Live UGC Stream Selection Using Quality Metadata

Marcus Thaler, Andras Horti, Albert Hofmann, Stefanie Wechtitsch, Werner Bailer
JOANNEUM RESEARCH, DIGITAL – Institute for Information and Communication Technologies
Steyrergasse 17, 8010 Graz, Austria
{firstname.lastname}@joanneum.at

Wolfram Hofmeister, Jameson Steiner, Reinhard Grandl
Bitmovin
Lakeside B01
9020 Klagenfurt, Austria
{firstname.lastname}@bitmovin.com

ABSTRACT
Many types of events such as concerts, festivals or sports events are spatially spread out, and user generated content (UGC) can be used to complement traditional broadcast capture. We address the issue of integrating user generated live streams into production environments by performing automatic content selection based on quality metadata. We demonstrate a system for capturing content together with sensor metadata and perform automatic quality analysis. A web application performs content selection based on the analysis metadata and visualises time-based metadata.

ACM Classification Keywords
H.5.1 Information Interfaces and Presentation: Multimedia Information Systems; I.4.1 Image Processing and Computer Vision: Digitization and Image Capture

Author Keywords
user generated content, mobile, content selection, live

INTRODUCTION
Many types of events such as concerts, festivals or sports events are spatially distributed, so that they are not easy to cover with traditional broadcast capture setups. User generated content (UGC) is a valuable source for improving the coverage of such events and can complement the professional coverage. Recently, smart-phone apps like Meerkat[1] and Periscope[2] have raised the interest for live UGC. However its integration in production systems is still challenging. In particular, live UGC needs to be filtered in order not to overwhelm the production team. Quality is one important criterion for filtering, and the location of the recording is another. Due to time constraints, it is not feasible to perform the filtering task manually, but one must rely on automatic tools. The metadata needed by these tools must either be captured with the content (e.g. using the sensors of the mobile device), or extracted from the content. The obtained quality metadata can then be used to select the best of the available streams at the same or nearby locations.

This paper is organised as follows. In Section Analysis System we describe a system for capturing content and metadata, and performing quality analysis on the mobile device and further analysis on the server side. We then present in Section Content Selection and Data Visualisation how content selection is performed and visualised. We then describe the demonstration setup and conclude the paper.

ANALYSIS SYSTEM
We demonstrate a system for capturing live audio and video streams on a mobile device, performing automatic metadata extraction in real-time and indexing the metadata for access by a production system. The system receives an audio, video and metadata stream from the mobile device, and extracts additional metadata from the audiovisual content. All metadata is available as a stream (with low latency from the extraction), and is indexed in a metadata store in the processing system. Metadata needed in the real-time process can be read directly from the stream, and earlier metadata can be queried from the store. The metadata is used to automatically filter content that matches defined quality levels, to select the best stream among alternative ones and to provide a set of content options. We decided to build on an existing framework with many standard components and able to handle the decoding of commonly used media formats. The GStreamer[3] open source multimedia framework was chosen for this purpose.

The system consists of a dedicated capture app, which sends video, audio and metadata as separate streams. This saves the muxing/demuxing effort and also facilitates processing different modalities on different machines in the cloud. All data are provided as RTP streams. The processing system performs the necessary decoding and transformation for the content, and also includes a set of interconnected analysis modules. These modules may not only use the content as input, but may also use metadata from the device or from other modules. All extracted metadata are provided as streams again, and a logging module listens to these streams and indexes data in the metadata store. The audiovisual streams can be connected to viewers or to an editing system. A web application performs content selection and displays the audiovisual data together with the extracted metadata.

The integrated capture application for Android enables users to perform quality analysis while capturing sensor data and streaming captured video. The application continuously measures sharpness, noise, luminance, exposure and detects the use of brightness compensation before streaming captured video [1]. The main features are: (a) audio and video recording, via the built-in microphone and camera respectively, (b)

[1] https://meerkatapp.co
[2] https://www.periscope.tv
[3] http://gstreamer.freedesktop.org

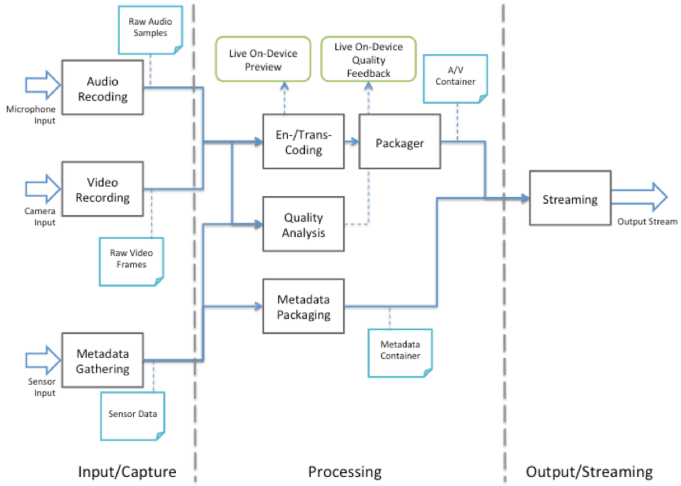

Figure 1. High-level architectural overview of the capture application

metadata capturing from different sensors available on the device, (c) on-device analysis of captured essence to meet quality constraints, (d) en-/transcoding and packaging of recorded content and (e) the up-streaming functionality to servers for processing. The high-level architecture of the capture application is shown in Figure 1.

Raw video and audio data is captured through the camera and microphone of the device and encoded using Android's *MediaCodec API*, while at the same time the quality of encoded video frames is analysed. Once the encoding for a frame has finished, it is committed into the buffer and/or sent to an RTP packager. Synchronization is done by keeping track of the latest PTS for each stream.

The server-side processing system is based on GStreamer, using existing decoding modules, and implementing new modules for metadata extraction, synchronisation and metadata handling. The metadata extraction functionality is provided by existing content and quality analysis algorithms (cf. [1]), which have been integrated into the real-time framework. The functionalities include cut detection, key frame detection, detection of MPEG-7 visual descriptors, sharpness estimation (more precise than on the mobile device) and macroblocking detection. A GStreamer module at the end of the processing chain consumes all metadata streams and indexes data in the metadata store. In order to obtain an overall quality measure of a user generated video for a specific time segment, all available measures are considered. Therefore, the metadata received from the mobile device directly as well as the more complex quality measures obtained after transmitting to the server are fused.

CONTENT SELECTION AND DATA VISUALISATION

The data exchange between the analysis platform and the operator's backend is realised via a metadata store. This metadata store is a persistent repository accessible over a REST interface. To support fast in- and output transactions, the database is a Redis in-memory data structure store. Metadata for the current time window is kept in Redis for efficient insert and querying of recent metadata. For live productions, all required metadata can be kept in this store. To reduce the load on the Redis store, older data is archived into a MySQL database. This database can be queried using the same interface, but with slower response time. In a live production scenario this may be needed for querying metadata of clips for replay or for collecting material for a summary to be included at the end of the production.

Having multiple concurrent live streams, a requirement resulting from the use of user generated content and relevant for the demonstration is to select the best-quality stream out of these. For such a decision, we can rely on the rich set of metadata, the generated overall quality measures for each live stream and a set of rules for content selection. If applied for a production system, multiple criteria need to be evaluated in order to present a restricted and pre-filtered list of suitable streams to an operator for final selection. This would also include location based filtering. In the demonstration we focus on selecting the best-quality stream. Additional temporal filtering of selection decisions prevents switching continuously between similarly ranked streams.

The visualisation of a live video stream and time-aligned metadata is done by the HTML5 metadata viewer (see Figure 2). The metadata store is polled in defined intervals for recent data and the UI is updated accordingly. Both quality annotations done at the mobile client and server are shown. For each annotation type, a line chart with the continuous evolution of the quality measure is shown, combined with an additional event view to spot segments that do not meet predefined quality standards. This high level of detail is shown for the currently selected stream out of the list of concurrent live streams for an event. If other concurrent streams are available, the web front-end is capable of continuously monitoring the overall quality measures of these streams and of automatically switching to the best-quality stream. These overall quality measures of the alternative streams are also visualised in a compact line chart at the bottom of the user interface.

Since the metadata store demonstrator is implemented as a HTML5 viewer, the incoming media stream (see top-right of Figure 2) is re-streamed by the analysis platform. This can be done as RTP stream with very low latency (requiring a browser plugin) or providing a stream for consumption by an HTML video player, with possibly higher latency.

DEMO SETUP

The demo shown at the workshop allows users to get hands-on experience of the system described in the previous chapters. Mobile devices running the integrated capture and streaming application for Android are provided. The application package is also available for download to users who want to test the system on their own devices. In order to enable contributing users to perform quality analysis while capturing sensor data and streaming captured video, an Android operating system with an application programming interface (API) level 19 or higher is required (Android 4.4 KitKat). For each quality measure, an overlay including a descriptive icon and message is displayed to immediately notify the user if the quality of the captured content is not within the expected limits (see Figure 3). In this way, users can interactively experience how

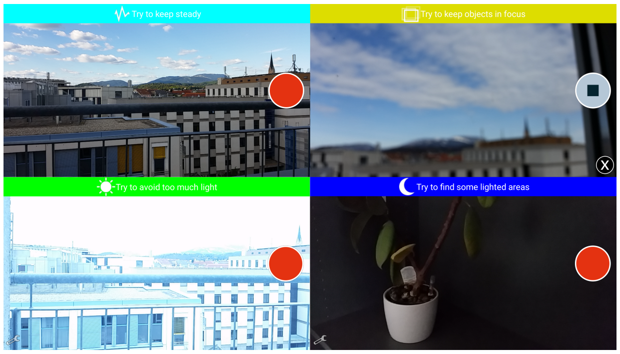

Figure 3. Screenshot of the Android capture app, including quality feedback. In case the quality metric violates the thresholds a message is shown: instability (upper left), out of focus (upper right), too bright (lower left), too dark (lower right)

various factors contribute to objective quality degradations by getting immediate feedback on the device. For example, by shaking the mobile phone or generating bad lighting conditions visual quality impairments can be forced.

For the demonstration, also machines for real-time processing of the captured live streams with the advanced algorithms described in Section Analysis System are provided – one high-performance notebook suffices for handling a limited number of mobile devices as used in the demo, including metadata storage. By using the HTML5 metadata viewer described in the previous section, we can demonstrate our approach of selecting the best quality content out of multiple streams. Users can test on how to gradually diminish the quality of their streams until the system switches to another stream. The web front-end is shown in a standard web browser. Since this front-end is HTML5 compatible, it can be viewed on any device connected to the network of the demonstration system.

CONCLUSION

In this paper, we have described a technical demonstration for automating content selection in order to complement the professional coverage of live events such as concerts, festivals or sports events with user generated content. We have described the demo setup and provided an overview of the used system for capturing content together with sensor metadata and performing automatic quality analysis. The system creates additional metadata from the audiovisual content, and all available metadata are then used to automatically select and rank streams using overall quality measures. A web application performs content selection based on the analysis metadata and visualises time-based metadata. Users can interactively participate in the demonstration and get immediate response on a mobile device regarding quality impairments in different recording situations. This can be valuable feedback if including user generated content is planned in live production processes. While the demonstrated content selection functionality focuses on quality only, the approach is suitable for including various other cues (e.g., location) for content selection.

ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 610370, ICoSOLE ("Immersive Coverage of Spatially Outspread Live Events", `http://www.icosole.eu`).

REFERENCES

1. Stefanie Wechtitsch, Hannes Fassold, Marcus Thaler, Krzysztof Kozłowski, and Werner Bailer. 2016. Quality Analysis on Mobile Devices for Real-Time Feedback. In *MultiMedia Modeling*. Springer, 359–369.

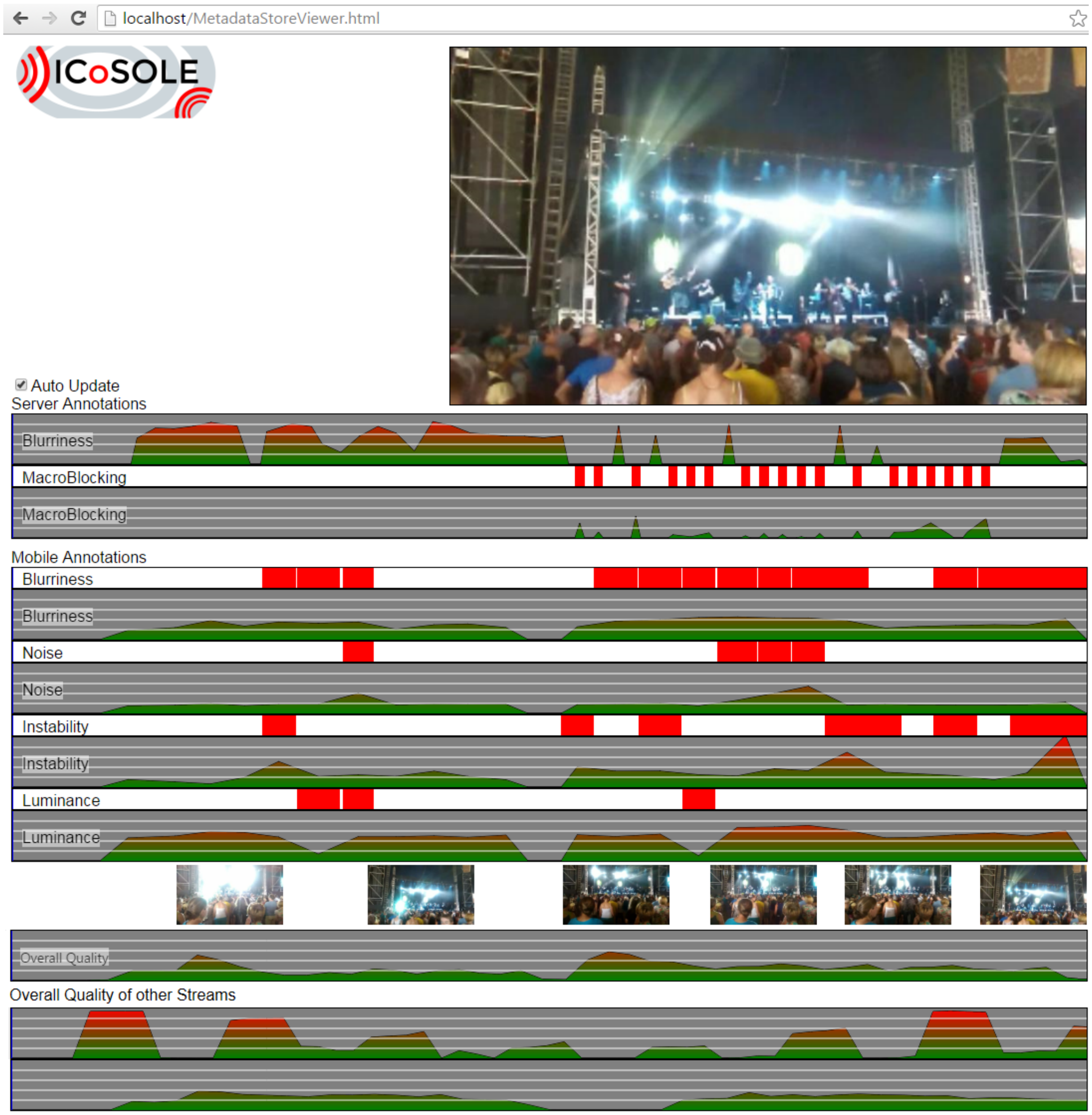

Figure 2. Web-based content and metadata visualisation. For each quality metric, a line chart with the continuous evolution of the measurement is shown. An additional event view on the top of each quality metric highlights segments that do not meet predefined quality standards (indicated by a red bar).