Unsupervised QoE field study for mobile YouTube video streaming with YoMoApp

Michael Seufert, Nikolas Wehner, Florian Wamser, Pedro Casas, Alessandro D’Alconzo, Phuoc Tran-Gia

Angaben zur Veröffentlichung / Publication details:

Seufert, Michael, Nikolas Wehner, Florian Wamser, Pedro Casas, Alessandro D’Alconzo, and Phuoc Tran-Gia. 2017. “Unsupervised QoE field study for mobile YouTube video streaming with YoMoApp.” In *Ninth International Conference on Quality of Multimedia Experience (QoMEX), 31 May - 2 June 2017, Erfurt, Germany*, edited by Stephan Werner, Steve Göring, Glenn van Wallendael, and Werner Robitza, 1–6. Piscataway, NJ: IEEE. https://doi.org/10.1109/qomex.2017.7965688.

Unsupervised QoE Field Study for Mobile YouTube Video Streaming with YoMoApp

Michael Seufert*, Nikolas Wehner*, Florian Wamser*, Pedro Casas†, Alessandro D'Alconzo†, Phuoc Tran-Gia*

*University of Würzburg, Institute of Computer Science, Würzburg, Germany
{seufert | nikolas.wehner | florian.wamser | trangia}@informatik.uni-wuerzburg.de
†AIT Austrian Institute of Technology, Vienna, Austria
{pedro.casas | alessandro.dalconzo}@ait.ac.at

***Abstract*—YoMoApp (YouTube Monitoring App) is an Android app to monitor mobile YouTube video streaming on both application- and network-layer. Additionally, it allows to collect subjective Quality of Experience (QoE) feedback of end users. During the development of the app, the stable versions of YoMoApp were already available in the Google Play Store, and the app was downloaded, installed, and used on many devices to monitor streaming sessions. As the app was not advertised in special campaigns or used for dedicated QoE studies, the monitored streaming sessions of this period compose the data set of a large unsupervised field study. The collected data set is evaluated to characterize current mobile YouTube streaming on both application and network layers. Furthermore, the problems and methodology to obtain QoE results from such unsupervised field study are discussed together with the actual QoE results. Correlations between QoE factors are investigated, and the QoE of clusters of similar streaming sessions is analyzed.**

I. Introduction

With mobile video streaming being one of the most popular and most demanding Internet services, it poses huge challenges to mobile network operators. They strive to deliver the video data efficiently within the constrained cellular networks, but have to achieve a high Quality of Experience (QoE) to satisfy their customers. Thus, it is of paramount importance to mobile network operators to understand the demands of mobile video streaming and the perceived streaming quality of end users.

Nowadays, almost all mobile video streaming services utilize HTTP adaptive video streaming (HAS) technology to align the video demands to the network conditions. This means, the client-side adaptation logic can change the video bit rate to reflect the fluctuating throughput in the network. The ultimate goal is to avoid stalling, i.e., a playback buffer underrun due to insufficiently downloaded data, which is the worst QoE degradation of video streaming. To avoid stalling, the bit rate of the video is reduced, which was shown to have a smaller negative impact on the QoE [1]. For example, the popular streaming portal YouTube switches the resolution of videos when the network conditions change.

This paper presents YoMoApp (YouTube Monitoring App), an Android application to monitor mobile YouTube streaming on application and network layers. The app gives insights into the streaming process and can also be used to obtain subjective QoE ratings from end users. Thus, the app is a valuable tool for studying the QoE of HAS in detail. The app was published in the Google Play Store and more than 1250 sessions have been monitored since July 2014 until November 2016. As the app was not advertised in special campaigns or used for dedicated QoE studies, the monitored streaming sessions compose the data set of a large unsupervised field study. In this paper, YoMoApp will be presented and the unsupervised field study will be described. Furthermore, the collected data set is evaluated in terms of application- and network-layer characteristics of mobile YouTube streaming. The problems and methodology to obtain QoE results from such data will be discussed together with the actual QoE results, and the lessons learned from the study will be summarized.

This paper is structured as follows. Section II will outline related work on QoE of HAS and QoE monitoring. In Section III, YoMoApp is presented and the unsupervised field study is characterized. Afterwards, Section IV describes the evaluation of the data set and presents the gained insights. Finally, the paper is concluded in Section V.

II. Related Work

QoE assessment of HTTP video streaming is a well known and widely researched problem. [2]–[4] showed that initial delay and stalling events are the key factors influencing the QoE of video streaming. While most users were indifferent to moderate initial delays, already little stalling severely decreased the QoE. With the growing commercial usage of HTTP adaptive video streaming, stalling can be traded off for quality adaptation. [5] revealed that, in a mobile environment with fluctuating bandwidth, stalling could be reduced up to 80%, and the available bandwidth could be better utilized. Nevertheless, [6] found that quality switches also impact the QoE depending on the switching direction, i.e., the increase or decrease of the video quality. [7] showed that the number of quality switches can be neglected, while the time on each quality layer has to be considered as a QoE factor [8]. [9] found that, in mobile devices with small screens, the impact of resolution switches on the QoE is rather low. A survey on the QoE of adaptive video streaming was conducted in [1].

Apart from application layer QoE factors, also several papers focused on estimating the QoE from network layer measurements. In [10], authors introduced QoE Doctor, a tool to measure and analyze mobile app QoE, based on active measurements at the network and the application layers.

[11] gathered passive in-network measurements and applied machine learning methods to find correlations between QoS and QoE of mobile video applications. The authors of [12] used a decision tree based on network statistics and flow records to predict the termination of the video session by the user. [13] gathered stalling information, average video quality, and quality variations by applying random forests that take network features, such as round-trip time and packet loss, into account. The authors of [14] used different machine learning methods to classify the QoE from network parameters.

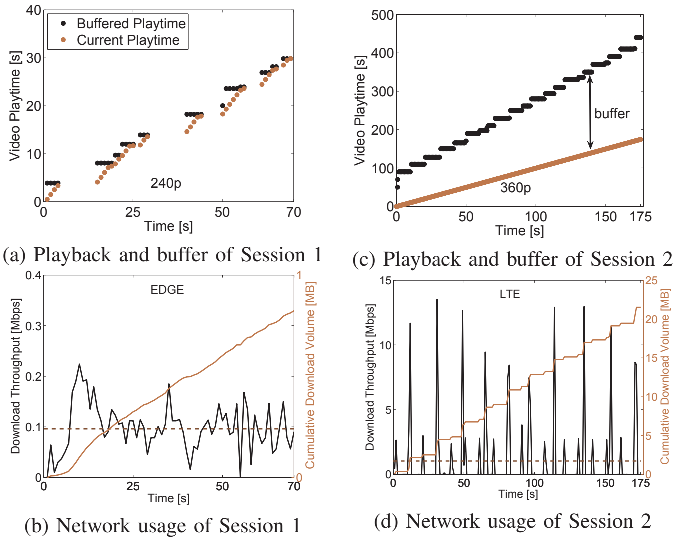
III. Study Description

A. *YoMoApp*

YoMoApp (YouTube Monitoring App) is based on an Android WebView browser element, in which the mobile YouTube website is loaded. The video playback on the mobile YouTube website is integrated via an HTML5 video element and uses DASH technology. With the Android app, JavaScript functions are injected to the website, similar to the browser plugin of YoMo [15]. These functions detect the video element in the DOM tree and perform the monitoring of the playback. Therefore, event listeners are added to the HTML5 video element to monitor all changes of the player state, and the height and width of the video element, which correspond to the video resolution. Periodically every second, the current playback time and buffered playtime are polled. Additionally, the YouTube ID, video title, video duration, and statistics about precedent advertisement clips are collected. The data is sent to the Android app for postprocessing and logging. Due to inconsistencies and errors, such as missing or incorrect values, which may arise from the usage of JavaScript, the postprocessing is required to ensure a high consistency of the resulting streaming logs.

The previous version of YoMoApp [16] has been improved to also monitor network and context parameters in the native Android part of YoMoApp. The network usage, i.e., the total amount of uploaded and downloaded data, is logged periodically for both mobile and WiFi networks. The app also logs changes of operator, cell ID, signal strength, RAT, and GPS position. Moreover, it retrieves several device characteristics and monitors their changes. These include screen size, screen orientation, player size, player mode (normal/full screen), and volume. The monitored data are stored in separate log files for each video session. If the video has ended or was aborted after a minimum session length of 20 s, the user is asked a single question to rate the QoE of the streaming session on a continuous MOS scale ranging from 1 (bad) to 5 (excellent). However, the user is not required to submit a rating, but he can also close the rating dialogue. All log files are locally cached on the device and they are transmitted to an external database when the app is closed, at fixed time intervals, or if triggered manually by the user.

To encourage the usage of the app, several aggregate statistics about each streaming session as well as a visualization are available to the user. Moreover, a map view including all subjective ratings is included. The overlay heat map shows how each network operator performs in terms of subjective QoE ratings, and can therefore be used to benchmark operators. Finally, for researchers using YoMoApp, the log files of the streaming sessions can be accessed on the YoMoApp web portal[1] for further evaluations. Thus, YoMoApp is a valuable tool to accurately monitor application and network layer streaming parameters and subjective QoE ratings, and use the monitored data to study the QoE of HAS.



(a) Playback and buffer of Session 1

(c) Playback and buffer of Session 2

(b) Network usage of Session 1

(d) Network usage of Session 2

Fig. 1. Illustration of some of the monitored parameters for two exemplary video streaming sessions on application and network layer.

Figure 1 shows an illustration of the streaming behavior on application layer and network layer for two sessions. In Figure 1a, the buffered playtime (black) and the current playtime (orange) of Session 1 are depicted. It can be seen that a video of around 30 s length is watched. The video is played back in 240p and the playback suffers from multiple stalling events, which occur when the buffer runs empty, i.e., the buffered playtime equals the current playtime. Figure 1b shows the corresponding average download throughput (black, left axis) and the cumulative download volume (orange, right axis). It can be seen that data are almost constantly downloaded. The detected RAT is EDGE, which is responsible for the session's low average throughput of 0.1 Mbps, which is indicated by the dashed line. Figure 1c shows that, in Session 2, a video is watched for 175 s in 360p before the playback is aborted. The buffered playtime and also the resulting buffer steadily increase, such that no stalling occurs. At the time of abortion the buffer contains 265 s of additional playtime. The corresponding network log is visualized in Figure 1d. LTE was the used RAT, which resulted in a high average throughput of 1.0 Mbps. Thereby, segments are downloaded with a high throughput of up to 13.5 Mbps, and the requesting of new segments is periodically paused for several seconds. This download behavior results in the stepwise increase of the cumulative download volume in this figure, and also of the buffered playtime in Figure 1c.

B. *Field Study*

The field study of the YoMoApp application resulted in 1266 streaming sessions generated by 196 different users in the period from July 16, 2014 to November 1, 2016. The participants were distributed all over the world with the

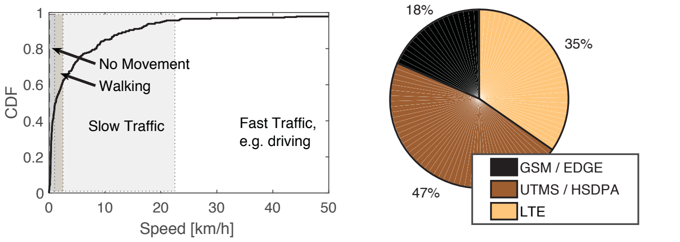
[1] http://yomoapp.de/dashboard

top 5 countries Germany: 28 %, US: 12.5 %, India: 10.9 %, Pakistan: 2.7 %, and Vietnam: 2.2 %. All participants used their own smartphone or tablet, and WiFi or cellular ISPs to stream videos using YoMoApp. Exactly 5 participants used a tablet. All other participants had a smartphone. 51 % of participants used Android 6.0, followed by Android 5.1 with 14 %. According to statistics of the Google Play Store, users were 39 % German-speaking and 37 % English-speaking. In terms of the display size, the participants used very different devices. 28 % of users had a resolution of 1920x1080 pixels at 480 or 560 dpi, followed by 13 % with a significantly lower resolution of 976x600 pixels at only 160 dpi.

Before the evaluation, the log files were filtered and preprocessed. Invalid logs were removed and logs without video statistics were sorted out. After the filtering, 674 sessions could be used for evaluation purposes in the results section. Compared to the study performed in [17], which also examined YoMoApp data, current evaluations also analyze network layer statistics and subjective ratings. The most important difference to the aforementioned paper is that the number of users and the analyzed sessions available for the evaluation have increased significantly. Figure 2a shows the average speed within a video session in km/h, which was obtained from the GPS locations. On average users moved forward at a speed of 6 km/h, with about 48 % of users not moving at all, i.e. the speed was slower than 1 km/h during the video session. About 14 % of the users were *walking* with a speed of less than 2.5 km/h. Furthermore, *slow traffic* is considered as 2.5 km/h to 20 km/h, and *fast traffic*, i.e., probably on a train or even driving, as more than 20 km/h. 33 % of the sessions can be counted as slow traffic. Around 5 % of users drove faster than 20 km/h. The maximum detected speed was at 135.7 km/h. As depicted in Figure 2b, 35 % of the users used LTE, 47 % were connected with WCDMA (e.g., UMTS or HSDPA), and the rest used GSM or EDGE. The analysis of the field study participants, device characteristics, mobility, and network access shows that the collected data set covers a huge range of different streaming sessions. In the following, the technical characteristics of the sessions and the perceived quality are investigated.

IV. Results

Figure 3 depicts temporal statistics of the QoE factors for all 674 evaluated mobile YouTube streaming sessions. The CDFs for initial delay (brown), total stalling time (yellow), and playback time (black) are shown in Figure 3a. The mean of the played back time of a single video is 78.6 s. The playback times ranged from around 5 s up to around 480 s. About 46% of the videos have a playback time greater than 30 s. Thus, future QoE assessments should especially consider longer video clips also for mobile devices. The average length of the initial delay is 2.6 s, with a maximum of 11.9 s. About 70% of the streaming sessions show an initial delay less than 3 s, suggesting that initial waiting times for mobile YouTube have a negligible QoE impact [4]. This notion is also supported by the total stalling time, which does not include the stalling during the initial delay. The average total stalling time is 0.93 s and the maximum total stalling time 12.68 s. Around 65% of the sessions have a smooth playback without stalling. The CDF of the number of stalling events is depicted in black in Figure 3b. The maximum number of monitored stalling events is 9, while the average number of stalling events is 0.58. This confirms the findings of [5] that adaptive video streaming is mostly able to avoid stalling in mobile sessions.

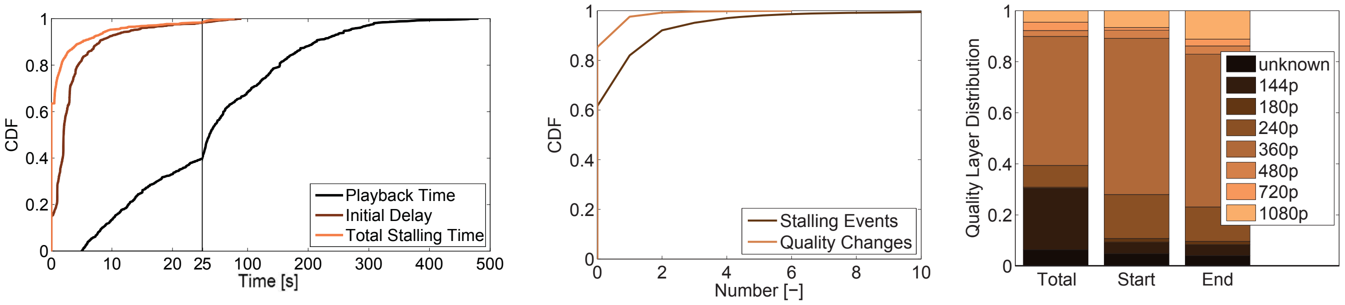


(a) Average speed within a session

(b) Distribution of video sessions by radio access technology (RAT)

Fig. 2. Information about the speed and access technology of the users.

The CDFs of the number of quality changes is shown in Figure 3b in brown. In the context of YouTube, the switching between two different quality layers is implemented as switching between two different video resolutions. Although stalling was avoided in most mobile video sessions, also nearly 87% of the streaming sessions showed no quality changes, hinting at a rather conservative adaptation logic on the part of YouTube. This means, YouTube cautiously chooses an appropriate start quality for the current estimation of the network conditions, such that quality switches and stalling can be avoided to the greatest extend. However, this might result in the selection of a lower video quality than supported by the network and underutilized bandwidth. In Figure 3c, distribution of the playout time of the different video qualities (time on layer) is illustrated. Additionally, the distributions of the quality played out at the start and the end of a streaming session are shown. As depicted in the legend, all qualities were used. For some videos the video resolutions could not be determined by YoMoApp (unknown). The dominating resolution for the



(a) Time-related streaming parameters

(b) Event-related streaming parameters

(c) Played out video quality

Fig. 3. Monitoring of streaming during the field study: (a) distribution of playback time, initial delay and total stalling time; (b) distribution of stallings and quality changes; (c) distribution of video quality layers.

time on layer is with over 60% 360p, followed by 240p and 144p. Further, HD content (720p or 1080p) is only streamed in ca. 8% of the sessions. Again the start qualities support the conservative behavior of YouTube as mainly 240p and 360p is streamed from the start. The end quality distribution approximately resembles the distribution of the time on layer, and argues for an improvement from the low start qualities if the network conditions permit.

A. *Correlations Between Monitored Parameters*

Table I investigates the correlation between the monitored parameters. The cells of the table show the Spearman rank order correlation coefficient between the two parameters in the respective row and column. Parameter A-G cover application layer parameters, namely, initial delay (A), stalling ratio (B), which is the total stalling time divided by the total playback time, number of stalling events (C), number of quality changes (D), start quality (E), end quality (F), and weighted time on layer (G), which is the time-weighted mean of all played out quality levels. Moreover, the network parameters average throughput (H), maximum throughput (I), and flow volume (J) are considered. Finally, the perceived quality is measured in terms of user engagement (K), which is the ratio of watched playtime and video duration, and the actual video quality subjective rating given by the user (L). As the participation in YoMoApp is voluntary, the collected ratings were not filtered a priori, but will be analyzed in the following sections.

It can be seen that only few parameters show non-negligible correlations. These high correlations can be observed in natural clusters, e.g., stalling ratio (B) has a very high correlation of 0.93 to number of stalling events (C). Similarly, the quality layer parameters have high or even very high correlations (E, F, G). The last cluster of very high correlation larger than 0.91 can be found for the network parameters (H, I, J). The correlations of other parameter combinations are little if any. Interestingly, not a single parameter has a correlation of 0.30 or higher to the subjective quality ratings of the users (L). Only the directions of correlation to the user ratings are as expected, such as negative correlations of initial delay (A) and stalling (B, C), and positive correlations of video quality layers (E, F, G). Also for user engagement (K), which is widely considered as a QoE metric, only low correlations can be observed. In contrast to subjective ratings, some correlations with respect to user engagement are even unintuitive, e.g., the small positive correlations of stalling parameters (B, C). The very low correlation of 0.01 between user engagement and subjective ratings suggests that both QoE metrics should be considered uncorrelated. This indicates that user engagement rather measures the interest or motivation of the users to watch a certain content, while the subjective rating is the only QoE indicator in this unsupervised field study.

B. *Clustering Streaming Sessions*

To analyze the perceived quality, in a first step, similar streaming sessions have to be identified. Therefore, the sessions will be characterized by common attributes, i.e., a feature vector. Each feature vector consists of several perceivable application layer metrics for a single session, namely, number of stalling events, stalling ratio, and weighted time on layer. The number of quality changes is not included in the vector, because quality changes occurred very seldom, and therefore, have no influence on the results. Instead, user engagement was added to the feature vector as it could indicate the motivation and interest of the users with the video content. Each metric was normalized to the unit interval.

These 674 feature vectors were clustered with the well known DBSCAN algorithm [18], with neighborhood distance $\epsilon = 0.1$ and minimum number of cluster points 15. All in all, nine clusters emerged from this process. As the set of outliers contained a huge number of sessions (216), it was again clustered with neighborhood distance $\epsilon = 0.3$ and minimum number of cluster points 15. As a result, two additional clusters and a smaller set of outliers showed up. Altogether, ten clusters were generated, which are listed in Table II. Each row shows the generated cluster with the corresponding metrics. Note the split between clusters 1-8 and 9-11, as it represents the two steps of the clustering process. Cluster 11 represents the remaining set of outliers.

C. *Subjectively Perceived Quality and User Engagement*

In this section, the aforementioned session clusters are analyzed with respect to the subjectively perceived quality in terms of mean opinion score (MOS) and user engagement. Note that the MOS was only analyzed for clusters with 10 or more ratings. Cluster 1 contains the sessions with HD content (1080p), no stalling events, and a very short mean user engagement (4.60%). Of the 18 clustered sessions, 10 sessions were rated, which resulted in an average MOS of 3.85. The average MOS seems contradictory compared to the video quality and the stalling ratio. However, the mean user engagement indicates that the users had no intent to actually watch the video, which might have negatively influenced the

TABLE I. Correlations between monitored parameters, A=Initial Delay, B=Stalling Ratio, C=Num. of Stalling Events, D=Num. of Quality Changes, E=Start Quality, F=End Quality, G=Weighted Time on Layer, H=Avg. Throughput, I=Max. Throughput, J=Flow Volume, K=User Engagement, L=Subjective Quality Rating

Parameter	A	B	C	D	E	F	G	H	I	J	K	L
A	1.00	0.01	0.02	0.17	-0.11	-0.10	-0.11	0.00	0.03	0.06	-0.14	-0.17
B	0.01	1.00	0.93	0.09	-0.08	-0.10	-0.11	0.20	0.19	0.21	0.22	-0.10
C	0.02	0.93	1.00	0.10	-0.15	-0.17	-0.18	0.17	0.18	0.21	0.28	-0.15
D	0.17	0.09	0.10	1.00	-0.10	0.19	0.12	-0.27	-0.25	-0.23	0.11	-0.21
E	-0.11	-0.08	-0.15	-0.10	1.00	0.72	0.73	-0.06	-0.07	-0.09	-0.04	0.29
F	-0.10	-0.10	-0.17	0.19	0.72	1.00	0.93	-0.20	-0.19	-0.21	-0.08	0.16
G	-0.11	-0.11	-0.18	0.12	0.73	0.93	1.00	-0.19	-0.20	-0.21	-0.10	0.18
H	0.00	0.20	0.17	-0.27	-0.06	-0.20	-0.19	1.00	0.92	0.91	0.08	0.06
I	0.03	0.19	0.18	-0.25	-0.07	-0.19	-0.20	0.92	1.00	0.95	0.19	-0.15
J	0.06	0.21	0.21	-0.23	-0.09	-0.21	-0.21	0.91	0.95	1.00	0.27	-0.18
K	-0.14	0.22	0.28	0.11	-0.04	-0.08	-0.10	0.08	0.19	0.27	1.00	0.01
L	-0.17	-0.10	-0.15	-0.21	0.29	0.16	0.18	0.06	-0.15	-0.18	0.01	1.00

TABLE II. CLUSTERING RECORDED FOR THE FEATURE VECTORS

Cluster	Sessions	Sessions with MOS	Stalling Events	Mean Stalling Ratio	Quality	Mean User Engagement	Mean Opinion Score
1	18	10	0	0%	1080p	4.60%	3.85
2	84	38	0	0%	360p	97.23%	4.44
3	234	71	0	0%	360p	11.77%	4.18
4	38	2	0	0%	240p	5.30%	N/A
5	48	0	0	0%	108-180p	6.83%	N/A
6	18	11	1	0.5%	360p	98.77%	4.22
7	21	11	1	29.4%	360p	7.96%	4.47
8	20	6	2	22.2%	360p	5.50%	N/A
9	21	13	0.24	0.07%	108p-240p	96.76%	3.89
10	36	12	1	2.9%	240p-360p	14.21%	4.36
11	159	88	1.9	4.45%	108p-1080p	49.34%	3.77

quality rating. Moreover, in case of video abortion, they might have wanted to quickly proceed to a different video, and thus, were not much concerned with spending too much time and giving a reliable quality rating. Clusters 2 and 3 consist of the sessions with 360p resolution and no stalling events. While Cluster 2 has a very high mean user engagement (97.23%), sessions in Cluster 3 were aborted at a mean of 11.77% of the video length. Also, Cluster 3 contains almost three times as many sessions (234) as Cluster 2 (84). The MOS for the cluster with the high user engagement is around 4.44, while the MOS for Cluster 3 is only slightly worse (4.18).

Clusters 4 and 5 are very similar in that there is no stalling, a poor video quality, and a low mean user engagement. The only difference is that the played out resolutions in Cluster 5 (108p - 180p) are even worse than in Cluster 4 (240p). The low user engagement and the missing quality ratings indicate that the streaming or content quality in these clusters is especially low, such that users aborted early and even did not want to rate. Note that these two clusters account for 12.76% of the monitored sessions, thus, the poor streaming performance or poor content cannot be neglected. Also Clusters 6 and 7 are about the same size, and both have 360p resolution and exactly one stalling event. While Cluster 6 accommodates a low mean stalling ratio (0.5%) and a high mean user engagement (98.77%), the mean stalling ratio of Cluster 7 is 29.4%. It could be argued that the higher mean stalling ratio causes the low mean user engagement of 7.96%. However, when rated subjectively by the users, Cluster 7 exhibits a better MOS than the cluster with the low mean stalling ratio and the high mean user engagement. The last cluster generated by the first clustering step is portrayed by two stalling events, a high mean stalling ratio (22.2%), a quality level of 360p, and a short mean user engagement.

The original set of outlier sessions, which is not listed in Table II, was clustered again and could be split into three more clusters. Clusters 9 and 10 could be formed because of the higher neighborhood distance, and Cluster 11 comprises the remaining set of outlier sessions. All three clusters contain sessions with and without stalling events. Still, the mean stalling ratio is relatively low for all of them. The main differences are the mean user engagement and the played out video quality. Cluster 9 offers a video quality between 108p and 240p and has a mean user engagement of 96.76%, while Cluster 10 has a higher video quality between 240p and 360p, but a lower mean user engagement of 14.21%. It can be seen from Cluster 9 that many users do not abort their streaming sessions although they face a poor video quality and some stalling. The reason might be that users know about their poor network access or are really interested in the video content, and thus, tolerate the bad streaming quality. Cluster 11 contains all remaining sessions, which were not similar enough to the above presented clusters. All kinds of resolutions and stalling occur. The mean user engagement of the remaining sessions is 49.34% and the MOS is 3.77.

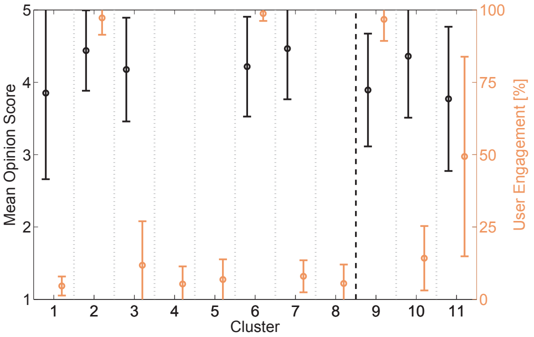


Fig. 4. MOS, mean user engagement, and their standard deviations of clusters.

Figure 4 summarizes the above findings about subjectively perceived quality and user engagement. The figure shows the different clusters on the x-axis, and their MOS and standard deviation in black on the left y-axis. Note that the MOS of Clusters 4, 5, 8 with no or insufficient ratings is omitted. The right y-axis shows the mean user engagement and standard deviation in orange. It can be seen that the MOS of the clusters are in a similar range, and the obtained ratings for each cluster contain a significant variance, which is indicated by the large standard deviations. Nevertheless, the pairwise comparison results of a one-way analysis of variance (ANOVA) confirms that only the means of Cluster 2 and the outlier Cluster 11 are significantly different. The exclusion of Cluster 11 in the ANOVA shows that there is no significant difference among the means of the ratings of the ten clusters with a p-value of 0.12. This high amount of variance can be attributed to the unsupervised nature of the field study. The figure shows that the clusters are nicely separated into either a very high user engagement, i.e., videos are entirely watched, or very low user engagement, i.e., videos are aborted early. As mentioned above, user engagement is not correlated to the streaming or network parameters, and therefore, could be influenced by the users' motivation or interest in the content. A lack of these eventually could have forced users to quickly abort the video to find better content, thereby also quickly performing the rating, which could result in unreliable ratings.

When analyzing only the clusters with high user engagement 2, 6, and 9, the MOS values reflect the streaming quality. Cluster 2 with 360p and no stalling has a MOS of 4.44, Cluster 6 with 360p and 1 stalling has a MOS of 4.22, and Cluster 9 with lower resolution and small stalling has MOS of 3.89. The one-way ANOVA has a p-value of 0.03 and pairwise comparison confirms that the means of Cluster 2 and Cluster 9 are significantly different. This confirms findings on the impact of stalling [2] and time on quality layer [7], [8], and suggests that an unsupervised field study can be used for QoE research, when focusing on users with a high engagement.

V. Conclusion

This work is the first to conduct an unsupervised field study on the QoE of mobile adaptive video streaming on YouTube. An Android-based monitoring app, YoMoApp, was implemented and published on the Google Play Store. During the development more than 1250 streaming sessions could be monitored, thereby logging network layer parameters, application layer streaming parameters, device characteristics, and subjective ratings. After the filtering of incompletely monitored sessions, 674 sessions were evaluated to gain insights into the streaming context and streaming behavior of mobile YouTube.

The correlation between the monitored parameters were investigated, which showed that none had a high correlation to user engagement or subjective quality rating. Also user engagement, which is widely considered a QoE metric, was uncorrelated to the subjective rating. This could be due to the fact that it rather measures the interest or motivation of the users to watch a certain content. When clustering all streaming sessions according to perceived streaming parameters and user engagement, ten clusters could be identified. The characteristics of the clusters and the remaining set of outlier sessions were analyzed. User ratings showed a very high variance within the ten clusters, such that their MOS values were not significantly different. The high variance might be attributed to the user engagement, which nicely separated the clusters into very high and very low user engagement. In case of low motivation or interest to watch the video, users could have quickly aborted to find better content and quickly performed the rating, which could result in unreliable feedback.

After excluding the clusters with low mean user engagement, the ratings of the remaining clusters reflected the streaming quality. Thus, QoE research could rely on an unsupervised field study if the behavior and engagement of the users with the test is monitored. In future work, additional means will be added to YoMoApp to improve the user engagement. In particular, users will be asked before the start of the video streaming session if they want to rate the subjective quality afterwards. Moreover, the rating dialogue will be extended to feature additional questions, which can be used for consistency checks, and the rating time will be monitored.

Acknowledgment

This work was partly funded in the framework of the EU ICT projects INPUT (H2020-2014-ICT-644672) and MONROE (H2020-2014-ICT-644399, through open call project Mobi-QoE). The authors alone are responsible for the content.

References

[1] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2015.

[2] T. Hoßfeld, R. Schatz, M. Seufert, M. Hirth, T. Zinner, and P. Tran-Gia, "Quantification of YouTube QoE via Crowdsourcing," in *International Workshop on Multimedia Quality of Experience - Modeling, Evaluation, and Directions (MQoE)*, Dana Point, CA, USA, 2011.

[3] R. K. P. Mok, E. W. W. Chan, X. Luo, and R. K. C. Chan, "Inferring the QoE of HTTP Video Streaming from User-Viewing Activities," in *1st ACM SIGCOMM Workshop on Measurements Up the STack (W-MUST)*, Toronto, Canada, 2011.

[4] T. Hoßfeld, S. Egger, R. Schatz, M. Fiedler, K. Masuch, and C. Lorentzen, "Initial Delay vs. Interruptions: Between the Devil and the Deep Blue Sea," in *4th International Workshop on Quality of Multimedia Experience (QoMEX)*, Yarra Valley, Australia, 2012.

[5] J. Yao, S. S. Kanhere, I. Hossain, and M. Hassan, "Empirical Evaluation of HTTP Adaptive Streaming Under Vehicular Mobility," in *10th International IFIP TC 6 Networking Conference: Networking 2011*, Valencia, Spain, 2011.

[6] B. Lewcio, B. Belmudez, A. Mehmood, M. Wältermann, and S. Möller, "Video Quality in Next Generation Mobile Networks – Perception of Time-varying Transmission," in *2011 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, Naples, FL, USA, 2011.

[7] T. Hoßfeld, M. Seufert, C. Sieber, and T. Zinner, "Assessing Effect Sizes of Influence Factors Towards a QoE Model for HTTP Adaptive Streaming," in *6th International Workshop on Quality of Multimedia Experience (QoMEX)*, Singapore, 2014.

[8] M. Seufert, T. Hoßfeld, and C. Sieber, "Impact of Intermediate Layer on Quality of Experience of HTTP Adaptive Streaming," in *11th International Conference on Network and Service Management (CNSM)*, Barcelona, Spain, 2015.

[9] P. Casas, R. Schatz, F. Wamser, M. Seufert, and R. Irmer, "Exploring QoE in Cellular Networks: How Much Bandwidth do you Need for Popular Smartphone Apps?" in *5th ACM SIGCOMM Workshop on All Things Cellular: Operations, Applications and Challenges (ATC)*, London, UK, 2015.

[10] Q. A. Chen, H. Luo, S. Rosen, Z. M. Mao, K. Iyer, J. Hui, K. Sontineni, and K. Lau, "QoE Doctor: Diagnosing Mobile App QoE with Automated UI Control and Cross-layer Analysis," in *Internet Measurement Conference (IMC)*, Melbourne, Australia, 2014.

[11] V. Aggarwal, E. Halepovic, J. Pang, S. Venkataraman, and H. Yan, "Prometheus: Toward Quality-of-Experience Estimation for Mobile Apps from Passive Network Measurements," in *15th Workshop on Mobile Computing Systems and Applications (HotMobile)*, Santa Barbara, CA, USA, 2014.

[12] M. Z. Shafiq, J. Erman, L. Ji, A. X. Liu, J. Pang, and J. Wang, "Understanding the Impact of Network Dynamics on Mobile Video User Engagement," in *ACM SIGMETRICS*, Austin, TX, USA, 2014.

[13] G. Dimopoulos, I. Leontiadis, P. Barlet-Ros, and K. Papagiannaki, "Measuring Video QoE from Encrypted Traffic," in *Internet Measurement Conference (IMC)*, Santa Monica, CA, USA, 2016, pp. 513–526.

[14] I. Orsolic, D. Pevec, M. Suznjevic, and L. Skorin-Kapov, "YouTube QoE Estimation Based on the Analysis of Encrypted Network Traffic Using Machine Learning," in *5th IEEE International Workshop on Quality of Experience for Multimedia Communications (QoEMC)*, Washington, DC, USA, 2016.

[15] B. Staehle, M. Hirth, R. Pries, F. Wamser, and D. Staehle, "YoMo: A YouTube Application Comfort Monitoring Tool," in *1st Workshop of Quality of Experience for Multimedia Content Sharing (QoEMCS)*, Tampere, Finland, 2010.

[16] F. Wamser, M. Seufert, P. Casas, R. Irmer, P. Tran-Gia, and R. Schatz, "YoMoApp: a Tool for Analyzing QoE of YouTube HTTP Adaptive Streaming in Mobile Networks," in *European Conference on Networks and Communications (EuCNC)*, Paris, France, 2015.

[17] M. Seufert, P. Casas, F. Wamser, N. Wehner, R. Schatz, and P. Tran-Gia, "Application-Layer Monitoring of QoE Parameters for Mobile YouTube Video Streaming in the Field," in *IEEE 6th International Conference on Communications and Electronics (ICCE)*, Ha Long, Vietnam, 2016.

[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *2nd International Conference on Knowledge Discovery and Data Mining (KDD)*, Portland, OR, USA, 1996.