Towards A Format-agnostic Approach for Production, Delivery and Rendering of Immersive Media

Omar A. Niamut
TNO
Technical Sciences
Delft, The Netherlands
omar.niamut@tno.nl

Rene Kaiser
JOANNEUM RESEARCH
Institute for Information and Communication Technologies
Graz, Austria
rene.kaiser@joanneum.at

Gert Kienast
JOANNEUM RESEARCH
Institute for Information and Communication Technologies
Graz, Austria
gert.kienast@joanneum.at

Axel Kochale
Technicolor Research and Innovation
Hannover, Germany
axel.kochale@technicolor.com

Jens Spille
Technicolor Research and Innovation
Hannover, Germany
Jens.spille@technicolor.com

Oliver Schreer
Fraunhofer Heinrich-Hertz Institute,
Berlin, Germany
Oliver.Schreer@hhi.fraunhofer.de

Javier Ruiz Hidalgo
Universitat Politècnica de Catalunya
Barcelona, Spain
j.ruiz@upc.edu

Jean-Francois Macq
Alcatel-Lucent Bell Labs
Antwerp, Belgium
jean-francois.macq@alcatel-lucent.com

Ben Shirley
University of Salford
Manchester, UK
B.G.Shirley@salford.ac.uk

ABSTRACT
The media industry is currently being pulled in the often-opposing directions of increased realism (high resolution, stereoscopic, large screen) and personalization (selection and control of content, availability on many devices). We investigate the feasibility of an end-to-end format-agnostic approach to support both these trends. In this paper, different aspects of a format-agnostic capture, production, delivery and rendering system are discussed. At the capture stage, the concept of layered scene representation is introduced, including panoramic video and 3D audio capture. At the analysis stage, a virtual director component is discussed that allows for automatic execution of cinematographic principles, using feature tracking and saliency detection. At the delivery stage, resolution-independent audiovisual transport mechanisms for both managed and unmanaged networks are treated. In the rendering stage, a rendering process that includes the manipulation of audiovisual content to match the connected display and loudspeaker properties is introduced. Different parts of the complete system are revisited demonstrating the requirements and the potential of this advanced concept.

Categories and Subject Descriptors
H.5.1 [**Information Interfaces And Presentation**]: Multimedia Information Systems – *video, immersive media, interactive media*

General Terms
Experimentation, Verification.

MMSys'13, February 27 – March 1, 2013, Oslo, Norway.

Keywords
Immersive media, ultra-high definition, panoramic imaging, spatial audio, content analysis, media aware networking, gesture-based interaction, virtual director.

1. INTRODUCTION
Within the broadcast industry it is an often-expressed view that a common video production format is adopted, one that is unified across the world and supports a wide range of applications. Although some systems already allow repurposing for other devices, they are not ideal for supporting extreme variations in viewing device, e.g. from mobile phones to ultra-high-resolution immersive projection systems with 3D audio support. Audiences increasingly expect to be able to control their experience, by selecting one of several suggested areas of interest, or even by freely exploring the scene themselves. Traditionally-produced content offers very limited support for such functionality. Whilst such a degree of freedom may not be appropriate for all kinds of content, it has the potential to add useful interactivity to any kind of programme where no single 'best' shot will satisfy all viewers.

The so-called *'format agnostic'* approach [1] can help in overcoming the limitations of current production systems, as it allows to merge video signals from different sensors with different spatial and temporal resolution. In such a new production system, the resolution, field-of-view, aspect ratio, frame rate and colour depth of the captured image are chosen based on the requirements of the particular production, rather than being tailored to a particular delivery format. This approach requires a paradigm shift in video production, towards capturing a format-agnostic representation of the whole scene from a given viewpoint, rather than the view selected by a cameraman based on assumptions about the viewer's screen size and interests. Ideally, a format-agnostic representation of a scene involves capturing a wide angle view of the scene from each camera position, sampled at a sufficiently high resolution so that any desired shot framing and resolution can be obtained. However, this is impractical and wasteful, as less interesting areas of the scene would be captured at the same high resolution as the key areas of interest.

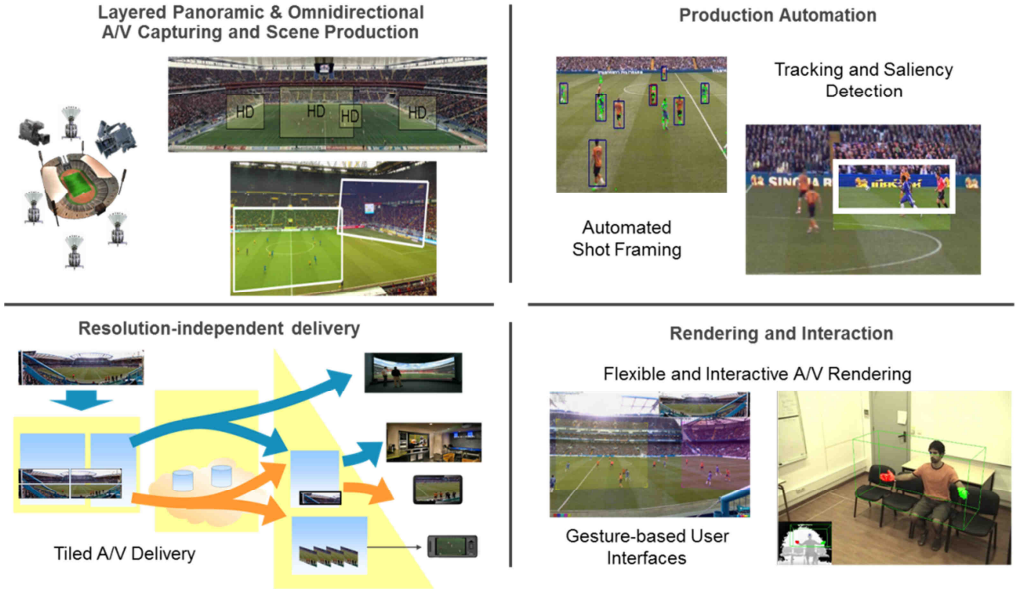


Figure 1 – Key innovation areas and system aspects.

This leads to the concept of a *'layered'* scene representation, where several cameras with different spatial resolutions and fields-of-view can be used to represent the view of the scene from a given viewpoint. The views from these cameras can be considered as providing a 'base' layer panoramic image, with 'enhancement layers' from one or more cameras more tightly-framed on key areas of interest. Other kinds of camera, such as high frame-rate or high dynamic range, could add further layers in relevant areas. This layered concept can be extended to audio capture, by using a range of microphone types to allow capture of the ambient sound field, enhanced by the use of additional microphones to capture localized sound sources at locations of interest. This allows an audio mix to be produced to match any required shot framing, in a way that can support reproduction systems ranging from mono, through 5.1, to higher order Ambisonics or wave field synthesis.

This paper presents the current state of a capture, delivery and reproduction system that is being developed by a consortium of 11 European partners from the broadcast, film, telecoms and academic sectors within the *'FascinatE'* project, to evaluate the concepts outlined above. The project addresses several different levels of format-agnosticity and interactivity: at simplest, the production tools developed could be used to allow local or specialist broadcasters to customize and tailor coverage of live events for a specific audience, rather than for specific device capabilities. At the other extreme, all captured content could be delivered to the user, regardless of device capabilities. This would allow them to switch between a number of shot sequences selected by the director, optimized locally for their particular screen size. Users could even construct and define their own shot selection and framing, with matching audio that they could further customize, for example by adding various commentary channels.

This paper is organized as follows. In section 2, we shortly discuss the main innovative aspects of the proposed system, as well as the main foreseen usage scenarios. Then, in section 3, we consider the layered acquisition and productions of audiovisual content. In section 4 we treat production automation, including feature tracking and saliency detection. Section 5 considers the role and functionality of the delivery network to cope with the demanding bandwidths of layered content. Then, in sections 6 and 7 we discuss how audiovisual content can be rendered on arbitrary display and loudspeaker setups. Finally, in section 8 we reflect on the whole format-agnostic approach and the potential of the proposed system.

2. SYSTEM ASPECTS AND USAGE

New kinds of ultra-high resolution sensors and ultra large displays are generally considered to be a logical next step in providing a more immersive experience to end users. High resolution video immersive media services have been studied by NHK in their Super Hi-Vision 8k developments [2]. In the international organization CineGrid.org [3], 4k video for display in large theaters plays a central role. At Fraunhofer HHI, a 6k multi-camera system, called the OmniCam, and an associated panoramic projection system was recently developed [1]. On the other hand, personalized media delivery in multi-screen environments is an established consumer market. However, the notion of immersive media with high resolution video, stereoscopic displays and large screen sizes seems contradictory to leveraging the user's ability to select and control content and have it available on personal devices. The proposed system aims to improve upon these current immersive media developments by focusing on four key innovation areas, as shown in Figure 1. Within these areas, we identify the following innovative system aspects:

1. **Layered Scene Production**; where audiovisual scenes are captured in multiple resolutions, frame rates and dynamic ranges with A/V sensor clusters consisting of multiple cameras and microphones;

2. **Production Automation**; providing knowledge to steer further processing and adaptation of the content within the network and on the terminal, based on production knowledge and metadata;

3. **Resolution-independent Delivery;**, enabling the efficient delivery and media-aware network-based processing that is required for the support of low-end terminals and bandwidth limitations in the access networks;

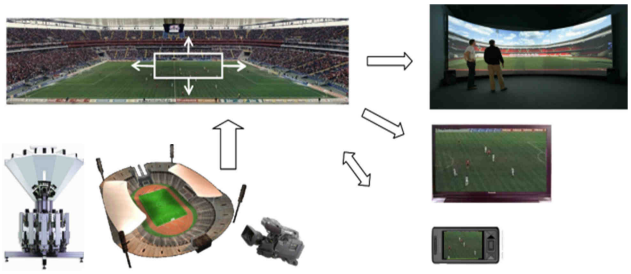


Figure 2 - FascinatE main use cases and terminals.

4. **Interactive Audio/Video Rendering;** adapting the content to the end-user terminals with their associated screen and speaker set-ups;
5. **Gesture Based User Interaction;** enabling natural end-user navigation based on simple and intuitive gestures.

Leveraging the system aspects, we foresee three main usage scenarios, each with associated devices and screens (see Figure 2). First, in a theatre case the captured content is transmitted to and displayed on a large panoramic screen, with the associated 3D audio being presented through a multi-loudspeaker set-up. This enables multiple viewers to simultaneously see the content and interact with it. Second, in the home viewing situation a limited number of viewers consumes content via a primary TV screen and interacts using gestures, e.g. by selecting players to follow when watching a sports game and zooming in on interesting events. Lastly, in a mobile use case, users employ their individual devices, such as smart phones and tablets, to personalize their views at e.g. live concerts. Various other combinations of system aspects lead to additional usage that fit in e.g. automated production scenarios for director support, or second, complementary screen scenarios with navigation on tablets.

3. ACQUISITION AND PRODUCTION

The format agnostic nature on the production side requires a large variety of audio-visual sensors. In order to allow the succeeding modules to access content in the most efficient way, a '*Layered Scene Representation'* (LSR) has been developed. This description contains all the available data, but also the relationship between them in the form of metadata. In terms of visual information, this contains either geometrical information (i.e. which view of the scene), the spatial and temporal resolution or dynamic range. The metadata for audio sensors contains the geometrical information i.e. position and orientation, number of capsules, capsules arrangement and transport format (mono, stereo, surround, etc.).

A set of different audio-visual sensors are currently part of the proposed system. The core of the visual capturing system is a new ultra-high resolution omni-directional camera that is now equipped with 6 high-dynamic range Alexa M[1] cameras. This new camera consists of a separated head that is mounted in a specifically designed mirror rig. It allows acquisition of a 180° panoramic view of the scene at a resolution of ~7K x 2K pixels (see Figure 3, left). The body of the camera is connected via fibre connection in order to minimize the weight at the front end in the camera rig. In addition to that, a number of HD broadcast cameras are available allowing pan/tilt and zoom into special parts of the scene. Cameras with different dynamic range or frame rate are envisaged as well. The audio scene is captured by a number of spot microphones, shot gun microphones, stereo microphones, soundfield microphones and Eigenmike® microphones (see Figure 3, right). The captured raw content needs to be further processed according to the layered scene description. Hence a complete stitched panoramic view is produced out of the individual HD cameras of the omni-directional camera. A registration of the different views (e.g. panoramic and zoom in view) is performed to offer the user additional information of the scene at higher resolution, better dynamic range or even higher frame rate. The set of audio signals is processed to achieve a soundfield description, which can then be used to drive a large variety of sound systems in a format agnostic manner.

The complete acquisition and production pipeline has been investigated and tested in a first test shoot at a UK premier league soccer match in October 2010. The new omni-directional camera rig has been used in a second test shoot that was performed at the concert hall Arena in Berlin in May 2012, during the production of a dance project by the Compagnie Sasha Waltz & Guests and the Education Programme by the Berlin Philharmonic Orchestra conducted by Sir Simon Rattle performing a choreography of the Carmen-Suite by Rodion Schtschedrin. A third test shoot has been performed at Royal Albert Hall in London in August 2012, capturing a classical concert from the BBC Proms series. See Figure 4 for examples of the stitched panoramic image output. The last two test shoots offer the project content at 50 fps and with high dynamic range due to the usage of the new Alexa M. Further experiments on format agnostic usage of different type of video in terms of frame rate, dynamic range and resolution, will now be investigated. Regarding audio capture, in the second and third test shoots classical music was recorded using 20 to 50 spot microphones, up to two Eigenmikes and one SoundField microphone. Depending on the position of the Eigenmikes, the results are well suited to support the stereo microphones at the conductor, as well as for documenting the ambience of the event.

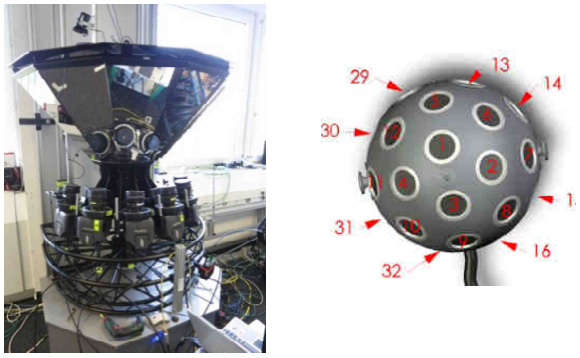


Figure 3 - Omni-directional camera by Fraunhofer HHI equipped with ARRI Alexa M (left), Eigenmike® (right).

4. PRODUCTION AUTOMATION

A major benefit of the proposed system is to reason in parallel for different viewer groups and end user devices. This is achieved by automating the content selection process by means of a *'Production Scripting Engine'* (PSE) which is capable of choosing camera views based on results from automatic video content analysis – possibly supported by live human annotation.

[1] ARRI Alexa M camera, see http://www.arri.com/camera/digital_cameras/cameras.html

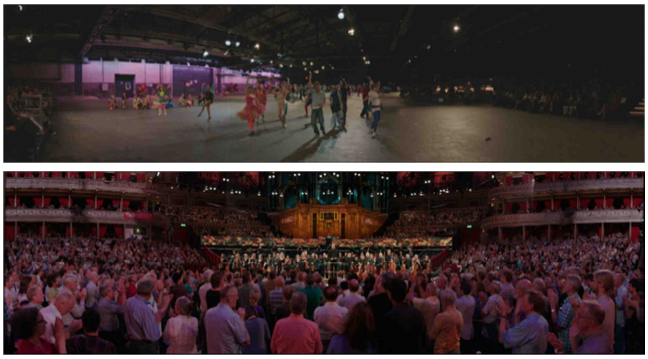


Figure 4 – footage from 2nd and 3rd test shoot. The upper picture shows the dance project by Compagnie Sasha Waltz & Guests / BPO; the lower picture shows the BBC Prom concert.

This section describes the PSE and the content analysis algorithms it is informed by. These are person detection and tracking, and spatio-temporal detection of salient regions. The criteria for selection of these algorithms were the applicability to a wide range of content scenarios and a realistic possibility to achieve real-time performance on the high-resolution panorama images.

4.1 Person detection and tracking

One of the key elements of automated metadata extraction in the proposed system is a real-time person detection and tracking algorithm [4]. Developed to detect and track persons in different types of image sequences from live events captured by the system, the real-time capability is the most important requirement of the proposed algorithm. It is not feasible to perform the A/V content analysis on the full resolution panorama on a single computer due to limitations regarding both the bandwidth of network interfaces and the computational requirements of the A/V content analysis itself. Person detection and tracking is therefore performed independently for each of the HD resolution tiles of the omni-directional panorama, in parallel. Graphics processing units (GPUs) are increasingly used as co-processors allowing superior performance for massively parallelisable tasks. Therefore we utilise GPU implementations in order to meet the real-time requirements. As illustrated in Figure 5 below, the implemented approach consists of two main steps.

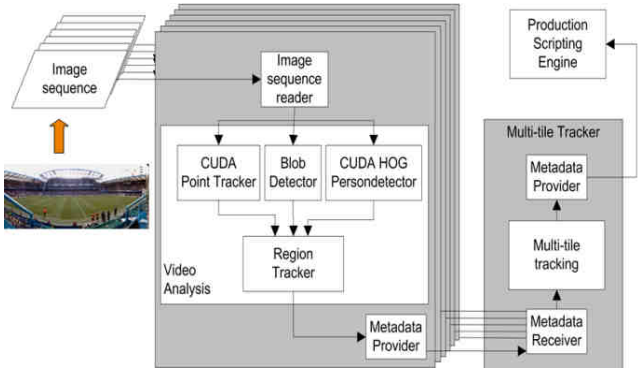


Figure 5 - AV content analysis using region tracker and multi-tile tracker.

The *'region tracker'* detects and tracks persons in the image sequence from one HD resolution tile by using a CUDA[2] based accelerated feature point tracker, a blob detector and a CUDA based HOG person detector. The *'multi-tile tracker'* fuses the resulting person tracks from the different tiles for the full resolution panorama.

To detect the persons in the scene a GPU implementation of the HOG [5] person detector, the fastHOG [6] is used. To achieve real-time performance a coarser scale sampling of the pre-trained SVM classifier with a scale ratio of 1.3 (instead of 1.05 as in the original implementation) is adjusted. Due to the given speed up factor of 3 - 4 a small decrease of the detection rate is acceptable for our application. In some cases we have to deal with motion blur due to fast and sudden movement changes of the persons to track. For these situations we use the OpenCV[3] blob detector, a CPU implementation. The detected regions of both detectors are illustrated in the left hand side of Figure 6 The results of the person detector are indicated by the blue bounding boxes and the results of the blob detector by the red ones. The resulting detected regions for further tracking are indicated by the black bounding boxes on the right hand side. In order to track persons in real-time we use a KLT algorithm based feature point tracker implemented for the recent NVIDIA GPUs based Fermi architecture [4]. The very popular KLT algorithm proposed by Kanade, Lucas and Tomasi [7] is used for the detection and tracking of salient points in image sequences. The GPU accelerated feature point tracker is able to handle 5-10k feature points on two full HD video streams in real- time. The region tracker combines the detected regions with the extracted feature point in order to track all detected persons within the image sequence from one HD resolution tile. The extracted points not located in a detected person region (indicated by the yellow circle in Figure 6) are used to reduce the number of missed person detections by distance clustering. The resulting new detected regions are added to the already detected regions of both detectors as basis for the region tracker. To track all detected regions person IDs are linked to these regions with their corresponding, therein located feature points (see Figure 6).

Furthermore, for extending the duration of continuous tracks, a propagation of the previously detected person regions is performed based on the age of detected region, the number of therein located feature points and their direction of movement. If detection fails in the area of a region already being tracked, the region's position is updated by the new overall location of the corresponding feature points. In the case of an insufficient number of feature points the new position is calculated from the region's previous direction of movement depending on the age of the track. The moving direction, velocity and history of the detected regions and extracted feature points are additionally used for occlusion handling. In the metadata interface all tracked regions, their locations and person IDs of all separate camera sequences are delivered with their specific image-sequence ID to process the results in multi-tile tracking component. To track persons over the stitched full resolution panorama the resulting person tracks are fused by merging the corresponding regions and updating the linked person IDs between adjacent sequences. The centre of the new reported detected region's position is given by the border of the two adjacent images.

[2] Compute Unified Device Architecture, see http://www.nvidia.com/object/cuda_home_new.html

[3] http://opencv.willowgarage.com/wiki/VideoSurveillance

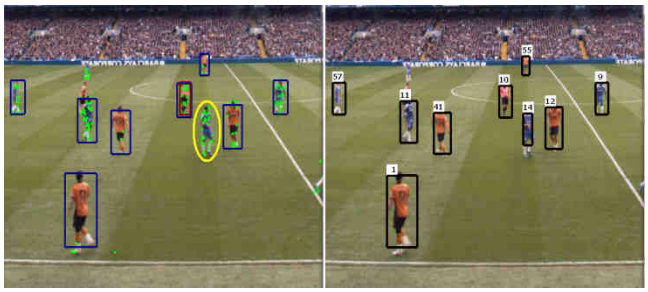


Figure 6 - The left image shows the detected person regions described by bounding boxes and tracked feature points. The resulting tracked persons with their bounding boxes and IDs are shown on the right.

The total runtime for each image tile the region tracking is about 129 ms, consisting of 70 ms for the improved fastHOG, 9 ms the feature point tracker, 35 ms for the blob detector and 15 ms for result fusion. The effort for merging and updating the results of the different workstation is negligible. The result of person detection and tracking is an MPEG-7 AVDP (Audiovisual Description Profile [8]) document describing the segments in which persons are visible and the location of the bounding box in each frame. The description is organized so that bounding boxes per frame can be accessed separately in order to stream it as a sequence of update MPEG-7 documents, using the same format between content analysis and scripting engine.

4.2 Spatiotemporal visual saliency estimation

We consider visual saliency in spatiotemporal space as a generic measure that can be applied to any content domain. It does not provide semantic meaning on its own, but rather indicate where a human observer of the scene might naturally look. We use the feature maps calculated from basic saliency measures in order to integrate intensity and colour histograms calculated in a recent time window. These reference histograms represent the content of the scene recently seen by the viewers. Significant changes in the current frame mean a new visual stimulus and indicate the regions of interest. Thus, the histogram differences between the current frame and the previously stored reference model can be used as saliency indicator. In our implementation, we divide the image area into grids of size 40x40 pixels. Each grid cell provides its own reference histogram with 256 bins. To build the reference model, the recent 20 frames are used. Note that the time range over which the reference model is built/updates does not directly influence the latency. To determine a current saliency estimate, the saliency features of a much shorter time window (i.e., a few or only a single frame) are matched against the reference model.

We provide two types of outputs from the calculated saliency information, both visualised in Figure 7. The saliency map is provided to delivery for calculating of a saliency values for each of the tiles in their segmentation. The bounding boxes are represented in the same MPEG-7 format as the person tracks, and provided to the PSE as candidate framing areas.

4.3 Automatic Camera Selection

The PSE is a distributed component that takes decisions on automatic camera selection. It continuously decides what is visible and audible at each playout device, taking individual preferences of the viewers and capabilities of their devices into account. A metaphor for such systems is *'Virtual Director'*. The PSE's research problem of automatic camera selection and framing is multifaceted. Certain aspects of our approach can be compared to previous work on sports event summarization using multiple camera views [9][10]. Automatic execution of cinematographic principles has also been investigated in the domain of videoconferencing [10]. Our aim is to work towards a generic framework that can be adapted to different production system increments, and also to different production genres. While our first prototype was developed for the domain of soccer, we are now targeting a dance performance to investigate to which degree production rule re-use is feasible.

Figure 7 - Spatiotemporal saliency: brighter blocks indicate higher saliency; red boxes mark candidates of salient regions.

The PSE software component is distributed to form a chain through the production network. The minimal configuration is to have a primary PSE instance at the production end, which has the following specific tasks:

- The primary PSE processes the real-time stream of low-level *'cues'* as extracted by audiovisual content analysis;
- It is integrated with an *'EditorUI'* toolset, i.e. an interface for production professionals that allows manual annotation and decision intervention;
- It uses a knowledgebase for spatiotemporal queries and to retrieve metadata for e.g. replays;
- It informs the delivery network via a delivery scripting engine (DSE) about its shot candidates. This information can be used for content transmission optimization.

An instance at the terminal end is also required, as it is responsible for taking final decisions and for instructing the renderer. Any number of PSE instances might be added between them, with the specific purpose to filter and re-prioritize shot candidates with respect to a certain aspect, e.g. privacy or content rights. Editing decisions will be more and more restricted down the PSE chain and in addition to a list of prioritized shot options, metadata is passed from one instance to another. These messages, as well as final instructions for the renderer, are called *'scripts'*.

While traditional TV broadcast produces a singular video stream where every viewer gets to see the same edit, one of the key advantages of the PSE is the ability to service a large number of individual preferences. A central aspect of the format-agnostic vision is to realize personalized audiovisual streams that respect the viewers' connection and device capabilities, and their domain-dependent preferences. Examples for the latter are selection between several cinematographic styles, focus on certain persons, groups, or types of actions. This automatic process is informed by the content analysis algorithms and a set of dedicated production tools that allow manual support of the decision making sub-processes. The EditorUI toolset is designed to enable basic features such as manual annotation of high-level concepts, including properties such as their location within the video panorama and their temporal extent.

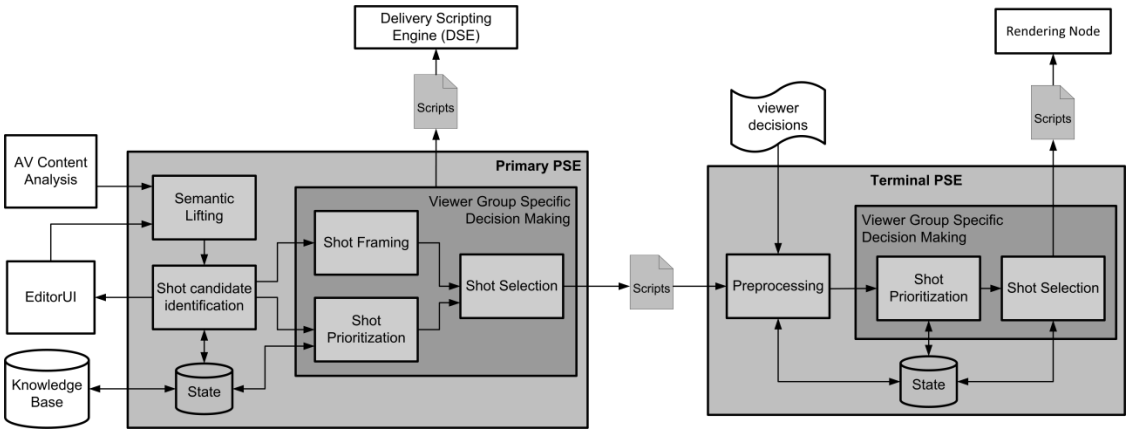


Figure 8 - Internal architecture of the distributed PSE. The left block depicts the primary instance at the production end with interfaces to content analysis, the EditorUI toolset and a knowledgebase. The right area depicts the PSE instance at the terminal end which sends final instructions to a rendering node.

The PSE's logic is executed by a rule engine that is also an event processing engine, mainly for performance reasons, as it is required to react to input and to take decisions in real-time. The PSE's behavior is defined by a set of domain-dependent production principles, implemented in a format specific to the rule engine. These principles define how the PSE is automatically framing virtual cameras within the omni-directional panorama, how camera movements are smoothed, when cuts and transitions to other cameras are issued etc. They define both the pragmatic scope of how to capture domain-dependent actions, and the cinematographic scope how to do that in a visually aesthetic manner. However, general production rules are not independent. The engineering effort necessary to structure their interplay and to balance their effects and side-effects is non-trivial. Competing and contradicting principles need to be resolved. The internal architecture of the PSE in a 2-instance configuration is depicted in Figure 8. Further details can be found in [12]. the following describes the three main sub-processes in more detail.

4.3.1 Semantic Lifting

A key sub-process of the PSE is called *'Semantic Lifting'*. It deals with the problem that the incoming information about the scene is on a different semantic level than the production rules for camera selection decisions. In order to bridge the gap, this component aims to achieve an understanding of what is currently happening. From a technical point of view, it aims to derive domain-specific higher-level concepts. It does so by looking for certain spatiotemporal constellations of low-level cues in the streams as triggers. As an example, in the domain of soccer, the PSE receives a real-time stream with person tracking cues. A certain motion pattern of the players, fused with the location of the ball, the audience volume level and the amount of visual saliency in the audience regions could be used to detect goal situations, or fouls. The component emits a range of high-level events to inform subsequent components. We chose to implement Semantic Lifting with JBoss Drools[4], a hybrid rule engine that is also a Complex Event Processing (CEP) engine.

4.3.2 Shot Candidate Identification

This component creates and maintains a list of usable shot candidates, i.e. a list of real and virtual (cropped) views from the omni-directional panorama and broadcast cameras. Its output are not final decisions, but options that subsequent components use to take personalized camera decisions. Shot Candidate Identification builds on the high-level understanding achieved by Semantic Lifting to decide which views to select as candidates, while also keeping its options balanced. Some candidates can be directly derived from inputs of the EditorUI. It makes use of the definition of annotation concepts and shot properties that are loaded from a domain-dependent model. The component determines at least one candidate at all times; the actual number depends on the scenario. In the previous soccer example, if a foul was detected, this would create new virtual cameras to frame it.

4.3.3 Decision making

The Production Scripting Engine's decision making mechanism is distributed. Each instance creates a prioritized decision list including metadata, which can be updated by subsequent instances. The decision making components work in parallel per viewer group, i.e. per group with the same personalization and device properties. The rules are triggered by the occurrence of higher-level cues and states that are specific to each viewer group. They determine how and when to cut. Since we are dealing with a real-time broadcast system, decisions have to be taken fast and with constant delay.

5. RESOLUTION-INDEPENDENT DELIVERY NETWORKS

In order to deliver the LSR to end-devices, the network needs to ingest the whole set of audiovisual (A/V) data produced to support immersive and personalized applications. This typically translates into very demanding bandwidth requirements. As an example, the live delivery of the immersive A/V material in the LSR that is currently used within the project would require an uncompressed data rate of 16 Gbps. In situations where the full LSR is to be received by an end-user terminal, say in the case of a theatre with large-scale immersive rendering conditions, the delivery requires massive end-to-end bandwidth provisioning.

[4] JBoss Drools, http://www.jboss.org/drools/

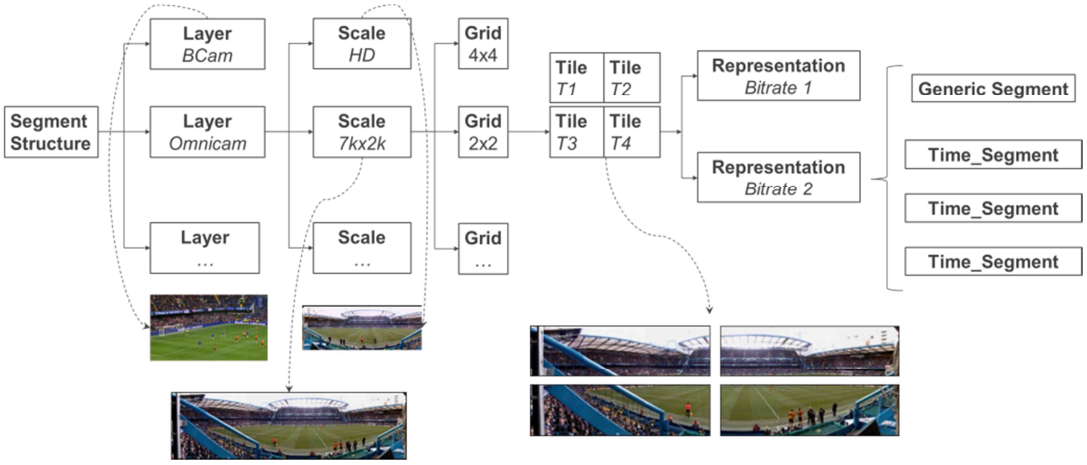


Figure 9 – Hierarchical structure of spatially segmented content. For every LSR layer, multiple resolution scales are created. This scales are tiled according to a grid layout. Each tile is then encoded individually, possibly in multiple quality layers.

But the proposed system also aims at delivering immersive video services to terminal devices with lower bandwidth access or less processing power. In particular, a high-end home set-up capable of processing the full LSR for interactive rendering, but with typical residential network access, may be unable to receive the data rate of the complete LSR. Finally, in case of low-powered devices, such as mobile phones or tablets, the system provides for media proxies, capable of performing some or all rendering functionality on behalf of the end-client.

5.1 Content Segmentation

A key realisation in developing the delivery network architecture is the fact that inside the delivery network, the adaptive delivery of parts of the content based on the viewing behaviour of the client (or the user) can be supported by spatially segmenting the A/V data into tiles that relate to a specific spatial region of a video frame. In most cases, tiles are grouped for a certain time period, in which case they are called segments. The particular grouping can be dependent on the transport protocol used, but globally, the delivery network is aimed at delivery of tiled and segmented content. This content segmentation is required to recast the LSR content into segments that are suitable for network encapsulation and further transport functions. The general concept behind spatial segmentation is to spatially split each video frame into numerous pieces, or tiles. All frames representing a single area of the video are taken together, encoded and stored as a new independent video stream, or spatial segment. Thus, the delivery of video becomes resolution independent, as high-resolution video can be reassembled from lower-resolution tiles. The concept of tiled streaming was previously explored in [13] and [14].

Content segmentation leads to a set of video files, each representing a specific area of the original video file. Encoding each spatial segment as an independent video stream allows an end-device to only request a subset of segments, based on the region-of-interest (ROI) selected by the user for which it performs the spatial recombination. Upon reception of the individual spatial segments, the end-device can then recombine them with a content reassembling operation. In certain cases, a user may also want to see an overview of the entire video. In order to do so, the end-device would need to receive all spatial segments, resulting in enormous bandwidth requirements.

Furthermore, it would need to downscale the resulting video, e.g. in order to be able to present it on a small smartphone display. To solve these inefficiencies we create multiple resolution scales. Each scale is a collection of spatial segments that together encompasses the entire video. However, each scale does so in a different resolution. For example, the top scale might consist of 144 segments, together encompassing the original omni-directional panoramic video resolution of 7680x4320, while the bottom scale might only consist of 4 segments, with a combined resolution of 1280x720. It is even possible to create a scale consisting of only a single segment, with a 640x360 resolution, still showing the entire video but at a much lower quality. The resulting spatial segmentation system provides an efficient method -in terms of required bandwidth- for receiving parts of an ultra-high resolution video. By only having to receive those areas of a video in which a user is interested, combined with support for a wide variety of display sizes and resolutions, it is possible to exploit next-generation ultra-high resolution camera systems with current generation delivery networks. Note that the combined usage of spatial segments and multiple resolution scales may lead to a significant number of video files. Figure 9 shows the hierarchy of segmented content and some example segments.

This shows that, starting from an LSR, video content is segmented at the following levels:

- For every layer in the LSR, one or more resolution scales are created;
- For every scale, one or more MxN tiling grids are created, leading to a set of tiled video streams per scale;
- For every tile, one or more representations at different quality settings are created;
- For every representation, the associated tiled video stream can be temporally segmented.

5.2 Delivery Components and Mechanisms

When taking into account current and near-future home viewing situations based on e.g. IPTV networks over xDSL or cable, we expect bandwidths ranging between 20-100 Mbps, which is not enough to transmit the full LSR in its uncompressed form.

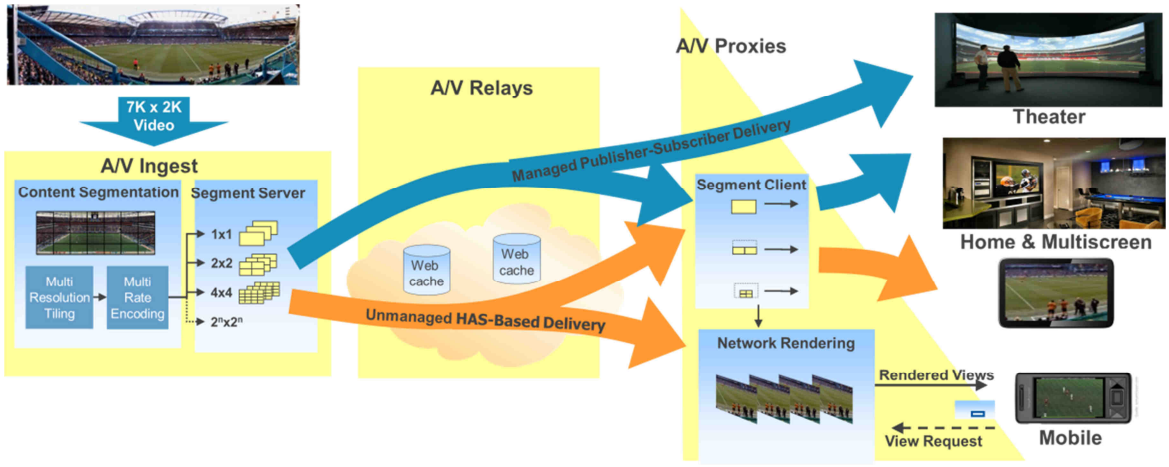


Figure 10 – A resolution independent delivery network, including both managed and unmanaged delivery mechanisms.

Furthermore, in the case of mobile broadband networks, bandwidths of up to only 20 Mbps are foreseen. Hence, in-network rendering, in-network content adaptation and tiled transmission are required within the delivery network, as visualized in Figure 10. Three high-level active delivery components provide this functionality at specific stages of the delivery, namely, ingest, storage and forwarding, and rendering.

- A/V Ingest: receives as input the full LSR and performs A/V processing that is applicable for all or a large fraction of the end-users. Furthermore, content is prepared for the actual delivery by the content segmentation operation, resulting in media delivery units that we refer to as segments.
- A/V Proxy: at the other end of the network, this block is responsible for ensuring that the A/V segments required by one user or a local set of end-users are delivered and reassembled according to their interactivity requests. The proxy can also perform in-network A/V processing using a rendering node to adapt to personalized requests and/or personalized delivery conditions, such as access bandwidth and device capabilities.
- A/V Relay: in between these two network demarcation points, the transport of A/V segments needs to act as an end-to-end filter that accommodates the network capabilities as well as the aggregated requests of the deployed A/V proxies. This can be ensured by intermediate transport nodes, that can aggregate, cache and/or relay segment requests at the transport protocol control level, and also serve as demarcation points between delivery modes for the downstream A/V flows.

The actual transport of A/V segments takes place between Segment Transport Servers and Clients. Two specific delivery mechanisms are developed between the A/V Ingest and the A/V Proxy, catering for different usage scenarios and network deployments. On the one hand, an adapted flavour of HTTP Adaptive Streaming (HAS) mechanism that is suitable for todays' web-based over-the-top delivery, in e.g. CDN or cloud video deployments. On the other hand, a Publisher/Subscriber (PUB/SUB) mechanism fits the requirements of future managed delivery networks such as IPTV over xDSL and cable.

5.2.1 HAS and Tiled Streaming

Current HAS solutions [15] focus on temporal-segmentation. HTTP adaptive streaming can however also be combined with the spatial content segmentation operation. Each video tile is individually encoded and temporally segmented according to common HAS solutions. An advantage of using HAS for the delivery of spatial tiles is that the inherent time-segmentation makes it relatively easy to resynchronize different spatial tiles. That is, all HAS tiles are temporally aligned such that segments from different tiles can be easily recombined to create the reassembled picture. As long as the time segmentation process makes sure that time-segments between different spatial tiles have exactly the same length, the relative position of a frame within a time segment can be used as a measure for the position of that frame within the overall timeline. In HAS solutions such as MPEG-DASH [16], a manifest file is used to describe the structure of the segmented content. This manifest is referred to as a Media Presentation Description (MPD). The MPD includes all information that a HTTP client needs to retrieve the media segments corresponding to a media session, such as the Media Presentation, alternative representations of the media, specific groupings of media and segment and media information, e.g. segment length, resolution, audio and video codecs and the container format. The MPD as defined in MPEG DASH can be readily extended with resolution scale and spatial tiling information.

Figure 11 shows an instance of the delivery network based on tiled HAS delivery [17][18]. It incorporates a set functions for the A/V Ingest and A/V Proxy components based on the tiled HAS mechanism, and a regular Content Delivery Network (CDN) as the A/V Relay. The main functions at the A/V Ingest, Relay and Proxy are the following:

- At the A/V ingest: the main component is the tiled HAS server that hosts the segmented LSR content.
- At the A/V relay: the main component is a live CDN delivery server, for scalable and distributed delivery of segments.
- At the A/V proxy: the main components are the tiled HAS segment client which requests the segments, and the frame combiner which performs the content reassembly function and adapts the reassembled view to the target device.

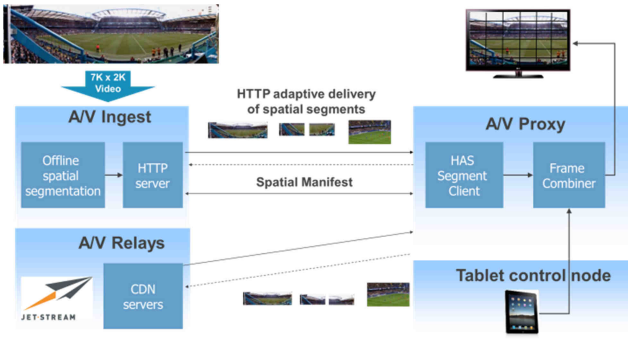


Figure 11 – Architecture for tiled HAS delivery mechanism.

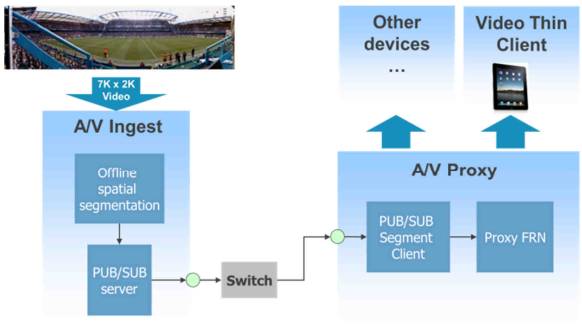


Figure 12 - Architecture for tiled PUB/SUB delivery mechanism.

5.2.2 Publisher/Subscriber mechanism

A main challenge for the delivery network is that the requirements on the type of transport technology seem contradictory depending on which end of the network one looks at:

1. At the network ingress where the whole LSR is ingested, the network elements are responsible for pushing the content through the network, agnostic to the actual user requests.

2. At the terminal side, user-specific portions of the layered scene may be requested. Therefore, if the capabilities of the end-to-end network and of the terminal cannot support a plain transmission of the full LSR, the terminal has to send some requests upstream to pull those parts of the LSR which are required for rendering.

To cover these two requirements, we propose to use a message-queue mechanism, which specifies '*publisher*' and '*subscriber*' functions that can work asynchronously at each end of the network. This approach fits well for a deployment in a managed network, such as next generation IPTV systems, that would be required to support a large number of end-devices with various bandwidth and processing capabilities. In this PUB/SUB mechanism, the published data is transported over a combination of unicast and multicast channels, organized according to the previously described multi-resolution hierarchy of spatial segments.

In addition to a better control of the transport channels, a managed network context also opens the possibility to put more processing functions into the network. Figure 12 shows a network instance where the rendering stage is performed in the network, so as to support thin clients. In this case, the thin client does not need to directly subscribe to the segmented data, but directly receives a pre-rendered video stream. This requires the network to include in the A/V Proxy rendering functions are responsible for making the received segments ready for delivery to the end-device. Such a video proxy has been developed so as to allow any thin client device to freely navigate into the LSR content. The end-device only has to send its pan-tilt-zoom navigation commands (e.g. from a touch-based user interface) to the proxy and receives back the requested sequence of views, fully pre-rendered by the network and delivered at a resolution and bandwidth that match the device capabilities. With this approach, high-resolution video content can be watched interactively in a natural manner, even on low-power and small-display devices, such as smartphones and tablets.

6. FORMAT AGNOSTIC AUDIO

A clear trend of future audio reproduction formats is an increasing number of loudspeakers to create a more immersive and engaging experience. Typically each audio channel carries a signal dedicated to a specific loudspeaker position (e.g. stereo: first channel = left loudspeaker, second channel = right loudspeaker). This approach requires a specific audio production for each targeted reproduction system which generates additional cost per production. Even where upmixing is used, for example from stereo to 5.1, results are often poor and periphonic ('with height') audio systems are not catered for. We are considering a new, format agnostic, approach: Instead of using the loudspeaker signals to define content representation, a sound scene description is utilised incorporating a combination of audio objects with location, and higher order Ambisonic audio capture. The sound scene can be transmitted using several audio channels with associated metadata descriptions of location and content of audio objects and sound field capture. The sound scene can then be reproduced on any reproduction system without the need for separate production. Two major sound field description technologies are implemented, the '*Wave Field Synthesis*' (WFS) and the '*Higher Order Ambisonics*' (HOA) techniques, but the format agnostic approach also allows reproduction on other systems. For example, using a sound scene description of audio objects and sound fields this is possible using various down-mixing algorithms pertaining to the techniques in question. The processing, is done at the user end where the audio objects and sound fields are mixed in real-time to match the dynamically updating visual content.

6.1 Sound Scene Descriptions And Rendering

The spatial audio reproduction system for FascinatE aims to replay the sound scene accompanying a visual scene where the user can determine the current visual viewpoint. In order to enable the sound scene to adapt to the user's viewpoint, and to ensure optimisation for any reproduction system, a flexible sound scene description is utilised. The sound scene is synthesised from 3D audio capture and from multiple audio objects that are positioned in a sound scene. Audio objects may be either explicit (directly captured) or implicit (derived from a microphone array(s)). The acoustical environment is auralised by additional objects that create reverberation and by superimposing a sound field captured by a multi-capsule microphone such as the Eigenmike®. For the sound scene creation, basic parameters like source positions and acoustical parameters need to be transmitted using metadata generated in (close to) real-time.

6.2 Wave Field Synthesis

Wave field synthesis (WFS) [19][20] uses a sound field description based on the Kirchhoff-Helmholtz integral. This means that the sound field within a volume can be accurately reconstructed using arrays of loudspeakers surrounding the listening area. Each source in WFS is rendered with the correct wave front shape, whether a point source, plane wave or sources with a more complicated directivity patterns. The sources are positioned in space such that a sound field can be accurately spatially reconstructed with the correct localization cues present.

Decoding the recorded sound scene to be reproduced over wave field synthesis requires some consideration of the different source types in the scene. As mentioned previously, the sound scene will consist of a collection of audio objects and one or more sound field components. As the audio objects have a defined source content and position in 2D or 3D space they can be rendered over a WFS system as (e.g.) simple point sources from the respective locations. However for the sound field components, it is not as obvious how best to perform the rendering and consequently there may be more than one flavor of the WFS rendering available within FascinatE. One possible solution is to use of virtual loudspeakers [21]. In this scenario, the sound field components would be decoded to an idealized Ambisonics loudspeaker system, and these signals would then be rendered as point sources (virtual loudspeakers) in the WFS rendering. Another way of accurately reproducing the sound field spatially would be to perform a plane wave decomposition and render the corresponding coefficients as plane waves from the correct locations.

One of the major problems with WFS reproduction is the computational overhead, which increases with each additional source that it added, thus there is a limitation on number of sources that can be reproduced for a given system, whilst there are techniques available that aim to reduce the computation overhead using fractional order delays and IIR filters [22] this is still an important issue and for FascinatE. The production however will not change; the same number of audio objects and sound fields will be extracted from the scene but at the user-end the WFS system will group together any close sources or reduce the resolution of the sound field to allow a full rendering of the sound scene with all sound sources present.

6.3 Higher-Order Ambisonics

Ambisonics is a mathematical soundfield description method [23], characterizing a soundfield in one reference point, often referred to as '*sweet spot*'. The basic idea of Ambisonics is to describe a soundfield, consisting out of sound pressure, particle velocities, etc., using the coefficients of a spatial Fourier transform [24]. FascinatE uses this representation together with an audio object presentation, as an intermediate format for transmission and processing. The advantage of this representation is a scalable spatial resolution of a soundfield. In other words, if a high spatial resolution is not required or cannot be retrieved at the receiving side, the coefficients of the highest order can be simply omitted. Moreover, the HOA representation is agnostic to the loudspeaker setup since the transmitted signals do not refer to specific loudspeaker positions, but describe the properties of a soundfield. On the recording side, spherical microphone arrays can be used to obtain the coefficients of an Ambisonics representation. Audio objects (sound sources) as used for WFS can be encoded in Ambisonics and mixed with other HOA representations.

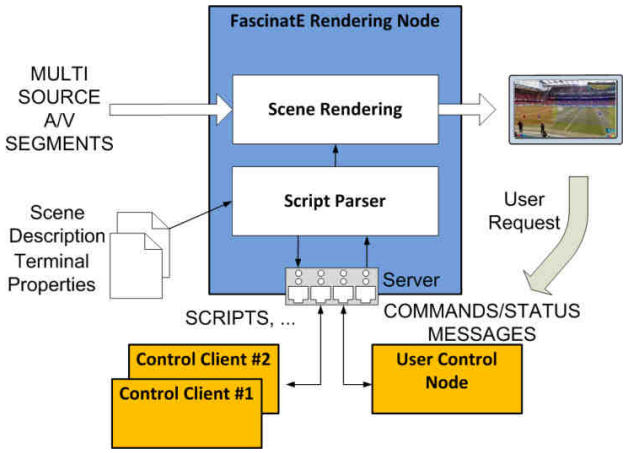


Figure 13 – A system Rendering Node (FRN).

7. RENDERING AND INTERACTIVITY

7.1 Format agnostic video rendering

Reproducing high quality pictures and sound for the large variety of available end terminals such as tablets, set-top boxes and cinema servers requires a flexible format agnostic production and scalable infrastructure. The LSR allows an individual to make a selection of scene elements matching the terminal in use and its connected displays and loudspeaker arrangements. The terminal prototype developed within the proposed system contains modules to render the individual perspective called a Rendering Node (FRN, see Figure 13). Other terminal functionalities are interfaces for real-time transmission of audio-visual content, interfaces for control information and configuration settings and elements to manage these control information. The rendering process includes the manipulation of video content to match the connected display properties. With the layered scene containing a cylindrical panorama this means geometrical corrections and camera data blending depending on the perspective of the rendered viewport have to be made. Ideally, a HD shot camera picture complementing the panorama capture can replace the omni-directional video. Typically, this requires combined rendering of different resolutions, zoom factors and blur levels.

Additionally, the area of available data for the video scene will not be exceedingly larger than the rendering target to limit system requirements such as bandwidth and response timing. The interaction with the delivery network and content production is implemented by adding control infrastructures to pass viewport identifications. This can be a region request for a video update sent to another FRN or to a network proxy element ensuring the in time delivery of required video content. Controlling this rendering process can be performed in different ways. Either by having a local or remote control entity passing commands or command lists in the form of scripts, or be having a connection to a User Control Node translating user requests to such commands. The developed terminal prototype implements a graphical user interface to indicate individual commands (see Figure 14).

7.2 Gesture interaction

The use of format-agnostic rendering allows new interaction possibilities to the end user. The control of the rendering is more complex and it is not limited to a few interactions, such as changing video channels, but richer ones like navigating (pan, tilt

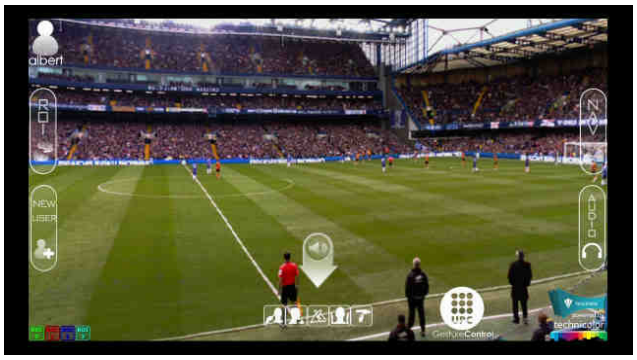


Figure 14 – Rendered image with graphical user interface overlay.

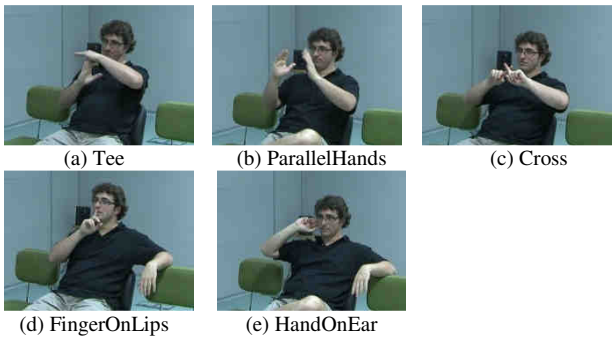
or zoom) in the panorama image or selecting automatic scripts generated by the PSE. Performing these new interactions naturally might not be very easy and intuitive using general devices like remote controls. Therefore, new devices or mechanisms might be studied in order to facilitate the interaction with the format-agnostic representation. At the same time, during the last years, device-less interaction has experienced an exponential growth mainly due to two factors: The appearance of new sensors that facilitate the recognition of human, hand and finger gestures, such as the Kinect by Microsoft, and the latest advances in image processing algorithms that have open the possibility of implementing real-time systems that recognize human gestures with high fidelity. In this work we propose a device-less and marker-less gesture recognition mechanism to allow users to interact as natural as possible with the format-agnostic representation thus providing a truly immersive experience. The gestures allow the user to perform interactions such as selecting menus presented on the screen (Figure 14), navigating through high resolution panoramic views of the scene, control the audio by changing the volume, muting or selecting the speaker, etc.

In order to interpret user gestures, the system employs a Kinect sensor that records the colour and depth video with VGA resolution. Colour and depth images are analysed to detect heads (oval areas of the same depth) [25] and faces by using a modified Viola-Jones detector [26]. Active users are automatically detected and segmented from the background (walls, sofas or chairs) [27]. Once heads are detected, hands are tracked using a 3D virtual box in front of the head of the user with control of the system. 3D blobs in the virtual box are segmented and treated as hands [25]. Other users in the field of view of the Kinect sensor are not tracked and their gestures do not interfere with the system. Detected gestures can be classified in two groups: dynamic and static. Dynamic gestures are performed moving your hands, such as waving or circling and the important information resides in the trajectory of the hand (not the position of it). Static gestures are not performed moving your hands and, in this case, the shape and position of the hands is crucial for the classification of the gesture. The classification of gestures is based on random forests, aiming at accurately localizing gesture and object classes in highly unbalanced problems, where positive classes barely appear.

Current implemented gestures include:

- *Swipe* (moving hand right to left) used to select channels;
- *Pointing* allowing user to select the menu on the screen;
- *Tee* for taking and releasing control of the system (Figure 15a);
- *ParallelHands* to pause or resume the reproduction or streaming of video content (Figure 15b);
- *Cross* to mute/activate the audio (Figure 15c);
- *FingerOnLips* to lower the volume (Figure 15d);
- *HandOnEar* to raise the volume (Figure 15e).

A communication channel using TCP/IP has been created between the FascinatE Rendering Node (FRN) and the gesture recognition system. Each time a gesture is recognized it is translated into an XML message that notifies the FRN to perform the associated interaction.



(a) Tee (b) ParallelHands (c) Cross

(d) FingerOnLips (e) HandOnEar

Figure 15 - Static gestures recognized by the system.

8. CONCLUDING REMARKS

We presented a format-agnostic approach for future immersive multimedia production, delivery and consumption based on a Layered Scene Representation. A complete end-to-end capture, analysis, delivery and rendering system has been proposed and several parts of the complete processing chain have been discussed in order to identify the necessary requirements or even to show first solutions towards a future multimedia framework. The format agnostic approach can be used in a variety of setups. Even if only a single view were to be produced for different playout devices in a single production chain, reduction of production cost can be achieved. Further added value is unlocked when using it for personalization features, where the viewers may not only decide between a fixed number of streams, but get an individual view that is tailored to their preferences. Key elements of the proposed system are the layered audiovisual capture, equipped with microphone arrays and an omni-directional panoramic camera, allow capturing the whole scene at all times; a Production Scripting Engine, making use of automatic content analysis and serving as the system's Virtual Director for automated production for a large number of viewers in parallel; a resolution-independent delivery network, based on tiled streaming over both managed and unmanaged networks; format-agnostic audio and video rendering, using both Wavefield Synthesis and Higher Order Ambisonics; and gesture-based interaction for intuitive system control.

Our future work will focus on the integration and further automatization of the system components. Interaction handling, delay management and syncing are complex challenges to be resolved when further developing the system as a whole. Evaluations will be conducted to answer scientific questions, infer further requirements, and to measure user behaviour in different scenarios, ranging from lean-back consumption of PSE-produced views to lean-forward navigation using gestures or tablets.

9. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 248138. Thanks to SIS Live, the Premier League, Compagnie Sasha Waltz & Guests, the Berlin Philharmonic Orchestra, KUK Filmproduktion and the BBC for their assistance during the test shoots and permission to use images. The authors would like their colleagues for co-editing of this paper: Martin Prins and Ray van Brandenburg from TNO, Werner Bailer and Marcus Thaler from JOANNEUM DIGITAL, Jan-Mark Batke from Technicolor and Rob Oldfield from Salford University. The authors would like to thank the FascinatE project partners for the insightful discussions and collaboration.

10. REFERENCES

[1] R. Schäfer, P. Kauff, C. Weissig, "Ultra high resolution video production and display as basis of a format agnostic production system", Proceedings of IBC 2010.

[2] M. Maeda, Y. Shishikui, F. Suginoshita, Y. Takiguchi, T. Nakatogawa, M. Kanazawa, K. Mitani, K. Hamasaki, M. Iwaki and Y. Nojiri. "Steps Toward the Practical Use of Super Hi-Vision". NAB2006 Proceedings, Las Vegas, USA, April 2006.

[3] P. Grosso, L. Herr, N. Ohta, P. Hearty and C. de Laat. "Super high definition media over optical networks", Future Generation Computer Systems, Volume 27, Issue 7, Pages 881-990, July 2011.

[4] R. Kaiser, M. Thaler, A. Kriechbaum, H. Fassold, W. Bailer and J. Rosner, "Real time person tracking in high-resolution panoramic video for Automated broadcast Production", Proceedings of the 8th European Conference on Visual Media Production (CVMP 2011), 2011.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", in Proc. IEEE Computer Vision and Pattern Recognition (CVPR), vol. 1, 2005.

[6] V. Prisacariu and I. Reid, " FastHOG - a realtime GPU implementation of HOG",Technical report, Department of Engineering Science, Oxford University, 2009.

[7] J. Shi and C. Tomasi, "Good features to track", In Computer Vision and Pattern Recognition, 1994. Proc. CVPR '94., 1994 IEEE Computer Society Conf., p. 593–600, 1994

[8] Information technology — Multimedia content description interface — Part 9: Profiles and levels, AM1: Extensions to profiles and levels. ISO/IEC 15938-9:2005/PDAM 1:2012.

[9] APIDIS project - Autonomous Production of Images based on Distributed and Intelligent Sensing, http://www.apidis.org

[10] F. Chen, C. De Vleeschouwer, "Automatic summarization of broadcasted soccer videos with adaptive fast-forwarding", IEEE International Conference on Multimedia and Expo (ICME), 2011.

[11] R. Kaiser, W. Weiss, M. Falelakis et al. (2012), "A Rule-Based Virtual Director Enhancing Group Communication", In 2012 IEEE International Conference on Multimedia and Expo Workshops, 187-192.

[12] R. Kaiser, W. Weiss, G. Kienast, "The FascinatE Production Scripting Engine", Lecture Notes in Computer Science, 2012, Volume 7131, Advances in Multimedia Modeling Advances in Multimedia Modeling - 18th International Conference, MMM 2012, Pages 682-692, 2012

[13] Mavlankar, A., "Peer-to-Peer Video Streaming with Interactive Regionof- Interest", Ph.D. Dissertation, Stanford University, April 2010

[14] Khiem, N., Ravindra, G., Carlier, A., and Ooi., W. 2010. Supporting zoomable video streams with dynamic region-of-interest cropping. In Proceedings of the first annual ACM SIGMM conference on Multimedia systems (MMSys '10). ACM, New York, NY, USA, 259-270.

[15] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP - Standards and Design Principles", MMSys'11, February 23–25, 2011, San Jose, California, USA.

[16] I. Sodagar, "The MPEG-DASH Standard for Multimedia Streaming Over the Internet", IEEE Transactions on Multimedia, Vol. 18, No.4, p.62-67, April 2011.

[17] O.A. Niamut, M.J. Prins, R. van Brandenburg, A. Havekes "Spatial Tiling And Streaming In An Immersive Media Delivery Network", in Adjunct Proceedings of EuroITV 2011, Lisbon, Portugal, June 2011.

[18] R. van Brandenburg, O.A. Niamut, M. Prins, H. Stokking, "Spatial segmentation for immersive media delivery," in Proc. of 15th Int. Conf. on Intelligence in Next Generation Networks (ICIN), Berlin, Germany, 4-7 October, 2011.

[19] A. Berkhout, "A holographic approach to acoustic control,", J. Audio Eng. Soc., 36(12), pp. 977-995, December 1988.

[20] A. Berkhout, D. de Vries and P. Vogel, "Acoustic control by wave field synthesis," , J. Audio Eng. Soc., 93(5), pp. 2664-2778, May 1993.

[21] G. Theile, H. Wittek and M. Reisinger, "Potential Wavefield Synthesis Applications in the Multichannel Stereophonic World", 24th Conf. Audio Eng. Soc., June 2003.

[22] C. D. Salvador, "Discrete Wave Field Synthesis Using Fractional Order Filters and Fractional Delays",128th Conv. Audio Eng. Soc., May 2010.

[23] J. Daniel, R. Nicol, and S. Moreau, "Further Investigations of High Order Ambisonics and Wavefield Synthesis for Holophonic Sound Imaging", 114th Conv. Audio Eng. Soc., March 2003.

[24] S. Spors and J. Ahrens. A comparison of wave field synthesis and higher-order Ambisonics with respect to physical properties and spatial sampling. In 125th AES Convention, San Fransisco, USA, 2008.

[25] X. Suau, J.R. Casas and J. Ruiz-Hidalgo, "Real-Time Head and Hand Tracking based on 2.5D data", IEEE Transactions on Multimedia , vol. 14, no. 3, p. 575-585 , 2012.

[26] P. Viola, M.J. Jones: "Rapid object detection using a boosted cascade of simple features", IEEE CVPR, 2001.

[27] J. Gallego, M. Pardàs, J.L. Landabaso: "Segmentation and tracking of static and moving objects in video surveillance scenarios", IEEE International Conference on Image Processing, 2008.