Service Quality Definition and Measurement
A Technical Report
by NGMN Alliance

Version: 1.0.4
Date: 27th August 2013
Document Type: Final Deliverable (approved)
Confidentiality Class: P - Public

Project: P-SERQU
Editor / Submitter: Michael Wennesheimer (Vodafone), Dave Robinson (Alcatel Lucent)
Contributors: Johan De Vriendt, Danny De Vleeschauwer, Charles Bahr (Alcatel Lucent), Karl Heise, Savvas Argyropoulos, Bernhard Feiten (Deutsche Telekom), Murat Bilgic, Tommi Raiha (Exfo), Sebastian Egger, Michael Seufert, Martin Slanina (FTW), Meik Kottkamp, Harald Gsödl, Harald Ibl (Rohde+Schwarz), Raquel García Pérez (Telefónica)

Approved by / Date: NGMN Board / 12th August 2013

This document contains information that is confidential and proprietary to NGMN Ltd. The information may not be used, disclosed or reproduced without the prior written authorisation of NGMN Ltd., and those so authorised may only use this information for the purpose consistent with the authorisation.

For all Confidential documents (CN, CL, CR):
For Public documents (P):
© 2013 Next Generation Mobile Networks Ltd. All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means without prior written permission from NGMN Ltd.

The information contained in this document represents the current view held by NGMN Ltd. on the issues discussed as of the date of publication. This document is provided “as is” with no warranties whatsoever including any warranty of merchantability, non-infringement, or fitness for any particular purpose. All liability (including liability for infringement of any property rights) relating to the use of information in this document is disclaimed. No license, express or implied, to any intellectual property rights are granted herein. This document is distributed for informational purposes only and is subject to change without notice. Readers should not design products based on this document.
1 Executive Summary

The NGMN Alliance identified Quality of Experience (QoE) as a key focus topic of the mobile telecommunications industry and further concluded to tailor its QoS/QoE research project around streamed video applications, as these are being used at a rapidly increasing rate over wireless networks. Particularly, the focus was set on HTTP Adaptive Streaming (HAS).

The P-SERQU project conducted by the NGMN alliance addresses a large audience in the multimedia delivery domain. Firstly, telecom operators and ISP providers can identify the expected range of QoE based on QoS KPIs. Moreover, the developers of media platforms based on HAS may exploit the results of the study to identify optimal adaptation strategies of the media clients and maximise the perceived quality of the end-users. Furthermore, the QoE research community may use the results of the subjective study (referred to as “mass-test” below) to analyse the perceptual effects of audio-visual temporal adaptation in the context of adaptive streaming over HTTP and the impact of end-devices (e.g. tablets vs smartphones) on the perceived quality.

This particular scope of research was to identify KPIs to monitor video QoE delivered over wireless networks. One of the original project objectives was to develop a correlation between radio and core network QoS KPIs to predict QoE of streamed video. Primary reason for that objective was to provide mobile network operators a mechanism to predict QoE based on the QoS measurements obtained with existing tools in their network.

In this case, a test lab was created. This environment consisted of an LTE network into which controlled impairments could be introduced. The lab was used for two purposes: First, for generating a number of HAS profiles by streaming HAS encoded videos over the LTE network instrumented with various radio scenarios. These profiles were then used for creating the corresponding video files. The second purpose was the calibration phase where 20 people viewed and rated a sequence of clips delivered over the test LTE network with the same impairments as used for the profile creation.

In order to get statistically significant results a large number of test subjects (or participants) downloaded the files and viewed them accordingly. As the files were fully downloaded before viewing them locally, the quality of the delivery network was not impacting the results. After watching each video the subjects issued a video quality rating. Supplementary objective measurement results were obtained, which served as an additional input throughout the analysis phase.

The results revealed that instead of making a one-to-one mapping between generic radio and core network QoS and QoE, it is less complex and more accurate to measure HAS bit-stream properties and predict QoE based on these properties. The biggest benefit of this approach is a simplification of the measurement framework, by not needing a reference video sample. Instead collecting HTTP header information on the server, or any point between the terminal and the server, is sufficient.
The subjective and objective results at hand were used to examine the accuracy of existing video quality estimation algorithms. This was done by applying these algorithms to the video files created according to the measured HAS profiles and comparing the algorithm outputs with the subjective ratings from the test subjects.

In summary, the P-SERQU project has shown that for HTTP Adaptive Streaming (HAS) content, conventional objective measurement techniques such as PSNR and SSIM \(^1\) give good results. However, by just examining the video quality levels requested in HTTP GETs, an accurate way to monitor video streams in a mobile network has been identified. The Linear MOS Predictor shows that an accuracy of 0.3 MOS points can be expected. This avoids the need to analyse the significant amount of measurement data generated at the wireless level. It is a non-intrusive approach. Periodic sampling of content from each source may be required to calibrate the quality levels. If the packet headers are in the clear (unencrypted), then analysing the MPEG-2 Transport Stream header information with P.1201.2 can further improve the results at a small computational overhead.

Using either the Linear Predictor Model or P.1201.2, a mobile network operator can effectively and efficiently monitor the quality of a large number of video streams. Where the quality of a stream is below the required quality threshold, the operator can further investigate the cause. This can include identifying the radio conditions of the client as well as the general congestion in the RAN cell. While there is no 1-to-1 relationship between radio conditions and QoE, NGMN should consider extending the P-SERQU work to study how it can help in radio network planning.

\(^1\) Peak Signal to Noise Ratio (PSNR) and Structural SIMilarity (SSIM) are two widely used objective metrics for assessing the quality of video content.

Service Quality Definition and Measurement,
Version 1.0.4, 27\(^{th}\) August 2013
2 Contents

1 Executive Summary .................................................................................................................................. 2
2 Contents ................................................................................................................................................ 4
3 Introduction ........................................................................................................................................... 7
  3.1 Business Rationale and problem statement .................................................................................. 7
  3.2 Scope ................................................................................................................................................. 8
    3.2.1 HTTP Adaptive Streaming ....................................................................................................... 8
    3.2.2 Research outline and deliverables ............................................................................................. 9
4 Data gathering phase ................................................................................................................................ 10
  4.1 Test Lab ............................................................................................................................................... 10
    4.1.1 Factors considered .................................................................................................................... 12
    4.1.2 Radio Scenarios ....................................................................................................................... 12
    4.1.3 Information Collected .............................................................................................................. 13
    4.1.4 Content Selection ..................................................................................................................... 13
    4.1.5 Viewing Sessions for Calibration ............................................................................................. 14
  4.2 Preparation for Mass Test .............................................................................................................. 15
    4.2.1 Creation of HAS Profiles .......................................................................................................... 15
    4.2.2 Details of creating a typical HAS profile ................................................................................... 15
  4.3 Mass Test ............................................................................................................................................ 17
    4.3.1 Phase 1 – Survey Monkey based approach ............................................................................. 18
    4.3.2 Phase 2 - NGMN project partner hosted website ...................................................................... 18
    4.3.3 Summary of the Mass Test Process ......................................................................................... 19
5 Analyses .................................................................................................................................................. 22
  5.1 Introduction to Individual Predictors ............................................................................................... 23
    5.1.1 Peak Signal to Noise Ratio (PSNR) ........................................................................................... 25
    5.1.2 Structural Similarity (SSIM) ..................................................................................................... 25
    5.1.3 J.144 (VQM) .............................................................................................................................. 25
    5.1.4 J.341 (VQUAD) .......................................................................................................................... 26
    5.1.5 ITU-Recommendation P.1201.2 ............................................................................................... 26
    5.1.6 HAS Parameters ....................................................................................................................... 26
3 Introduction

3.1 Business Rationale and problem statement

The NGMN Alliance identified Quality of Experience (QoE) as a key focus topic of the mobile telecommunications industry.

From a historical view mobile network operators traditionally looked at performance of individual network elements (Network Performance). The next level was becoming service-aware by combining individual performance metrics along the end-to-end chain into Quality of Service (QoS) indicators (Service Management). Finally, many operators invested in Customer Experience Management solutions that allowed network-based information to be captured and interpreted on an individual per-subscriber level.

![Quality of Experience (QoE) vs Quality of Service (QoS)](image)

**Figure 1: NP, QoS, QoE model and terminology**

However, today's demand to differentiate requires an increased knowledge or anticipation of the actual Quality of Experience (QoE), i.e. how the offered QoS is actually perceived by the end user.

The model in Figure 2 provides a view on the complexity of influencing factors on QoE – both, technically and non-technically. It is apparent that there is no simple correlation of (technical) QoS to QoE. There are still adjacent areas that can lead to an unsatisfactory customer experience despite positive QoS metrics. A generic correlation model between QoS and QoE therefore does not lead to any improvements.

There are minimum thresholds of (technical) QoS though, that need to be provided by mobile operators (as a hygiene factor/level) in order to prevent a significantly degraded QoE at the subscribers' end. Additionally, maximum thresholds exist for individual (technical) QoS which when exceeded, the QoE for that service does not improve any further.
The result of users constantly and subjectively evaluating the service provided is that Quality of Experience (QoE) is the second churn driver after price. Therefore, it is extremely important for an operator to have a high level of customer satisfaction by ensuring a good QoE. It should be noted that customer acquisition costs are worth 2 to 5 times more than retention costs. It is therefore in the best interest of any operator to make sure that customers experience an expected level of quality.

3.2 Scope

The NGMN Alliance decided to tailor its QoS/QoE research project around streamed video applications, as these are being used more frequently and at an increasing rate over wireless networks. Particularly, the focus was set on HTTP Adaptive Streaming (HAS)\(^2\).

3.2.1 HTTP Adaptive Streaming

HAS is a popular technique to deliver video streams over a best effort (BE) network which adapts to the available bandwidth and can stream to clients behind firewalls without additional configuration. HAS in its various forms is widely used by professional content distributors for streaming over wireline networks to PCs and over WiFi (wireless fidelity) and cellular networks to smartphones and tablets. Previous work has examined the subjective QoE of HAS streams delivered over wireline networks. This work has been to identify KPIs to monitor video QoE delivered over wireless networks.

In HAS delivery the video is segmented in intervals of between 2 and 10 seconds long and each video segment (i.e., consecutive non-overlapping interval of video) is encoded in multiple quality versions, where a higher quality version requires a higher bit rate to be delivered. The bit strings associated with these encodings are referred to as chunks. There are as many chunks associated with a video interval as there are bit rate versions. The HAS client uses HTTP to request chunks – the first ones usually at a low bit rate to build up a play-out buffer. If the play-out buffer is large enough and chunks

are (consistently) delivered in a time shorter than the video segment length, the RDA (rate decision algorithm) in the client selects a higher bit rate for the next chunk (see Figure 3), therefore the download time becomes about equal to the video segment length, keeping the play-out buffer more or less steady. In that way the RDA continually senses the available throughput and adapts the video rate accordingly. Any mismatch between video rate and throughput, is absorbed by the play-out buffer. Assuming the RDA is working properly, and there is enough bandwidth to support the lowest quality level, no stalling should occur. As it is running on TCP, packet loss is not hampering the video quality directly (as lost packets are retransmitted) and results in reduced throughput.

![Figure 3: HTTP Adaptive Streaming](image)

### 3.2.2 Research outline and deliverables

In this project, the influence of the radio parameters on the video quality as perceived by the end users was assessed. The correlation of the Quality of Experience as measured by MOS with some radio parameters was evaluated. The impact on the MOS of the individual radio parameters and the impact of the interdependency of these radio parameters were further analysed. An explanation is given on why a QoE model based on the radio parameters necessarily has a lower performance than a QoE model based on a HAS profile and an indication of how large this loss in performance is, is also provided. The use of some standard objective video quality metrics for HAS traffic over a wireless network were also compared.

---

Note, the term “radio parameters” is used to describe various factors that might impact the performance of an HAS application. These include fading and interference/noise, which are exclusively air interface factors. The amount of competing traffic can have an influence on both air interface and wired interfaces of an LTE network. RTT may be influenced by air interface variables such as scheduling as well as core network variables such as the distance between the RAN and the HAS server.
4 Data gathering phase

Conventional subjective testing involves getting a number of test subjects to view and rate clips. This, however, was not feasible in this project. There are many factors which influence or potentially influence the subjective quality resulting in a very large number of scenarios. Consequently a two phase approach was undertaken. Firstly a simulated LTE network was created into which controlled various radio level impairments could be introduced. A small number of test subjects were recruited to view and rate the quality of HAS delivered content. This was primarily to create a calibration set of results.

A property of HAS to get reviews from a much larger number of test subjects was then used. By streaming HAS content over various radio scenarios, it was possible to create representative HAS profiles. From these profiles, corresponding video files were created. These files were made available via download to a large number of test subjects and viewed as if they were in the test lab. As the files were downloaded and viewed locally, the quality of the network used for delivery did not impact the results.

In addition to the rating of videos for quality, a short questionnaire was conducted to understand video viewing habits of the test subjects. A similar project to test HTTP Adaptive Streaming when delivered over wireline is reported in [5].

4.1 Test Lab

A test lab at EXFO facilities in Oulu, Finland, was created. This environment consisted of an LTE network into which controlled impairments could be introduced. The lab was used both to produce a calibration set of viewings and to generate a number of HAS profiles. During the calibration phase, 20 people viewed and rated a sequence of clips delivered over the test LTE network. Controlled impairments were then introduced into the LTE network. An Apple iMac with an LTE dongle was used, as it was more practical to shield the dongle from interference and other signals in a shielded box where the eNodeB antennas were also placed. This section describes how the laboratory test was carried out.
The above diagram shows the key components in the test setup. At the bottom there are the essential parts of the wireless network including EAST LTE that provides the entire EPC and NSN eNodeBs. The Propsim F8 channel emulator allows the controlled introduction of radio impairments. Two mini aerials connected to the channel emulator are placed in a shielded box along with the USB dongle to eliminate issues with interference to/from the macro environment.

The next level in the diagram shows the components providing background traffic. The Shenick diversifEye can simulate multiple clients (UEs) and servers. The client requests are sent to the NetHawk East500 which converts them into RF (LTE air interface Uu) for forwarding via the eNodeBs. The other interface of the Shenick is connected to the gateway where it receives the HTTP and FTP requests. The gateway along with an impairment tool from Rugged Tooling allowed the controlled introduction of latency impairments.

The top level is the HAS server and client. Because of the popularity in professional streamed video over wireless networks to iPhones and iPads, the Apple HLS (HTTP live streaming) variant of the HAS protocol was chosen. As LTE versions of the iPhone and iPad were not available during the lab tests, they were simulated by using an HLS client running on an Apple iMac running OS X natively supporting HLS and connected to an LTE dongle. The HAS Server is a Red Hat Enterprise Linux server version 4 with Apache web server. HLS formulated content is loaded onto the HAS server.
4.1.1 Factors considered

Many factors affect end user QoE of streamed video in a wireless environment. Below is a list of main factors that influence end user QoE of streamed video and a range to consider for testing. A judicious selection of test cases based on these factors was necessary.

Device and Media Factors

- **Device** – The size of screen has a significant impact of impairments in the video quality and hence on QoE. Both smartphone and tablet sized screens were used in the lab experiment.
- **HAS Technology** – At the time of the experiment, most common widely deployed version of HAS technology for wireless was Apple HTTP Live Streaming (HLS). This is natively supported on iOS devices. Content shall be in the clear.
- **Content Type** – Some content is more difficult to encode (high motion and chaotic scenes) or impairments are more easily noticeable (e.g. slow panning shots). The content has to be free from copyrights and permission to edit.
- **Encoding Rate** – Suitable for both tablets and smartphones.

Radio Factors

- **Fading** – Effects of fading at mobility of 0.3Km/h and 30 Km/h
- **Round Trip Time** – Controlled round trip delay (20msecs, 100msecs) added to the SGi interface
- **Interference/Noise** – SINR both fixed (0, 2, 5 and 10 dB) and variable (increasing from 0 to 30 dB in 1 dB (per 2 sec) steps, decreasing from 30 to 0 dB in 1 dB (per 2 sec) steps, decreasing from 15 to 0 and increasing back to 15 dB in 1 dB (per 2 sec) steps)
- **Competing traffic** – Competing best effort FTP transfers of 5, 10, or 20 users

4.1.2 Radio Scenarios

It is clear from the previous section that there are a number of factors that can influence video QoE, and performing tests to cover every combination of these factors would lead to an extremely large number of radio scenarios. In order to keep the testing time reasonable, and keep the number of MOS testers required to provide statistical significance in the MOS results reasonable, it was decided that the number of radio scenarios should be limited to reduce the number of mass testers. This was accomplished by limiting the test devices to one smartphone and one tablet (both simulated on an Apple iMac).

The important parameters affecting QoE were identified as device type, content type, congestion, SINR, fading and RTT. Setting these parameters to values typical for wireless environments resulted in 168 potential scenarios:

- Device type (2): smartphone and tablet
- Content (2): low and high intensity (see below for details)
- Competing traffic (3): 5, 10 or 20 competing best effort FTP users
• Combined fading and SINR profiles (7)
  • Fading of 0.3 km/h combined with 4 fixed SINR profiles of 0, 2, 5 and 10 dB
  • Fading of 30 km/h combined with 3 variable SINR profiles: 30 to 0 dB; 0 to 30 dB and 15 to 0 to 15 dB all changing in steps of 1dB every two seconds
  • RTT (2) – 20 ms and 100 ms

Since testing every combination required a very large set of test subjects, we limited the number to 112 during the lab test and further reduced to 84 for the mass test. This latter reduction was because the same radio scenario was used for both iPhone and iPad.

4.1.3 Information Collected
HTTP GET requests and TCP PCAP files were collected from the Web Server Access Logs (Apache). This is used to create corresponding HAS profiles. From these the TCP performance could be determined and compared with the corresponding HAS profiles.

Note that 3GPP specification 26.247 defines several additional metrics including playout delay and buffer level. Jitter in playout is well researched and it is universally agreed that it should be avoided [6]. Therefore it was assumed that any professional mobile video offering will be able to maintain at least the lowest video quality.

During the tests LTE signalling between the UE (dongle) and EPC (EAST LTE simulator) set up a default bearer only with the QCI=9 (best effort traffic). No eNodeB scheduler related information was captured since competing FTP traffic was also using the same QCI value.

4.1.4 Content Selection
The content has a significant impact on the end user being able to detect imperfections. Firstly, the content has to be engaging to the viewers. For example a company promotion may not hold the attention of a young teenager.

Scenes with chaotic motion (e.g. waves and grass moving in the wind) are particularly difficult for digital encoding. Also high motion requires more bandwidth for an acceptable quality. Talking heads such as news program are generally easier to encode. However, hard edges and line – especially at 45 degrees – require more effort for a good encoding.

Imperfections in encoding are easier to spot in steady or slow pan shots. Whereas the video quality can be significantly lower when there are rapid scene changes.

A final consideration when selecting content is its legality. The content source should be as high a quality as possible. However, such content is often covered by copyright and other legal limitations.

For this test we selected two clips of nominally two minute’s length from Sintel www.sintel.org: the first one from the opening sequence panning over a mountainscape and the second one of a high motion clip which includes a chase through a market. The content was encoded in 6 levels (128, 210, 500, 1080, 2160, 4320).
350, 545, 876 and 1440 kbps) AVC and 64kbps stereo audio. The overall QoE depended on both the audio and video quality.

4.1.5 Viewing Sessions for Calibration

20 viewing sessions of one video viewer were conducted as part of calibration testing. During this phase scenarios with extremes such as 0 and 10 dB while skipping, and 2 and 5 dB for fixed SINR impairment were chosen.

The test condition sequence and order were arranged so that any effects on the grading, tiredness or viewer’s adaptation were balanced out from presentation to presentation. Some of the presentations were repeated from session to session to check coherence.

To minimise bias between the different sessions, the following script was used

“Good morning/afternoon and thank you for coming.

In this experiment, you will view some video clips on a MAC screen sized either for a smartphone or a tablet. The app will ask you to view a number of short video sequences. After each clip has finished, you will be asked to judge its quality by using one of the five levels of the following scale:

5 - Excellent
4 - Good
3 - Fair
2 - Poor
1 – Bad

Observe carefully the entire video sequence before making your judgment.”

The results were entered into an excel sheet for processing.

![Viewing Session for one HAS Solution Diagram](image)

- **P** = Pristine Content (pre HAS Encoding with no Impairment)
- **C** = Conditioned Content (post HAS Encoding with or without Impairment)
4.2 Preparation for Mass Test

4.2.1 Creation of HAS Profiles
HAS profiles for each of the 84 radio scenarios were identified as follows. The Propsim F8 was set to introduce the relevant network impairments, the NetHawk East T500 was configured to inject the background traffic and the HAS client in the iMac was launched. The corresponding HAS profile was extracted from the “Access Log” on the HAS Server. All of the 84 scenarios were run an average of eight times and a typical HAS profile for each scenario was selected – see section 4.2.2.

After this initial analysis, it was clear that in the set of typical HAS profiles some were similar, so there was no need to test all typical HAS profiles associated with all scenarios. The pruning of the set of typical profiles allowed us to introduce some synthetic profiles, i.e., HAS profiles that did not show up while using the Apple HLS client in the test network, but could occur with other RDAs. In total this resulted in 90 HAS profiles to be tested: 38 profiles (16 typical profiles and 22 synthetic ones) for clip 1, and 52 profiles (40 typical profiles and 12 synthetic ones) for clip 2.

Based on each HAS profile, a video file was created. This was done by concatenating the video chunks as indicated in the HAS profile. For example if the HAS profile was 3 4 1 ... segment 1 in quality 3, segment 2 in quality 4, segment 3 in quality 1, ... were concatenated.

The resulting 90 files were grouped into 20 packages of five clips each (18 packages of unique clips and two packages with clips already in other packages). The NGMN partners recruited some 500 volunteers prepared to take part in a brief survey and rate the clips of one package. The volunteers all had to have access to either an iPhone or iPad. A package was allocated to each volunteer as they signed up. They were asked to download the package as a PodCast and view the five clips using the built-in video app. This procedure yielded a MOS value per HAS profile (see section 5.4 and [15] for details).

4.2.2 Details of creating a typical HAS profile
The 84 radio scenarios were run an average of eight times. Each run is an instance of the scenario and can result in a different HAS profile. Therefore with each radio scenario corresponds a set (i.e., a cloud in the vector space) of possible HAS profiles. Based on the PCAP files the goodput⁴ that the client perceived during the run to assess the correlation of the goodput trace and the HAS profile was also investigated.

Some of these clouds of HAS profiles were very confined (in the vector space of HAS profiles), some cases resulted in each run having exactly the same HAS profile. For other scenarios, although the runs had a very similar TCP goodput and the resulting HAS profiles were quite different (Figure 5). In such cases the RDA decisions were very sensitive to very small changes in goodput and an extended cloud of HAS profiles resulted. In other scenarios although the goodput of the different runs was quite different, a confined cloud of possible HAS profiles resulted (Figure 6). The conclusion is that

---

⁴ Goodput is a measure of the TCP bandwidth and hence indicates what could be available to the application (HAS stream in this case)
depending on the scenarios (i.e., set of radio parameters) a cloud of HAS profiles results which can be either confined or extended; and how goodput changes from run to run does not give a clear indication for the size of the clouds.

Figure 5: Radio Scenario 82:  (top) goodput profiles for the different runs; (bottom) HAS profiles for the corresponding runs

Figure 6: Scenario 2:   (top) goodput profiles for the different runs; (bottom) HAS profiles for the corresponding runs
For any radio scenario, the corresponding HAS profile was determined using a distance squared method. The HAS profile with the minimum distance to the other HAS profiles (discarding outlier HAS profiles) was selected at the typical profile. For some profiles, this was quite easy to do. For others like the ones in Figure 6 sometimes an arbitrary choice was needed. Indeed, the average distance squared value was less than eight for 42% of the radio scenarios and only larger than 30 for 9% of radio scenarios.

4.3 Mass Test

As it is well known that reliability of test subjects is an issue in a crowd sourcing approach test users were mostly recruited from employees of partnering institutions of the NGMN P-SERQU project. Some test users were also recruited via Facebook and from universities. Using ‘friendly’ and ‘technically oriented’ volunteer subjects reliability is expected to be less of an issue compared to common crowd sourcing approaches where literally the whole internet population is permitted to participate in the test. To ensure that the users experienced the desired test conditions, the video profiles (i.e. videos including quality adaptations) were prepared offline and these videos were offered as Podcasts. In order to participate the test user had to download a Podcast containing five different video profiles onto their iPhone or iPad. During the download phase, a personal data questionnaire, including consistency questions, was completed by the participant. After that, the user had to click a button for starting the test, which appeared upon download completion. Then the user sequentially viewed all video clips. After the playback of each video, the user was asked to rate his or her current personal satisfaction with the video quality on a 5-point Absolute Category Reference (ACR) scale (see section 5.1).

As described in section 4.2, the team created videos comprised of different chunked sequences at different encoded bit-rates. These chunk sequences represented HTTP Adaptive Streaming conditions possibly experienced by clients of different mobile environments. The goal was to measure a statistically significant population of human mean opinion scores (MOS) but to avoid unintended interference from irrelevant factors. The factors that were taken into account were:

1. Respondent age, gender, previous digital video experience
2. Systematic errors from a “learning” effect as more videos were viewed
3. Variable errors from network conditions affecting the test conditions
4. Variable errors by misinterpreting screen differences as video differences
5. A friendly survey respondent audience

These considerations led to several requirements:

1. A survey had to collect respondent data and device data, and be easily readable on the device
2. The videos had to be watchable on the user device without any buffering delay in the test
3. The team needed a way to invite the respondents
4. The respondents had to be given a defined group (package) of videos to watch, and the group ID had to be recorded.
There were two phases to the Mass Test. Firstly a survey using Survey Monkey was created and volunteers were recruited through NGMN (see section 4.3.1). As the number of respondents was not sufficient for good statistical results, a web portal was created and the test was made available through the Internet using social media to promote (see section 4.3.2).

4.3.1 Phase 1 – Survey Monkey based approach

4.3.1.1 Video and survey delivery design
Several methods were considered in delivering the video viewing experience: (1) video delivery through a web page, (2) video delivery through downloaded ZIP archive files, (3) video delivery through a podcast, (4) video delivery through a custom web site, and (5) video delivery through a custom app, (but this was not attempted because of the limited resources). Several methods were considered in associating user IDs, invitations, video package ID, and survey response, including (1) web-based surveys and (2) e-mail based surveys.

Details of the survey and invitation letters can be found in Appendix 8.1

4.3.1.2 Volunteer management
To conduct the video delivery and survey, NGMN first sent invitations by email to a list of known colleagues and other volunteers. When a response was received, a single registrar recorded the email address and assigned a user ID and video package ID. Based on the video package ID, one of 20 different instruction letters was sent to that volunteer. It was found to be more convenient to cycle package IDs across different volunteers using Microsoft Outlook than by using the built-in mailer provided by Survey Monkey. Volunteers would be instructed to subscribe to a specific linked podcast (which required the ability to embed URLs in the email), and they would be further instructed to watch videos and record responses in the survey. Volunteer demographics and other response data are described elsewhere in this report.

4.3.2 Phase 2 - NGMN project partner hosted website
The main part of the test regarding the download of videos, rating scales etc. was based on the given infrastructure and questionnaires as described in section (cf. 4.3.1), with the questionnaire forms being hosted on a NGMN project partner server and not on the Survey Monkey server. The main difference was the initiation of the user involvement: While in phase one the user had to send an e-mail expressing his interest for participation to the test supervisor which allocated a certain video package to the respective user and sent the corresponding link via e-mail to him, in phase two the video package selection was automatic so that the user only had to click a link in the invitation e-mail which forwarded him / her to the start page of the video quality survey. On this page the user got the link to the video package to download (packages were automatically selected such that packages having the least number of ratings were used first) as well as the instructions to take the test, and subsequently the questionnaires.

Details of the portal and invitation letter can be found in Appendix 8.2
4.3.3 Summary of the Mass Test Process

Figure 7 shows the split in respondents between male and female. The bias towards male respondents reflects the way volunteers were recruited through work contacts.

![Figure 7: Gender of Respondents](image)

Figure 7 shows the split in respondents between male and female. The bias towards male respondents reflects the way volunteers were recruited through work contacts.

Figure 8 shows the age and occupation split between respondents. This shows a reasonably well balanced response from 21 to 65. The dominance of students in the lower age categories is at least explained by the push in universities. As we recruited through work contacts, this explains the dominance of employed people.

![Figure 8: Age Range and Employment of Respondents](image)
Figure 9 shows the number of hours each day people use the Internet for all purposes with between 1 and 5 hours dominating. However, there is a good fraction of people (male and female) that are using the Internet for 7 hours or more on a daily basis.

![Figure 9: Daily Internet Use (Hours)](image)

86% of the respondents said they watched short form video (<10 mins) over the Internet. YouTube was the almost universal response except for the Chinese respondents who often used YouKu. However, only 25% of the respondents watched long form content over the Internet. For those who did, the favourite service largely depends on country, with BBC iPlayer, Netflix, Hulu and Amazon frequently mentioned.

Figure 10 shows the number of responses per package run on the iPhone and iPad respectively.

![Figure 10: Rating distribution across video packages](image)
For each of the 90 videos (corresponding with the 90 HAS profiles) mentioned in Section 4.2.1 we obtained between 11 and 16 subjective scores using a tablet (iPad) and between 4 and 10 scores using a smartphone (iPhone). The number of scores is disappointingly low, especially for the smartphone, and this limits the accuracy that can be obtained. The average (taken over the 90 profiles and two devices) of the standard deviation of the scores per profile is about $\sigma_S=0.84$. Taking into account the number of scores obtained per profile the error in the subjective MOS ($\sigma_S/\sqrt{N}$, with N being the number of scores per profile) was estimated on average to be 0.24 for the tablet, and 0.32 for the smartphone. These numbers give an indication of the accuracy that can be obtained with any model, as the accuracy will be a combination of the error in the MOS model and the error in the MOS values from the subjective tests.

![Stdev versus average QL (1 dot per profile)](image)

**Figure 11: Profile description in the average QL ($\mu$) and stdev QL ($\sigma$) space**
Figure 12: Profile description in the stdev QL and frequency of QL switches

Figure 11 and Figure 12 show that the 90 profiles are well spread over the $(\mu, \sigma, \phi)$ space, where $\mu$ is the average quality level across all chunks, $\sigma$ is standard deviation of quality level and $\phi$ is the number of changes in quality level. However, in the $\phi$ dimension the spread is relatively low, but this is consistent with the HAS profiles captured during the lab tests. The profiles with a larger $\phi$ are part of the synthetic profiles that were added.

5 Analyses

One of the original project objectives was to develop a correlation between radio and core network QoS KPIs and predicted QoE. The primary reason for that objective was to provide mobile network operators a mechanism to predict QoE based on the QoS measurements they are making using existing tools in their network. However, throughout the study it was realized that instead of making a one-to-one mapping between generic radio and core network QoS and QoE, it is simpler and more accurate to measure metrics from the HAS bit-stream and make a prediction on QoE. The biggest benefits of this approach are to simplify the measurement framework and to remove the need to have a reference video sample. Instead it would be sufficient enough to collect HTTP header information on the server or at any point between the terminal and the server.

In the course of the analysis, it was decided to include intrusive objective measurement methods that do require a reference video stream. The primary reason for this was to establish a benchmark point to measure the performance of HTTP-header based QoE predictor against the objective analysis methods included in the study. Furthermore a recent ITU-T non-intrusive objective measurement method was included for benchmarking.
5.1 Introduction to Individual Predictors

There are many approaches to determining the QoE of a video. Video QoE assessment methods can be classified as either subjective methods or objective methods. Objective methods can be grouped as intrusive or non-intrusive. Furthermore, non-intrusive methods can be signal-based or parametric-based. Figure 13 describes the taxonomy of various video quality assessment methods. It was adopted from [25] where an identical taxonomy was introduced for VoIP quality assessment methods.

![Video QoE Measurement Taxonomy](image)

ITU-T specifications P.910 [15] and P.911 [16] describe subjective video and audiovisual quality measurement methods, respectively. Both of these specifications’ approach is to get a group of people to rate the quality. While this is subjective, it is reasonably accurate when a sample of 15 or more people is used to rate the same content under the same conditions. Here a number of viewers are asked to rate the quality against using the following Absolute Category Rating (ACR) scheme (Figure 14). The Mean Opinion Score (MOS) is formed from the average of the individual ratings.
The problem with MOS and subjective schemes in general is that they are not automatic and require many viewers for accurate results. Hence objective schemes are preferred for automatic monitoring of video streams. Objective measurements can be done intrusively by comparing the reference video stream with the degraded video stream or non-intrusively using quality parameters extracted from the bit stream carrying the video stream to the receiver or the degraded video stream itself (signal) without using the reference video stream. This section introduces some of the main objective schemes.

Objective quality assessment simulates the opinions of human testers using computational models. Subjective methods are used to calibrate objective methods. The accuracy of objective methods is determined by the correlation with the subjective MOS scores. As discussed later, the accuracy of various objective methods in the analysis were evaluated against the subjective results collected from the mass tests.

Multimedia quality assessment models can be described in terms of the level at which input information is extracted. For example, information may stem from protocol headers or payload-related bit stream data. In general, models can be categorised into planning, parametric, signal-based, bit stream-based, or hybrid models, as detailed in the following sections. In addition, the models can be classified according to the amount of information they need from the original signal into no reference (NR), reduced reference (RR), and full reference (FR) models. FR models have access to the original source sequence, which is compared with the processed sequence. RR models use the processed sequence together with a set of parameters extracted from the source sequence. NR models calculate quality predictions based only on the processed sequence.

Signal-based models exploit the decoded signal as input to calculate a quality score. Many of these models include aspects of human perception. Several ITU recommendations for signal-based models exist. In the video quality domain, J.144 [17] specifies models for FR SDTV quality assessment, J.341 [18] for FR HDTV quality assessment, and J.246 [19] and J.247 [20], for RR and FR video quality assessment, respectively. Moreover, two popular signal-based quality metrics which mainly stem from the image quality domain are the Peak Signal to Noise Ratio (PSNR), and Structure Similarity

<table>
<thead>
<tr>
<th>Score</th>
<th>Quality</th>
<th>Impairment</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Excellent</td>
<td>Imperceptible</td>
</tr>
<tr>
<td>4</td>
<td>Good</td>
<td>Perceptible but not annoying</td>
</tr>
<tr>
<td>3</td>
<td>Fair</td>
<td>Slightly annoying</td>
</tr>
<tr>
<td>2</td>
<td>Poor</td>
<td>Annoying</td>
</tr>
<tr>
<td>1</td>
<td>Bad</td>
<td>Very annoying</td>
</tr>
</tbody>
</table>

**Figure 14: ACR category for MOS**

Service Quality Definition and Measurement,
Version 1.0.4, 27th August 2013
Index (SSIM). In the following sections, a brief introduction to the quality metrics employed in this project is provided. All of these models are intrusive measurement methods since they need access to the original reference video stream.

Non-intrusive video quality assessment models have the distinct advantage of not needing access to the original video stream. Recent ITU activities have resulted in the development of two parametric non-intrusive video quality assessment models. P.1201.1 [22] and P.1201.2 [22] use packet header information for generating video quality assessment of lower and higher resolution video streams, respectively. Another recent ITU recommendation P.1202 [23] uses bitstream to compute video quality non-intrusively. In this study we used P.1201.2 [22] as a parameter-based non-intrusive quality assessment model.

During the course of the study, it was shown that knowing just the HAS profile provided an accurate way to predict the quality of a video stream. This only requires monitoring of the HTTP GET requests and hence is non-intrusive. It is also very easy to process a large number of streams.

5.1.1 Peak Signal to Noise Ratio (PSNR)
PSNR is a calculation of the Mean Square Error (MSE) between the original and decoded image. It is popular because it is simple to compute, parameter-free and memory-less, and as a result, it can be calculated locally without consideration of other source samples. Nevertheless, it is often criticized due to its poor correlation with subjective perceived quality due to the fact that it does not take into account the properties of the human visual system.

Further details of PSNR can be found in appendix 9.1

5.1.2 Structural Similarity (SSIM)
Since human visual perception is highly adapted for extracting structural information from an image, SSIM was introduced for image/video quality assessment. It is based on the degradation of structural information and it requires an initial uncompressed or distortion-free video as reference, therefore is a full-reference signal-based video quality metric. It follows a picture-based engineering approach, which is based on the extraction and analysis of certain features or artefacts in the video.

Further details can be found in appendix 9.2

5.1.3 J.144 (VQM)
The Rec. J.144, also known as Video Quality Metric (VQM), is a reduced-reference metric that combines different parameters using linear models to produce estimates of video quality that closely approximate subjective scores. It divides sequences into spatio-temporal blocks, and a number of features measuring the amount and orientation of activity in each of these blocks are computed from the spatial luminance gradient. The features extracted from test and reference videos are then compared using a process similar to masking.

Further details can be found in appendix 9.3
5.1.4 J.341 (VQUAD)
The Rec. J.341, also known as VQuad, is a full-reference picture-based metric which improves and extends several detectors for individual degradation. It is more robust to the latest coding technologies and less content dependent. It also provides a set of additional results giving more details about the type of distortions that came up during the analysis of the video sequences. Therefore an easier interpretation and localisation of potential problems in quality estimation is available.

Further details can be found in appendix 9.4

5.1.5 ITU-Recommendation P.1201.2
Many existing methodologies for assessing video quality of a signal are full reference or at best reduced reference. This makes them less than ideal for monitoring a large number of video streams. ITU-T P.1201.2 overcame this problem by providing a no-reference approach suitable for IP delivered content which only examines information contained in packet headers (e.g. MPEG-2 TS/RTP/UDP/IP).

Further details can be found in appendix 9.5

5.1.6 HAS Parameters
As described earlier, HTTP Adaptive Streaming relies on coding the video stream in six different quality levels, each corresponding to a specific bitstream rate. Each chunk of the video stream is transmitted at a certain quality level. Three parameters can be used as the basis of a parameter-based non-intrusive video quality assessment model:

1. Nominal Bit rate of each chunk in Mbps
2. MOS score of each chunk as measured from constant quality HAS profiles
3. Quality level (1-6) with the assumption that the content is encoded with constant change in MOS score between quality levels

Usage of these three parameters as the basis of a MOS predictor is described in section 5.4.
5.2 Model Assessment Metrics

A prediction can be measured accurately by using a model. Three metrics are used for doing this – Root Mean Square Error (RMSE Section 5.2.1), Pearson Correlation Coefficient (PCC section 5.2.2) and Spearman Rank Order Correlation Coefficient (SROCC section 5.2.3). These metrics were used to calculate the accuracy of the prediction models introduced in section 5.1 against the results of the mass test we described in section 4.3.

5.2.1 Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) is a frequently used metric to give the error of a model. It measures the differences between values predicted by a model or an estimator and the values actually observed. These individual differences are called residuals when the calculations are performed over the data sample that was used for estimation, and are called prediction errors when computed out-of-sample. A small value of RMSE is an indicator that the model is a good predictor.

\[ \text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \]

5.2.2 Pearson Correlation Coefficient (PCC)

The correlation between the test results and the predicted values from the model is a useful indicator of the stability of the model when applied to different values. The Pearson Correlation Coefficient (PCC) is defined as the covariance of the two variables divided by the product of their standard deviations. A strong correlation is close to 1 or to -1.

\[ P_{xy} = \frac{\text{cov}(x, y)}{\sigma(x) \cdot \sigma(y)} \]

5.2.3 Spearman Rank Order Correlation Coefficient (SROCC)

Spearman Rank Order Correlation Coefficient (SROCC) is a non-parametric measure of statistical dependence between two variables. It is the PCC between ranked variables. For a sample of n pairs \( X,Y \) are converted into ranks \( x_i,y_i \) and SROCC (\( \rho \)) calculated

\[ \rho = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 \cdot \sum_{i=1}^{n} (y_i - \bar{y})^2}}. \]

A value of \( \rho \) close to 1 or -1 indicates a strong SROCC. And a value of 1 means a monotonic relationship where an increase in \( X \) (the MOS from the mass test) results in an increase in \( Y \) (the predicted value).
5.3 Applying PSNR and SSIM

5.3.1 Test setup

Figure 15 shows the test setup to obtain the PSNR and SSIM values for each frame of the test clips. They were obtained using the video distortion analysis function offered by the R&S®VTE Video Tester device or the R&S®VTC Video Test Center:

![Test Setup Diagram]

**Figure 15: Test Setup**

5.3.2 Analysis using PSNR

The simplest and quite natural temporal pooling approach is to calculate the mean value of the objective metric output over all frames of the video clip. The user can then calculate the correlation between these mean values and the MOS scores gathered in the crowd sourcing experiment. After a quick look at the mean values of PSNR for different profiles evaluated for the two clips (see Figure 16), it was obvious that the mean PSNR values for all the profiles of the two clips are in a different range: For Clip 1, the mean PSNR overall quality profiles was in the range [34.7, 41.2] dB while for Clip 2, the mean values were in the interval [30.7, 37.0] dB. This is caused by the different spatio-temporal characteristics of the two clips.

As a result, the mean objective metric values, as well as outputs of other temporal pooling algorithms, cannot be directly mapped to MOS scores for the two sequences at once.

Assuming that the encoder is optimised for perceptual quality and that the perceptual quality at level 6 (cf. section 4.2) can be considered as Excellent (5) on the MOS scale, the difference in objective metric values can be compensated by subtracting the mean values of the level 6 sequence from all per-frame PSNR values for all sequences. The input to the temporal pooling algorithms can then be expressed as *objective metric value with respect to mean value of highest available quality in the sequence* as shown in Figure 17.
Figure 16: MOS vs PSNR for all profiles Clip 1 (blue), Clip 2 (red).

Figure 17: MOS vs dPSNR for all profiles Clip 1 (blue), Clip 2 (red)
5.3.3 Analysis using SSIM

5.3.3.1 Objective quality distribution within profiles

The focus of this chapter is to compare the given specification of quality levels (QL) for the individual profile chunks with the objective video quality results.

The profiles used for clip 1 and clip 2 are sorted according to the MOS ratings from the mass test, i.e. the first coloured row in Figure 18 and Figure 19 represent the profile with the highest average MOS rating, followed by the second highest MOS rating and so on. The columns represent the video quality of each chunk (0...19 for clip1 and 1...22 for clip2).

In order to ease the analysis, the results – either quality level (QL 1 to 6 according to the coding rate used for that specific chunk) or average SSIM – have been colour-coded from green (best value for QL or SSIM) to red (lowest value for QL or SSIM). The objective video quality focus is restricted to SSIM results, because as already discussed SSIM presents better correlation with human perception than PSNR values.

Figure 18: QL vs. average SSIM values for sorted clip1 profiles (from good to bad MOS)
5.3.3.2 Correlation

The focus of this chapter is to analyse the correlation of the objective video quality results with the subjective MOS rating taken from the mass test. In the first place, this requires deriving a single representative value for the whole profile from the individual SSIM results of each chunk. From the manifold pooling algorithms possible, the classic average calculation was chosen. Thus, no training to either the properties of clip1 or clip2 was applied.

The Pearson Correlation Coefficient (PCC) between subjective MOS and average SSIM is calculated to 0.880 (clip1) and 0.869 (clip2).

Running the same analysis for averaged QL values results in 0.861 (clip 1) and 0.867 (clip 2).

Figure 20 and Figure 21 represent this analysis graphically.
5.3.4 Summary

The objective SSIM analysis in section 5.3.3.1 affirms with regards to the content selection criteria in section 4.1.4, the applicability of the chosen test clips for mass test evaluation. The video content of certain chunks is complex enough to get impaired as intended when coded according to the related QL assignment.

Service Quality Definition and Measurement,
Version 1.0.4, 27th August 2013
Section 5.3.3.2 validates a high correlation between subjective MOS and average SSIM values independent of the clip content. Thus, objective video quality measurements can provide similar results as an alternatively time consuming subjective mass test, providing a reference rating for the test clips at the push of a button. This allows faster validation of simpler prediction algorithms and trimming of its input values. The latter is beneficial with regards to high correlation outcome of the straight forward algorithm based on mean QL calculation, since its key to success lies in the initial finding of a QL level assignment proportionally to the true video quality. The inferior result when evaluating the mean bitrates instead shows the influence of a suboptimal mapping.

5.4 Linear MOS predictor

From previous observations, it was noted that the QoE of video streaming depends on both the average of the quality levels and on change in quality level. Therefore, subjective MOS scores from the mass test were used to validate a linear model combining the average quality level, standard deviation in quality level and in the number of quality level changes.

$$M_{\text{pred}} = \alpha \cdot \mu - \beta \cdot \sigma - \gamma \cdot \phi + \delta$$

Where $\alpha$, $\beta$, $\gamma$, and $\delta$ are tuneable parameters and $\mu$ is the average quality level across the chunks, $\sigma$ is standard deviation of quality level and $\phi$ is the number of changes in quality level.

Five measures of quality level of each chunk were used:

1. Nominal Bit Rate (labelled BR) of each chunk in Mbps
2. MOS score of each Chunk (labelled CM) as measured from constant quality HAS profiles
3. Quality Level (1-6) (labelled QL) with the assumption that the content is encoded with constant change in MOS score between quality levels
4. PSNR of each chunk
5. SSIM of each chunk

Using the data set of 90 HAS profiles and corresponding MOS score, the RMSE (Root Mean Square Error) was used to tune $\alpha$, $\beta$, $\gamma$, and $\delta$ per device and clip pair. As part of the sensitivity analysis, the results were divided into a training set and a test set per device and clip pair. The training set was used to produce the optimal values (minimum RMSE) for the tuneable parameters. These were then used to test the error when applied to the test set. As well as using RMSE to train the model parameters, the Pearson Correlation Coefficient (PCC) and Spearman Rank Order Correlation Coefficients (SROCC) were also checked.

5.4.1 Parameter tuning

The parameters of each of the models have been trained by minimizing the RMSE over the subset of profiles corresponding with one clip and one device. This allows comparing the different models optimized for each device and clip pair.
The results for the RMSE are shown in Figure 22. This figure shows that when the model parameters are optimised for each device and clip pair the chunk-MOS model gives the best performance for the tablet for both clips and for the smartphone and clip 1. The SSIM model gives the best performance for the smartphone and clip 2. These conclusions are valid for all three metrics: RMSE, PCC and SROCC. The results for PCC and SROCC are shown in Figure 22 and Figure 23. The SSIM model is always better than the PSNR model. The QL model is inferior to the chunk-MOS model (which is obvious, because QL model is a special case of chunk-MOS model). The BR model gives the worst performance. This can be explained by the fact that the relation between bit rate and MOS cannot be accurately modeled by a linear relation. A modified BR model where a non-linear transformation of the bit rates is used will give better results.

When analysing the obtained results it was also observed that the impact of the profile metric $\phi$ (number of QL switches) is small, hence $\gamma$ can be set equal to 0. This is a general conclusion across all models, clips and devices. Consequently, it can be concluded that the number of bit rate switches is not a key parameter to predict the MOS for the profile domain that was investigated (as long as the standard deviation is taken into account). Investigating the space of profiles with a much larger number of switches (profiles not seen frequently in the lab test) may result in a different conclusion.

The RMSE values for the chunk-MOS models are close to the estimated error in the subjective MOS values of the dataset. This gives an indication that the chunk-MOS model is quite good, and that the main contribution to the RMSE stems from the noisy subjective MOS values. It was also expected that with a dataset that has many more scores per video, the RMSE of the chunk-MOS model will be lower than the RMSE values obtained here.
This shows that the linear algorithm is a good predictor of QoE. The standard error for tablets is better than 0.3 MOS-points when using the actual MOS of each chunk. When this is not possible, using the HAS video quality level is a good proxy. The bit rate is not useful because there is no linear mapping to quality. While PSNR and SSIM show good response, the parameters are very sensitive to the content. Hence parameters trained for clip 1 do not work very well on clip 2 and vice versa. The error in the iPhone results is higher because of the low number of responses.

5.4.2 Sensitivity Analysis
It is important to know how sensitive the model is. The model parameter tuning was done based on the training set data, and the performance was then calculated for both the training set and the test set (all profiles except those of the training set). The size of the training set was varied between 11
and 35 (for clip 1) and 11 and 48 (for clip 2). For each size (T) of a training set, 30 different training sets were randomly picked from the complete set and the rest of profiles were used as test set. The performance was averaged over the 30 instances. This analysis was done for all (clip, device) combinations and qualitatively very similar results were obtained.

The parameters trained on one clip/device pair to the other three clip/device pairs was then applied to understand how good a predictor of MOS is when used on other content. As expected, using the MOS of each chunk, the sensitivity was least with RMSE less than 0.5. This rises to a maximum of 0.75 when the quality level (QL) is used. For a compromise setting of the model parameters, the RMSE was less than 0.39 and 0.55 respectively.

The PSNR and SSIM models were both very sensitive to the model parameters. As an example, using the optimal SSIM model parameters for clip 1 on a tablet device for clip 2, also on a tablet device (for which it was not tuned) yields an RMSE of larger than 2. In fact the predictions of MOS do not even stay within the [1, 5] range. The reason for this poor performance of the PSNR and SSIM models (despite sometimes having the best performance for the optimal parameter set), is that these measures (PSNR, SSIM) are not good indicators for absolute quality. PSNR and SSIM give a good assessment of relative quality of different encodings of the same video, but are not good in assessing the absolute quality of different videos. Full analysis can be found in [15].

5.4.3 Summary

The chunk-MOS model has proven to be a good MOS estimator resulting in an accuracy of about 0.3 MOS-points for tablet and 0.4 MOS-points for smartphone. The QL model is also a good MOS estimator, and is equivalent with the chunk-MOS model when the quality levels have been encoded at equidistant MOS. The accuracy of the model is currently (with the available dataset) mainly limited by the error (uncertainty) of the obtained subjective MOS values in the dataset which are of similar order. The model itself has not yet reached its limits of accuracy. It is hard to predict the lower boundary of accuracy that could be reached with less noisy data, but we expect that considerable further improvement is possible.

The chunk-MOS model is well positioned to be used in a video QoE assessment module that, by inspecting HTTP requests for HAS content, predicts the quality delivered to the user. The model can be used for individual users, but can also be the basis to give a quality indicator per cell or per larger geography.

5.5 Spatial Temporal Pooling

5.5.1 Overview

One unique feature of HAS content is abrupt changes in quality level. Many of the existing quality assessment techniques work well for determining the QoE of content encoded at constant quality. These allow us to determine the QoE (MOS) of each quality level. However, a HAS stream can have several sudden and significant changes in quality. Therefore, a model which pooled the MOS values from the individual chunks into a single MOS value for the entire clip is needed. This is known as
temporal pooling. There are several temporal pooling algorithms, from simple approaches which consider just the maximum, minimum, or mean video quality, to those which consider how recent high quality or low quality was experienced. A full list of approaches can be found in Appendix 10.

5.5.2 Performance Comparison
As described above, most of the metrics require one or more input parameters to account for recency effects and/or to emphasise low quality which both have a higher impact on subjective quality. But no a priori limitation of the parameters is suitable because it is the aim of the experiment to examine the usability of each temporal pooling algorithm for adaptive video streaming. Thus, all the parametric temporal pooling algorithms are tested for a number of parameter values. This assumption leads to an optimisation problem: the optimal parameter values for a given temporal pooling algorithm which maximise the correlation of the temporal pooling output with the subjective scores needed to be found.

As the dataset is limited to two video clips only, the optimisation needs to be cross-validated in order to make sure that the pooling algorithm is performing well independently of the training set (used to adjust the parameters) and the evaluation set (used to check the performance of the trained algorithm) selection. For cross-validation, the “leave one out” algorithm was used, which is known to be the most exhaustive among all cross-validation algorithms. Its principle is such that during the training, one sample is removed from the whole set of sequences and the training is done on the rest. Then, the output for the single left out sample is evaluated. Using this approach as many times as there are samples in the whole dataset, researchers get the same number of cross-validated outputs as there are samples in total. A good temporal pooling algorithm should exhibit high correlation of the cross-validated values with the MOS scores for the corresponding profiles.

In the training phase, two different optimisation criteria were used - the Pearson Correlation Coefficient (PCC) and Spearman Rank Order Correlation Coefficient (SROCC), which were also used in the evaluation to assess the performance of the different temporal pooling algorithms.

The results of the training with cross-validation are given in Table 1 for temporal pooling of per-frame PSNR and SSIM values. The best performing methods for both PSNR and SSIM were Percentile, Mean, VQA, SequenceLevel, and Hysteresis.

With Minkowski and SoftMax pooling the special case exists that they can be identical to Mean pooling for certain parametrisations and thus could yield better results, (which can be seen in the Mean row). However, to keep the specific characteristics of the methods visible this special case was excluded in the table. Also with LocalMinimum the special case was excluded so that the minimum of the mean of all successive frames was the mean of the last frames. Thus, also LocalMinimum could yield better results, which can be seen in the MeanLastFrames rows.

Finally, Table 2 presents the result achieved when quality levels or MOS values of constant quality levels (i.e. MOS of constant level profiles) were used as inputs to the temporal pooling.
For the temporal pooling in this case, only the scheme of quality level switching along all the profiles and optionally the six representative MOS values were required. The results are comparable to those achieved from temporal pooling of per-frame objective scores, in some cases even better. Only KMeans pooling failed to classify the input of quality levels probably because of the very small number of distinct values. In addition the worst PSNR and SSIM pooling methods performed quite well with quality level and constant quality level MOS pooling and more trivial metrics also outperformed the pooling of chunk bitrates which only reached correlations up to 0.79 for the best methods. It is worth mentioning that simple mean pooling of quality levels is one of the best performing methods although it is the most general approach which includes almost no information about the underlying content. As the output of pooling of constant quality level MOS score is on the same scale as the subjective ratings, the scatter plot diagram representing the SequenceLevel pooling, reaching the highest correlation in this scenario is represented in Figure 25. It illustrates that a good performance can be achieved by temporal pooling for all 90 quality profiles without any extreme outliers.

<table>
<thead>
<tr>
<th>Pooling algorithm</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCC</td>
<td>SROCC</td>
</tr>
<tr>
<td>Percentile</td>
<td>0.854</td>
<td>0.835</td>
</tr>
<tr>
<td>Mean</td>
<td>0.851</td>
<td>0.837</td>
</tr>
<tr>
<td>VQA</td>
<td>0.846</td>
<td>0.833</td>
</tr>
<tr>
<td>SequenceLevel</td>
<td>0.834</td>
<td>0.777</td>
</tr>
<tr>
<td>Hysteresis</td>
<td>0.826</td>
<td>0.802</td>
</tr>
<tr>
<td>Histogram</td>
<td>0.782</td>
<td>0.781</td>
</tr>
<tr>
<td>LogExp</td>
<td>0.571</td>
<td>0.538</td>
</tr>
<tr>
<td>ExpMinkowski</td>
<td>0.565</td>
<td>0.549</td>
</tr>
<tr>
<td>LocalMinimum</td>
<td>0.555</td>
<td>0.447</td>
</tr>
<tr>
<td>MeanLastFrames</td>
<td>0.485</td>
<td>0.477</td>
</tr>
<tr>
<td>KMeans</td>
<td>0.353</td>
<td>0.334</td>
</tr>
<tr>
<td>SoftMax</td>
<td>0.259</td>
<td>0.297</td>
</tr>
<tr>
<td>Minkowski</td>
<td>-0.058</td>
<td>0.368</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pooling algorithm</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCC</td>
<td>SROCC</td>
</tr>
<tr>
<td>Percentile</td>
<td>0.854</td>
<td>0.835</td>
</tr>
<tr>
<td>Mean</td>
<td>0.851</td>
<td>0.837</td>
</tr>
<tr>
<td>VQA</td>
<td>0.846</td>
<td>0.833</td>
</tr>
<tr>
<td>SequenceLevel</td>
<td>0.834</td>
<td>0.777</td>
</tr>
<tr>
<td>Hysteresis</td>
<td>0.826</td>
<td>0.802</td>
</tr>
<tr>
<td>Histogram</td>
<td>0.782</td>
<td>0.781</td>
</tr>
<tr>
<td>LogExp</td>
<td>0.571</td>
<td>0.538</td>
</tr>
<tr>
<td>ExpMinkowski</td>
<td>0.565</td>
<td>0.549</td>
</tr>
<tr>
<td>LocalMinimum</td>
<td>0.555</td>
<td>0.447</td>
</tr>
<tr>
<td>MeanLastFrames</td>
<td>0.485</td>
<td>0.477</td>
</tr>
<tr>
<td>KMeans</td>
<td>0.353</td>
<td>0.334</td>
</tr>
<tr>
<td>SoftMax</td>
<td>0.259</td>
<td>0.297</td>
</tr>
<tr>
<td>Minkowski</td>
<td>-0.058</td>
<td>0.368</td>
</tr>
</tbody>
</table>

Table 1: Correlation between the pooled objective values (PSNR, SSIM) and subjective MOS after training and cross validation (sorted by PCC of PSNR pooling).
Figure 25: Scatter plot diagram of MOS versus model output based on pooling of MOS values for all profiles under consideration.

<table>
<thead>
<tr>
<th>Pooling algorithm</th>
<th>Quality Level</th>
<th>Constant QL MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCC</td>
<td>SROCC</td>
</tr>
<tr>
<td>SequenceLevel</td>
<td>0.861</td>
<td>0.851</td>
</tr>
<tr>
<td>VQA</td>
<td>0.867</td>
<td>0.860</td>
</tr>
<tr>
<td>Mean</td>
<td>0.856</td>
<td>0.837</td>
</tr>
<tr>
<td>Percentile</td>
<td>0.824</td>
<td>0.806</td>
</tr>
<tr>
<td>Minkowski</td>
<td>0.820</td>
<td>0.800</td>
</tr>
<tr>
<td>MeanLastFrames</td>
<td>0.829</td>
<td>0.805</td>
</tr>
<tr>
<td>ExpMinkowski</td>
<td>0.822</td>
<td>0.802</td>
</tr>
<tr>
<td>Histogram</td>
<td>0.785</td>
<td>0.789</td>
</tr>
<tr>
<td>LogExp</td>
<td>0.746</td>
<td>0.711</td>
</tr>
<tr>
<td>LocalMinimum</td>
<td>0.831</td>
<td>0.803</td>
</tr>
<tr>
<td>Hysteresis</td>
<td>0.766</td>
<td>0.726</td>
</tr>
<tr>
<td>KMeans</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>SoftMax</td>
<td>0.656</td>
<td>0.598</td>
</tr>
</tbody>
</table>

Table 2: Correlation between the pooled quality levels and pooled constant quality level MOS scores and subjective MOS after training and cross validation (sorted by PCC of Constant QL MOS pooling).
5.5.3 Temporal pooling of objective metrics

For the assessment of the perceived quality of the audio-visual sequences, a temporal pooling scheme is proposed in this section. It is based on the computation of standardised video quality metrics, such as J.144, J.341, and P.1201.2, in video chunks of short duration, typically from 5 to 15 seconds, and the aggregation of these scores using several temporal pooling schemes.

The first step for the computation of the short-term objective video quality metrics is the splitting of the video sequence into chunks of the same video representation, as depicted in Figure 26. Emphasis is given in the concatenation of those segments which belong to the same representation to ensure that there are no large quality fluctuations within one concatenated chunk. Moreover, no more than three segments, i.e. 15 seconds are concatenated. The main reason for that is that the employed metrics have been designed, developed, and tested only for short durations, between 5 and 15 seconds.

Once the objective metrics of the concatenated segments have been computed, the next step is the application of the temporal pooling scheme. The pooling of the individual scores should account for the perceptual effects in human quality assessment, such as the well-known “forgiveness” recency effect, memory effects, and perceived quality due to quality changes within the video sequence. Prior research [11],[13] indicates that perceived quality drops quickly when there is a distortion while it slowly increases when the video quality increases. Therefore, the proposed method employs an auto-regressive moving-average (ARMA) model to simulate the adaptation of perceived quality over time. Let \( x(i) \) denote the objective quality score which is computed at chunk \( i \), then the predicted perceived quality of the video sequence at time \( i \) is given by:

\[
y(t) = \sum_{i=1}^{k} a_i \cdot y(t-i) + \sum_{j=1}^{l} b_j \cdot x(t-l)
\]

where \( k, l \) denote the number of previous scores that are considered for the estimation of \( y \) at each time, and they are both selected equal to 3 in the proposed method. The coefficients \( a_i \) and \( b_j \), \( i \in \{1, \ldots, k\}, j \in \{1, \ldots, l\} \) are selected during the training phase.

Moreover, a penalty parameter is introduced into the model to take into consideration the abrupt quality degradation within the sequence and the frequency in representation switches. This parameter, called Abrupt Quality Changes (AQC), denotes the quality adaptation between
representations with very different perceived quality and occurs when there is a non-smooth adaptation in profile switching. It highlights the fact that users are expected to be very annoyed when quality suddenly and significantly drops but also when it suddenly increases. The concept of the use of AQC is illustrated in Figure 27; in this sequence, the representation is changing from profile #1 to profile #4, then back to #1 and finally to #5. As can be seen from the predicted values of each of the concatenated segments, the switching from representation #1 to #4 will result in a quality change of 3.8-1.63=2.17 MOS scores, which is an indication of an abrupt, highly noticeable difference for the viewer. Likewise, the other two switches result in a similarly large quality difference, therefore, the AQC value in this sequence is 3. More specifically, AQC is given by:

\[ AQC = \sum_{i=2}^{k} u(|x(i) - x(i - 1)|) \]

where \(x(i)\) denotes the predicted objective quality score at chunk \(i\), and the function \(u(.)\) is defined as:

\[ u(i) = \begin{cases} 0, & \text{if } i \geq T \\ 1, & \text{if } i < T \end{cases} \]

Based on the subjective data, \(T\) was selected equal to 2.1. Then, the quality degradation associated with the computed value of AQC is given by:

\[ I_{AQC} = c_1 - c_2 \cdot \exp(-c_3 \cdot AQC) \]

Finally, the overall predicted quality of the video sequence is given by:

\[ MOS = \left( \frac{1}{p} \sum_{i=1}^{m} y^p(i) \right)^{1/p} - d_1 \cdot I_{AQC} \]

Figure 27: Abrupt quality changes within a sequence. The switching between representations of substantially different quality results in very annoying artifacts. The AQC value in this example is equal to 3.
where \( m \) denotes the total number of concatenated chunks, \( p \) is the exponent of the Minkowski summation (selected equal to 0.8), and \( d_1 \) is a coefficient. The motivation of using Minkowski summation for the temporal pooling is that higher distortions (lower \( y \) values) should be weighted more than lower distortions, as it is known that they have a higher impact on user perception.

### 5.5.3.1 Performance evaluation

The proposed method was evaluated on the subjective data that were obtained as described in Section 4.3 against five other popular techniques, namely “Mean”, “Median”, “Max”, “Min”, and “Minkowski summation”, which were explained in Section 5.5.1. In the “Max” “Min” methods, the quality of the sequence is determined by the quality of the maximum or minimum predicted quality of a concatenated chunk in the video sequence. Moreover, for the predicted quality of each concatenated chunk three objective metrics were used: the full-reference VQM and VQuad, and the no-reference P.1201.2. The performance of the different pooling schemes and the different objective metrics in terms of correlation coefficient and RMSE is presented below in Table 3 and Table 4, respectively.

It should be noted that since VQM does not directly provide scores in the MOS scale, but rather in the DMOS scale, the scores need to be mapped to the MOS scale. For this, the subjective scores of the sequences with only one representation throughout the whole sequence (i.e., without representation switches) and the corresponding subjective data were used. From Figure 28 and Figure 29 it can be seen that the VQM results can be linearly mapped to the subjective scores.

For the performance evaluation, half of the sequences from clip 1 and clip 2 were used in the training set, and the remaining were used as the evaluation (or “test”) set. In this fashion, the results are reported on sequences which are completely unknown to the prediction models. However, it must be stressed that the evaluation set contains the same source sequences (clip 1 and clip 2) as the training set, therefore, the video content at the evaluation set is not unknown. This is imposed by the fact that the subjective test included only two source sequences. 20 different instances were randomly selected for the training set (which corresponds to 20 different evaluation sets with the remaining data). All results reported below refer to the average of the evaluation sets.
First of all, it can be observed that the performance of the “Min” method is higher than the “Max” method; this result is expected since it is well-known that users are more sensitive to quality degradation than quality improvement. Surprisingly, however, the “Mean” method performs better than the “Minkowski summation” (it must be noted that for $p=1$, the Minkowski summation method is identical to the “Mean” method, however the value $p=1$ is excluded for Minkowski summation). The best performance is achieved for the proposed method which incorporates a memory element to simulate the process of user perception and adapts smoothly to every quality adaptation within the sequence.

Moreover, it must be noted that the performance of VQM is constantly better than the two other objective measures. This can be justified by the fact that it has been designed to address mainly coding distortions, whereas the two other metrics have been designed in order to address other
types of visual distortions, such as slicing artefacts and freezing. Furthermore, the subjective data of the sequences without representation switches were used to map the VQM scores to subjective scores, which means that the VQM results are not completely “blind” of the subjective data. Nevertheless, the P.1201.2, despite being a no-reference metric, can achieve better results than the full-reference VQuad.

Table 3: Pearson Correlation Coefficient of the temporal pooling schemes and the different objective quality metrics.

<table>
<thead>
<tr>
<th>Pooling method</th>
<th>VQM</th>
<th>VQuad</th>
<th>P.1201.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.8641</td>
<td>0.8173</td>
<td>0.8358</td>
</tr>
<tr>
<td>Median</td>
<td>0.7567</td>
<td>0.7375</td>
<td>0.6925</td>
</tr>
<tr>
<td>Max</td>
<td>0.6030</td>
<td>0.5435</td>
<td>0.5632</td>
</tr>
<tr>
<td>Min</td>
<td>0.7567</td>
<td>0.7224</td>
<td>0.7738</td>
</tr>
<tr>
<td>Minkowski summation</td>
<td>0.8147</td>
<td>0.7843</td>
<td>0.8024</td>
</tr>
<tr>
<td>ARMA</td>
<td>0.9064</td>
<td>0.8451</td>
<td>0.8742</td>
</tr>
</tbody>
</table>

Table 4: RMSE of the temporal pooling schemes and the different objective quality metrics.

<table>
<thead>
<tr>
<th>Pooling method</th>
<th>VQM</th>
<th>VQuad</th>
<th>P.1201.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.3198</td>
<td>0.3660</td>
<td>0.3467</td>
</tr>
<tr>
<td>Median</td>
<td>0.5069</td>
<td>0.5746</td>
<td>0.5542</td>
</tr>
<tr>
<td>Max</td>
<td>0.4153</td>
<td>0.4324</td>
<td>0.4539</td>
</tr>
<tr>
<td>Min</td>
<td>0.5069</td>
<td>0.4831</td>
<td>0.5327</td>
</tr>
<tr>
<td>Minkowski summation</td>
<td>0.3655</td>
<td>0.3982</td>
<td>0.3821</td>
</tr>
<tr>
<td>ARMA</td>
<td>0.2832</td>
<td>0.3226</td>
<td>0.2964</td>
</tr>
</tbody>
</table>
6 Conclusions & Recommendations

This work has been to identify KPIs to monitor video QoE delivered over wireless networks.

One of the original project objectives was to develop a correlation between radio and core network QoS KPIs to predict QoE of a streamed video. The primary reason for that objective was to provide mobile network operators a mechanism to predict QoE based on the QoS measurements they are obtaining with existing tools in their network. However, through the study it was realised that instead of making a one-to-one mapping between generic radio and core network QoS and QoE, it is simpler and more accurate to measure metrics from the HAS bit-stream and make predictions on QoE. The biggest benefits of this approach were to simplify the measurement framework and to remove the need to have a reference video sample. Instead it would be sufficient to simply collect HTTP header information on the server or any point between the terminal and the server.

One challenge in validating a model to predict the QoE of a video stream is obtaining sufficient experimental results. For statistically meaningful results, a minimum of 15 or more individuals are required to rate the quality of a session. Wireless delivered video causes two particular problems – the screens are limited in size so a single stream can only be viewed by one person at a time. Secondly, the variability in radio conditions means it is difficult to be sure that two people have seen the same quality. Fortunately, concentrating on HTTP Adaptive Streaming (HAS) content, helped to bypass these issues by downloading the video test clips to the user devices. It also meant it was possible to recruit a large number of test subjects through NGMN supported advertising as described in section 4.

The video industry has developed several mechanisms and techniques to quantify the quality of a video stream. These are discussed in section 5.1. Test results from this project were applied to examine how accurate these techniques are in predicting the quality of a video stream. In particular the error in prediction (Root Mean Square Error – RMSE) and in correlation (both Pearson - PCC and Spearman Rank Order - SROCC) was investigated. The suitability of the technique for monitoring of video streams by the operator of a mobile network was also examined. Some techniques are intrusive or require information about the original stream which is not normally available to the mobile operator. Fortunately, we know the HAS profile actually gives sufficient information the majority of the time.

As discussed in section 4.1.4, the selection of content can influence the accuracy of the results. The objective SSIM analysis in section 5.3.3.1 affirms the suitability of the chosen test clips for the mass test evaluation. The video content of certain chunks is complex enough to get impaired as intended when encoded at the different quality levels.

Section 5.3.3.2 validates a high correlation between subjective MOS and average SSIM or PSNR values. SSIM has proven to be less dependent of the clip content than PSNR. Under the proviso that the reference content is available, objective video quality measurements can provide similar results as an alternatively time consuming subjective mass test, providing a reference rating for the test.
clips at the push of a button. This allows faster validation of simpler prediction algorithms and trimming of the input values. The latter is beneficial with regards to high correlation outcome of the straightforward algorithm based on mean QL calculation, since its key to success lies in the initial finding of a QL level assignment proportionally to the true video quality. The inferior result when evaluating the mean bitrates instead shows the influence of a suboptimal mapping.

One unique feature of this work is that it examined the impact on QoE of abrupt changes in quality level which is characteristic of HAS. Many of the existing quality assessment techniques work well for determining the QoE of content encoded at constant quality. These allow us to determine the QoE (MOS) of each quality level. However a HAS stream can have several sudden and significant changes in quality. Therefore a model which pools the MOS values from the individual chunks into a single MOS value for the entire clip is required. This is known as temporal pooling. There are several temporal pooling techniques described in literature (see section 5.5.1). In addition a Linear MOS Predictor (see section 5.4) was introduced using the mean and standard deviation of the quality levels.

Section 5.5.3 shows the results of applying some temporal pooling approaches to some standard objective video quality metrics (VQM, VQUAD, and P.1201.2). First of all, it can be observed that the performance of the “Min” method is higher than the “Max” method; this result is expected since it is well-known that users are more sensitive to quality degradations than quality improvements. Surprisingly, however, the “Mean” method performs better than the “Minkowski summation”. The best performance is achieved for the proposed method using an Auto-Regressive Moving Average (ARMA) which incorporates a memory element to simulate the process of user perception and adapts smoothly to every quality adaptation within the sequence. VQM performs better than the other methods but the VQM score needs to be mapped to MOS for each clip. Surprisingly, the no-reference metric P.1201.2 performs better than the full-reference VQUAD.

The Linear MOS Predictor model (section 5.4) demonstrates that just considering the HTTP GET requests provides an accurate prediction of MOS. It uses the mean and standard deviation of the quality levels. The chunk-MOS model (MOS of each quality level is known or is sampled) has proven to be a good MOS estimator resulting in an accuracy of about 0.3 MOS-points for tablet and 0.4 MOS-points for smartphone. If, as most deployments will do, the quality levels are encoded at equidistant MOS, just using the Quality Level is also a good MOS predictor. The accuracy of the model is currently (with the available dataset) mainly limited by the error (uncertainty) of the obtained subjective MOS values in the dataset which are of similar order. The model itself has not yet reached its limits of accuracy. It is hard to predict the lower boundary of accuracy that could be reached with less noisy data, but it is expected that considerable further improvement is possible.

The chunk-MOS model is well positioned to be used in a video QoE assessment module that, by inspecting HTTP requests for HAS content, predicts the quality delivered to the user. The model can be used for individual users, but can also be the basis to give a quality indicator per cell or per larger geography.
Different content sources will use different encoder settings (codec, bit rate, GOP size, aspect ratio, frame rate etc.), which impacts the QoE. Also, different content types have different encoding requirements. For example a News channel of mainly talking heads is much easier to encode than a sports channel with active motion content. Therefore, it is advisable to periodically calibrate the quality of content from different sources using either subjective methods or intrusive techniques.

The results should also be used to help network planning and radio capacity. For any set of radio conditions, there is a spread of MOS scores. An operator can decide the probability that the MOS score is above 2 and from that ensure there is sufficient capacity in the Radio Access Network.

In summary, the P-SERQU project has shown that for HTTP Adaptive Streaming (HAS) content, conventional objective measurement techniques such as PSNR and SSIM give good results. However, by just examining the video quality levels requested in HTTP GETs, there is also an accurate way to monitor video streams in a mobile network. The Linear MOS Predictor shows that an accuracy of 0.3 MOS points can be expected. This avoids the need to analyse the significant amount of measurement data generated in the wireless level. It is a non-intrusive approach. Periodic sampling of content from each source may be required to calibrate the quality levels. If the packet headers are in the clear (unencrypted), then analysing the MPEG-2 Transport Stream header information with P.1201.2 can further improve the results at a small computational overhead.

Using either the Linear Predictor Model or P.1201.2, a mobile network operator can effectively and efficiently monitor the quality of a large number of video streams. Where the quality of a stream is below the required quality threshold, the operator can further investigate the cause. This can include identifying the radio conditions of the client as well as the general congestion in the RAN cell. While there is no 1-to-1 relationship between radio conditions and QoE, NGMN should consider extending the P-SERQU work to study how it can help in radio network planning.
7 References


8 Appendix – Surveys

8.1 Phase 1

The first step was to recruit volunteers. Below is the letter sent from NGMN (ngmn.org) to potential volunteers. These were generally those suggested by member companies. As people were asked to download content to their devices as well as answer a questionnaire, they had to be sure that the content was from a source they trust.

---

**Video Quality Test Invitation**

The Next Generation Mobile Networks (NGMN) alliance is conducting a test of mobile video networks and needs your help. We are looking for individuals willing and able to watch a few video clips and answer questions about the video playback experience. We will use the results in our research so that wireless network providers can upgrade their networks and improve tomorrow’s networks for a better video streaming experience. We will not share any personal information such as your name, e-mail address, or location. Help us improve the networks for better video!

To qualify for this test survey, you are required to use an iPhone or iPad using iOS 5 for the test. You will need to receive an email message containing instructions on your iPhone or iPad. The test will require up to 15 minutes of uninterrupted attention, and should be performed in a quiet place with high speed Wi-Fi access and a strong signal.

If you are willing to participate and feel that you qualify, please apply by sending an email message to video@ngmn.org with “Volunteer” in the subject, and your email address in the body of the message (upper or lowercase are not important). On receipt of your email, NGMN will email you detailed instructions. The survey, which is in English, will be available until end of August.

Many thanks for considering taking part in the survey.

Michael Wennesheimer
On behalf of the NGMN Video Quality Project

---

Those wishing to volunteer would send an email with the string ‘volunteer’ in the subject line. This provided a semi-automated process to allocate a specific package to each volunteer. As there are 20 packages, they were allocated on a round robin basis on the arrival time of the email from the
volunteer. Assuming everyone who volunteered would also complete the questionnaire; this would give even coverage of packages. By return, each volunteer would be given the below instruction letter with a personalised URL for their package. This allowed tracking of volunteers’ response to packages and send reminders to those who were slow in completing the questionnaire.

Instructions are shown here:

To: test-volunteer@domain
[note: this email field is populated using Mail Merge]
Re: Video Quality Test Instructions

Before taking the survey

The SERQU (Mobile Video Quality) test and survey requires several steps to be taken, using your preferred mobile device, either iPhone or iPad.

1. You should be reading this email on the mobile device of your choice (either iPhone or iPad) and the device should not be “jailbroken”.
2. You should be connected to a WiFi network so that you will not incur data charges during the test.
3. You will need about 80 megabytes of free storage on your device (the same as about 20-30 songs).
4. To prepare for this survey, please visit the following link to subscribe to a video podcast containing 5 videos: http://itunes.apple.com/us/podcast/ngmn-video-quality-package-20/id543051102?mt=2
   [note: this URL refers to a package number ranging from 1-20]
5. Please touch each video download button to download each video, and then wait for all to download completely.
6. Then check to see if all videos have been downloaded by opening the Videos app.

Service Quality Definition and Measurement,
Version 1.0.4, 27th August 2013
The principle of preparing packages of five clips which can be downloaded to the iPhone or iPad turned out to be more tricky than expected. In particular, it was necessary to ensure the process of downloading, viewing and recording the rating was as easy and foolproof as possible. Options as compressing the content into a ZIP package and downloading using an application such as iZIP were explored. However, finally it was agreed to uploading the packages as podcasts to iTunes and using the built in video application. However, this also caused some issues as it was essential that people waited for the clips to download. We were also further hampered by Apple releasing iOS6 before the questionnaire phase was completed. This meant the handling of podcasts changed and iOS6 specific instructions had to be released including downloading the podcast application. Below are some more details of creating the packages and questionnaire.

1. The first attempt at creating an integrated video delivery and survey package was not successful. These are the steps taken: prototype web pages were designed using the
2. The second method tested involved video delivery using ZIPped archives containing complete packages of up to five videos in each archive. This method was satisfactory from the perspective of controlling video playout, however it required each survey respondent to install a ZIP app on their iDevice. Because of the additional user burden, this method was not selected.

3. The third method tested, and eventually used for the majority of responses, provided video delivery through a podcast, supported by the standard Apple iTunes Podcast infrastructure. In general, Apple podcasts are implemented through RSS, with the Apple iTunes store controlling publication of the RSS link. The media supplier (e.g., this team) is then responsible for hosting the files containing the XML definitions of the RSS (podcasts) and the media files. The iTunes store only stores thumbnail icons and catalogue descriptions. A truncated XML file referring to video files is shown below. The media files were stored on a private Amazon S3 bucket (internet file hosting solution). Because of the fairly modest download traffic and storage requirements, the media
storage cost was negligible.

4. The fourth survey method was provided later in the project, with a custom, mobile-compatible website. This enabled us to expand the volunteer base and as a result a better survey experience was offered - using a custom web site capable of collecting more volunteer data (e.g., type of iDevice). Sample web pages from that site are shown in the next section.

8.2 Phase 2

Figure 30: Sample Screenshot of video questionnaire
The questionnaire was preceded with a special e-mail offer of iTunes vouchers for randomly selected participants. Invitation E-Mail Text is shown here.

Dear colleague,

The Next Generation Mobile Networks (NGMN) alliance is conducting a test of mobile video networks and needs your help. We are looking for individuals willing and able to watch a few video clips and answer questions about the video playback experience. We will use the results in our research so that wireless network providers can upgrade their networks and improve tomorrow’s networks for a better video streaming experience. Help us improve the networks for better video! It simply requires you to access a link (see below) from your mobile (Apple) device to participate.

If you have access to a recent iPhone (4, 4s, 5) or iPad (2, 3, 4) you simply need to go to

https://portal.ftw.at/ngmn-survey/

and follow the instructions straight from your phone/tablet.

As a special thank you for taking part in this study we will enter your name in a lottery. Within this lottery we draw 10 participants who win a €25.- voucher for the iTunes store. The winners will be contacted via e-mail. The survey will be online until 17th December. The winners of the lottery will be informed and announced in due course after.

Feel free to distribute this link within your network to enable us gathering a solid number of survey statistics.

Thanks in advance for your support!
9 Appendix – Video Quality Assessment Methodologies

There are several methodologies already defined to assess the quality of video. This appendix lists those used in the NGMN project.

9.1 Peak Signal to Noise Ratio (PSNR)

Let \( I(i,j) \) denote the intensity value of the original frame and \( I'(i,j) \) the intensity of the distorted frame at the pixel position \((i,j)\). Then, the Mean Squared Error (MSE) is defined as:

\[
MSE = \frac{\sum_{i=1}^{R} \sum_{j=1}^{C} (I(i,j) - I'(i,j))^2}{R \cdot C}
\]

and the PSNR of an image is defined as:

\[
PSNR = 10 \log_{10} \frac{L^2}{MSE}
\]

where \( R, C \) is the number of rows and columns in the image, respectively, and \( L \) is the maximum intensity value; typically \( L=255 \) for an 8-bit image. The PSNR of a video sequence is typically computed as the average of the individual PSNR of the images comprising the video sequence.

MSE is quite popular in image/video quality assessment due to its inherent properties: it is simple to compute, parameter-free and memory-less, and as a result, it can be calculated locally without consideration of other source samples. Nevertheless, it is often criticised due to its poor correlation with subjective perceived quality due to the fact that it does not take into account the properties of the human visual system.

9.2 Structural Similarity (SSIM)

SSIM is computed based on three individual components: luminance, contrast, and structure, as detailed below. For the luminance component, the mean and standard deviation of the luminance of the reference and the distorted image are computed as:

\[
\mu_I = \frac{1}{R \cdot C} \sum_{i=1}^{R} \sum_{j=1}^{C} I(i,j)
\]

\[
\mu_{I'} = \frac{1}{R \cdot C} \sum_{i=1}^{R} \sum_{j=1}^{C} I'(i,j)
\]

\[
\sigma_I = \left( \frac{1}{R \cdot C} \sum_{i=1}^{R} \sum_{j=1}^{C} (I(i,j) - \mu_I)^2 \right)^{1/2}
\]

\[
\sigma_{I'} = \left( \frac{1}{R \cdot C} \sum_{i=1}^{R} \sum_{j=1}^{C} (I'(i,j) - \mu_{I'})^2 \right)^{1/2}
\]

\[
\rho_{IJ} = \frac{\mu_I \mu_{I'} + \gamma}{\sqrt{\sigma_I \sigma_{I'}}}
\]

where \( \gamma \) is a small constant to avoid division by zero (typically \( \gamma = 0.01 \)). The SSIM of a video sequence is typically computed as the average of the individual SSIM of the images comprising the video sequence.

SSIM is considered to be a more accurate measure of image/video quality than PSNR due to its ability to take into account the properties of the human visual system.
Finally, the SSIM index is computed as:

\[
SSIM = \frac{2\mu_I\mu_{I'} + C_1}{\mu_I^2 + \mu_{I'}^2 + C_1} \cdot \frac{2\sigma_I\sigma_{I'} + C_2}{\sigma_I^2 + \sigma_{I'}^2 + C_2} \cdot \frac{2\sigma_{II'} + C_3}{\sigma_I\sigma_{I'} + C_3}
\]

where the constants \(C_1, C_2,\) and \(C_3\) are selected to avoid division by zero. If the parameter \(C_3\) is selected equal to \(C_2^2/2\) then, the above equation becomes:

\[
SSIM = \frac{2\mu_I\mu_{I'} + C_1}{\mu_I^2 + \mu_{I'}^2 + C_1} \cdot \frac{2\sigma_{II'} + C_2}{\sigma_I^2 + \sigma_{I'}^2 + C_2}
\]

For video sequences, the SSIM index is typically computed as the average of the indices of SSIM index of the images comprising the video sequences.

### 9.3 ITU-T Recommendation J.144 (VQM)

The VQM utilises reduced-reference picture-based, following an engineering approach, parameters that are extracted from optimally-sized spatial-temporal (S-T) regions of the video sequences. These features are extracted from the source and distorted video streams. Therefore a calibration for comparing both videos in the same environment is needed. VQM and its associated calibration techniques follow a complete automated process. The calibration of the original and distorted video streams consist of four parts: spatial alignment, valid region estimation, gain and level offset calculation and temporal alignment. Then the perception-based features are extracted and the video quality parameters are computed. Finally these parameters are combined to obtain the estimate of video quality.

VQM is composed by seven independent parameters. Four are based on features extracted from spatial gradients of the Y component, two on features extracted from the vector formed by the two U and V chrominance components, and one on the product of features that measure contrast and motion. The seven parameters are briefly described below:

- **si_loss**
  
  This parameter detects a decrease or loss of spatial information. It uses a 13 pixel spatial information filter (SI13), which was specifically developed to measure perceptually significant edge impairments. SI13 utilises 13x13 pixel horizontal and vertical filter masks.

- **hv_loss**

Service Quality Definition and Measurement, Version 1.0.4, 27th August 2013
This parameter detects a shift of edges from horizontal and vertical orientation to diagonal orientation. It uses the horizontally and vertically filtered H and V images from the SI13 filter.

- **hv_gain**  
  This parameter detects a shift of edges from diagonal to horizontal and vertical.

- **chroma_spread**  
  This parameter detects changes in the spread of the distribution of two-dimensional color samples.

- **si_gain**  
  This is the only quality improvement parameter in the model. The **si_gain** parameter measures improvements to quality that result from edge sharpening or enhancements.

- **ct_ati_gain**  
  This metric is computed as the product of a contrast feature, measuring the amount of spatial detail, and a temporal information feature, measuring the amount of motion present in the S-T region. Impairments will be more visible in S-T regions that have a low product than in S-T regions that have a high product. The parameter **ct_ati_gain** identifies moving-edge impairments that are nearly always present.

- **chroma_extreme**  
  This metric detects severe localised color impairments, such as those produced by digital transmission errors.

Finally, VQM is computed by a linear combination of the parameters described above, as:

\[
VQM = -0.2097 \cdot si\_loss + 0.5969 \cdot hv\_loss + 0.2483 \cdot hv\_gain + 0.0192 \cdot chroma\_spread - 2.3416 \cdot si\_gain + 0.0431 \cdot ct\_ati\_gain + 0.0076 \cdot chroma\_extreme
\]

It must be noted that VQM is clipped at a lower threshold of 0.0. If \( VQM \leq 1.0 \), then:  
\[
VQM = (1 + c) \cdot VQM / (c + VQM), \text{ where } c = 0.5.
\]

Moreover, lower values correspond to lower degradations (better quality) whereas higher values correspond to higher degradations (lower quality).

### 9.4 ITU-T Recommandation J.341 (VQuad)

It is based on the computation of the following properties:

- **Blockiness**  
  The blockiness value is estimated by measuring the luminance differences at block borders. It is related to the amount of spatial detail, since a block border has a stronger visibility in
the absence of spatial details. Blocks might have different sizes; however the borders will always be horizontally or vertically oriented and therefore form a right angle. In very bright or dark areas the degradation between borders is less visible even though it is clearly measureable. The main reason for the blockiness effect is the strong compression during the encoding processes. Furthermore, packet loss during transmission could increase blockiness.

- **Tiling**
  Tiling is the effect of visible tile-like artefacts in the video frames. It can occur due to either the encoding process or the transmission. The tiling value is based on the distortions at block borders caused by transmission errors. This type of error can be handled by the receiving decoder following different strategies. The simplest one is just to display the erroneous data, which leads to very strange effects. A more developed strategy is to freeze the last successfully updated video frame up to the next key-frame providing a complete image at once. Another way to deal with this kind of errors is to replace the erroneous transmitted parts of the video frame by the same area of the previous video frame, this is known as slicing. Advanced strategies predict the missing data by the neighboring blocks. Since no concealment strategy is perfect, the residual error will be propagated by the following differential frames up to the next key-frame.

- **Blurring**
  Blurring is measured indirectly by the measuring of sharpness, which measures the luminance offset at edge borders in the video frames and relates this to the local contrast at the edge location. Sharpness tries to avoid edges which form block borders as a result of blocking or tiling. The blurring value is the decrease of the sharpness of the reference video frame to the transmitted one. Sharpness is strongly content dependent; therefore the sharpness is measured at the position of the sharpest edges in the video frame.

- **Jerkiness**
  Jerkiness is the result of bad representation of moving objects in a video sequence. Jerkiness is a perceptual degradation, which measures the loss of information due to freezing or low frame rates. Therefore freezing and the ‘Dominating Frame Rate’ along with this anticipated loss of information forms the jerkiness.

The above described perceptual degradation measures are used to estimate a MOS prediction. As those are non-additive and therefore non-linear, the most important degradation will determine the objective quality, while the less important distortions will have a smaller weight in the predicted MOS.

\[
MOS = f_{\text{temporal}}(\text{video}) \cdot f_{\text{spatial}}(\text{video})
\]

where \( f_{\text{temporal}} \) is a function of temporal degradations and \( f_{\text{spatial}} \) is a function of spatial degradations dominated by the perceptual difference measure.
9.5 ITU-T Recommendation P.1201.2

Recommendation ITU-T P.1201 provides an overview of algorithmic models for non-intrusive monitoring of the audio, video, and audiovisual quality of IP-based video services based on packet-header information. The ITU-T P.1201 model algorithm is a no-reference (i.e. non-intrusive) model which operates by analysing packet header information as available from respective packet trace data, provided to the model algorithms in the packet capture format (PCAP). The model is restricted to input Information from decoding the bit-stream or parsing the packet payload is not used. Further input information on more general aspects of the stream, such as the video resolution, which may not be available from packet header information, is provided to the model algorithm out-of-band, for example in the form of stream-specific side information. The block diagram of the P.1201.2 model is provided in Figure 31.

As output, the model algorithms provide individual estimates of audio, video and audiovisual quality in terms of the five-point absolute category rating (ACR) mean opinion score (MOS) scale. Further, diagnostic information on causes of quality degradations can be made available, too.

![Figure 31 - Block diagram of the ITU-T P.1201.2 model](image)

In ITU-T Rec. P.1201.2 49[22] the quality impact of video compression \(\text{CompDeg} \) is given by:

\[
\text{CompDeg} = a_1 \cdot \exp(a_2 \cdot \text{BitPerPixel}) + a_3 \cdot \text{SceneComp} + a_4
\]

where \(\text{CompDeg} \in [0, 100]\), with lower \(\text{CompDeg}\) values corresponding to higher perceived quality. The parameters \(a_i, i \in \{1,2,3,4\}\), are curve fitting coefficients which depend on the video resolution. The \(\text{BitPerPixel}\) is the normalised video bitrate, given by:

\[
\text{BitPerPixel} = \frac{\text{bitrate} \cdot 10^6}{nx \cdot fr}
\]

where \(\text{bitrate}\) is the video bitrate in Mbps, \(nx\) denotes the number of pixels per frame, and \(fr\) the video frame rate in frames per second. Finally, \(\text{SceneComp}\) captures the spatio-temporal complexity of the content from scene-related parameters extracted from the packet headers, and is given by:
The parameter $S_{sc}^I$ is the average I-frame sizes for the given scene $sc$. The first I-frame of the first scene is ignored in the case of single-pass encoding. $Z$ denotes the number of scenes in the video sequence, and $N_{sc}$ is the number of GOPs in scene $sc$. For the scene having the lowest $S_{sc}^I$ value, $w_{sc}$ is set to 16, otherwise it is set to 1. For more information on ITU-T Rec. P.1201.2, the interested reader is referred to [21].

Equation 1 reflects that the I-frame sizes play an important role in the determination of the content complexity. In particular, video contents with low $S_{sc}^I$ values yield lower perceived quality. Moreover, the scene with the lowest quality, i.e. with the lowest $S_{sc}^I$ value, will have more influence ($w_{sc} = 16$) on the overall perceived quality than the other scenes. See [24] for more detailed explanations on the parameter $SceneComp$.

10 Appendix Temporal Pooling techniques

Objective methodologies for video quality assessment typically output quality scores per frame. As analysed video sequences consist of $T$ frames, a certain (temporal) pooling metric is needed to combine these $T$ scores to an overall score for the sequence. An early comparison of simple pooling methods is given in [6]. More sophisticated pooling methods are presented in [7], [8], [9], [10] and [11]. A different pooling approach using spatial pooling is described in [12]. However, the mentioned temporal pooling methods are mainly targeted towards video sequences of 10 to 15 sec and can hence not be used for video sequences of longer durations in the order of minutes.

In this comparison, 13 different pooling methods were used and applied to the objective metrics $OM$. Six methods are described in [1]. For histogram pooling (Histogram) the $k$-th percentile of the cumulative histogram values is used. Low values of $k$ express the influence of the lowest quality frames on viewers. For Minkowski summation (Minkowski, $\left[\frac{1}{T} \sum_{t=1}^{T} OM^P(t)\right]^{1/p}$) and exponentially-weighted Minkowski summation (ExpMinkowski, $\left[\frac{1}{T} \sum_{t=1}^{T} \exp\left(\frac{T-t}{\tau}\right) OM^P(t)\right]^{1/p}$) high values of $p$ emphasize the influence of highest quality frames. ExpMinkowski additionally accounts for recency effects by the exponential weighing factor with parameter $\tau$. Mean pooling (Mean, $\frac{1}{T} \sum_{t=1}^{T} OM(t)$) and last frames mean pooling (MeanLastFrames, $\frac{1}{F} \sum_{t=T-F}^{T} OM(t)$) simply compute the mean of all or respectively the most recent $F$ frames’ objective metrics. Local minimum of mean values of $N$ successive frames (LocalMinimum, $\min\left(\frac{1}{N} \sum_{i=1}^{N} OM(t + i)\right)$) emphasises the influence of the
poorest quality section on the overall score. A related approach not described in [1] simply computes the mean of the $p$ percent of overall frames with lowest quality (Percentile).

To account for the hysteresis effect, two elements are employed in [8] at each time instance. The first element is computed as the minimum of the quality scores over the last $\tau$ seconds, while the second element is computed as a weighted sum of the quality scores, sorted in ascending order, in the next $\tau$ seconds. Finally, the two elements are linearly combined at each time instance, and the overall quality is computed as the mean over time. Hysteresis is described in [7]. In [8] two parametric functions are described which, similar to Minkowski pooling, transition continuously from mean ($p=0$) to max pooling ($p \to \infty$): SoftMax ($\frac{\sum_{t=1}^{T} \exp(p \cdot OM(t))}{\sum_{u=1}^{T} \exp(p \cdot OM(u))} \cdot OM(t)$) and LogExp ($\frac{\sum_{t=1}^{T} \exp(p \cdot OM(t))}{\sum_{u=1}^{T} \exp(p \cdot OM(u))}$).

In [9] objective metrics are clustered by k-means algorithm into two clusters, containing frames of lower and higher quality respectively. After reducing the impact of the less important higher quality frames by multiplying a weight, the scores are combined (KMeans, see detailed description in [9]). A similar approach is sequence level pooling (SequenceLevel, see detailed description in [10]) which divides the frames according to a percentage parameter instead of clustering. In [11] the score is computed from the mean of the objective metrics and the differences between successive frames. This score emphasizes quality deteriorations and can decrease down to a saturation threshold (VQA, based on detailed description in [11] but slightly modified).

### 11 List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACR</td>
<td>Absolute Category Rating</td>
</tr>
<tr>
<td>AQC</td>
<td>Abrupt Quality Change</td>
</tr>
<tr>
<td>ARMA</td>
<td>Auto-Regressive Moving Average</td>
</tr>
<tr>
<td>AVC</td>
<td>Advanced Video Coding</td>
</tr>
<tr>
<td>BE</td>
<td>Best Effort</td>
</tr>
<tr>
<td>BR</td>
<td>Nominal Bit Rate</td>
</tr>
<tr>
<td>CM</td>
<td>MOS Score of each Chunk</td>
</tr>
<tr>
<td>DASH</td>
<td>Dynamic Adaptive Streaming over HTTP</td>
</tr>
<tr>
<td>DMOS</td>
<td>Degradation Mean Opinion Score</td>
</tr>
<tr>
<td>eNodeB</td>
<td>E-UTRAN Node B</td>
</tr>
<tr>
<td>EPC</td>
<td>Evolved Packet Core</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>FR</td>
<td>Full Reference</td>
</tr>
<tr>
<td>HAS</td>
<td>HTTP Adaptive Streaming</td>
</tr>
<tr>
<td>HDS</td>
<td>Adobe HTTP Adaptive Streaming</td>
</tr>
<tr>
<td>HLS</td>
<td>HTTP Live Streaming</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>iOS device</td>
<td>Apple Device using Apple-specific iOS Operating System</td>
</tr>
<tr>
<td>ISP</td>
<td>Internet Service Provider</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>KPI</td>
<td>Key Performance Indicator</td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
</tr>
<tr>
<td>MOS</td>
<td>Mean Opinion Score</td>
</tr>
<tr>
<td>MPEG</td>
<td>Motion Picture Expert Group</td>
</tr>
<tr>
<td>MPEG-DASH</td>
<td>See DASH</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Square Error</td>
</tr>
<tr>
<td>NP</td>
<td>Network Performance</td>
</tr>
<tr>
<td>NR</td>
<td>No Reference</td>
</tr>
<tr>
<td>NSN</td>
<td>Nokia Siemens Networks</td>
</tr>
<tr>
<td>PCAP</td>
<td>Packet Capture Application Programming Interface for Capturing Network Traffic</td>
</tr>
<tr>
<td>PCC</td>
<td>Pearson’s Correlation Coefficient</td>
</tr>
<tr>
<td>PLMN</td>
<td>Public Land Mobile Network</td>
</tr>
<tr>
<td>P-SERQU</td>
<td>Project Service Quality Definition and Measurement</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>QCI</td>
<td>QoS Class Index</td>
</tr>
<tr>
<td>QL</td>
<td>Quality Level</td>
</tr>
<tr>
<td>QoE</td>
<td>Quality of Experience</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RAN</td>
<td>Radio Access Network</td>
</tr>
<tr>
<td>RDA</td>
<td>Rate Decision Algorithm</td>
</tr>
<tr>
<td>RHEL4</td>
<td>Red Hat Enterprise Linux Version 4</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Square Error</td>
</tr>
<tr>
<td>RR</td>
<td>Reduced Reference</td>
</tr>
<tr>
<td>RTT</td>
<td>Round Trip Time</td>
</tr>
<tr>
<td>SDTV</td>
<td>Standard Definition Television</td>
</tr>
<tr>
<td>SGi</td>
<td>Reference point between the EPC based PLMN and the packet data network</td>
</tr>
<tr>
<td>SINR</td>
<td>Signal to Interference Noise Ratio</td>
</tr>
<tr>
<td>SROCC</td>
<td>Spearman Rank-Order Correlation Coefficient</td>
</tr>
<tr>
<td>SSIM</td>
<td>Structural SIMilarity</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>VQA</td>
<td>Video Quality Assessment</td>
</tr>
<tr>
<td>VQM</td>
<td>Video Quality Metric</td>
</tr>
<tr>
<td>VQUAD</td>
<td>Objective Perceptual Multimedia Video Quality Measurement</td>
</tr>
<tr>
<td>WiFi</td>
<td>Wireless Fidelity</td>
</tr>
</tbody>
</table>