate streaming," *IEEE Access*, vol. 9, pp. 15530–15547, 2021, doi: 10.1109/ACCESS.2021.3052552.

[36] M. Li and C.-Y. Lee, "A cost-effective and real-time QoE evaluation method for multimedia streaming services," *Telecommun. Syst.*, vol. 59, no. 3, pp. 317–327, Jul. 2015, doi: 10.1007/s11235-014-9938-8.

[37] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a 'completely blind' image quality analyzer," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, Mar. 2013, doi: 10.1109/lsp.2012.2227726.

[38] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Trans. Image Process.*, vol. 21, no. 12, pp. 4695–4708, Dec. 2012, doi: 10.1109/TIP.2012.2214050.

[39] *Parametric Bitstream-Based Quality Assessment of Progressive Download and Adaptive Audiovisual Streaming Services Over Reliable Transport*, document ITU-T, AH Recommendation, 2016, p. 1203.

[40] Y. Liu, S. Dey, F. Ulupinar, M. Luby, and Y. Mao, "Deriving and validating user experience model for DASH video streaming," *IEEE Trans. Broadcast.*, vol. 61, no. 4, pp. 651–665, Dec. 2015, doi: 10.1109/TBC.2015.2460611.

[41] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 325–338, Sep. 2015, doi: 10.1145/2829988.2787486.

[42] Z. Akhtar, Y. S. Nam, R. Govindan, S. Rao, J. Chen, E. Katz-Bassett, B. Ribeiro, J. Zhan, and H. Zhang, "Oboe: Auto-tuning video ABR algorithms to network conditions," in *Proc. Conf. ACM Special Interest Group Data Commun.*, Aug. 2018, pp. 44–58.

[43] The Federal Communications Commission. *Measuring Broadband America Mobile Data*. Accessed: Sep. 20, 2023. [Online]. Available: https://www.fcc.gov/reportsresearch/reports/measuring-broadband-america/measuring-broadbandamerica-mobile-data

[44] J. van der Hooft, S. Petrangeli, T. Wauters, R. Huysegems, P. R. Alface, T. Bostoen, and F. De Turck, "HTTP/2-based adaptive streaming of HEVC video over 4G/LTE networks," *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2177–2180, Nov. 2016, doi: 10.1109/LCOMM.2016.2601087.

[45] *NGINX Web Server*. Accessed: Aug. 10, 2022. [Online]. Available: https://www.nginx.com/

[46] *MIT Mahimahi Tool*. Accessed: Aug. 10, 2022. [Online]. Available: http://mahimahi.mit.edu/

TIEN VU HUU was born in Hanoi, Vietnam, in 1979. He received the B.E. and M.E. degrees in electronics and telecommunications engineering from Hanoi University of Science and Technology (HUST), Vietnam, in 2002 and 2004, respectively, and the D.Eng. degree in electronics and telecommunications engineering from Chulalongkorn University, Thailand, in 2011. He is currently a Lecturer with the Posts and Telecommunications Institute of Technology (PTIT), Vietnam. He is also the Head of the Multimedia Application Development Department, PTIT. He is also the Deputy Head of the PTIT-Naver AI Center, PTIT. He is the first author of one book and more than 30 articles. His research interests include video quality assessment, video coding, image processing, computer vision, and machine learning.

VAN SU PHAM was born in Hanoi, Vietnam, in 1976. He received the B.E. and M.E. degrees in electronic and telecommunication engineering from the Posts and Telecommunications Institute of Technology (PTIT), Hanoi, in 1999 and 2004, respectively. He is currently pursuing the Ph.D. degree in computer engineering. He is a Lecturer with the Department of Electronics Engineering, PTIT. His research interests include signal and image processing, computer vision, and machine learning/deep learning.

THAO NGUYEN THI HUONG was born in Thái Bình, Vietnam. She received the B.E., M.E., and D.Eng. degrees in electronic and telecommunication engineering from the Posts and Telecommunications Institute of Technology (PTIT), Vietnam, in 2003, 2007, and 2021, respectively. Currently, she is a Lecturer with the Telecommunications Faculty, PTIT. She is also a Key Researcher with the PTIT-Naver AI Center, PTIT. Her research interests include distributed video coding, rate control for video coding, image processing, and machine learning.

HAI-CHAU LE (Member, IEEE) was born in Hanoi, Vietnam. He received the B.E. degree in electronics and telecommunications engineering from the Posts and Telecommunications Institute of Technology (PTIT), Vietnam, in 2003, and the M.Eng. and D.Eng. degrees in electrical engineering and computer science from Nagoya University, Japan, in 2009 and 2012, respectively. From 2003 to 2006, he was a Lecturer with the Telecommunications Faculty, PTIT. He was a Researcher with Nagoya University, from 2012 to 2014. He was a Research Scientist with the University of California at Davis. He is currently an Associate Professor and the Head of the Data Engineering Department, PTIT, Hanoi Campus. He is also the Head of the Data and Intelligent Systems Laboratory, PTIT. He is the author of two books and more than 70 articles. His research interests include optical network technologies, network design and optimization, future internet technologies, and machine learning. He was a recipient of the Tenth Conference on Lasers and Electro-Optics Pacific Rim and the 18th OptoElectronics and Communications Conference/Photonics in Switching (CLEO-PR&OECC/PS 2013) Best Paper Award, in 2013, and the IEEE Ninth International Conference on Communications and Electronics Best Paper Award, in 2022.