Crowdsensed QoE for the Community

Florian Metzger
Chair of Communication Networks
University of Würzburg, Germany
florian.metzger@uni-wuerzburg.de

Tobias Hoßfeld
Chair of Communication Networks
University of Würzburg, Germany
tobias.hossfeld@uni-wuerzburg.de

Abstract—In recent years several community testbeds as well as participatory sensing platforms have successfully established themselves to provide open data to everyone interested. Each of them with a specific goal in mind, ranging from collecting radio coverage data up to environmental and radiation data. Such data can be used by the community in their decision making, whether to subscribe to a specific mobile phone service that provides good coverage in an area or in finding a sunny and warm region for the summer holidays.

However, the existing platforms are usually limiting themselves to directly measurable network QoS. If such a crowdsourced data set would provide more in-depth derived measures, this would enable an even better decision making. A community-driven crowdsensing platform that derives spatial application-layer user experience from resource-friendly bandwidth estimates would be such a case, video streaming services come to mind as a prime example. In this paper we present a concept for such a system based on an initial prototype that eases the collection of data necessary to determine mobile-specific QoE at large scale. In addition we reason why the simple quality metric proposed here can hold its own.

Preface

This paper was originally written in 2017, but never published. We think, the idea to engage the community in crowdsensing with easy-to-understand and practical information without burdening them has even increased in merit today, and we strife to continue this work in the future.

Therefore, we wanted to preserve this work in the form it was originally prepared in 2017, with updated affiliations, restored links, and this preface. Past reviewers of this manuscript mostly criticized the small scale of this study and the lack of details and evaluation with regards to the bandwidth estimation, the QoE model, and the geographical visualization. Reviewers praised the dissemination of easily interpretable information from science to citizens using a decentralized approach.

I. Introduction

Be honest: Did you ever try using your flavor-of-the-month messaging or social app and just couldn't get it to work? Despite all information available to you claiming otherwise. Or did you want to watch a stream but only received a video in abysmal quality despite having an excellent connection?

An example: One of the author's routinely commutes to work by tram. While waiting at the underground tram station he tends to read his Twitter stream. In about 50% of cases the author's device does not receive any data despite a full signal strength and data indicator. Only after putting a significant distance between himself and the station the data transmission resumes.

If this is the case then you are in the same boat as the authors. and many other people. And lo and behold! This situation can be improved upon, continue reading to find out how.

We — as in we, the research community — have reached a point where we have a rather good understanding of video streaming Quality of Experience (QoE). Many a user study has been conducted to ascertain the subjective aspects of video quality, both in lab environments as well as through crowdsourcing. From the results of these, application layer Quality of Service (QoS) and QoE models have been derived that affix some quality rating to certain objectively measurable QoS metrics. This begs the question: What does one actually do with this information? How can it be used to one's benefit? One such benefit will be the contribution of this paper in the form of a mapping service displaying service-specific quality information derived from community-driven network measurements.

In the past video streaming QoE research was often incentivised by both video streaming service providers and home Internet access providers. The former obviously desired to understand how their service is received and thus improved their system based upon this information. The latter entities aspire to understand the traffic composition flowing through their network and the resulting direct implications on their customers, not in terms of easy-measurable-but-hard-to-interpret network QoS metrics but rather as direct customer satisfaction, e.g. QoE measures.

However, there is another, third, party that has seen not much direct benefit from these endeavors: the actual consumers of video streams. Currently, if a user watches a video she can typically only assess her own subjective video quality herself. The results from user studies and QoE models are mostly not readily available to be applied to home users directly. And this situation becomes even more interesting when looking at mobile users, where one often desires to know if one's mobile device works at a specific location. But these current mobile network conditions are very location-dependent and might fluctuate based on a great many factors. This includes spatio-temporal properties, but also radio propagation effects as well as mobility issues and the other usual kinks of a shared medium (as the multiplexed resource allocation in

today's mobile radio technologies provides).

Now take a look at the recent developments around mobile crowdsourcing, participatory crowdsensing and community-run testbeds and data-collection programs. You can partake in projects like radio coverage collection[1], current weather information[2], air quality data[3], community-run Internet access[4], or even environmental radiation monitoring[5] with little to no effort, using either one's own mobile phone or, alternatively, affordable sensor boards. Such projects flourish both through the immediate benefits one can gain through the available data and the participatory motif.

However, most of these community-driven efforts concern basic, directly measurable data. Only very rarely [2], [3] have application-layer QoS and QoE models and mappings been applied to such data for an immediate benefit of the participants. Exploring the opportunities of this crossover is the goal of this work. Specifically, for this paper we want to tackle the question of YouTube's streaming quality on-the-go on the basis of location-specific bandwidth estimates collected from participating mobile devices.

In order to do so, after highlighting all the related and similar projects as well as the necessary research foundations in Section II, we describe a mobile crowdsensing prototype — aimed at low resource usage to have a negligible performance impact in order to be scalable — that collects the network QoS data required for input to an appropriate QoE mapping in Section III. In the following Section IV the collected data is fed into a simple, but effective QoE mapping and the results evaluated. We further discuss potential approaches how these quality ratings can be appropriately visualized, especially with geospatial and other context data in mind. The work is concluded in Section V.

II. RELATED WORK AND BACKGROUND

In order for such an approach to succeed quite a few different lines of work need to be reconciled here. Namely:

- Crowdsourcing and participatory sensing methods,
- the intricacies of community networks and testbeds,
- subjective video streaming user studies,
- Resource-friendly network QoS measurements
- the relationship of network QoS to video QoE and mappings thereof.

Each of these individual sectors and work essential for this QoE-crowdsensing approach are briefly covered in this section.

A. Crowdsourcing, Crowdsensing and Experiences with Existing Community Campaigns

Targeting a participatory crowdsourcing approach for large-scale QoE-mappings is a challenging endeavor, therefore building on past experiences and best practices is crucial. For

[1] E.g. https://opensignal.com/, https://radiocells.org/, and [1].
[2] E.g. https://web.archive.org/web/20170703191919/http://weathersignal.com/about/.
[3] E.g. http://airqualityegg.com/.
[4] E.g. the German Freifunk project.
[5] E.g. https://safecast.org/.

example, crowdsourced network measurements raise the issue of trust in the collected data, as it could have been, willingly or unwillingly, falsified [4], [5]. This needs to be dealt with the proper verification and statistical methods.

The concept of participatory sensing (e.g. [6], [7]) has now been around for a few years and quite a number of projects exist now that aim to gather and make available sensed data to the community, e.g. air quality data in [8] or vehicular traffic [9]. Campaigns that aim to put a data-collecting application unto one's mobile phone have to deal with significant privacy concerns, as they usually not only have access to the devices current location but also to all communication, sensor readings and files stored on the device. One of the ways to resolve this is to give the user fine-grained control over the data to be used in the campaign, and also ensure its secure and anonymized transmission [1], [10], [11].

B. Resource-Friendliness and Mobile QoS-Measurements

In order to encourage a wide-spread proliferation of the campaign — an absolute must when dealing with location-specific cellular measurement data — the data-collecting mobile phone application should adhere to stringent resource-restrictions. This would enable an installation on a wider selection of devices as well as allows it to run in the background without interfering with other daily activities of the participants. This especially concerns both the battery life as well as the used radio resources (which in turn generates high energy usage as well) and traffic data caps. To tackle the latter, direct throughput measurements need to be avoided. Instead, various bandwidth estimation methods come to mind that attempt to compute the actual currently possible throughput through specific transmission side effects — like the inter-arrival time of two consecutively sent TCP segments in the Packet Pair method [12]–[14]. This works quite well in stable wired networks, but can have issues in radio networks (cf. also [15] and Section III-A). Concerning energy consumption, [16], [17] gives some pointers as to the root causes of these in relation to smartphone applications.

C. Subjective Studies on Video Streaming Quality and Quality Models

After the network QoS data has been collected it needs to be transformed to the desired target QoE metric. The goal in this work was to project YouTube streaming quality. So, for an adequate mapping of measured transmission characteristics to be defined, first the service's properties have to be fully understood and described. Thankfully, many prior publications on YouTube's adaptive streaming mechanisms readily exist, e.g. [18], [19]. Indeed, many interactive YouTube subjective quality user studies have been conducted in the past [2], [3], and guidelines to conduct further crowdsourced subjective video quality studies also exist [20]. And they all paint a good picture of YouTube's streaming experience. However, they are hard to scale up to much more participants, due to the time and resource investments required by the users, amongst other reasons. Of special interest to this work are

studies that link the user experience to bandwidth as is for example done in [21]. This study finds, that $4\,\mathrm{Mbit/s}$ produce a near-optimal experience in their scenario. Yet other studies (e.g. [22]) suggest that it is not the value of throughput itself that is decisive, but instead the throughput variations that have a large impact on the user experience. However, other work also suggests that the correlation of throughput could actually be almost neglible [23]. Indeed, many other operational factors and also especially anomalies can have a large impact on the QoE (as e.g. shown in the case of YouTube in [24]).

To achieve independence of this interactive user component, models that appropriately map network QoS to an application layer representation of quality are necessary. While there are models for non-adaptive streaming that map the number and length of stalling events to a Mean Opinion Score (MOS) value [25], research on good adaptive streaming quality models has not yet progressed very far. However, works exist that discuss QoE modeling for video streaming in general [26]. A further survey [27] describes the intricacies of HTTP Adaptive Streaming (HAS) that are necessary to understand in order to appropriately conduct QoE modeling. For the purposes of the work in this manuscript we aim for a simple available-throughput-to-achievable-video-quality mapping. On this matter it is also interesting to find *acceptable* levels of QoE, or specifically of MOS, to a user group in order to link this level to metrics that a user of a participatory crowdsensing platform can easily understand.

For a mobile QoE-crowdsensing campaign it is not only beneficial to conduct regular, location-based network QoS measurements, but also to couple this with further context data, i.e. device information and sensor readings. The importance of such *context factors* in QoE monitoring has been discussed in, e.g., [28]. In this specific case context information can be used to accurately handle the "tunnel scenario" [29], [30] to provide even better quality predictions to the community in the future.

III. The Crowdsensing Mobile App Prototype

A community crowdsensing service faces several challenges in its task of gathering and processing the data before making it available in a concise and easy-to-understand visualized way. The goal for the gathering portion is to measure in an automated, non-interactive fashion in order to be non-intrusive to the participants as well as reach a larger audience. This allows for a larger geographical coverage of the project service quality values while simultaneously being more conservative with the available resources. The implementation of these aspects is discussed in this section.

A. The Issue of Bandwidth Measurements

The most important directly measurable network QoS metric for a TCP-based video streaming service is arguably the achievable goodput of this flow. Typically, this would be calculated by downloading a file from a remote location and dividing its size by the time it took to transmit it. This is inherently an average value over a certain time period, which makes

it tricky to pinpoint it to specific spatial coordinates when this is employed during a mobile crowdsensing campaign.

Furthermore, it is especially crucial to not waste the participants monthly data caps solely on the estimation of video streaming quality. In some countries the monthly contractual data caps are set extremely low. For example, in Germany postpaid contracts with a data cap of $1\,\mathrm{GB}$ currently cost a monthly fee of around $25\,€$. On the other hand, only measuring the user's actually watched video streams would provide an insufficient amount of samples for providing accurate data for mobile networks. This means that in order to widen the possible audience, direct throughput measurements of streaming services are out of the question. Our measurement prototype attempts to solve these issues by implementing several bandwidth estimation methods, which trade a loss of accuracy for much less data and time usage. The rest of this section aims to evaluate if the precision of these methods is sufficient for the QoE crowdsensing endeavor.

1) Bandwidth Estimation Methods: Basics and Variants: These bandwidth estimators can be roughly divided into two different categories. Approaches from the first category, namely *Cross-Traffic Estimation*, actively send a small amount of probe packets with fixed inter-packet times. On the bottleneck link the timing of these are altered through network throughput limitations as well as traffic from other sources, from which the bandwidth can then be calculated. One well-known representative of this category is Packet Pair [31], [32]. The second set of approaches, termed *Self-Induced Congestion*, relies also on active probes, but here they are meant to briefly congest the network themselves in order to observe the resulting path characteristics. Packet Train [33] is one such variant implemented here.

It should be noted that some methods require the presence of an actively sending server under the control of the measurement client with the possible benefit of a better measurement accuracy, making it unfeasible to measure public services like YouTube with them. Packet Pair is one of them. However, it is also usually possible to circumvent this requirement by exploiting known properties of TCP. For example, some approaches [34], [35] force the transmission of Transmission Control Protocol (TCP) RST packets through specifically crafted TCP SYN packets and then time the interaction.

B. Prototype and Estimator Evaluation

To evaluate the feasibility of these methods as well as to establish a foundation for the participatory crowdsensing aspects a prototype was set up. This was realized as a Java-based Android application. Both the prototype, with the bandwidth estimation and data collection and upload methods[6], and the server-side components required for some of the bandwidth estimation methods[7], are available online.

Evaluating the addressed estimation methods is not as straight-forward as one might expect in a mobile environment

---

[6]https://github.com/mas-ude/bw-estimation-App
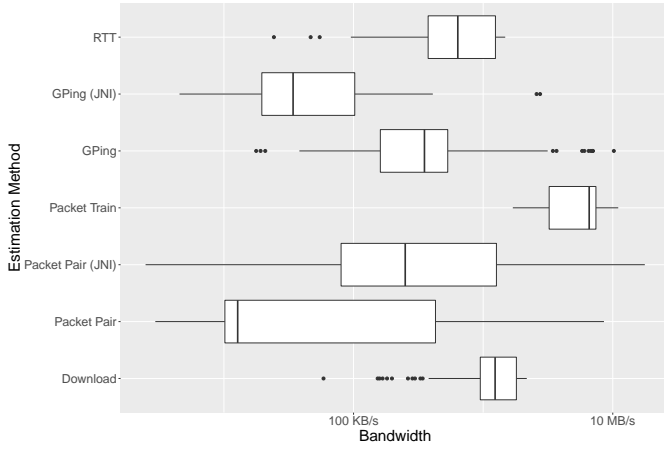[7]https://github.com/Nobodi/Bandwith-Estimation

Figure 1: Boxplots of the various tested throughput estimation methods in the evaluation scenario. Entries denoted with JNI use system-level packet and timing facilities in an attempt to improve accuracy.

Table I: Data usage of the estimation methods.

	Used Volume (kB)	Duration (s)
Download	10 240	9.42
RTT	256	0.36
Packet Pair	6	0.37
GPing	2.5	0.14
Packet Train	437	0.89

due to the many factors that can influence the results to a small or large degree. To compensate for these factors this evaluation was conducted in a more controlled setting. All evaluations ran on a single mobile phone, an *LG G3*, and were fixed to a single location to avoid varying radio conditions. Similarly, all corresponding runs were closely grouped together to ensure reasonable temporal coupling. These runs were conducted in five-minute intervals for roughly one day. Various methods described in the literature (as mentioned above) were tested in comparison to the results from simple download tests with a large file. The method dubbed 'RTT' is a variant of download, but is using very small files, transferred from a controlled server environment, to calculate its throughput estimate, making it potentially less accurate.

Figure 1 depicts the results attained from this scenario. The inter-quartile ranges of the accumulated samples reveal, with every method except RTT, rather large deviations from the download results. The root cause is unclear in this limited evaluation, but this kind of estimation error does not come entirely unexpected due to the aforementioned temporal insta-bility of mobile networks, even in a static location. All of the applied methods generally expect certain network properties to hold that are unfortunately only loosely applicable to mobile networks, with their strong form of a signaling plane. On ap-proach to remedy this situation is to apply an adjustment factor to the individual methods if the offset to the download results proves to be stable enough in a wider range of scenarios.

Table II: Recommended YouTube video bitrates for 30 fps video, mean stalling frequency, and ratio of bandwidth samples above $1.25 \times V$ in the measurement data.

Res.	Video Bitrate V (Mbit/s)	Stalling Frequency F	Ratio of B above $1.25 \times V$
2160p	45	0.35	0
1440p	16	0.23	0
1080p	8	0.09	0.34
720p	5	0.04	0.67
480p	2.5	0.01	0.90
360p	1	0.00	0.98

However, one other benefit is clear when looking at the data and time used during the measurements in Table I. The intended goal of being more resource-friendly in order to target a wider audience can be easily fulfilled with these methods, especially with the single-pair approaches Packet Pair and GPing, allowing for more frequent probing and thus more coverage on a crowdsensed community QoE map.

IV. QUALITY MODEL, MAPPINGS AND MAP

Speaking of community maps, this bring us to the final tasks of interpreting the collected data as YouTube quality and presenting it in an easy-to-grasp way to the community. As discussed in Section II there do exists metrics that derive a higher layer abstraction from directly measurable values, such as the bitrate. E.g. [36] fitted the **stalling frequency** F from throughput samples as

$$F(X) = -1.09e^{-1.18\frac{V}{B}} + 0.36$$

for transmission bandwidths B and video bitrates V. But a stalling frequency is nothing that can be easily understood in terms of YouTube quality by the community, and therefore a simpler metric is desired. For this let's consider the **reception rate** $\rho = \frac{B}{V}$ [25]. This means that as long as the transmission rate is above the video rate the video streaming should work well, bar any variations. This is even more so if you include a safety margin as evaluated in, e.g., [37]. A 25 % surplus of the transmission rate should therefore take care of this and ensure the video to be playable at the selected quality level at all times. This then yields a definition of the playable video quality $v_{playable}$ for our system as

$$v_{playable} = \max\left(\{v | v \in \mathbb{V} \wedge 1.25 \cdot v < B\}\right).$$

Used as the set of V is here simply YouTube's recom-mended video encoding bitrates[8] as listed in Table II.

Each spatial region now has to have a series of bandwidth samples attached to it, taken at different points of time. Figure 2 demonstrates such a time series for one location as both the stalling frequency and the reception rate with the discussed threshold of 1.25. This also clearly demonstrates

[8]Based on information from https://support.google.com/youtube/answer/1722171?hl=en. Do note that these are recommendations for the video upload before they are transcoded by YouTube. But this gives a rough estimate for the download bitrate as well.

(a) Stalling frequency (according to [36]) time series.

(b) Reception ratio (according to [25]) time series.

(c) Corresponding CDF of the stalling frequency.
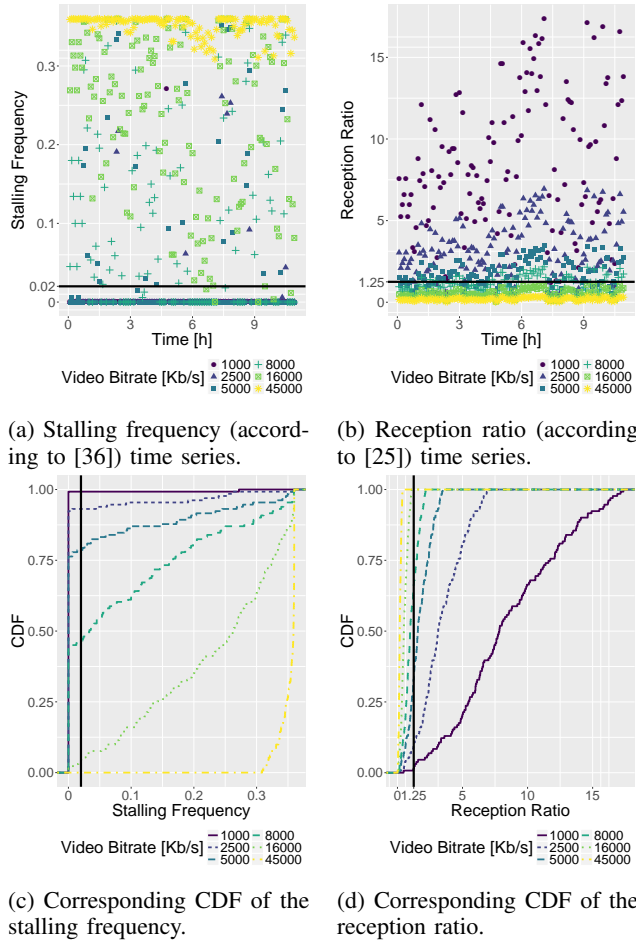
(d) Corresponding CDF of the reception ratio.

Figure 2: Time series and CDF depictions of the two derived video streaming quality metrics, matching the throughput to the desired video video bitrate. Horizontal — respectively vertical — lines denote the selected cutoff threshold for an acceptable streaming quality.

large temporal fluctuations for the playable video quality, which should also be exposed in the community quality map, e.g. through some additional, advanced information overlays for each location. Yet, the default geographical map should be kept free of such temporal data to avoid overwhelming the users.

In order to provide a clutter-free representation the mapping service needs to decide for just one specific video quality rating at each location. This requires some form of temporal averaging to be conducted. Here, the system will color the map in the highest video quality of which $90\,\%$ of the bandwidth samples over the measured period were above $1.25 \times V$.

Both the more complex stalling frequency — using a similar threshold of $2\,\%$ here — and the reception ratio representation result in the same quality decisions here. Moreover, the two metrics have a Pearson correlation coefficient of -0.92. This means that even this simple metric is sufficiently able to visualize the service's quality, and no more complex metric needs to be employed here. With this, the results also become easily
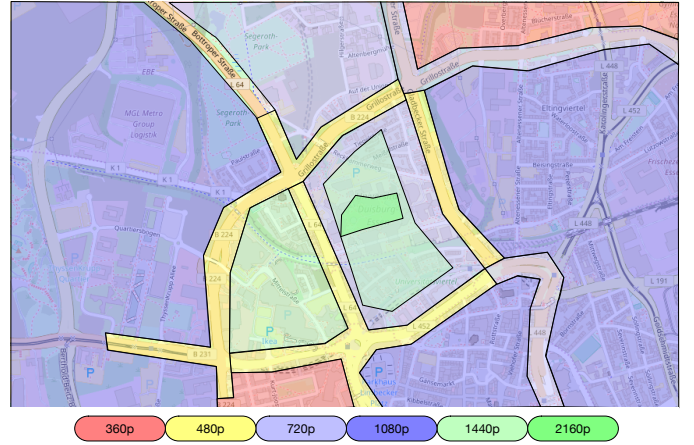

Figure 3: Mockup map of achievable YouTube video resolution for specific regions for a mobile provider. Regions should be grouped and interpolated from individual participants' measurements.

comparable to other approaches (e.g. YouSlow[2]) that rely on directly measuring the rebuffering rate (which is akin to the stalling frequency F). A mockup of the resulting geographical map and video quality overlay is shown in Figure 3. Common caveats of geographical grouping and interpolation of samples also apply to this type of video quality map and still need to be solved.

Now consider this map for a hypothetical use case of someone having to move to a new place. It is common today to check the new apartment's location for good internet access, both wired as well as mobile. Using our approach that person would then not just be able to tell if there is connectivity in that area — as in most other participatory sensing approaches — but also if a specific service works in an acceptable quality there, especially at the time of day when one is at home.

While YouTube does serve as the example here, it is important to note that, with the available throughput data, similar mappings and visualizations can easily be achieved for any other kind of service. This is just a matter of switching the mapping function to another, appropriate one for that service. This would allow the community crowdsensed map to be easily extensible and able to be personalized to show the quality of one's favorite services. All while still avoiding to collect resource-intense service-specific data and therefore able to easily scaled up to a large community.

V. CONCLUSION

The approach taken here attempts to combine the strengths of participatory crowdsensing with the knowledge garnered by the QoE community in recent years in order to provide the public with a type of information — the attainable YouTube video quality at a specific location in a mobile network— that has much more immediate utility than providing simple measured network QoS samples. Such a large-scale crowdsensing endeavor can only be achieved through some compromises

as they have been taken here. Actively conducting subjective video QoE user studies at each location is equally unfeasible as is being wasteful with the participants' devices' resources, especially with stringent bandwidth data caps in place.

All in all, we think that our approach of conducting large scale low-volume bandwidth estimation measurements and transforming them through adequate QoE mapping might be much more engaging for a wider audience and thus indicative of the future to come.

REFERENCES

[1] A. Rafetseder *et al.*, "Sensorium – a generic sensor framework," *PIK - Praxis der Informationsverarbeitung und Kommunikation*, vol. 36, no. 1, p. 46, Feb. 2013.

[2] H. Nam *et al.*, "YouSlow: A performance analysis tool for adaptive bitrate video streaming," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, ser. SIGCOMM '14, Chicago, Illinois, USA: ACM, 2014, pp. 111–112.

[3] F. Wamser *et al.*, "YoMoApp: A tool for analyzing QoE of YouTube HTTP adaptive streaming in mobile networks," in *2015 European Conference on Networks and Communications (EuCNC)*, Jun. 2015, pp. 239–243.

[4] C. Tanas and J. Herrera-Joancomartí, "Crowdsensing simulation using ns-3," in *Citizen in Sensor Networks: Second International Workshop, CitiSens 2013, Barcelona, Spain, September 19, 2013, Revised Selected Papers*, J. Nin and D. Villatoro, Eds. Cham: Springer International Publishing, 2014, pp. 47–58.

[5] M. Hirth *et al.*, "Crowdsourced network measurements: Benefits and best practices," *Computer Networks*, vol. 90, pp. 85–98, 2015, Crowdsourcing.

[6] P. Dutta *et al.*, "Common sense: Participatory urban sensing using a network of handheld air quality monitors," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09, Berkeley, California: ACM, 2009, pp. 349–350.

[7] K. Shilton, "Four billion little brothers?: Privacy, mobile phones, and ubiquitous data collection," *Commun. ACM*, vol. 52, no. 11, pp. 48–53, Nov. 2009.

[8] J. A. Burke *et al.*, *Participatory sensing*, 2006.

[9] J. C. Herrera *et al.*, "Evaluation of traffic data obtained via GPS-enabled mobile phones: The mobile century field experiment," *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 4, pp. 568–583, 2010.

[10] A. Rafetseder *et al.*, "Sensibility testbed: A programmable sensor laboratory for android devices," in *2016 Microelectronic Systems Symposium (MESS 16)*, 2016.

[11] Y. Zhuang *et al.*, "Privacy-preserving experimentation with Sensibility Testbed," in *;login: the USENIX magazine*, vol. 40, Berkeley, CA: USENIX Association, Aug. 2015, pp. 18–21.

[12] K. Lai and M. Baker, "Nettimer: A tool for measuring bottleneck link bandwidth," in *In Proceedings of the USENIX Symposium on Internet Technologies and Systems*, 2001, pp. 123–134.

[13] V. J. Ribeiro *et al.*, "Spatio-temporal available bandwidth estimation with STAB," in *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '04/Performance '04, New York, NY, USA: ACM, 2004, pp. 394–395.

[14] J. Strauss *et al.*, "A measurement study of available bandwidth estimation tools," in *Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '03, Miami Beach, FL, USA: ACM, 2003, pp. 39–44.

[15] M. Zhong *et al.*, "Revisited: Bandwidth estimation methods for mobile networks," in *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, Jun. 2014, pp. 1–6.

[16] C. Schwartz *et al.*, "Angry apps: The impact of network timer selection on power consumption, signalling load, and web QoE," *Journal of Computer Networks and Communications*, vol. 2013, p. 13, 2013.

[17] C. Schwartz *et al.*, "Smart-phone energy consumption vs. 3G signaling load: The influence of application traffic patterns," in *2013 24th Tyrrhenian International Workshop on Digital Communications - Green ICT (TIWDC)*, Sep. 2013, pp. 1–6.

[18] C. Moldovan *et al.*, "Youtube can do better: Getting the most out of video adaptation," in *2016 28th International Teletraffic Congress (ITC 28)*, vol. 03, Sep. 2016, pp. 7–12.

[19] C. Sieber *et al.*, "Sacrificing efficiency for quality of experience: YouTube's redundant traffic behavior," in *2016 IFIP Networking Conference (IFIP Networking) and Workshops*, May 2016, pp. 503–511.

[20] T. Hoßfeld and J. Redi, "Journey through the crowd: Best practices and recommendations for crowdsourced QoE," in *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, May 2015, pp. 1–2.

[21] P. Casas *et al.*, "Exploring QoE in cellular networks: How much bandwidth do you need for popular smartphone apps?" In *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*, ser. AllThingsCellular '15, London, United Kingdom: ACM, 2015, pp. 13–18.

[22] P. Casas *et al.*, "On the quest for new KPIs in mobile networks: The impact of throughput fluctuations on QoE," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, Jun. 2015, pp. 1705–1710.

[23] M. Seufert *et al.*, "Application-layer monitoring of QoE parameters for mobile youtube video streaming in the field," in *2016 IEEE Sixth International Conference on Communications and Electronics (ICCE)*, Jul. 2016, pp. 411–416.

[24] P. Casas *et al.*, "When youtube does not work — analysis of QoE-relevant degradation in google CDN traffic," *IEEE Transactions on Network and Service Management*, vol. 11, no. 4, pp. 441–457, Dec. 2014.

[25] T. Hoßfeld *et al.*, "Internet video delivery in YouTube: From traffic measurements to quality of experience," in *Data Traffic Monitoring and Analysis: From Measurement, Classification, and Anomaly Detection to Quality of Experience*, E. Biersack *et al.*, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 264–301.

[26] P. Reichl *et al.*, "Towards a comprehensive framework for QoE and user behavior modelling," in *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, May 2015, pp. 1–6.

[27] M. Seufert *et al.*, "A survey on quality of experience of http adaptive streaming," *Communications Surveys Tutorials, IEEE*, vol. 17, no. 1, pp. 469–492, 2015.

[28] T. Hoßfeld *et al.*, "Can context monitoring improve QoE? a case study of video flash crowds in the Internet of services," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, May 2015, pp. 1274–1277.

[29] E. Liotou *et al.*, "Enriching HTTP adaptive streaming with context awareness: A tunnel case study," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6.

[30] F. Metzger *et al.*, "TCP video streaming and mobile networks: Not a love story, but better with context," *Computer Networks*, vol. 109, Part 2, pp. 246–256, 2016, Traffic and Performance in the Big Data Era.

[31] J.-C. Bolot, "End-to-end packet delay and loss behavior in the Internet," *SIGCOMM Comput. Commun. Rev.*, vol. 23, no. 4, pp. 289–298, Oct. 1993.

[32] K. Lai and M. Baker, "Measuring bandwidth," in *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, Mar. 1999, 235–245 vol.1.

[33] M. Jain and C. Dovrolis, "Pathload: A measurement tool for end-to-end available bandwidth," in *In Proceedings of Passive and Active Measurements (PAM) Workshop*, 2002, pp. 14–25.

[34] S. Saroiu *et al.*, "Sprobe: A fast technique for measuring bottleneck bandwidth in uncooperative environments," in *IEEE INFOCOM*, 2002.

[35] S. Chakravarty *et al.*, "LinkWidth: A method to measure link capacity and available bandwidth using single-end probes," Tech. Rep., 2008.

[36] T. Hoßfeld *et al.*, "Transport protocol influences on YouTube QoE," *University of Würzburg, Tech. Rep*, vol. 482, 2011.

[37] T. Zinner *et al.*, "Requirement driven prospects for realizing user-centric network orchestration," *Multimedia Tools and Applications*, vol. 74, no. 2, pp. 413–437, 2015.