Quantification of YouTube QoE via crowdsourcing

Tobias Hoßfeld, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, Raimund Schatz

Angaben zur Veröffentlichung / Publication details:

Hoßfeld, Tobias, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, and Raimund Schatz. 2011. "Quantification of YouTube QoE via crowdsourcing." In *IEEE International Symposium on Multimedia, 5-7 December 2011, Dana Point, CA, USA*, 494–99. Piscataway, NJ: IEEE. https://doi.org/10.1109/ism.2011.87.

Quantification of YouTube QoE via Crowdsourcing

Tobias Hoßfeld, Michael Seufert,
Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia
University of Würzburg, Institute of Computer Science
D-97074 Würzburg, Germany
tobias.hossfeld@uni-wuerzburg.de

Raimund Schatz
Telecommunications Research
Center Vienna - ftw
A-1220 Vienna, Austria
schatz@ftw.at

***Abstract*—This paper addresses the challenge of assessing and modeling Quality of Experience (QoE) for online video services that are based on TCP-streaming. We present a dedicated QoE model for YouTube that takes into account the key influence factors (such as stalling events caused by network bottlenecks) that shape quality perception of this service. As second contribution, we propose a generic subjective QoE assessment methodology for multimedia applications (like online video) that is based on crowdsourcing - a highly cost-efficient, fast and flexible way of conducting user experiments. We demonstrate how our approach successfully leverages the inherent strengths of crowdsourcing while addressing critical aspects such as the reliability of the experimental data obtained. Our results suggest that, crowdsourcing is a highly effective QoE assessment method not only for online video, but also for a wide range of other current and future Internet applications.**

I. Introduction

Video streaming dominates global Internet traffic and is expected to account for 57 % of all consumer Internet traffic in 2014 generating over 23 exabytes per month [1]. The most prominent video streaming portal is Youtube which serves more than two billion videos daily. However in practice, many users face volatile performance of the service, e.g. due to bad network conditions or congested media streaming servers. Such adverse conditions are the main causes for bad online video QoE.

In the domain of video streaming, traditional UDP-based services like IPTV or Real Media streaming typically do not guarantee packet delivery. Thus, congestion in the network or at the multimedia servers leads to lost packets causing visual artifacts, jerky motion or jumps in the stream, forms of degraded media quality which have been extensively studied in previous video quality research. In contrast, delivery of YouTube video to the end user is realized as progressive download using TCP as transport protocol. The usage of TCP guarantees the delivery of unaltered video content since the protocol itself cares for the retransmissions of corrupted or lost packets. Further, it adapts the transport rate to network congestion, effectively minimizing packet loss. However, if available bandwidth is lower than the video bit rate, video transmission becomes too slow, gradually emptying the playback buffer until underrun occurs. If rebuffering happens, the user notices interrupted video playback, commonly referred to as *stalling*. In this respect, YouTube QoE is different from traditional UDP-based video streaming, since with TCP only the video playback itself is disturbed while the transmitted audiovisual content remains unaltered.

Due to the current lack of QoE models that identify key influence factors for YouTube (e.g. demographics of users, Internet application usage habits, content types, network impairments) and explicitly address stalling effects in the context of TCP-based online video, subjective user studies need to be performed. Such studies are typically carried out by a test panel of real users in a laboratory environment. While many and possibly even diverging views on the quality of the media consumption can be taken into account – entailing accurate results and a good understanding of the QoE and its relationship with QoS – lab-based user studies can be time-consuming and costly, since the tests have to be conducted by a large number of users to obtain statistically relevant results. Because of the costs and time demands posed by laboratory tests, only a limited set of influence factors can be tested per test session. In related work [2], the correlation between network QoS in terms of delay, packet loss and throughput, application QoS in term of stalling, and QoE was evaluated for HTTP video streaming in a lab test. However, only a single video clip was used and for each test condition only ten subjects rated their experienced quality. Therefore, [2] is quite limited with respect to reliability and QoE influence factors, e.g. video content type, resolution, etc., under investigation.

For deriving a QoE model, crowdsourcing therefore seems to be an appropriate alternative approach. Crowdsourcing means to outsource a task (like video quality testing) to a large, anonymous crowd of users in the form of an open call. Crowdsourcing platforms in the Internet, like Amazon Mechanical Turk or Microworkers, offer access to a large number of internationally widespread users in the Internet and distribute the work submitted by an employer among the users. The work is typically organized at a finer granularity and large jobs (like a QoE test campaign) are split into cheap (micro-)tasks that can be rapidly performed by the crowd.

With crowdsourcing, subjective user studies can be efficiently conducted at low costs with adequate user numbers for obtaining statistically significant QoE scores [3]. In addition,

This work was conducted within the Internet Research Center (IRC) at the University of Würzburg. The work has been supported by COST TMA Action IC0703, COST QUALINET Action IC1003, and the project G-Lab, funded by the German Ministry of Educations and Research (Förderkennzeichen 01 BK 0800, G-Lab). The authors alone are responsible for the content of the paper.

the desktop-PC based setting of crowdsourcing provides a highly realistic context for usage scenarios like online video consumption. However, reliability of results cannot be assumed because of the anonymity and remoteness of participants (cf. [4] and references therein): some subjects may submit incorrect results in order to maximize their income by completing as many tasks as possible; others just may not work correctly due to lack of supervision. Therefore, it is necessary to develop an appropriate methodology that addresses these issues and ensures consistent behavior of the test subjects throughout a test session and thus obtain reliable QoE results.

The contribution of this paper is twofold. Firstly, we provide a YouTube QoE model taking into account stalling as key influence factor based on subjective user studies. Second, we develop a generic subjective QoE testing methodology for Internet applications like YouTube based on crowdsourcing for efficiently obtaining highly valid and reliable results.

The remainder of this paper is structured as follows. Section II gives a background on crowdsourcing and the platform used in this work. The subjective test methodology is presented in Section III aiming at an appropriate test design to detect unreliable user ratings. In Section IV, the test results are statistically analyzed. In particular, we apply different results filtering levels and assess the reliability of the data set. The YouTube QoE is then quantified in Section V for a realistic impairment scenario, where the YouTube video is streamed over a bottleneck link. Finally, Section VI concludes this work, discussing the potential of the crowdsourcing method.

II. Crowdsourcing and Microworkers Platform

Crowdsourcing can be understood as a further development of the outsourcing principle by changing the granularity of work [5] and the size of the outsourced tasks, as well as the administrative overhead. A microtask can be accomplished within a few minutes to a few hours and thus does not need a long term employment. Further, it is irrelevant to the employer who actually accomplishes the task and usually the task has to be repeated several times. The repetitive tasks are combined in a *campaign*, which the employer submits to the crowdsourcing platform. The workforce in the crowdsourcing approach is not a designated worker but a large, anonymous human crowd of workers. The *crowdsourcing platform* acts as a mediator between the employer and the crowd.

In this work, we use the Microworkers[1] crowdsourcing platform, since Microworkers allows to conduct online user surveys like our YouTube QoE tests. Microworkers supports workers internationally in a controlled fashion, resulting in a realistic user diversity well-suited for QoE assessment. The Microworkers platform had about 80,000 registered users end of 2010 (see [6] providing also a detailed analysis of the platform and its users).

In general, every crowdsourcing task suffers from bad quality results. Therefore, different task design strategies have been proposed to improve the quality of the output. Using the example of an image labeling task, Huang et al. [7] demonstrated that the results quality of a task can be influenced by its design. They varied the payment per task, the number of requested tags per image, the number of images per task and the tasks per campaign in order to maximize the number of unique labels or the number of labels corresponding with their gold standard.

However, even if the task is designed effectively, workers might still submit incorrect work. Thus, tasks can be equipped with verification questions [8] to increase the quality, the workers input can be rechecked by others as e.g. in [9], [10], or iterative approaches can be used [11], [12]. If the workers input is not wrong but only biased, there also exist methods to eliminate these biases [13]. Based on these insights and suggestions, we developed a new, improved QoE assessment method for crowdsourcing.

III. Subjective Crowd Test Methodology

The test methodology developed throughout this work allows experimenters to conduct subjective user tests about the user perceived quality of Internet applications like YouTube by means of crowdsourcing and to evaluate the impact of network impairments on QoE. For the necessary quality assurance of the QoE test results themselves including the identification of unreliable user ratings, we apply different task design methods (cf. Section III-A), before the subjective users tests are conducted by the crowd (cf. Section III-B). Different user study campaigns are designed (cf. Section III-C) according to the influence factors under investigation.

A. Task Design Based Methods

The task design methods described in the following paragraphs can be used for different crowdsourcing tasks. Nonetheless, we describe their applicability in the context of evaluating the QoE for YouTube videostreaming.

Gold Standard Data: The most common mechanism to detect unreliable workers and to estimate the quality of the results is to use questions whereof the correct results are already known. These gold standard questions are interspersed among the normal tasks the worker has to process. After results submission by the worker, the answers are compared to gold standard data. If the worker did not process the gold standard questions correctly, the non-gold standard results should be assumed to be incorrect too.

Since for subjective quality testing personal opinions are asked for, the gold standard data approach has to be applied with care since user opinions must be allowed to diverge. Still, in our tests we included videos without any stalling and additionally asked participants: “Did you notice any stops to the video you just watched?”. If a user then noticed stops, we disregarded his ratings for quantification of QoE. We additionally monitored the stalling events on application layer to exclude any unwanted stops, see Section III-B.

Consistency Tests: In this approach, the worker is asked the same question multiple times in a sightly different manner. For example, at the beginning of the survey the worker is

[1] http://www.microworkers.com

asked how often she visits the YouTube web page, at the end of the survey she is asked how often she watches videos on YouTube. The answers can slightly differ but should be lie within the same order of magnitude. Another example is to ask the user about his origin country in the beginning and about his origin continent at the end. The ratings of the participant are disregarded, if not all answers of the test questions are consistent.

Content Questions: After watching a video, the users were asked to answer simple questions about the video clip. For example, "Which sport was shown in the clip? A) Tennis. B) Soccer. C) Skiing." or "The scene was from the TV series... A) Star Trek Enterprise. B) Sex and the City. C) The Simpsons." Only correct answers allow the user's ratings to be considered in the QoE analysis.

Mixed Answers: This method is an extension to consistency tests to detect workers using fixed click schemes in surveys. Usually, the rating scales on surveys are always structured in the same way, e.g. from good to bad. Consequently, workers using fixed click scheme might bypass automated consistency tests, as always selecting the first or the middle answer results in a consistent survey. An easy way to avoid this is to vary the structure of the rating scales. For example the options of the first quality question "Did you notice any stops while the video was playing?" has the order "No", "Yes", whereas in the following question "Did you experience these stops as annoying?" the order is "Extremely","Fairly",..., "Not at all". Now, following a fixed clicking scheme results causes inconsistencies and identifies unreliable participants.

Application Usage Monitoring: Monitoring users during the tasks completion can also be used to detect cheating workers. The most common approach here is measuring the time the worker spends on the task. If the worker completes a task very quickly, this might indicate that she did the work sloppy.

In this work, we monitored browser events in order to measure the focus time, which is the time interval during which the browser focus is on the website belonging to the user test. In order to increase the number of valid results from crowdsourcing, we displayed a warning message if the worker did not watch more than 70 % of the video. The users could decide to watch the video again or to continue the test. When workers became aware of this control mechanism, the percentage of completely watched videos doubled and almost three times more workers could be considered reliable than without the system warning.

For the subjective crowd tests, we recommend to combine all above mentioned task designs, i.e. gold standard data, consistency checks, content questions, mixed questions and application monitoring.

B. Implementation and Execution of Experiments

The aim of the experiments is to quantify the impact of network impairments on QoE. For YouTube video streaming, network impairments result into related stalling patterns. As the video experience should be as similar as possible to a visit of the real YouTube website, the application should run on the users' default web browser. To this end, an instance of the YouTube Chromeless Player was embedded into dynamically generated web pages. With JavaScript commands the video stream can be paused, a feature we used to simulate stalling. YouTubes standard animated icon was used as visual indicator that the video is being buffered. In addition, the JavaScript API allows monitoring the player and the buffer status, i.e. to monitor stalling on application layer. In order to avoid additional stalling caused by the test users' Internet connection, the videos had to be downloaded completely to the browser cache before playing. This enables us to specify fixed unique stalling patterns which are evaluated by several users.

During the initial download of the videos, a personal data questionnaire was completed by the participant which also includes consistency questions from above. The user then sequentially viewed three different YouTube video clips with a predefined stalling pattern. After the streaming of the video, the user was asked to give his current personal satisfaction rating during the video streaming. In addition, we included gold standard, consistency, content and mixed questions to identify reliable subjective ratings. The workers were not aware of these checks and were not informed about the results of their reliability evaluation. Users had to rate the impact of stalling during video playback on a 5-point absolute category rating (ACR) scale [14] with the following values: (1) bad; (2) poor; (3) fair; (4) good; (5) excellent.

C. Design of Campaigns with Respect to Influence Factors

For deriving the impact of various influence factors, we conducted individual crowdsourcing campaigns in which only a single parameter is varied, while the others are kept constant. This strict separation helps for a proper QoE analysis and deriving adequate QoE models. In this work, we focus on the quantification of network impairments on YouTube QoE. Since YouTube videos are delivered via TCP, any network impairments appear as stalling to the end user.

For obtaining realistic stalling patterns we first studied the relationship between network QoS and stalling events. To this end, several YouTube videos were requested with a downlink bandwidth limitation of the used browser. On network layer, packet traces were captured, while on application layer, the YouTube player status (i.e. playing or stalling) was monitored by using the YouTube Javascript API. In case of a bottleneck, i.e. if the available bandwidth is lower than the video bandwidth, the video play back stalls several times. For example, we requested a 54 s video clip with an average video bit rate of 489 kbps or 61 kBps. We varied the bottleneck bandwidth b between 20 kBps and 65 kBps. As a result, we found that the stalling events occur periodically. For the example trailer, the number N of stallings can be approximated by $N(b) = \max\{-0.467 \cdot b + 27.616, 0\}$, while the total stalling time T follows as $T(b) = \max\{1\,237e^{2.323/x} - 1\,286, 0\}$. The average length L of a single stalling event follows as $L(b) = {}^{T(b)}/_{N(b)}$. We found that for our videos, a bandwidth of about 59 kBps was sufficient to play out the video without any interruptions,

TABLE I
PARAMETERIZATION OF THE SEVEN CROWDSOURCING CAMPAIGNS

id	number N of stallings	length L of stalling event
$C1$	$0, 1, 2, 3, 4, 5, 6$	4 s
$C2$	1	$2, 4, 8, 16, 32, 64$ s
$C3$	$0, 1, 2, 3, 4, 5, 6$	1 s
$C4$	$0, 1, 2, 3, 4, 5, 6$	2 s
$C5$	2	$1, 2, 4, 8, 16, 32$ s
$C6$	3	$1, 2, 4, 8, 16$ s
$C7$	$0, 1, 2, 3, 4, 5, 6$	3 s

since an initial buffering process prevents stalling in this case. Details can be found in the technical report [15].

As a result of this analysis, we parameterized our crowdsourcing campaigns $C1 - C7$ as outlined in Table I, varying either length or number of stalling events while keeping the other parameter constant.

IV. Statistical Analysis of Test Results

Throughout our measurement campaign, 1 349 users from 61 countries participated in the YouTube stalling test and rated the quality of 4 047 video transmissions suffering from stalling. Statistical analysis of the demographics of the users can be found in [15]. We first identify unreliable users and filter the data from the user studies accordingly. Then, we show that the (inter-rater and intra-rater) reliability of the filtered data is improved significantly.

A. Unreliable Users and Filtering of Data

The task design based methods as defined in Section III-A allow a three level filtering of the users. The first level identifies crowdsourcing users that gave wrong answers to content questions, that provided different answers to the same rephrased consistency questions, or that often selected the same option during the test. Thus, the first level applies consistency tests, content questions and mixed answers. The second level checks additionally whether participants who watched a video with stops noticed the stalling and vice versa, i.e., gold standard data is included in the test. The third level extends the previous filter level by additionally monitoring the application usage. All users are removed that did not watch all three videos completely.

Figure 1 shows the percentage of users passing the three filter levels for the different crowdsourcing campaigns $C1, \ldots, C7$ we performed. In each of the user study campaigns we only varied a single test condition (either the number of stallings or the duration of a single stalling event), while the remaining test conditions like video contents were kept equal. Level 0 refers to the unfiltered data from all users.

Interestingly, each filter technique reduces the number of valid crowdsourcing workers by approx. 25 % on average over all campaigns. This indicates that the consistency tests are quite useful for identifying spammers clicking random answers as well as video content questions and monitoring task specific parameters (like the focus time) for identifying sloppy workers who do not watch the video carefully enough. Due to our restrictive filtering, only about one fourth of the subjective ratings were finally considered for the analysis.

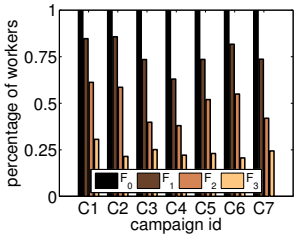

Fig. 1. Percentage of remaining participants per filter level F_i which are defined in Section IV-B

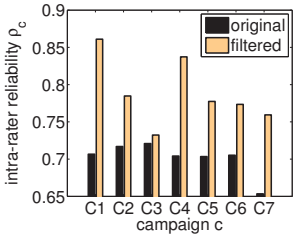

Fig. 2. Intra-rater reliability of filtered data (filter level F_3) compared to original data (filter level F_0)

B. Reliability of Filtered Data

We consider two different types of reliability of the user studies: intra-rater and inter-rater reliability. Firstly, *intra-rater reliability* determines to which extent the ratings of an individual user are consistent. In a measurement campaign C, an individual user u sequentially views three different YouTube video clips with a predefined stalling pattern x_i for $i \in \{1, 2, 3\}$ and rates the QoE accordingly with y_i. In each campaign, we only vary a single test condition (either the number of stalling pattern or the length of a single stalling event) and keep the others constant. Hence, we assume that worse stalling conditions $x_j > x_k$ will be reflected accordingly by the the user ratings with $y_j \leq y_k$. Therefore, we can apply the Spearman rank-order correlation coefficient $\rho_{C;u}(x_u, y_u)$ for ordinal data between the user ratings y_u and the varied stalling parameter x_u. Spearman rank correlation considers only that the items on the rating scale represent higher vs. lower values, but not necessarily of equal intervals. We define the intra-rater reliability ρ_C of a campaign C by averaging over all users $\mathcal{U}$, i.e. $\rho_c = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \rho_{C,u}$.

Secondly, *inter-rater reliability* denotes the degree of agreement among raters. For a campaign c, we define it as Spearman rank-order correlation coefficient κ_c between all user ratings y_C and the varied stalling parameter x_C for all user ratings in a campaign. It has to be noted that the applied filter levels are independent of the actual stalling conditions, hence, the above defined reliability metrics are valid.

As illustration, Figure 2 shows the intra-rater reliability ρ_C of the different campaigns for the original data and the filtered data (level 3), respectively. It can be seen that the intra-rater reliability is increased in all campaigns, thus, the filtering succeeds in identifying unreliable users. On average, both ρ_C and κ_C (inter-rater reliability) are increased by 0.0879 and 0.2215, respectively. The three level filtering of the users from campaign $C3$ only leads to a slight increase of the intra-rater reliability. This is due to the fact that $C3$ investigates the influence of very short stallings of length 1 s and it seems to be more difficult for individual users to rate the influence on the 5-point ACR scale appropriately. Nevertheless, the inter-rater reliability of campaign $C3$ still is significantly improved

by the filtering. For a more detailed elaboration of intra- and inter-rater reliability please refer to [15].

V. Quantification of YouTube QoE

The quantification of YouTube QoE aims at inferring the subjective user rating from the stalling parameters. This includes an analysis of the user diversity conducted by means of the SOS hypothesis, before the key influence factors on YouTube QoE are investigated. Finally, a mapping between the user ratings and the key influence factors are presented. Together with the quantification of user diversity, the mapping function provides a complete picture of YouTube QoE.

A. User Diversity and the SOS Hypothesis

The reliability of the data indicates to which extent the users give consistent QoE ratings. However, a certain heterogeneity of the test subjects' opinions on the quality experienced remains, caused by several psychological influence factors such as individual expectations regarding quality levels, type of user and sensitivity to impairments, uncertainty how to rate a certain test condition, etc. Therefore, we investigate this diversity among users and show that the filtered data leads to valid results. To this end, we analyze quality ratings where users experience the same individual test conditions, i.e. the same number of stalling events and the same length of single stalling events. The SOS hypothesis as introduced in [16] postulates a square relationship between the average user ratings $\mathrm{MOS}(x)$ and the standard deviation $\mathrm{SOS}(x)$ of the user ratings for the same test condition x: $\mathrm{SOS}(x)^2 = a\left(-\mathrm{MOS}(x)^2 + 6 \cdot \mathrm{MOS}(x) - 5\right)$. Then, the SOS parameter a is characteristic for certain applications and stimuli like waiting times. Web surfing is closely related to YouTube video streaming due to the TCP-based delivery of data and the resulting waiting times due to network impairments. For web surfing, the SOS parameter is about 0.3 according to [16].

For the unfiltered YouTube user ratings, we obtain a SOS parameter of 0.4592 which is very large and shows an even larger user diversity than for complex cloud gaming [17]. Thus, the unfiltered data do not seem to be valid from this perspective. Considering the filtered data, we obtain an SOS parameter of 0.3367 which lies in the range of web surfing. This clearly indicates the validity of the filtered data. Consequently, we consider only filtered data in the following because of their reliability and validity.

B. Key Influence Factors on YouTube QoE

In the crowdsourcing campaigns, we focused on quantifying the impact of stalling on YouTube QoE and varied 1) the number of stalling events as well as 2) the length of a single stalling event, resulting in 3) different total stalling times. We also considered the influence of 4) the different crowdsourcing campaigns, 5) the test video id in order to take into account the type of video as well as the resolution, used codec settings, etc. Further, we asked the users to additionally rate 6) whether they liked the content (using a 5-point ACR scale). We collected additional data concerning the background

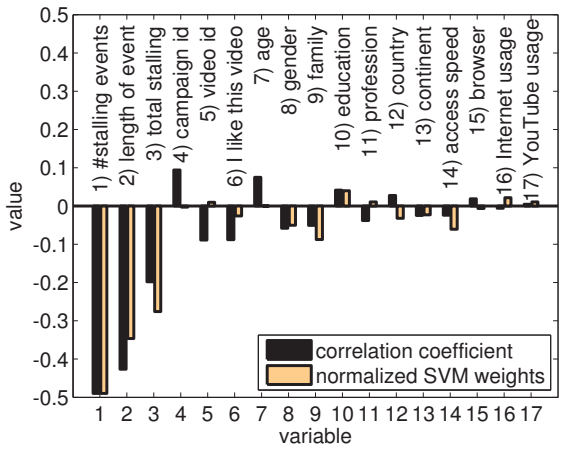

Fig. 3. Identification of key influence factors on YouTube QoE

of the user by integrating demographic questions including 7) age, 8) gender, etc. (9-13) as well as questions regarding their Internet application usage habits (16-17) in the survey. Furthermore, we additionally collected data such as access network speed (14) and browser used (15) in order to identify potential influence factors on YouTube QoE (see Figure 3).

Finally, the key influence factors on YouTube QoE are identified by means of (a) correlation coefficients and (b) support vector machine (SVM) weights. We compute the Spearman rank-order correlation coefficient between the subjective user rating and the above mentioned variables. In addition, we utilize SVMs as machine learning approach to make a model for classification. Every variable gets a weight from the model indicating the importance of the variable. However, SVMs are acting on two-class problems only. For this, we take the categories 1 to 3 of the ACR scale to class "bad quality" and the categories 4 to 5 to class "good quality". We choose the implementation of SMO (Sequential Minimal Optimization [18]) in WEKA [19] for analysis.

Figure 3 shows the results from the key influence analysis. On the x-axis, the different influence factors ν_i are considered, while the y-axis depicts the correlation coefficient α_i as well as the SVM weights β_i which are normalized to the largest correlation coefficient for the sake of readability. We can clearly observe from both measures α_i and β_i, that the stalling parameters dominate and are the key influence factors. Surprisingly, the user ratings are statistically independent from the video parameters (like resolution, video motion, type of content like news or music clip, etc.), the usage pattern of the user, as well as its access speed to reflect the user's expectations. As future work, we will further investigate such influence factors by considering more extreme scenarios (e.g. very small resolution vs. HD resolution).

C. Mapping between MOS and Stalling

The analysis in the previous subsection has shown that YouTube QoE is mainly determined by stalling and both stalling parameters, i.e. frequency and length. For quantifying

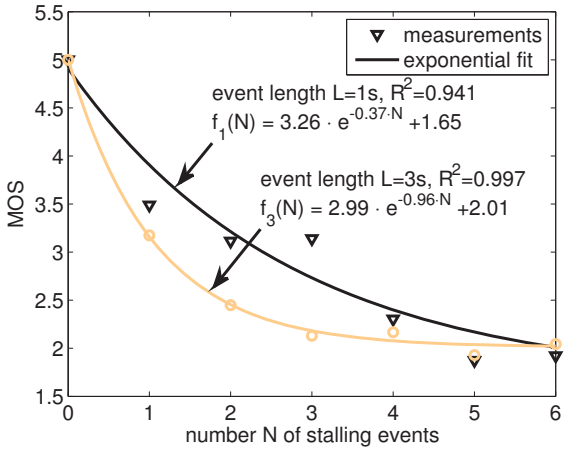

Fig. 4. Mapping functions of stalling parameters to mean opinion scores

YouTube QoE, concrete mapping functions depending on these two stalling parameters have to be derived. To this end, Figure 4 depicts the MOS values for one and three seconds stalling length for varying number of stalling events together with exponential fitting curves (as discussed in [20]). The goodness of fit is quantified by coefficient of determination R^2 and close to perfect match. The x-axis again denotes the number of stalling events, whereas the y-axis denotes the MOS rating. The results show hat users tend to be highly dissatisfied with two ore more stalling events per clip. However, for the case of a stalling length of one second, the user ratings are substantially better for same number of stallings. Nonetheless, users are likely to be dissatisfied in case of four or more stalling events, independent of stalling duration.

VI. Conclusions and Outlook

In this paper we have quantified QoE of YouTube on behalf of the results of seven crowdsourcing campaigns. We have shown that for this application, QoE is primarily influenced by the frequency and duration of stalling events. In contrast, we did not detect any significant impact of other factors like age, level of internet usage or content type. Our results indicate that users tolerate one stalling event per clip as long as stalling event duration remains below 3 s. These findings together with our analytical mapping functions that quantify the QoE impact of stalling can be used as guidelines for service design and network dimensioning.

Furthermore, we demonstrated how crowdsourcing can be used for fast and scalable QoE assessment for online video services, since testing is parallelized and campaign turnaround times lie in the range of a few days. We also showed that results quality are an inherent problem of the method, but can be dramatically improved by filtering based on additional test design measures, i.e. by including consistency, content, and gold standard questions as well as application monitoring. Albeit such filtering can result in a 75 % reduction of user data eligible for analysis, crowdsourcing still remains a cost-effective testing method since users are typically remunerated with less than 1 $. Nevertheless, sophisticated methods are required to reduce or avoid rejection of user results, e.g. by utilizing reputation systems of existing crowdsourcing platforms. For these reasons we believe that crowdsourcing has high potential not only for testing online video usage scenarios, but also for QoE assessment of typical Internet applications like web surfing, file downloads and cloud gaming.

References

[1] Cisco Systems Inc., "Cisco Visual Networking Index: Forecast and Methodology, 2010-2015," June 2011.

[2] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, "Measuring the quality of experience of http video streaming," in *IEEE/IFIP IM (Preconf Session)*, Dubland, Ireland, May 2011.

[3] K. Chen, C. Chang, C. Wu, Y. Chang, C. Lei, and C. Sinica, "Quadrant of Euphoria: A Crowdsourcing Platform for QoE Assessment," *IEEE Network*, vol. 24, no. 2, Mar. 2010.

[4] M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Cost-Optimal Validation Mechanisms and Cheat-Detection for Crowdsourcing Platforms," in *Workshop on Future Internet and Next Generation Networks*, Seoul, Korea, Jun. 2011.

[5] T. Hoßfeld, M. Hirth, and P. Tran-Gia, "Modeling of Crowdsourcing Platforms and Granularity of Work Organization in Future Internet," in *International Teletraffic Congress (ITC)*, San Francisco, USA, Sep. 2011.

[6] M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Anatomy of a crowdsourcing platform - using the example of microworkers.com," in *Workshop on Future Internet and Next Generation Networks*, Seoul, Korea, Jun. 2011.

[7] E. Huang, H. Zhang, D. Parkes, K. Gajos, and Y. Chen, "Toward Automatic Task Design: A Progress Report," in *ACM SIGKDD Workshop on Human Computation*, Washington, USA, Jul. 2010.

[8] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing User Studies with Mechanical Turk," in *ACM SIGCHI Conference on Human Factors in Computing Systems*, Florence, Italy, Apr. 2008.

[9] L. Von Ahn and L. Dabbish, "Labeling Images with a Computer Game," in *ACM SIGCHI Conference on Human Factors in Computing Systems*, Vienna, Austria, Apr. 2004.

[10] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "reCAPTCHA: Human-Based Character Recognition via Web Security Measures," *Science*, vol. 321, no. 5895, Sep. 2008.

[11] G. Little, L. Chilton, M. Goldman, and R. Miller, "Turkit: Tools for Iterative Tasks on Mechanical Turk," in *ACM SIGKDD Workshop on Human Computation*, Paris, France, Jun. 2009.

[12] P. Dai, Mausam, and D. S. Weld, "Decision-Theoretic Control of Crowd-Sourced Workflows," in *24th. AAAI Conference on Artificial Intelligence*, Atlanta, USA, Jul. 2010.

[13] P. G. Ipeirotis, F. Provost, and J. Wang, "Quality Management on Amazon Mechanical Turk," in *ACM SIGKDD Workshop on Human Computation*, Washington, DC, USA, Jul. 2010.

[14] International Telecommunication Union, "Subjective video quality assessment methods for multimedia applications," *ITU-T Recommendation P.910*, April 2008.

[15] T. Hoßfeld, T. Zinner, R. Schatz, M. Seufert, and P. Tran-Gia, "Transport Protocol Influences on YouTube QoE ," University of Würzburg, Tech. Rep. 482, Jul. 2011.

[16] T. Hoßfeld, R. Schatz, and S. Egger, "SOS: The MOS is not enough!" in *QoMEX 2011*, Mechelen, Belgium, Sep. 2011.

[17] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hoßfeld, "An Evaluation of QoE in Cloud Gaming Based on Subjective Tests," in *Workshop on Future Internet and Next Generation Networks*, Seoul, Korea, Jun. 2011.

[18] J. C. Platt, "Using Analytic QP and Sparseness to Speed Training of Support Vector Machines," in *Conference on Advances in Neural Information Processing Systems 11*, vol. 11, Dever, USA, Nov. 1998.

[19] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," *SIGKDD Explorations*, vol. 11, no. 1, 2009.

[20] M. Fiedler, T. Hoßfeld, and P. Tran-Gia, "A Generic Quantitative Relationship between Quality of Experience and Quality of Service," *IEEE Network Special Issue on Improving QoE for Network Services*, vol. 24, Jun. 2010.