NEW RESULTS IN FREE-VIEWPOINT TELEVISION SYSTEMS FOR HORIZONTAL VIRTUAL NAVIGATION

Marek Domański, Maciej Bartkowiak, Adrian Dziembowski, Tomasz Grajek, Adam Grzelka, Adam Łuczak, Dawid Mieloch, Jarosław Samelak, Olgierd Stankiewicz, Jakub Stankowski, Krzysztof Wegner

Poznań University of Technology, Chair of Multimedia Telecommunications and Microelectronics, Poznań, Poland
domanski@et.put.poznan.pl, {adziembowski, agrzelka, dmieloch, ostank} @multimedia.edu.pl

ABSTRACT

The paper presents the concept of a practical free-viewpoint television system with purely optical depth estimation. The system consists of camera modules that contain pairs or triples of cameras together with the respective microphones. The camera modules can be sparsely located in arbitrary positions around a scene. Each camera module is equivalent to a video camera with a depth sensor and microphones. The hardware requirements, the video and audio processing algorithms and the preliminary experimental results are reported. In particular, for such systems, a compression technique is discussed that is more efficient than the new 3D-HEVC technology. A set of new test sequences obtained with the use of camera pairs are presented.

Index Terms— virtual navigation, free-viewpoint television, multiview video.

1. INTRODUCTION

The virtual navigation is a functionality of future interactive video services where a user is able to navigate freely around a scene. The systems that provide such a functionality are often called free-viewpoint television (FTV) [1,22,24]. The prospective FTV will be an interactive internet-based system.

The goal of this paper is to present new results in the design and practical implementations of the FTV systems that overcome some problems related to earlier approaches as described e.g. in [1-4,22]. This paper is focused on the approaches that should lead to practical applications in the next very few years. In particular, we study efficient sparse camera setups, audio processing for FTV and appropriate extensions of 3D-HEVC video compression technology [9].

2. FTV SYSTEM STRUCTURE

Recently, the generic structure of FTV systems has been proposed [5] as shown in Fig.1. Throughout this paper we are going to use this structure that consists of the following functional blocks:

- The video and audio acquisition system,
- The representation server that produces a visual representation of the spatial dynamic scene,
- The rendering servers that serve the requests for synthesis of video and audio at particular virtual locations around a scene,
- The user terminal.

The video and audio acquisition system has to provide data necessary to compute the spatial representation of a scene. Except of video and audio, the data include also some depth information obtained either from pure multiview video analysis or also from depth sensors. The depth acquisition using the depth sensors is conceptually very attractive [e.g. 6,7], but its practical application still faces severe problems related to limited resolutions of the acquired depth maps, limited distance ranges, additional infrared illumination of the scene, synchronization of the video and depth cameras, and sensitivity to the environmental factors including solar illumination. In particular, in this paper we focus on the multiview recording of real events where additional infrared illumination might be unacceptable. Therefore, the considerations in this paper base on the assumption that the depth information is obtained by the video analysis only, and the special depth sensors are not used.

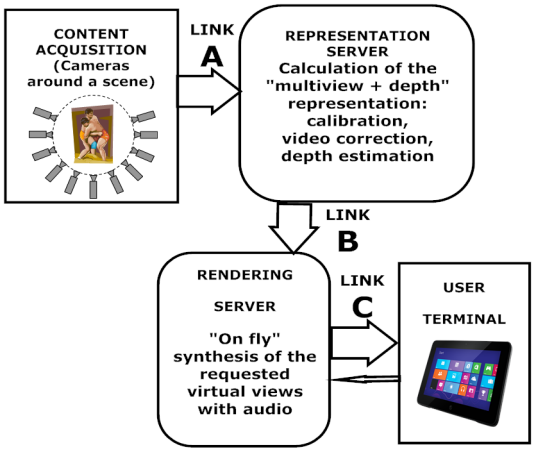

Fig. 1. The general structure of an FTV system (modified from[5]).

The video and audio data together with the system calibration data are transmitted via Link A that belongs to the contribution environment, thus needs the high–fidelity compression. As the video data in Link A are yet neither calibrated nor corrected, for video, a standard single-view compression techniques may be used, including both intraframe techniques like M-JPEG 2000 [8] or HEVC All Intra [9], or interframe studio profiles of AVC [13]

Reseach project was supported by The National Centre for Research and Development, Poland. Project no. TANGO1/266710/NCBR/2015.

or HEVC [9]. Note that simple FTV systems will probably rarely use nonlinear edition as the FTV material does not need any choice of the camera during the production process. The FTV video material does not need camera changes and zooming, as that is done individually by a viewer. If the nonlinear edition is not needed, there is also no need for the random frame access and no need for small error accumulation in the multiple encoding–decoding cycles. Therefore, the requirement to use the intraframe coding may be released, and the standard interframe compression techniques may be used for video. In that way the requested bitrate may be significantly reduced but still the total bitrate will be determined by simulcasting the video streams from multiple cameras plus the audio streams from many microphones.

In particular, especially for the initial phase of the FTV development, the audio and video hard-disk delivery to the representation server may be acceptable for the off-line viewing[4].

The representation server is responsible for calibration, correction of the video (lens aberration correction, illumination compensation, equalization of the color characteristics of sensors etc.) and depth estimation (e.g. [3,4,37-39]). The output is the model of the visual scene. The following scene representation types are mostly considered in the references: object-based [10, 11], ray space [1, 22], point-based [18], and multiview plus depth (MVD) [12]. As the first types of models are related to quite complex calculations, the MVD representation is used most often and its compression has been already standardized both for AVC [13] and for HEVC [9]. Currently, further standardization of MVD compression is also considered [5,14]. Therefore, the MVD representation is also considered in this paper.

In principle, in the representation server, the audio processing is limited to preprocessing of individual signals from microphones. This includes the conditioning, consisting of dynamic processing and equalization.

The compressed MVD representation with the camera parameters and the audio data is transmitted via Link B (Fig. 1). If the representation server and the rendering server are in distant locations, video compression is needed. For the MVD representation, the technology is available and standardized as the 3D extensions of the AVC [13] and the HEVC [9,15] standards. Unfortunately, these 3D extensions have been designed and tested for cameras located on a line. For cameras located around a scene, they exhibit compression performance only slightly higher than individual coding (the simulcast coding) of the views and the depth maps [3,16]. A more efficient MVD compression method is considered in Section 7. In Link B, each audio stream from each microphone may be compressed independently.

The sink of Link B is in the rendering server as we opt for the centralized model [3,17] of view synthesis. In this model, the views requested by the viewers are synthesized in the servers of the service provider, i.e. in the rendering servers. The number of the rendering servers depends on the number of the user terminals, as each such server may serve a limited number of the user terminals.

Another option would be a distributed model [2,19,20] where virtual views are synthesized in each user terminal. Such a model requires high transmission bandwidth in order to transmit the MVD representation directly to the user terminals. This model also requires significant processing power in the user terminals. Furthermore, we are going to avoid the problems related to sophisticated video streaming (see e.g. [19,20]), and we opt for the centralized model, following also the conclusions from paper [3]. For more details please refer to Section 9.

In the centralized model, the user terminal sends the requests for the current virtual positions, and the rendering server responds with the video and audio streams synthesized for the requested position. The free navigation service is available as a website where a user logs in. The proxy rendering server streams video and audio to the user terminals (Link C in Fig. 1). A user terminal may be as simple as a smartphone or a tablet equipped with any standard video decoder (AVC or HEVC) and any audio decoder. The virtual view position requests are defined by sliding the touchscreen horizontally for going around the scene or vertically for going inside or outside the scene (or zooming in and out).

This paper describes a practical and simple FTV system. Other descriptions from the references are either less complete [3,4] or aim at much more sophisticated systems [1,22]. Further in this paper, we are going to describe new and original results concerning selected parts of the system.

3. MULTIVIEW VIDEO ACQUISITION USING MULTIPLE-CAMERA MODULES

Let us consider the depth estimation (e.g. [1,3,38]) using a camera pair with the focal length f and the base distance b. A depth of a point object is z and the disparity of the object images is d measured on the camera sensors. Assuming $f \ll z$ and using the rules of the multiple view geometry [21] we get

$$z = \frac{fb}{d} \ . \tag{1}$$

Assume two objects with the depths z_1 and z_2, respectively. Their positions may be distinguished if the respective disparity difference $(d_1 - d_2)$ exceeds a minimum value Δ, i.e. 2-3 distances between the centers on the pixels in the sensors. From (1) we get

$$z_1 - z_2 = \frac{fb}{d_1} - \frac{fb}{d_2} = \frac{fb(d_2 - d_1)}{d_1 d_2} \ , \tag{2}$$

but $d_1 = \frac{fb}{z_1}$, $d_2 = \frac{fb}{z_2}$, and we denote $z = \sqrt{z_1 z_2}$.

Therefore, z_1 and z_2 may be distinguished by video analysis if

$$|z_1 - z_2| \geq \frac{z^2}{fb}\Delta \ . \tag{3}$$

For example, let consider a 1/1.2" HD sensor with 1920 pixels per line (like the Sony IMX 174 CMOS sensor from the Basler acA 1920-155uc camera) and $\Delta \approx 15$ µm, $f = 16$ mm.

For an average depth $z = 25$ m, for $b = 40$ cm we have the resolution $|z_1 - z_2| \geq 1.5$ m while for $b = 4$ m we have $|z_1 - z_2| \geq 0.15$ m. For an average depth $z = 10$ m, these numbers are 0.23m and 0.023 m, respectively.

The abovementioned example illustrates the fact that a large camera base b provides high resolution of the depth values. Unfortunately, the large camera base implies severe occlusions and many points are not visible by both cameras. On the other hand, a small camera base b yields much less occlusions but also much lower depth resolution. Therefore, we propose that the depth is estimated always from at least two camera pairs: a pair with a small base and with a large camera base. For the sake of reduction of the total number of video cameras, we propose to group the cameras into pairs in such a way that each scene point is visible by cameras from at least two pairs. Thus, the proposed system consists of camera modules sparsely distributed around a scene. For the practical reasons, the locations usually exhibit some irregularities due to the limitations of a real scene, e.g. the requirements to leave free communication routes, windows, commercial banners and displays etc. Each camera module includes the video cameras, the microphones, the control and synchronization hardware and the video storage.

The camera pairs may be replaced by the camera triples (Fig. 2) that usually yield even better depth estimation as they

additionally provide some vertical parallax that is very helpful when all camera modules are not ideally located on the same plane.

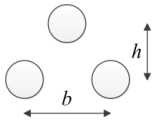

Fig. 2. Triple of cameras from a single camera module.

For the experimental system, we use HD cameras and the HD 1920×1080 frames are used for the depth estimation. In order to provide the ability of the virtual walking into the scene (similar to zooming in), we propose to use one UHD ("4K") camera per module. In such a case, at the output of the representation server the MVD representation consists of "4K" 3840×2160 views (one per each camera module) and 1920×1080 depth maps (one per each camera module). Nevertheless, only HD or even SD views are synthesized and delivered to the user terminals.

4. NEW MULTIVIEW-VIDEO TEST SEQUENCES FROM CAMERA PAIRS AND TRIPLES

Hitherto, the multiview video test material is available for uniformly spaced cameras, and only few sequences are available for cameras located on an arc [23]. Therefore, we produced new multiview test sequences (Figs. 3 and 4, Table 1) acquired using camera pairs or triples located on an arc. To our best knowledge these are the first such sequences granted to the research community (for access please contact the authors).

Fig. 3. Two views from the sequence *Poznan Fencing*.

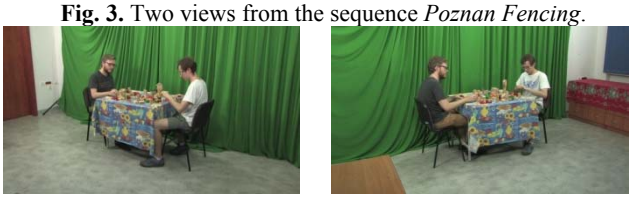

Fig. 4. Two views from the sequence *Poznan Blocks 3*.

Table 1. The parameters of the test sequences.

Parameter	*Poznan Fencing*	*Poznan Blocks 3*
Frame resolution and rate	1920 × 1020, 25 fps	
Sequence length	500 frames	
Contents (2 persons)	fencing	table game
Number of views	10	9
View organization	5 pairs	3 triples
Angle between the axes of the camera modules	15°	30°
Radius of the arrangement	3.5 m	3.0 m
The camera module parameters (cf. Fig. 2)	b = 22 cm	h = 29 cm b = 25 cm

5. COMPARISONS OF THE SYSTEMS WITH DIFFERENT NUMBERS OF CAMERAS PER MODULE

In Section 3 we have discussed the advantages of video acquisition using pairs or triples of cameras sparsely distributed around a scene. Here, we report the experiments where the depth is estimated from uncompressed video acquired from the modules with 1, 2 or 3 cameras, respectively. For such systems, we compare the fidelity of the synthesized views. This fidelity is estimated by calculation of the luma PSNR with respect to the real view captured in the same location (Fig. 5 and Tables 2 and 3). For the virtual views, a rough subjective quality assessment was done by 6 viewers who watched the video clips on an LCD HD display and were using the single stimulus method.

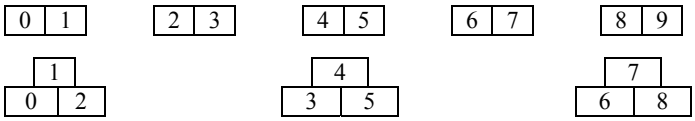

Fig. 5. The cameras in *Poznan Fencing* (top) and *Poznan Blocks 3*.

Table 2. The objective and the rough subjective quality measurements for synthesized views from *Poznan Fencing*.

Number of cameras per module	View 1 synthesized from views 0 and 3	View 4 synthesized from views 0 and 7
	PSNR [dB] for a synthesized view	
1	28.9	24.3
2	30.7	25.6
	MOS for a synthesized view	
1	6.7	3.3
2	8.0	5.3

Table 3. The objective and the rough subjective quality measurements for synthesized views from *Poznan Blocks 3*.

Number of cameras per module	View 2 synthesized from views 0 and 3	View 5 synthesized from views 0 and 8
	PSNR [dB] for a synthesized view	
1	21.7	19.8
2	30.3	25.2
3	30.9	24.8
	MOS for a synthesized view	
1	3.2	1.7
2	6.3	4.3
3	7.5	4.5

The results from Tables 2 and 3 demonstrate superior performance of the system with camera pairs as compared with mono-camera modules. Replacing the camera pairs by the camera triples provides a slight quality improvement for the synthesized views.

6. DESIGN OF AN FTV ACQUISITION SYSTEM

The FTV system is proposed for the university sports hall and the basketball games. The hall dimensions are 34 × 52 meters, and the basketball field is 15 × 28 meters. The horizontal field of view of a camera was chosen as 44 degrees (16 mm focal length, Basler acA 1920-155uc cameras with common trigger). The locations of the camera modules are chosen in such a way that all points in the field are visible from at least two modules. This assumption yielded 28 camera modules including 4 corner modules added for smooth virtual navigation (Fig. 6, where red dots denote camera modules, green cones denote their fields of view, the white rectangle is the standard basketball court). The height of camera modules over the floor is 4.5 meters. This height ensures that the basketball fans will not occlude the scene. The vertical field of view of 25 degrees is well enough to cover the whole court with the player even jumping.

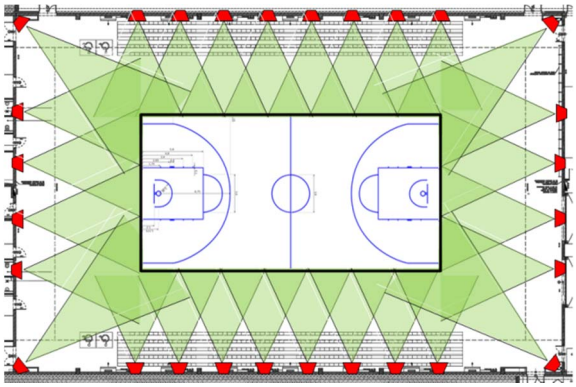

Fig. 6. The sports hall with an FTV acquisition system.

7. 3D HEVC EXTENSION FOR LINK B OF FTV SYSTEMS

As mentioned in Section 2, when the rendering server is distant from the representation server, in Link B (Fig. 1), the MVD representation should be compressed but the standard 3D extensions of AVC and HEVC are not efficient for cameras sparsely distributed around a scene. A more efficient extension has been already proposed in [16]. Here, we use the approach from [16] to develop a more efficient MVD codec for arbitrary located cameras. This codec exploits the derivation of the disparity vectors with nonzero vertical components. This implies also modification of the following tools: Disparity Compensated Prediction, Neighboring Block Disparity Vector (NBDV), Depth-oriented NBDV, View Synthesis Prediction, Inter-view Motion Prediction, Illumination Compensation. These modifications are similar as in [16] but they are embedded into another implementation.

Unfortunately, for compression efficiency, quite few results are available for higher numbers of cameras sparsely located on an arc [25]. We examine three available techniques (MV-HEVC, 3D-HEVC, [9,15] and our implementation (based on [16]) in the conditions similar to those in Link B.

For the experiments, we use HTM 13.0 software [26] for 3D-HEVC and MV-HEVC, and our implementation built on the top of HTM 13.0. The coding experiments are done for 7 views with the corresponding depth maps for the test sequences: *Poznan Blocks* (all views except the utmost left and right) [27], *Big Buck Bunny Flowers* (views 6,19,32,45,58,71,84) [28], *Ballet* and *Breakdancers* (all views) [29]. The configuration for all codecs is similar as in [30], i.e. Main Profile, GOP size = 8, intra period = 24, hierarchical GOPs on, 4 reference frames, Neighboring Block Disparity Vector on, Depth oriented NBDV on, View Synthesis Prediction on, Inter-view Motion Prediction on, Illumination Compensation on but View Synthesis Optimization for Depth Coding switched off. The comparison of the compression performance is made using PSNR for luma (Tables 4 and 5).

Table 4. Average luma bitrate reductions calculated according to the Bjøntegaard formula [31].

	Our vs 3D-HEVC	Our vs MV-HEVC	3D-HEVC vs MV-HEVC
Poznan Blocks	-6.44%	-4.20%	2.37%
BBB_Flowers	-3.03%	-2.80%	0.21%
Ballet	-8.64%	-12.55%	-4.32%
Breakdancers	-9.79%	-13.71%	-4.39%
Average	-6.97%	-8.32%	-1.53%

Table 5. Compression of 7 views with the depth maps.

Sequence	QP	MV-HEVC		3D-HEVC		Our	
		Bitrate [Mbps]	PSNR [dB]	Bitrate [Mbps]	PSNR [dB]	Bitrate [Mbps]	PSNR [dB]
Poznan Blocks [27]	25	6.85	43.0	6.81	42.9	6.61	43.0
	30	3.76	40.4	3.75	40.2	3.59	40.3
	35	2.14	37.6	2.11	37.4	2.01	37.5
	40	1.22	34.7	1.21	34.5	1.13	34.6
BBB_Flowers [28]	25	6.19	40.5	6.10	40.4	6.01	40.4
	30	3.25	37.7	3.20	37.6	3.13	37.6
	35	1.80	35.0	1.76	34.9	1.71	34.9
	40	1.01	32.2	0.99	32.1	0.95	32.1
Ballet [29]	25	2.06	41.4	1.89	41.4	1.82	41.4
	30	1.05	39.9	0.95	39.7	0.91	39.8
	35	0.59	37.9	0.52	37.6	0.50	37.8
	40	0.33	35.6	0.30	35.4	0.28	35.5
Breakdancers [29]	25	4.66	39.0	4.31	39.0	4.15	39.0
	30	1.96	37.6	1.76	37.4	1.67	37.5
	35	1.02	35.8	0.92	35.7	0.85	35.8
	40	0.54	33.8	0.49	33.7	0.45	33.8

The results demonstrate average bitrate reduction of the proposed technique versus the state-of-the-art 3D-HEVC of order of 6% (similar like in [16]). This result encourages further research on the MVD compression for FTV.

8. AUDIO PROCESSING FOR FREE NAVIGATION

A desirable feature of a FTV system is a realistic reconstruction of the auditory scene corresponding to the virtual position of the observer. This issue was not addressed by many works yet. Some advanced results are already reported in [1,32] where the blind source separation was proposed for the whole scene and also the transmission of such an acoustic scene representation through Link C was considered. Here, according to the experiments already done, we aim at simple systems. Therefore, we achieve the goal by simulating a virtual stereo pair of microphones that moves around and in the scene synchronously with the virtual camera. In this way, the spatial cues represented by inter-channel intensity and phase differences create a credible auditory image at the receiver, similarly to the traditional stereophonic setup which is reproducing a spatially-static content prepared in a studio [33]. The auditory scene is captured by the system through sampling of the spatio-temporal sound wave with a set of microphones positioned together with camera modules.

Every camera module is equipped with a pair of microphones in the classic MS (middle-side) configuration [34]. One microphone, of a cardioidal sensitivity pattern, has its maximum of sensitivity coherent with the field of view of the main camera in the camera module. The second microphone, having the polar pattern of figure-of-eight, is operating perpendicularly to the first one. The purpose of this setup is to allow for capturing independently both the sound produced by object targeted by the camera and the side sounds representing the objects surrounding the virtual listener. The signals from both microphones are subject to conditioning, consisting of dynamic processing and equalization, before they are mixed into a stereo pair.

Two degrees of freedom of the movement around and in the scene can be addressed by signal processing: zoom and panning. Zooming is achieved by a simple manipulation of the stereo field obtained by mixing of the M and S signals. As the virtual listener approaches the middle of the scene, a head-related transfer

function HRTF approximating filter is applied to the S signal in a way that yields attenuation and loss of high frequency components, simulating the effect of leaving the background sound sources behind. An additional diffusion procedure is applied to the M signal before it is mapped to a stereo pair in two incoherent versions, simulating the effect of widening the auditory image related to the target object [35].

Moving around the scene (auditory panning) requires an effective interpolation of the stereo images captured by the virtual stereo pairs. Such interpolation needs to take into account the different time of arrival of different sounds at the microphones, in order to avoid undesirable echos and comb filtering when signals are combined in the interpolation procedure. This is achieved by manipulation of the content in time-frequency domain. The signal is split into subbands corresponding to the critical bands of the human auditory system. In each band, the dominant ICTD (inter-channel time difference) is determined based on the correlation method [36]. For the purpose of spatial interpolation, these subband signals are temporally shifted by the part of delay corresponding to the intermediate position, thus forming two temporally adjusted versions of the signals captured by both microphones. These two signals are combined by linear mixing with weights depending on the intermediate position. The procedure is repeated in each of the subbands, and resulting signals are obtained in the synthesis filterbank (Fig. 7). Preliminary experiments demonstrate satisfactory subjective quality.

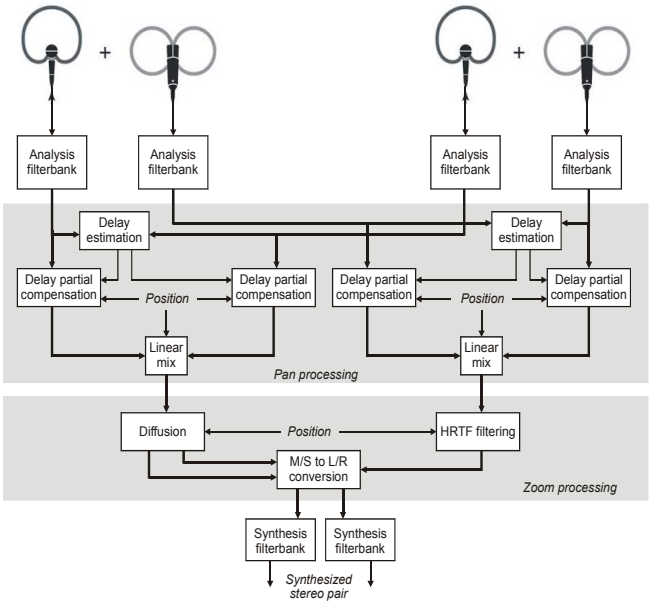

Fig. 7. The block diagram of the FTV audio processing: synthesis of virtual stereo audio for a position in between two modules. The position information comes from the video path.

In the proposed system, the stereophonic audio is synthesized at an arbitrary position from the audio signals from two modules only. This processing is performed in the rendering server with exception of the audio preprocessing in the representation server.

9. RENDERING SERVER

The rendering server responds to the requests from a user and streams video and audio for the requested viewpoint. This needs that the video and audio frames are synthesized according to the current viewpoint defined by a user. Unlike some other works [17], currently we aim at internet delivery only as the terrestrial and satellite broadcasting are too expensive for a small number of initial users.

For linear camera arrangements, the view-synthesis real-time implementations are known for graphical processing units (GPUs) [42,43]. For camera located on an arc, the synthesis is significantly more complex [40,41] but still doable on a GPU in real time. Thus, we designed the video processing as a set of GPUs each serving some users at a time. The connection request service, the position calculations and the connection and processing control are implemented in software. In parallel to video, stereo audio is also synthesized in software for the current viewpoint as described in Section 8. For a single user, the piece hardware for view synthesis and coding as well as the software for audio processing form a virtual processing block (Fig. 8) that is hired for a user for the time of a viewing session. The indicative latency budget is set to 350 ms including 150 ms given to the position calculation, view synthesis, video coding and buffering, 100ms for video decoding and buffering, and 100 ms for the round-trip packet travel time including operational system response times. The low-latency requirements imply the low-delay video coding to be applied. Therefore, for basketball games, the AVC bitrates are about 15-20 Mbps for HD and 4-6 Mbps for SD for fast virtual walking.

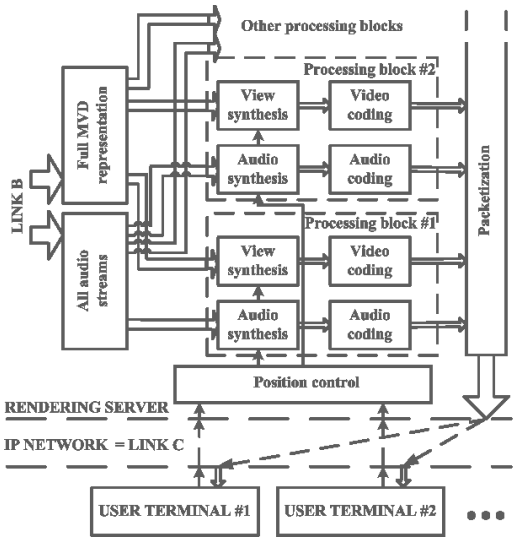

Fig. 8. The rendering server. The user entitlement control and the user connection control blocks are not shown.

10. CONCLUSIONS

In the paper we consider an original entire structure of a simple low-cost FTV system in contradiction to the sophisticated systems usually considered in the references. Unlike to most FTV contributions, both the video and audio parts are considered, and an original simple audio processing technique is proposed for low-cost FTV systems. Also straightforward schemes for the rendering server, and for video and audio streaming are proposed. The novelty of the paper is related also to the proposal to build the acquisition system using the two- or three-camera modules. The advantages of such a system are demonstrated empirically using the first test video sequences acquired from such camera modules. The paper also provides the description of these test sequences granted for the use within the research community. The paper also provides the original results on the improvements of the state-of-the-art 3D-HEVC compression technology. All these results

together with the other results cited in the paper encourage us to believe that the development of usable FTV systems will be possible within the next very few years.

REFERENCES

[1] M. Tanimoto, et al., "FTV for 3-D spatial communication", *Proc. IEEE*, vol. 100, pp. 905-917, 2012.

[2] E. Bondarev, R. Miquel, M. Imbert, S. Zinger, P. de With, "On the technology roadmap of free-viewpoint 3DTV receivers", *IEEE Int. Conf. Consumer Electronics*, Las Vegas, pp. 687 – 688, 2011.

[3] M. Domański, et al., „A practical approach to acquisition and processing of free viewpoint video," *Picture Coding Symposium PCS 2015*, Cairns, pp. 10-14.

[4] M. Domański, et al., "Experiments on acquisition and processing of video for free-viewpoint television", *3DTV-CON*, Budapest 2014.

[5] M. Domański, et al., "Comments on further standardization for free-viewpoint television," *MPEG* M35842 Geneva 2015.

[6] Stamos, P.K. Allen, "Integration of range and image sensing for photo-realistic 3D modeling", *IEEE Int. Conf. Robotics and Automation ICRA 2000*, vol. 2, pp. 1435-1440.

[7] D. Sandberg, P.-E. Forssen, J. Ogniewski, "Model-based video coding using colour and depth cameras," *2011 Int. Conf. Digital Image Computing Techn. .Appl.,* pp.158-163.

[8] *ISO/IEC IS 15444-3, ITU-T Rec. T.802*, "JPEG 2000 image coding system, Part 3: Motion JPEG2000," 2007.

[9] *ISO/IEC IS 23008-2, ITU-T Rec. H.265*, "High efficiency coding and media delivery in heterogeneous environments -- Part 2: High Efficiency Video Coding," 2015.

[10] G. Miller, J. Starck, A. Hilton, "Projective surface refinement for free-viewpoint video," *3rd European Conf. Visual Media Production, CVMP 2006,* pp.153-162.

[11] A. Smolic, et al., "3D video objects for interactive applications." *European Signal Proc. Conf. EUSIPCO 2005*.

[12] Müller K., Merkle P., and Wiegand T., "3D Video Representation Using Depth Maps", *Proc. IEEE*, vol. 99, pp. 643–656, Apr. 2011.

[13] *ISO/IEC IS 14496-10*, "Coding of audio-visual objects, Part 10: Advanced Video Coding," 2014.

[14] M. Tanimoto, et al., "Proposal on a new activity for the third phase of FTV", *MPEG* M30232, Vienna, 2013.

[15] G.J. Sullivan, et al., "Standardized Extensions of High Efficiency Video Coding (HEVC)", *IEEE Journal Selected Topics Signal Proc.*, vol. 7, pp. 1001–1016, Dec 2013.

[16] J. Stankowski, et al., „3D-HEVC extension for circular camera arrangements," *3DTV-CON*, Lisbon 2015.

[17] J. Kim, J. Jang, D. Ho Kim, "Design of platform and packet structure for the free-viewpoint television", *18th IEEE Int. Symposium Consumer Electronics*, Jeju Island, 2014.

[18] K.-Ch. Wei, Y.-L. Huang, S.-Y. Chien, "Point-based model construction for free-viewpoint tv," *IEEE Int. Conf. Consumer Electronics ICCE 2013*, Berlin, pp.220-221.

[19] L. Toni, G. Cheung, P. Frossard, „In-network view resampling for interactive free viewpoint video streaming", *IEEE Int. Conf. Image Proc. ICIP 2015*, pp. 4486-4490.

[20] T. Fujihashi, Z. Pan, T. Watanabe, "UMSM: A traffic reduction method on multi-view video streaming for multiple users", *IEEE Trans. Multimedia*, vol. 16, 2014, pp. 228-241.

[21] R. Hartley, A. Zisserman, "Multiple view geometry in computer vision" (2nd ed.), Cambridge Univ. Press, 2015.

[22] M. Tanimoto, "Overview of free viewpoint television", *Signal Proc.: Image Communic*., vol. 21, 2006, pp. 454-461.

[23] T. Senoh, K. Wegner, G. Lafruit, "Status of test sequences for free-viewpoint television (FTV)", *MPEG* M35804, Geneva, Feb. 2015.

[24] G. Lafruit et al, "FTV software framework", *MPEG* N15349, Warsaw, 2015.

[25] G. Lafruit, K. Wegner, M. Tanimoto, "Call for Evidence on Free-Viewpoint Television: Super-Multiview and Free Navigation", *MPEG* N15348, Warsaw, June 2015.

[26] 3D HEVC reference codec online https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-13.0

[27] M. Domański, et al., „Poznan Blocks – a multiview video test sequence and camera parameters for FTV", *MPEG* M32243, San Jose, Jan. 2014.

[28] P. T. Kovacs, "[FTV AHG] Big Buck Bunny light-field test sequences", *MPEG* M35721, Geneva, Feb. 2015

[29] C.L. Zitnick, et al., "High-quality video view interpolation using a layered representation," *ACM Trans. Graphics*, vol. 23, pp. 600-608, Aug. 2004.

[30] K. Müller, A. Vetro, "Common Test Conditions of 3DV Core Experiments", *JCT3V* G1100, San José, Jan. 2014

[31] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," *VCEG* M33*,* Austin, Apr. 2001.

[32] M. Tehrani et al., "3DAV integrated system featuring arbitrary listening-point and viewpoint generation", *IEEE Workshop Multimedia Signal Proc.*, 2008, p. 855-860.

[33] J. Jeppesen, H. Møller, Cues for localization in the horizontal plane, *Proc. 118th Convention of the Audio Engineering Society*, Paper 6323, Barcelona, 2005.

[34] W.L. Dooley, R. D., Streicher, M-S Stereo: A powerful technique for working in stereo, *Journal of the Audio Engineering Society*, vol. 30, pp. 707-18; October 1982.

[35] J. Vilkamo, V. Pulkki, Adaptive optimization of interchannel coherence with stereo and surround audio content, *Journal of the Audio Engineering Society*, vol. 62, pp. 861-869; 2014.

[36] C. H. Knapp, G. C. Carter, The generalized correlation method for estimation of time delay, *IEEE Trans. Acoustics, Speech Signal Proc.,* vol. 24, No. 4, pp. 320-327, 1976.

[37] M. Daribo, I. Cheung, G. Frossard, "Navigation domain representation for interactive multiview imaging", *IEEE Trans. Image Proc.,* vol. 22, pp. 3459 – 3472, Sept. 2013.

[38] O. Stankiewicz, K. Wegner, M. Tanimoto, M. Domański, "Enhanced Depth Estimation Reference Software (DERS) for Free-viewpoint Television", *MPEG* M31518, 2013.

[39] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations", *IEEE Int. Conf. Computer Vision*, vol. 1, pp. 666-673, 1999.

[40] O. Stankiewicz, K. Wegner, M. Tanimoto, M. Domański, "Enhanced view synthesis reference software (VSRS) for Free-viewpoint Television", *MPEG* M31520, 2013.

[41] L. Jorissen, P. Goorts, B. Bex, N. Michiels, S. Rogmans, P. Bekaert, G. Lafruit, "A qualitative comparison of MPEG view synthesis and light field rendering", *3DTV-CON*, Budapest 2014.

[42] L. Do, G. Bravo, S. Zinger, P.H.N. de With, "Real-time free-viewpoint DIBR on GPUs for large base-line multi-view 3DTV videos", *Visual Comm. Image Proc. (VCIP),* 2011.

[43] A. Akin, et al., "Real-time free viewpoint synthesis using three-camera disparity estimation hardware", *IEEE Int. Symp. Circuits Syst. ISCAS*, 2015.