2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX)

Assessing Interactive Gaming Quality of Experience Using a Crowdsourcing Approach

Steven Schmidt*, Babak Naderi*, Saeed Shafiee Sabet*‡, Saman Zadtootaghaj*, Sebastian Möller*†
*Quality and Usability Lab, Technische Universität Berlin, Germany, forename.surname@tu-berlin.de
†DFKI Projektbüro Berlin, Germany, sebastian.moeller@dfki.de
‡SimulaMet Oslo, Norway, saeed@simula.no

***Abstract*—Traditionally, the Quality of Experience (QoE) is assessed in a controlled laboratory environment where participants give their opinion about the perceived quality of a stimulus on a standardized rating scale. Recently, the usage of crowdsourcing micro-task platforms for assessing the media quality is increasing. The crowdsourcing platforms provide access to a pool of geographically distributed, and demographically diverse group of workers who participate in the experiment in their own working environment and using their own hardware. The main challenge in crowdsourcing QoE tests is to control the effect of interfering influencing factors such as a user's environment and device on the subjective ratings. While in the past, the crowdsourcing approach was frequently used for speech and video quality assessment, research on a quality assessment for gaming services is rare. In this paper, we present a method to measure gaming QoE under typically considered system influence factors including delay, packet loss, and framerates as well as different game designs. The factors are artificially manipulated due to controlled changes in the implementation of games. The results of a total of five studies using a developed evaluation method based on a combination of the ITU-T Rec. P.809 on subjective evaluation methods for gaming quality and the ITU-T Rec. P.808 on subjective evaluation of speech quality with a crowdsourcing approach will be discussed. To evaluate the reliability and validity of results collected using this method, we finally compare subjective ratings regarding the effect of network delay on gaming QoE gathered from interactive crowdsourcing tests with those from equivalent laboratory experiments.**

***Index Terms*—crowdsourcing, gaming, QoE, evaluation**

I. Introduction

For many research purposes, there is an interest in gathering a large amount of data in a short time frame of a demographically diverse audience. To assess the Quality of Experience (QoE) of multimedia services, traditionally laboratory studies are conducted. While this offers a controlled environment, these experiments are often time-consuming and expensive. Therefore, the method of Crowdsourcing (CS) has become very popular in the recent years. Participants of such tests, referred to as (crowd) workers, will typically be recruited via platforms such as Amazon Mechanical Turk (MTurk) and will solve short mini-tasks requiring some human intelligence compensated with monetary reward. However, as there is no direct contact to the workers, obtaining valid and reliable results is very challenging and strongly depends on the purpose and content of the experiment. While CS can be used to debug applications, to gather data about network connections and localization data, and for labelling tasks, it recently gained also attention for the quality assessment of diverse media contents such as speech, audio, and video quality [1]–[5]. The CS approach in the gaming domain, referred to as crowd gaming in the following, could be used for subjective interactive and passive quality assessment, usability tests, and playtesting.

In this paper, we investigate the impact of network and encoding parameters, namely delay, packet loss, and framerate, as well as changes in the game design on gaming QoE using a CS approach. Therefore, we developed an evaluation method based on the recently published ITU-T Recommendations P.808 and P.809. We will explain which steps we followed to investigate appropriate participation of workers, to increase their motivation to focus on the rating task, and how to control typically considered system influence factors for gaming research. The results of a series of studies will be discussed in respect to the expected influence of manipulated system factors. Finally, we will compare results of an experiment investigating the effect of delay on gaming QoE assessed in a CS test with those from a traditional lab study.

The remainder of the paper is structured as follows: Section II will summarize related work about CS and gaming QoE assessments. In Section III, an experimental methodology for assessing interactive gaming QoE using CS is described. Section IV shows results of conducted experiments and a comparison of test methods. Finally, Section V concludes the paper with a discussion and possible future work.

II. Related Work

Recently CS has been frequently used for media quality assessment as shown in [5], [6]. In these research activities, workers participate in subjective tests from their own working environment while using their own hardware which differs from the controlled laboratory studies. This approach provides higher validity as the situation is more realistic than the laboratory environment. However, in a price of being vulnerable to effects of uncontrolled influence factors. To overcome the multitude of challenges to conduct CS tests offering reliable and valid results, a variety of influencing factors and methods for media quality assessment have been investigated and different guidelines were provided in last years [4], [7], [8]. The lessons learned from recent work led to the ITU-T Rec. P.808 on the use of CS for subjective

978-1-7281-5965-2/20/$31.00 ©2020 IEEE

evaluation of speech quality. The recommendation describes the creation of test materials, experimental designs, and the procedure for conducting listening tests in the crowd, as well as how to report the result. In regard to gaming QoE research, the main reference for quality assessment is the ITU-T Rec. P.809, presenting methodologies for subjective quality assessment for gaming applications. It includes information about gaming QoE aspects, test set-up, stimulus duration, what to assess in pre-test, in-game, and post-test questionnaires, and which test paradigm, i.e., passive viewing-and-listening and interactive test, should be selected for which purpose. With respect to crowdsourced quality assessment of gaming applications, there have been only a few researches carried out. In [9] a few recommendations on a CS approach for online gaming tests are given. The authors of [10] present a CS game platform that can be used to create and share simple games, and collect data for different purposes.

III. Methodology

In this section, all components of the interactive crowd gaming test framework, as shown in Figure 1, will be described.

A. Game Implementation

The game is a central aspect of an interactive gaming study. It is not only the stimulus to investigate, it can also be the bridge between the user, server, and CS platform. We developed and modified simple web-compatible JavaScript games and hosted them on a web server, which workers can access via an URL available on a crowd platform. For the development, we used the p5.js library, which offers a set of drawing functions and add-ons for interaction with other HTML5 objects. One alternative would be the use of the cross-platform game engine Unity to create a WebGL game. The open-source nature of this approach offers several important advantages, which will be described in the following.

1) Game Introduction: The ITU Rec. P.809 suggests that players must learn the controls and rules of the game before rating the first condition. Therefore, workers had to pass a training session. Before each game scenario, a screenshot of the game with labeled heads-up display (HUD) (e.g., timers, scores) and game elements (e.g., character, enemies, targets), controls, and a description of the rules of the game was shown.

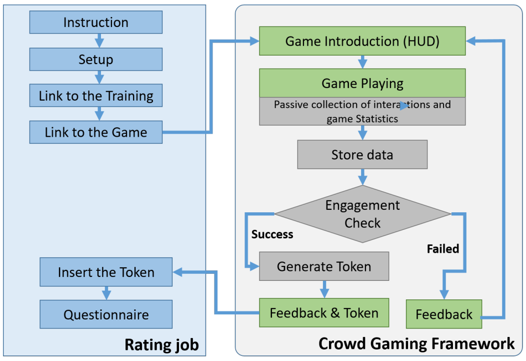

Fig. 1: Components of Crowd Gaming Framework.

B. Token System

Per se there is no information available to find out if someone who is playing a game on the web server is also participating in the survey on the CS platform. For this reason, a 36 character long universally unique identifier (UUID) was generated after each gaming session and served as a token. The token was stored in server logs among other information, and workers were asked to copy this token and paste it back to the survey at the beginning of the rating process. If a valid token was used, the rating scale was shown. This method ensured that workers really played the game until the end. It also enables us to know which information stored on the server belong to which worker ratings. While well-versed workers may figure out the method of the token creation, a mismatch of a potentially manipulated token and those stored on the server would lead to discarding the ratings of such workers.

C. Stimulus Generation

Essential for conducting an interactive gaming QoE assessment test is the generation of a stimulus. While from a technical point of view, the content of a game does not play an important role for conducting a crowd gaming test, there are a few aspects which should be considered. Firstly, the duration of a stimulus should be limited and clearly indicated to participants. A timer of the remaining playing time was added to the HUD of each game. Secondly, an automated restarting of the game scenario should be implemented in case of a defeat of the player. Thirdly, an open-source game offers means to artificially add network impairments or encoding artefacts to a game. A network delay can easily be simulated by buffering input commands, input packet loss can be simulated by using a random number generator and discarding functions called for input events, and even frame rates can be changed by skipping the drawing function, which is called every frame. Using this approach, degradations can be simulated without controlling the network conditions of workers. However, it has to be noted that these degradations might have small differences to the real end-to-end delay or packet loss, they should be carefully designed similar to the real scenario. For our test, we applied the delay by buffering the input commands and the framerate was manipulated by changing the frequency of the function call to draw the game elements.

D. Game Design

Especially for fundamental research, it is highly beneficial to be able to change the design of a game. Different methods of controlling a game, interface design, balancing, or characteristics such as the pace or predictability, which may influence the impact of a network delay on a gaming QoE, can be investigated. Furthermore, information of the game state such as performance indicators, which can be added to the HUD, and logs of the player inputs can be generated.

E. Engagement Check

One challenge to overcome in a crowd gaming test is to find out whether a worker played a game scenario as intended.

While in a lab study this can be observed visually by the experimenter, information generated by interacting with the game can be used in the crowd. Therefore, we implemented an engagement check at the end of each stimulus. During a pre-test we logged the number of inputs, i.e., mouse clicks or keystrokes, for each game to set a threshold. It is also possible to derive a threshold by an expert judgement. Workers passed the engagement check if their number of inputs was higher than 20 percent of the typical number of inputs derived from the pre-test, scaled by ratio of stimulus duration and duration of pre-testing. If a worker failed this check, they were told that they did not put enough attention to the game and were asked to play the condition again. Not only does this method prevent workers from cheating, it also is of high value for the training session to make sure workers understood the rules and controls of the game in the short amount of time available. If knowledge about typically reached performance values such as points are available, also such information could be used in addition to the input information.

F. Crowdsourcing Workflow

The following steps are adapted from ITU-T Rec. P.808 to design the CS tests for gaming QoE assessment. For our tests, we used MTurk as the platform is the most widely used, offers a pre-selection of workers with diverse backgrounds, English speaking workers, dynamic content creation, and easy payment of participants. The task, i.e., playing some game scenarios followed by a rating task, is referred to as a Human Intelligence Tasks (HITs) on MTurk. In the following, the procedure of the CS procedure will be explained.

1) Recruitment: Depending on the purpose of the study, it may be beneficial to select a specific target group for the study. Therefore, a screening HIT can be published before the actual crowd gaming test. Here, aspects such as age, gender, playing frequency, gaming skills, as well as game and device preferences can be assessed (cf., ITU-T Rec. P.809) to create a user profile. If a profile is suitable for the research, the worker can later be invited to participate in the test based on the profile, which also contains the worker ID. For our test, the most important criteria were that workers like to play video games and that they can control them sufficiently. Additionally, some platforms offer worker profiles based on a variety of characteristics. We only recruited workers who fulfilled the following three criteria: their location is in the United States, their HIT approval rate is over 98 percent, and their number of approved HITs is greater than 500 (cf. ITU-T Rec. P.808).

2) Requirements: Every HIT started with a summary of requirements. Workers were asked to only participate in the test if they fulfill the following requirements: They should have played video games in the past year. They should be interested in playing video games. They are using a desktop (PC) or a laptop for the job. Their device has a keyboard and mouse connected. Their device is connected with power. Their device must be able to play stereo sound.

3) HIT Instruction: The procedure of the test, what is expected from the workers, and how to use the rating scale should be explained to the participants using short and clear sentences for each step. In our instruction we explained that they will play different simple game scenarios and rate their experience after each scenario and we recommended to use a modern web browser for the test. Next, it was clearly stated that responses will be used for scientific research and that especially the questionnaire should be treated very seriously. Afterwards, the estimated total duration of the HIT, the duration of each scenario and the structure of the HIT, which was split into several section, were listed.

4) Questionnaire Instructions: For the test, we used the 7-point continuous rating scale as recommended in ITU-T Rec. P.809 for the assessment of gaming QoE. For consistency, the scale was also used for the remaining items. An example of the scale is given in Figure 2. The usage of the scale, especially concerning the overflow area, was explained to the participants in the introduction section. Furthermore, it was mentioned that it may happen that the quality of a scenario is not ideal, and this is intended and not a bug in the system.

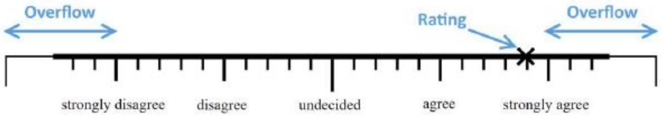

Fig. 2: 7-point continuous rating scale (cf. ITU-T Rec. P.851).

5) Payment: Presence of quality control system, and condition in which their answers would be rejected or selected for extra bonuses should be clearly explained. We promised bonuses for both quality and quantity of work they provide.

6) Worker Survey: The second section in the HIT was a short demographic questionnaire like the one in the recruitment HIT. The questions were derived from the pre-test questionnaire recommendations in ITU-T Rec. P.809.

7) Training: As suggested in ITU-T Rec. P.808 and P.809, before the first stimulus, a training scenario should be presented. Here, workers learned the rules and controls of the game. The duration was set to 30 seconds, and a token was generated at the end of the scenarios, if a participant passed the engagement check. Workers had to paste this token to the survey to proceed.

8) Rating Section: For each stimulus, workers were asked to play a game scenario by following a given link, and then copy the verification code that appears at the end of the scenario and use it for the HIT. If the worker passed the token check, the rating section became visible. In the rating section, workers were asked to indicate how much they agree or disagree with each of the following statements by clicking on the 7-point scale below as explained in the introduction. A dynamically generated slider provided workers always with a single item to prevent them from getting biased by their previous ratings. Once an answer was given, the next question was automatically shown.

9) Quality Control: It may happen that a worker despite the clear instructions take the rating process lightly or even attempt to cheat. Therefore, we added trapping questions (also known as gold standard questions) and consistency checks to the questionnaire [1]. In a test using a 31-item questionnaire, we added three trapping and two repeated questions. It should be avoided to add too many of these as it may show strong distrust to workers. For each condition, three different kinds of trapping questions were randomly assigned and kept the same for each condition used: (1) very obvious questions such as "Please select the answer "disagree" on the scale below.", (2) questions related to the current activity such as "Right now, I am answering a survey in MTurk.", and (3) a question related to the played game such as "In the game I played, I was able to talk to other players.". While the first kind should be a clear sign that a quality control is embedded in the questionnaire as it was told, the latter is most like only be answered correctly with proper attention.

10) Stimuli and Conditions: For our tests, we adhered to ITU-T Rec. P.809 and selected a stimulus duration of 90 seconds. However, a duration of 30 seconds was used for the training scenario. We aimed to keep the average duration of a HIT at around 15 minutes in order to avoid fatigue.

11) Web Server: Apart from providing access to the games, an API in the web server was used to save logging information of each played condition. Information such as game identification to prevent the cheating, game scores, users inputs and other statistics was stored into the server.

IV. Results

The main purpose of the conducted crowd gaming tests was to investigate the impact of delay, framerate, and packet loss as well as changes in the type of game feedback on gaming QoE. The results of five studies, in which in each case one factor was investigated, will be presented in this section. In addition, we evaluate validity and reliability of collected data of the CS tests by discussing if the results match with the expected influence of these factor (considering ITU-T Rec. G.1072 as a basis) and by comparing the results of a lab study and CS test for the investigation of the impact of delay on gaming QoE.

A. Experimental Design

We developed a dataset of six JavaScript games: Dodge, GTA, Shooting (Range), Flappy (Bird), Rocket (Escape), and T-Rex. For their implementation as well as for the experimental design, we followed the framework described in Section III. While in Dodge and T-Rex obstacles have to be avoided by well-timed keystrokes, Rocket and Flappy require a frequent player input to balance the position of the character. Finally, GTA and Shooting Range require in addition spatially accurate mouse inputs. As independent variables (IV), the network delays (0, 150, and 300 ms), different framerates (60, 30, 10 fps), input packet loss (0, 10, 30 %), and different feedback types (visual, audio-visual, audio-visual combined with progress) were used. In each HIT, participants assessed three game scenarios of one game which differ in the implementation to manipulate the IV. After each scenario, they answered a pool of items assessing first the overall gaming QoE using the item proposed in ITU-T P.809 followed by 3 trapping questions, 2 consistency questions, as well as 26 items measuring responsiveness, controllability, and (immediate) feedback. The mean of these three constructs describes in the following the Input Quality (IQ). In total, 571 workers participated in the tests, which resulted in 1713 ratings since each HIT contained three or four conditions. The estimated time to get the ratings from all workers was about three hours. In the end, a total of 571 subjects participated in the CS test, 245 females and 321 males with the age between 18 to 35 years. More than 42% of the test participants are experienced gamers.

B. General Findings and Dropout Rates

The average time spent on a HIT was 28 minutes. The trapping questions were answered incorrectly by many workers. Ratings of 28 workers were discarded due to failure of the consistency check for repeated items. Only considering those who did answer all trapping question correct and passed the consistency check, 1152 out of the initial 1713 ratings (67%) remained. Finally, an outlier detection using the outlier labeling method described in [11] was performed. It was decided to allow one non-extreme outlier per worker for any of the questionnaire items. As a result, 38 ratings were removed, which consequently lead to 1114 clean ratings, which were used in the following analysis.

C. Study 1: Delay

In the first study the influence of delay on gaming experience was investigated on six games. Three levels of delay (0, 150, and 300 ms) were simulated artificially on users' inputs. Figure 3 A) shows the gaming QoE and Figure 3 E) shows the Input Quality (IQ) of the test conditions. The influence of delay was depending on the games. Shooting games such as GTA and Shooting Range were more sensitive to delay than a racing game such as Rocket. For the jumping game T-Rex, the drop of the QoE was not high at 150ms as the players still could jump over obstacles properly. However, at 300ms delay the QoE had a significant drop as the interval to react to an obstacle was similar to the delay. A two-way Mixed Analysis of Variance (ANOVA) was conducted to compare the overall gaming QoE using delay as a within-subject variable, and game as a between-subject factor. The ANOVA yielded a significant main effect of delay, $F(2,178) = 168.54$, $p < .001$, $\eta_p^2 = .65$, a significant main effect of the game, $F(5,89) = 7.73$, $p < .001$, $\eta_p^2 = .30$, as well as an interaction effect of game and delay, $F(10,178) = 5.02$, $p < .001$, $\eta_p^2 = .22$. Also for the IQ, a significant interaction effect was found, $F(10,178)=4.99$, $p < .001$, $\eta_p^2 = .22$. This result is inline with the previous traditional lab studies which show the influence of delay on gaming QoE, which can also be mediated by the game type.

D. Study 2: Framerate

In the second study, the influence of framerates on gaming experience was investigated on two games. Three levels of framerates (60, 30, 10 fps) were simulated on two games by changing the game engine drawing rate. Figure 3 B) shows the gaming QOE and Figure 3 F) the IQ of the test

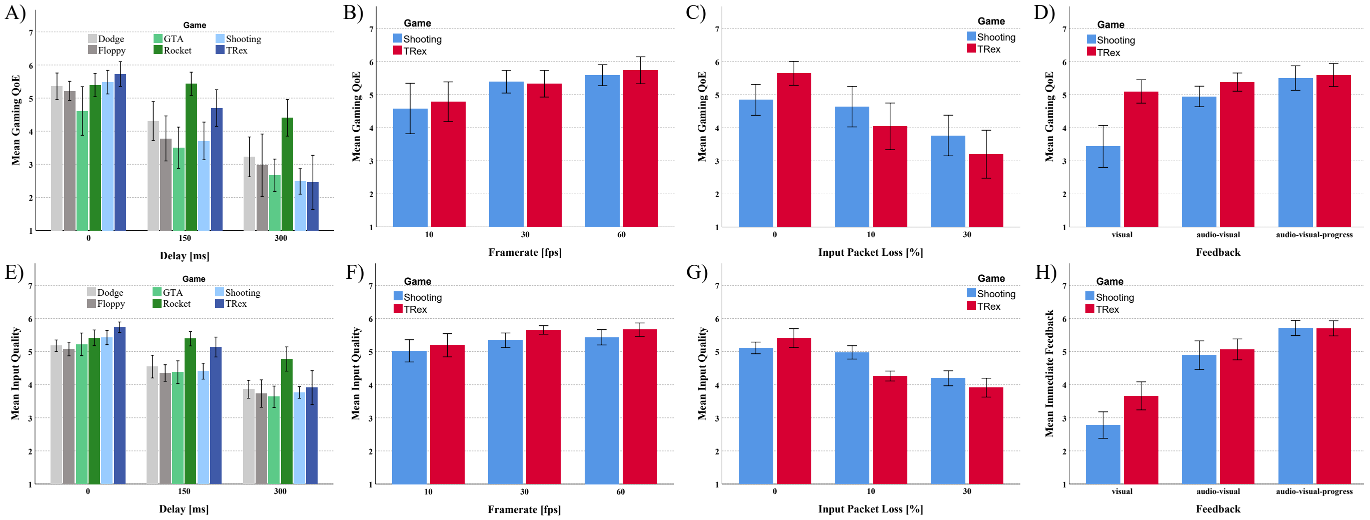

Fig. 3: Bar plots of gaming QoE and input quality mean values in CS studies 1 to 4.

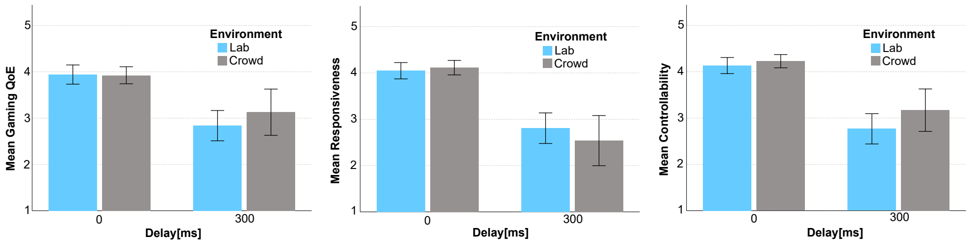

Fig. 4: Bar plots of gaming QoE, responsiveness, and controllability mean values in study 5 comparing CS and lab test.

conditions. Users could not see any difference between the 30 and 60 fps. However, reducing the framerate to 10 fps strongly reduced the gaming QoE on both games in the same way. A two-way Mixed Analysis of Variance was conducted to compare the overall gaming QoE using framerate as a within-subject variable, and game as a between-subject factor. The ANOVA yielded a significant main effect of framerate, F (1.33, 31.86) = 14.19, p < .001, $\eta_p^2 = .37$. Also for the IQ, a main effect of framerate was found, F (1.29,30.93) = 11.92, p < .001, $\eta_p^2 = .33$. In both cases, only the 10 fps condition is causing the effect. This result is inline with the previous traditional lab studies which showed that non-expert gamers do not rate 60 fps and 30 fps much differently.

E. Study 3: Input Packet Loss

The third study investigates the influence of packet loss linked to user inputs. Three levels of packet loss (0, 10, 30 %) were simulated on the user's inputs by discarding the inputs in case of a loss. Figure 3 C) shows the gaming QoE and Figure 3 G) shows the IQ of the test conditions. Similar to delay, the influence of the packet loss on the user input was dependent on the game. It has a stronger effect for T-Rex where missing a jump would lead to an immediate punishment as in the game Shooting where gamers always had a additional chances to shoot at the target. A two-way Mixed Analysis of Variance was conducted to compare the overall gaming QoE using packet loss as a within-subject variable, and game as a between-subject factor. The ANOVA yielded a significant main effect of packet loss, F (2,44) = 48.62, p < .001, $\eta_p^2 = .69$, as well as an interaction effect of game and packet loss, F (2,44) = 9.97, p < .001, $\eta_p^2 = .31$. Also for the IQ, a significant interaction effect was found, F (2,44) = 14.83, p < .001, $\eta_p^2 = .40$. The results confirm the finding of traditional lab studies.

F. Study 4: Feedback Type

Despite the three other studies that were mostly focused on the network degradation, the fourth study investigates changes on the game design. Three types of feedback (visual, audio-visual, audio-visual with progress) were developed for two games. Figure 3 D) shows the gaming QoE and Figure 3 H) shows the immediate feedback ratings of the test conditions. For both games, adding more feedback to the game resulted in enhancement on the immediate feedback ratings. The enhancement was stronger in the game Shooting as in the version with only visual feedback, users did not have a good insight whether they were successful on shooting the targets (missing bullet hole). A two-way Mixed Analysis of Variance was conducted to compare the overall gaming QoE using feedback type as a within-subject variable, and game as a between-subject factor. The ANOVA yielded a significant interaction effect of feedback type and game, $F(1.46, 59.90) = 14.67$, $p < .001$, $\eta_p^2 = .26$. Also for the immediate feedback, a significant interaction effect of feedback type and game was found, F (2,82) = 4.44, p = .015, $\eta_p^2 = .10$.

G. Study 5: Environment Comparison

The fifth study compares the collected results from the crowd gaming method with those from an equivalent lab study. We invited 27 gamers, ten females and 17 males, to our laboratories to play the game Rocket Escape which we also used in the crowd gaming tests. The methodology, also in terms of the test procedure, length of tests, and assessments, was in line with those used during the cs tests. The participants were aged between 20 years to 33 years. As independent variable, an artificially added input delay (0 and 300 ms) was used. In Figure 4 the ratings collected are shown in comparison with results from the crowd gaming tests. It must be noted that the data was transformed to a 5-point ACR scale according to [12]. One can observe that the ratings gathered in the lab study are comparable to those collected in the crowd gaming test. The overall gaming QoE, responsiveness, and controllability degrades substantially in case of the added delay, but are not near the saturation region, as the game is not highly sensitive. As the data was not normally distributed and we are only interested in a comparison of the test methods, a Mann-Whitney Test was performed for each delay condition. As evident in Table I, no significant differences for any of the assessed aspects could be found for the test method comparison for any of the two delay conditions.

TABLE I: Mann-Whitney Test results for method comparison.

Delay	**Quality Aspect**	U	z	p	r
0 ms	Gaming QoE	581	0.05	.962	.01
	Responsiveness	513	0.86	.394	.08
	Controllability	482	1.23	.221	.11
300 ms	Gaming QoE	222	1.28	.20	.12
	Responsiveness	229	1.13	.26	.10
	Controllability	221	1.51	.07	.14

V. CONCLUSIONS AND FUTURE WORK

In this paper, we investigated the impact of delay, framerate and input packetloss and feedback type on gaming QoE using CS. In addition, we described the structure of a framework for the use of CS for gaming QoE assessment.

We showed that with a proper stimuli design and controlling the environment and participants behaviour, we can get CS results which are similar to the lab study, as evident based on the study 5 investigating the influence of delay on gaming QoE. Additionally, we can confirm that we observed expected trends for the remaining variables such as input packet loss, feedback type, and framerate resulted from the crowd gaming tests. Thereby, we can conclude that the crowd gaming method is well suited for example for the development or validation of questionnaires, and that the work of the ITU-T Rec. P.808 and P.809 are of great use for crowd gaming tests.

In comparison to passive tests, including an engagement check in interactive test was very useful. It helped filtering data from workers who did not play the game as expected and ensured that workers learned to control the games during a training scenario. A training section was crucial to gather high quality data. The same applies to the quality control items added to the questionnaire. As suggested in ITU-T P.808, the number of additional items added for reliability checks should not be larger than 10% of the number of items in the questionnaire. In case of a short questionnaire, especially the game content related trapping questions should be considered. While the dropout rate of about 30% in our tests appears to be high, this value is also in line with CS tests for the assessment of speech quality we conducted in the past.

In future work, we plan to focus on enhancing the framework by gathering more information about the used gaming device properties, by considering performance metrics for the engagement check, and by considering other gaming QoE related influence factors. Finally extending the framework to handle cases in which the game source is not available, is an open question for future work.

ACKNOWLEDGEMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871793.

REFERENCES

[1] B. Naderi, *Motivation of workers on microtask crowdsourcing platforms*. Springer, 2018.

[2] T. Polzehl, B. Naderi, F. Köster, and S. Möller, "Robustness in speech quality assessment and temporal training expiry in mobile crowdsourcing environments," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[3] B. Naderi, T. Polzehl, I. Wechsung, F. Köster, and S. Möller, "Effect of trapping questions on the reliability of speech quality judgments in a crowdsourcing paradigm," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[4] T. Hoßfeld, M. Hirth, J. Redi, F. Mazza, P. Korshunov, B. Naderi, M. Seufert, B. Gardlo, S. Egger, and C. Keimel, "Best practices and recommendations for crowdsourced qoe-lessons learned from the qualinet task force" crowdsourcing"," 2014.

[5] D. Saupe, F. Hahn, V. Hosu, I. Zingman, M. Rana, and S. Li, "Crowd workers proven useful: A comparative study of subjective video quality assessment," in *QoMEX 2016: 8th International Conference on Quality of Multimedia Experience*, 2016.

[6] T. Hossfeld, C. Keimel, M. Hirth, B. Gardlo, J. Habigt, K. Diepold, and P. Tran-Gia, "Best practices for qoe crowdtesting: Qoe assessment with crowdsourcing," *IEEE Transactions on Multimedia*, vol. 16, no. 2.

[7] ITU-T Technical Report PSTR-CROWDS, *Subjective evaluation of media quality using a crowdsourcing approach*. Geneva: International Telecommunication Union, 2017.

[8] ITU-T Recommandation P.808, *Subjective evaluation of speech quality with a crowdsourcing approach*. Geneva: International Telecommunication Union, 2018.

[9] P. K. Paranthaman and S. Cooper, "Arapid: Towards integrating crowdsourced playtesting into the game development environment," in *Proceedings of the Annual Symposium on CHI in Play*, 2019.

[10] I. Guy, A. Hashavit, and Y. Corem, "Games for crowds: A crowdsourcing game platform for the enterprise," in *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, 2015, pp. 1860–1871.

[11] D. C. Hoaglin and B. Iglewicz, "Fine-tuning some resistant rules for outlier labeling," *Journal of the American statistical Association*, vol. 82, no. 400, pp. 1147–1149, 1987.

[12] F. Köster, D. Guse, M. Wältermann, and S. Möller, "Comparison between the discrete acr scale and an extended continuous scale for the quality assessment of transmitted speech," *Fortschritte der Akustik, DAGA*, 2015.