Virtual View Synthesis for 3DoF+ Video

Adrian Dziembowski
Chair of Multimedia Telecommunications and Microelectronics
Poznań University of Technology
Poznań, Poland
adrian.dziembowski@put.poznan.pl

Dawid Mieloch
Chair of Multimedia Telecommunications and Microelectronics
Poznań University of Technology
Poznań, Poland
dawid.mieloch@put.poznan.pl

Olgierd Stankiewicz
Chair of Multimedia Telecommunications and Microelectronics
Poznań University of Technology
Poznań, Poland
olgierd.stankiewicz@put.poznan.pl

Marek Domański
Chair of Multimedia Telecommunications and Microelectronics
Poznań University of Technology
Poznań, Poland
marek.domanski@put.poznan.pl

Gwangsoon Lee
Electronics and Telecommunications Research Institute
Daejeon, Republic of Korea
gslee@etri.re.kr

Jeongil Seo
Electronics and Telecommunications Research Institute
Daejeon, Republic of Korea
seoji@etri.re.kr

***Abstract*— The paper reports a new view synthesis method for omnidirectional video with the ability to slightly displace a virtual viewpoint, i.e. the paper describes a novel synthesis method for 3DoF+ 360 video. This new method is noteworthy because of its high versatility and reliability: the method is appropriate for both perspective and omnidirectional input views, is able to render both perspective and omnidirectional views, and the produced synthetic views differ from the respective ground truth images less than with other view synthesis methods. These important features result from several innovations, e.g., prioritization of input views, efficient inpainting adapted to equirectangular projections, and efficient color correction. The experimental results demonstrate that the new method outperforms the state-of-the-art methods used hitherto.**

Keywords*— *view synthesis, virtual reality, virtual view, equirectangular projection.

I. INTRODUCTION

Development of view synthesis methods has a long history, from early view synthesis applications to multiview video coding (e.g. [17, 18]) as well as to robotics, virtual navigation and free-viewpoint television (e.g. [8, 19, 20, 21]). Recently, view synthesis has also proven to be highly important for omnidirectional video, a field of extensive research at present. Currently, significant research efforts are focused on 3DoF+ video, i.e. omnidirectional video with the ability to slightly displace a virtual viewpoint. The huge research efforts coincide with emerging commercial applications as well as standardization activities, e.g., the MPEG-I project [24], and in particular in the recent MPEG standardization activities on 3DoF+. In MPEG standardization works, the view synthesis technique RVS (Reference View Synthesizer) [5, 16] is currently used. Its code is publicly available and the method is recently used as the reference state-of-the-art technique, and it will be also used as the reference technique in this work.

Synthesis methods for 3DoF+ video have to face new challenges, for example, allowing processing of data from both perspective cameras and omnidirectional cameras [6]. New methods should ideally allow combined processing of both types of input data, e.g. synthesis of the omnidirectional video using one omnidirectional camera and two supporting perspective cameras. The existing view synthesis techniques like RVS [5, 16] as well as VSRS (View Synthesis Reference Software) [13] and its extensions [1, 15] usually do not allow the simultaneous use of both formats of input and output views.

On the other hand, in more versatile methods such as the Reference View Synthesizer [5, 16], some parts of the processing are based on the geometry of perspective cameras (e.g. in the inpainting of disoccluded areas), so they may not provide good results for omnidirectional views. The need to adapt image processing and analysis methods to be compatible with equirectangular projection (ERP) images is already clear in new proposals for intra-prediction [2] and motion estimation [3] methods.

Given the limitations of existing view synthesis methods, a new synthesis method is proposed in this paper. This method is adapted to characteristics of omnidirectional views and supports all required synthesis cases. In order to increase the quality of synthesized views, prioritization of input views, depth-based inpainting for omnidirectional views and a fast color correction technique have been introduced as new and original tools.

The method proposed in this paper profits from the usage of several views and depth maps as well as reduction of inpainting, like the method already proposed in [8]. The method proposed in this paper, inherits from [8] that the virtual views are synthesized firstly using two neighboring real views, but the disoccluded areas are not inpainted, but filled by the information from further real views. The whole synthesis is performed with triangles rather than with individual pixels, and it also exploits additional steps of adaptive color correction, blurred edge removal and spatial edge blurring. For the sake of brevity, these tools are not described here (for description please see [8]) but they are used in the system described here.

In contrary to [8], here we adopt the synthesis technology to 3DoF+ video, and we use additional tools in order to reach high quality of virtual views. The main novelty of the paper is reporting a technique with new or extended tools that is universal for perspective and omnidirectional inputs and outputs freely mixed.

Work supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. 2018-0-00207, Immersive Media Research Laboratory).

II. Proposed Method

A. Overview

The proposed method uses the multiview-plus-depth (MVD) representations [22, 23], extends the triangle-based multiview synthesis method (where disoccluded areas are filled from the further input views rather than inpainted [8]), and employs adaptive color correction, blurred edge removal and spatial edge blurring. The current proposal is focused on omnidirectional video, nevertheless, both omnidirectional and perspective formats of input and output views (with depth) are allowed. Moreover, there is no constraint on the number of cameras or their positioning, so any number of arbitrarily placed cameras can be used.

The overall scheme of the proposed method is depicted in Fig. 1. Each input view is projected to a virtual view separately. Then, the data projected from all the views are merged (regarding processing priorities, see Section II B) in order to produce the final virtual view.

Depending on the format of the input and output views, projection of the points from the input view to the virtual view may be done in any combination to/from omnidirectional or perspective views.

Projection involves two steps: first, the projection from the input view to the three-dimensional space, second, the projection from this three dimensional space to the final virtual view. Because of the different geometries of these views, projections for perspective and omnidirectional views must be performed using different equations, described in detail in [1].

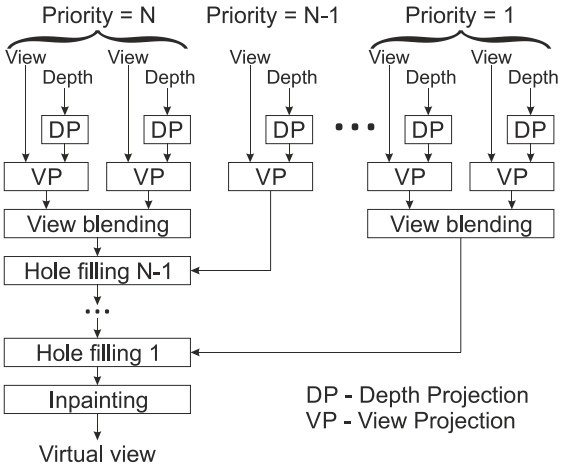


Fig. 1. The scheme of the proposed view synthesis method for omnidirectional video.

In the further paragraphs of Section II, selected tools are considered in details. Some of them are adopted and extended from previous works like [8, 10] (see Section IID), some other are new like that described in Sections IIB and IIC. The descriptions of the tools that are directly adopted from [8], like the triangle-based synthesis, blurred edge removal and spatial edge blurring are omitted for the sake of conciseness. The properties of the whole synthesis system are considered in Section III, where experimental results are reported.

B. Prioritization of input views

Some input views can more likely provide correct information for synthesis than others, e.g., the nearest view can more likely provide more valuable data for the virtual view synthesis than the more distant views. Therefore, the input views are prioritized for synthesis. The input views with the highest priority are used for synthesizing the virtual view, while the other input views may be used for disocclusion filling. The views with the same priority are blended before are used in the hole filling process. In general, the number of priorities is unlimited and holes in the virtual view are filled consecutively using points projected from the real views with lower priority.

The prioritization of input views is a new feature of the proposed method that is absent in the methods described in the references (e.g. [5], [15], [16]). In those methods, all of the views are treated equally, so the final color of each pixel is the weighted average of colors projected from different real views.

In our algorithm, the prioritization is performed automatically, and is based on the distance between real cameras and the virtual one – the nearest cameras receive the highest priority. The proposed approach provides improved quality of the virtual view by reducing the influence of color inconsistencies, depth artifacts and finite resolution of input views and depth maps [9].

C. Depth-based inpainting for omnidirectional views

In the most straightforward approach, inpainting of a disoccluded point uses two points projected to a synthesized virtual view, i.e. the nearest left and the nearest right point. Such an approach is not exact for equirectangular images where a sphere is projected onto a 2D image. It is well known that the shortest path between two points on the same circle of latitude is not a straight line on the 2D image if the circle of latitude is not the equator.

In omnidirectional images, the very important source of disocclusions is related to camera pan. Therefore the search for the nearest point is often the search for the neighbors on the respective circles of latitude, which not exactly mapped on a 2D image plane. Thus, the authors propose to use the transverse equirectangular projection (the Cassini projection [11] – see Fig. 2b), where always the shortest path between two points on a circle of latitude is mapped onto a straight line on 2D image.

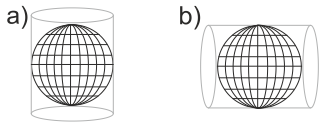


Fig. 2. a) Equirectangular projection and b) transverse equirectangular projection.

In order to facilitate the inpainting, the authors propose the fast approximate reprojection of an equirectangular image onto a transverse equirectangular image. First, the length of all rows in an equirectangular image is changed to correspond to the circumference of the corresponding circle on a sphere (Fig. 3a). In the second step, all columns of such image are expanded (Fig. 3b), to be of the same length (Fig. 3c).

After the two closest points are found in the transverse image, we perform a simple 2-way, depth-based inpainting of a disoccluded point. If the depth values of both closest points are similar, the color of the disoccluded point is the average color of its two nearest points, weighed by the

distances to these points. In the case of a greater difference of depth, only the color of point with greater depth value is used.

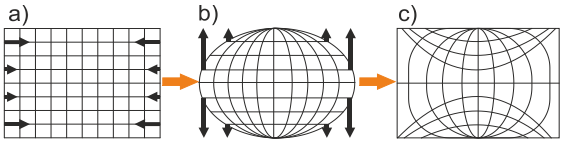


Fig. 3. Fast reprojection from an equirectangular image (a) to transverse equirectangular image (c). Black arrows show direction of change of size of respective rows and columns of images.

D. Fast color correction technique

Different views (especially natural ones, i.e. captured using physical cameras) may have inconsistent color characteristics, which may result in the appearance of color artifacts in the virtual view (Fig. 4b). These inconsistencies may result from diverse color characteristics of cameras as well as from different illuminations seen from individual directions of observation by individual cameras that capture the input views. The latter phenomenon is not an imperfection of the acquisition system but it often results in color artifacts when data from multiple input views are merged into one synthetic view.

In order to reduce this phenomenon, the authors propose a fast color correction technique, which equalizes global color differences for points projected from different real views. The global color difference between points projected from two real views is calculated as the mean ratio (averaged for the entire image) between color components projected from one view and color components projected from the other one. The algorithm is performed separately for all color components, e.g. Y, C_B, C_R.

In order to equalize colors of points projected from any real view i, color component values projected from view i are multiplied by the mean ratio between view i and the reference view (the view acquired by the closest real camera to the virtual one).

This technique is an adoption of the technique described in [10] that is extended for omnidirectional video here. For the sake of brevity, we omit the details of the core color correction technique that are already described in a previous paper of the authors [10].

III. Results

A. Goal of experimentation

The technique described in this paper and implemented in the respective software produced by the authors is very versatile. This technique is capable to work with arbitrarily mixed prospective and omnidirectional inputs and outputs.

In Section III, the results of performance comparisons are provided for four scenarios possible for Reference View Synthesizer (RVS) that was recently adopted by MPEG as the state-of-the-art technique for view synthesis. Indirectly it is a comparison to other relevant methods as RVS has already demonstrated its high performance in comparison to other methods [5], [21].

B. Test sequences

The experiments presented in this paper were performed using 5 sequences recommended by ISO/IEC MPEG for 3DoF+ research [6]. However, some cases, such as synthesis from an omnidirectional view to a perspective view, cannot be tested using these sequences due to a lack of reference views for both types. Therefore, four additional, highly diverse omnidirectional still pictures were added to the test set as proposed by Poznań University of Technology (PUT) and the Electronics and Telecommunications Research Institute (ETRI) [7].

In order to demonstrate the suitability of the proposed algorithm for different 3DoF+ applications, the test sequences were in two formats: equirectangular projection views from omnidirectional and hemispherical cameras (6 sequences) and perspective views from classical 2D cameras (3 sequences). In all tests, the first 100 frames of each sequence were processed (with exception of all *Poznan360* still images).

C. Comparison using the state-of-the-art WS-PSNR method

In the experiments, the proposed virtual view synthesis method was compared to the method implemented in the software provided by ISO/IEC MPEG (Reference View Synthesizer, RVS v.2.0) [16]. In the experiments, the methodology of MPEG-I [6] was observed.

The quality of synthesized virtual views was calculated using WS-PSNR (weighted-to-spherically-uniform PSNR) [12], which produces more reliable results than PSNR for omnidirectional video because of the reduction of the influence of resampling from a 3D sphere to a planar image.

1) View synthesis using two real views

In the first experiment, the virtual view was synthesized using two real views (with corresponding depth maps), which is the most common case for virtual reality systems with limited bandwidth [14].

a) Reference view b) RVS c) Proposed

Fig. 4. Result of the proposed color correction technique; fragment of *PoznanFencing2* sequence (for visualisation contrast increased by 20%).

TABLE I. View synthesis using 2 real views: the quality of the virtual views synthesized using RVS and the proposed method.

Sequence	Ref. view	Input views	WS-PSNR [dB]	
			RVS	Proposed
ClassroomVideo	0	9, 14	33.23	33.59
TechnicolorHijack	9	1, 4	38.76	40.06
TechnicolorPainter	5	4, 7	35.46	35.77
IntelFrog	7	4, 10	25.82	25.79
PoznanFencing2	4	3, 5	26.67	28.32
Average			**31.99**	**32.71**

The average quality improvement for 5 test sequences is 0.7 dB (see Table I). Notably, views in the *TechnicolorPainter* and *PoznanFencing2* sequences are not fully color consistent. Therefore, the proposed color correction technique allows the quality to improve even more. An example of color inconsistencies in a synthesized view is presented in Fig. 4.

2) View synthesis using all real views

In the second experiment, the virtual views were synthesized using all available real views (i.e. all excluding the real view in the position of the synthesized view). This is the most complex case, where all the views are transmitted to the receiver, which allows us to achieve the best possible quality (Table II).

TABLE II. VIEW SYNTHESIS USING ALL REAL VIEWS: QUALITY OF THE VIRTUAL VIEWS SYNTHESIZED USING RVS AND THE PROPOSED METHOD.

Sequence	Ref. view	WS-PSNR [dB]	
		RVS	Proposed
ClassroomVideo	0	34.80	34.93
TechnicolorHijack	9	43.74	43.73
TechnicolorPainter	5	37.58	37.61
IntelFrog	7	26.83	27.14
PoznanFencing2	4	26.77	28.32
Average		**33.94**	**34.35**

a) Reference view b) RVS c) Proposed

Fig. 5. Result of proposed real view prioritization technique; fragment of *PoznanFencing2* sequence.

On average, the proposed view synthesis method improves the quality by about 0.4 dB. The proposed view prioritization technique is the factor that yields the quality gain. In RVS, the color of each pixel is calculated as a weighted average of colors projected from all the real views, what causes artifacts if depth maps are inconsistent (Fig. 5).

3) View synthesis using one real view

In the third experiment, testing the view synthesis of each virtual view, only one real view was used. In that case, all the areas of the virtual view, that are not visible in the real view, are inpainted. Therefore, this experiment demonstrates how the proposed method handles the filling of disocclusions.

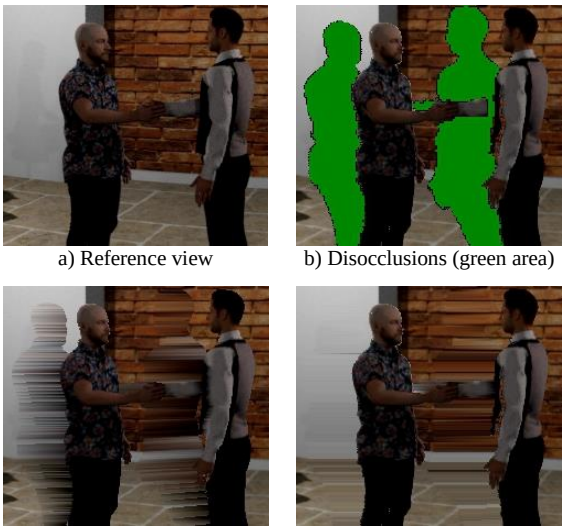
The experiment was performed only for omnidirectional sequences, because for that type of data non-projected areas represent disocclusions only, while perspective cameras also represent the regions close to the boundary of the image).

TABLE III. VIEW SYNTHESIS USING ONE REAL VIEW: THE QUALITY OF THE VIRTUAL VIEWS SYNTHESIZED USING RVS AND THE PROPOSED METHOD.

Sequence	WS-PSNR [dB]	
	RVS	Proposed
ClassroomVideo	29.77	31.29
PoznanBlocks360	27.82	28.76
PoznanHouse360	27.60	28.08
PoznanPeople360	31.39	32.31
PoznanSpace360	36.26	42.45
Average	**30.57**	**32.58**

The quality (Table III) was estimated for 5 test sequences. The sequence *TechnicolorHijack* was excluded from this test, since it is a hemi-spherical sequence, whereas the other sequences were filmed using perspective cameras.

The results clearly show that the proposed inpainting technique allows us to achieve a significantly better quality of virtual views as compared to RVS, where disocclusions are filled by interpolation within a triangle-based pixel projection. The visual comparison of disocclusion fillings in RVS and the proposed method is presented in Fig. 6.



a) Reference view b) Disocclusions (green area)

c) RVS d) Proposed

Fig. 6. Result of proposed inpainting technique; fragment of *PoznanPeople360* picture.

4) Omnidirectional to perspective view synthesis

In the last experiment, the projection of an omnidirectional video to a perspective video was analyzed. Unfortunately, none of the sequences from the test set recommended by MPEG for 3DoF+ research contains both omnidirectional and perspective views. Therefore, in order to calculate the quality of the perspective virtual view, 4 test 360-degree images were used. The results are presented in Table IV.

TABLE IV. OMNIDIRECTIONAL TO PERSPECTIVE VIEW SYNTHESIS: QUALITY OF VIRTUAL VIEWS SYNTHESIZED USING RVS AND THE PROPOSED METHOD.

Sequence	WS-PSNR [dB]	
	RVS	Proposed
PoznanBlocks360	21.17	21.30
PoznanHouse360	22.92	23.35
PoznanPeople360	24.60	25.28
PoznanSpace360	27.71	28.02
Average	**24.10**	**24.49**

a) Reference view b) RVS c) Proposed

Fig. 7. Omnidirectional to perspective view synthesis: proposed method vs. RVS; fragment of *PoznanHouse360* picture.

On average, the PSNR of virtual views synthesized using the proposed method is 0.4 dB higher than for views synthesized using RVS. Subjectively, the views generated using proposed methods have significantly fewer disturbing artifacts (Fig. 7).

IV. Conclusions

The novelty of the work is related to the development of the entire view synthesis system appropriate for versatile mixing of input and output perspective and omnidirectional video. The systems merges many approaches that were used (like those from the previous work of the authors [8]), adopted (like from [10]) or newly designed for the purpose of the work (e.g. see Section IIC).

In the proposal, the virtual views are synthesized firstly using two neighboring real views, but the disoccluded areas are not inpainted, but filled by the information from further real views. The whole synthesis is performed with triangles rather than with individual pixels, and it also exploits additional steps of adaptive color correction, blurred edge removal and spatial edge blurring (described in [8]). These tools add significantly to the superior overall performance of the proposed method. We introduce also new innovative tools focused on 3DoF+ video, such as prioritization of input views, efficient inpainting adapted to equirectangular projections, and efficient color correction.

Moreover, the paper provides the original experimental results obtained for this new view synthesis system in comparison to the state-of-the-art Reference View Synthesizer (RVS), recently adopted by MPEG as the reference technique for view synthesis. The results were obtained for four realistic scenarios of synthesis, and they prove that the proposed technique clearly outperforms RVS in average by 0.5 – 1.5 dB in WS-PSNR [12], when using the standardized MPEG-I experimentation methodology [6]. As already mentioned, RVS has already demonstrated its high performance in comparison to other methods [5], [21]. Therefore, one may conclude that the proposed method has proved to exhibit the top efficiency for omnidirectional video.

The computational cost of the proposed method is moderate. The rough comparisons run on PC computers demonstrate that the processing times for the proposed method are about 50 – 110 % of those for RVS.

References

[1] K. Wegner, D. Łosiewicz, T. Grajek, O. Stankiewicz, A. Dziembowski and M. Domański, "Omnidirectional view synthesis and test images," 2018 Int. Conf. .Signals .Electronic Systems (ICSES), Kraków, pp. 130-133.

[2] Y. Wang, Y. Li, D. Yang and Z. Chen, "A fast intra prediction algorithm for 360-degree equirectangular panoramic video," 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, 2017.

[3] Y. Wang, L. Li, D. Liu, F. Wu and W. Gao, "A new motion model for panoramic video coding," 2017 IEEE Int. Conf. . Image Processing (ICIP), Beijing, 2017, pp. 1407-1411.

[4] M. Domański, O. Stankiewicz, K. Wegner and T. Grajek, "Immersive visual media — MPEG-I: 360 video, virtual navigation and beyond," 2017 Int. Conf. . Systems, Signals and Image Processing (IWSSIP), Poznan, 2017.

[5] S. Fachada, D. Bonatto, A. Schenkels and G. Lafruit, "Depth image based view synthesis with multiple reference views for virtual reality," 2018 - 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), Helsinki.

[6] "Common Test Conditions on 3DoF+ and windowed 6DoF," ISO/IEC JTC1/SC29 WG11, MPEG 124th meeting, Macau, October 2018, Doc. N18089.

[7] O. Stankiewicz, K. Wegner, A. Dziembowski, M. Lorkiewicz, G. Lee, J. Seo, M. Domański, "Proposed test materials for 3DoF+ or Omnidirectional 6DoF," ISO/IEC JTC1/SC29 WG11 Doc. MPEG M44461, Macau, October 2018.

[8] A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, K. Wegner, M. Domański, "Multiview synthesis – improved view synthesis for virtual navigation," 32nd Picture Coding Symposium, PCS 2016, Nürnberg, Germany.

[9] A. Dziembowski, J. Samelak and M. Domański, "View selection for virtual view synthesis in free navigation systems," 2018 Int. Conf. Signals Electronic Syst. (ICSES), Kraków, pp. 83-87.

[10] A. Dziembowski, O. Stankiewicz, "[MPEG-I Visual] Fast color correction technique for view synthesis," ISO/IEC JTC1/SC29 WG11 Doc. MPEG M43694, Ljubljana, July 2018.

[11] J. Snyder, P. Voxland, "An album of map projections", US Government Printing Office, Washington, 1989.

[12] Y. Sun, A. Lu and L. Yu, "Weighted-to-spherically-uniform quality evaluation for omnidirectional video," IEEE Signal Processing Letters, vol. 24, no. 9, pp. 1408-1412, Sept. 2017.

[13] T. Senoh, K. Yamamoto, N. Tetsutani, H. Yasuda, and K. Wegner, "View synthesis reference software (VSRS) 4.2 with improved inpainting and hole filing," ISO/IEC JTC1/SC29/WG11 MPEG 118th meeting, Hobart, April 2017, Doc. M40657.

[14] J. Jeong, D. Jang, J. Son and E. Ryu, "Bitrate Efficient 3DoF+ 360 Video View Synthesis for Immersive VR Video Streaming," 2018 Int. Conf. Information and Communication Technology Convergence (ICTC), Jeju, 2018, pp. 581-586.

[15] M. Domański, D. Łosiewicz, T. Grajek, O. Stankiewicz, K. Wegner, A. Dziembowski and D. Mieloch, "Extended VSRS for 360 degree video," ISO/IEC JTC1/SC29 WG11 Doc. MPEG M41990, Gwangju, January 2018.

[16] "Reference View Synthesizer (RVS) 2.0 manual," ISO/IEC JTC1/SC29 WG11 MPEG 123th Meeting, Ljubljana, July 2018, Doc. N17759.

[17] E. Martinian, A. Behrens, J. Xin, A. Vetro, "View Synthesis for Multiview Video Compression," Picture Coding Symposium, 2006.

[18] S. Yea, A. Vetro, "View synthesis prediction for multiview video coding," Signal Processing: Image Communication, Volume 24, Issues 1-2, pp. 89-100, 2008.

[19] C. Zhu, S. Li, "Depth image based view synthesis: New insights and perspectives on hole generation and filling," IEEE Trans. Broadcasting, Vol. 62, pp. 82-93, 2015.

[20] T. Tezuka, M. Tehrani, K. Tkahashi, T. Fuji, "View synthesis using superpixel based inpainting capable of occlusion handling aand hole filling," Picture Coding Symposium, pp. 124-128, 2015.

[21] B. Ceulemans, Sh.-P. Lu, G. Lafruit, A. Munteanu, "Robust multiview synthesis for wide-baseline camera arrays," IEEE Trans. Multimedia, vol. 20, pp. 2235-2248, 2018.

[22] Müller K., Merkle P., and Wiegand T., "3D video representation using depth maps", Proc. IEEE, vol. 99, pp. 643–656, Apr. 2011.

[23] O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, J. Samelak, "A free-viewpoint television system for horizontal virtual navigation," IEEE Transactions on Multimedia, vol. 20, pp. 2182 – 2195, August 2018.

[24] MPEG-I, https://mpeg.chiariglione.org/standards/mpeg-i.