ADAPTIVE COLOR CORRECTION IN VIRTUAL VIEW SYNTHESIS

Adrian Dziembowski, Marek Domański

Chair of Multimedia Telecommunications and Microelectronics,
Poznan University of Technology, Poznań, Poland

ABSTRACT

In the paper an adaptive color correction method for virtual view synthesis is presented. It deals with the typical problem in free navigation systems – different illumination in views captured by different cameras acquiring the scene. The proposed technique adjusts the local color characteristics of objects visible in two real views. That approach allows to significantly reduce number and visibility of color artifacts in the virtual view. Proposed method was tested on 12 multiview test sequences. Obtained and presented in the paper results show, that proposed color correction provides increase of the virtual view quality measured by PSNR, SSIM and subjective evaluation.

Index Terms **—** virtual view synthesis, color correction, free-viewpoint television, free navigation

1. INTRODUCTION

Recently, attention is given to research on virtual navigation in real 3D scenes, where a viewer is able to virtually walk in a dynamic scene thus smoothly changing the virtual viewpoint and the virtual view direction [1]. Such an ability is characteristic for free-viewpoint television [2, 3] and virtual reality systems (cf. Fig. 1).

Figure 1. View synthesis idea

For the practical reasons, the virtual views are synthesized (rendered) from video sequences acquired using limited set of real cameras. Therefore, the linear or angular distances (cf. Fig. 1) between the real cameras are significant.. It is the most popular, and widely-used approach [4, 5, 6, 7, 8, 9].

In the paper, we assume that any virtual view is synthesized from two real views only, let us call them the reference views (cf. Fig. 2).

In fact, for distant real cameras, the two reference view items depict scene objects seen from different directions. Thus the angles between the optical axis of a camera and the directions from the illumination sources are different for the two views acquired by two real cameras. As the real object surfaces are mostly non-lambertian [10], the objects in both reference views appear different, i.e. their registered colors appear different even for the same point on the surface of a real object. Moreover, in particular the consumer-market cameras are highly automated, and their adjustments are done independently.

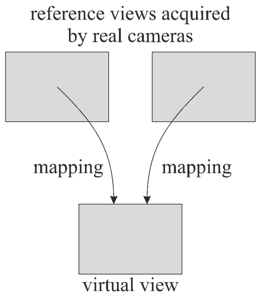

Fig. 2. Virtual view synthesis (rendering)

The two abovementioned observations lead us to the conclusion that the colors of pixels in the synthesized view depend on the reference view. In the virtual views, the luma and chroma samples can be categorized into 3 groups:

1. Samples are projected from both real views. Their actual values result from blending of the colors of the corresponding pixels from two views.
2. Due to occlusions, samples are projected from one real view only. In such a case, for a synthesized sample, the color depends on the single respective color from the reference view used. Nevertheless, at least some of the neighboring samples may be synthesized from i.e. two reference views. Thus colors of the neighboring samples may be different despite that they originate from the neighboring samples.
3. Due to occlusions, there is no direct mapping from the reference image samples to a given sample in the virtual view. The respective samples in the virtual view have to be inpainted and their average colors need to be adjusted to their neighborhood in the virtual view.

Number of color correction methods are known, parametric and non-parametric [11], including color transfer [12] or histogram matching method [13]. However, both methods cannot be implicitly used because of different objects visible in different real views.

Of course, there are methods calculating color differences only for common area of different views, using SIFT [14, 15], SURF [16] or optimization techniques [17, 18].

However, existing color correction methods adjust difference of characteristics of the entire view and do not allow to compensate local color characteristics, different for various objects of the acquired scene.

2. PROPOSED ALGORITHM

In presented approach, color correction is being performed only for the second group of points in the virtual view, i.e. points projected from one real view only.

For the sake of clarity, only luma processing is described below, but other color components are being processed in the same way.

In the first step, for every real view the **ratio map** is created. For every point from the first group, the ratio between blended luma value and the value projected from one real view is calculated. These ratios are inserted in proper ratio maps:

$$M_l(x,y) = \frac{c_V(x,y)}{c_l(x,y)}, \qquad M_r(x,y) = \frac{c_V(x,y)}{c_l(x,y)}, \tag{1}$$

where M_l and M_r are ratio maps for left and right real view, $c_l(x,y)$ and $c_r(x,y)$ are luma values projected from left and right real view into the point (x,y) in the virtual view; $c_V(x,y)$ is the blended luma value in the point at position (x,y). For points from groups other than 1, the ratios are not calculated.

In the second step, all the points from the second group are multiplied by locally-averaged ratio $\overline{M_l}(x,y)$ (points projected from left real view) or $\overline{M_r}(x,y)$ (points from right real view).

Locally-averaged ratio for left ratio map can be obtained by:

$$\overline{M_l}(x,y) = \frac{1}{|P|} \sum_{(w,h)\in P} M_l(w,h), \tag{2}$$

where P is the set of all these points in the neighborhood of point (x,y), for which the ratio map contains any value. 'Neighborhood' is the rectangular window of the width and height of over a dozen sampling periods, e.g. 13 x 13. Of course, locally-averaged ratio $\overline{M_r}(x,y)$ is calculated analogously.

Corrected luma value for the virtual view's point is calculated using:

$$c_V(x,y) = c_l(x,y) \cdot \overline{M_l}(x,y) \tag{3}$$

for points projected only from the left real view, and:

$$c_V(x,y) = c_r(x,y) \cdot \overline{M_r}(x,y) \tag{4}$$

for points from the right real view.

Let us consider an example. In Figs. 3A and 3B fragments of two instances of the virtual view are presented (each instance was created by projection of points from only one real view, e.g. the left view for Fig. 3A).

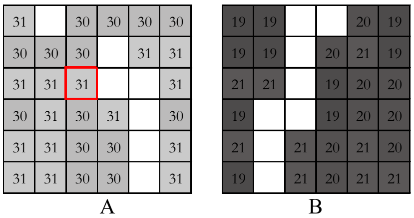

Figure 3. Virtual view instances (A, B), containing points projected from left and right real view; C – combined virtual view, containing points projected from both real views

For the example purposes it was assumed, that the depth value for all the points is equal and the distance between virtual camera and both real ones is the same. In that case, luma of the points of the virtual view is obtained by averaging luma values of the pixel in both instances of the virtual view.

Luma projected from both real views is significantly different, causing appearance of the artifacts in the virtual view (Fig. 3C) in the areas projected from only one real view. For example, point highlighted in red was projected from only left view, so without color correction it would have the same luma value in the combined virtual view as in its left instance (Fig. 3A): 31.

In the first step for left and right instance of the virtual view the ratio map is being calculated (Fig. 4).

Then, luma value of every point projected from only one real view is multiplied by local-averaged ratio. For example, for highlighted point in Fig. 2, all the ratio map's value in its neighborhood are averaged (for the sake of clarity, small size of neighborhood was chosen, 3 x 3) resulting in average ratio equal to 0.82. Therefore, luma value of analyzed point after color correction is 25 – according to (3): 25 = 31 · 0.82.

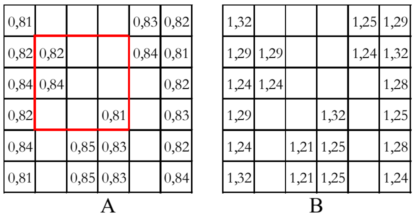

Figure 4. Exemplary ratio maps for left (A) and right (B) instance of the virtual view

In Figs. 5A and 5B corrected luma value for points projected from left and right view are presented. Fig. 5C presents combined virtual view after proposed color correction.

For greater areas projected from only one real view, presented approach would be insufficient, because for some of the points there would be no ratio values in their neighborhood (no point projected from both views within analyzed window).

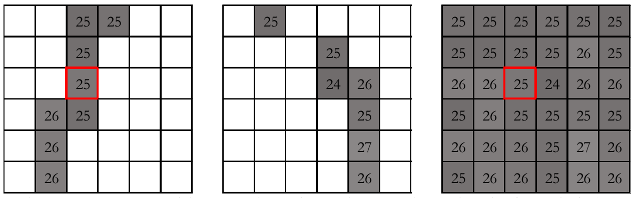

Figure 5. Corrected luma values for points projected only from left (A) and right (B) real view; C – combined virtual view after color correction

In order to eliminate that problem, presented approach is iterative and locally-averaged ratios (2) are inserted into ratio maps and used in the next iteration.

3. EXPERIMENTS

All the experiments were performed using MVS algorithm [19], because it provides higher quality of synthesized views than state-of-the-art technique – VSRS [20].

In order to estimate quality gain caused by proposed color correction method, set of 12 miscellaneous test sequences was chosen (Table I), including 9 sequences captured by sparse multi-camera systems with cameras placed on an arc and 3 linear sequences – *Soccer Linear*, *Poznan_Carpark* and *Poznan_Street*.

Table I – Test sequences

	Sequence	Source
A	Big Buck Bunny Butterfly	Holografika [21]
	Big Buck Bunny Flowers	Holografika [21]
	Poznan_Blocks	Poznan Univ. of Technology [22]
	Poznan_Blocks2	Poznan Univ. of Technology [23]
	Poznan_Fencing2	Poznan Univ. of Technology [23]
	Poznan_Service2	Poznan Univ. of Technology [23]
	Ballet	Microsoft Research [24]
	Breakdancers	Microsoft Research [24]
	Soccer Arc	Hasselt University [25]
L	Soccer Linear	Hasselt Univeristy [25]
	Poznan_Carpark	Poznan Univ. of Technology [26]
	Poznan_Street	Poznan Univ. of Technology [26]

A – cameras arranged on an arc, L – linear camera arrangement

The purpose of color correction is to increase **subjective quality** of synthesized views. To measure subjective quality gain, PC method [27] was chosen. In order to show that proposed method also increases the objective quality, two metrics were used: PSNR and SSIM [28]. For objective quality calculation virtual view was synthesized in a position of a real camera and then compared with a real view. In PC tests, viewers were evaluating the difference of quality of views synthesized with and without proposed color correction on the scale from -3 to 3. In subjective tests 8 experts were involved.

4. RESULTS

Table II contains objective quality of the virtual views measured as PSNR and SSIM. Higher values were bolded for both metrics. Average PSNR gain is 0.18 dB for all the test sequences and 0.23 dB for sequences captured by sparse multicamera systems. For SSIM, the quality increase is, on average, 0.001.

Table II – Objective quality of synthesized virtual views

Sequence	PSNR [dB]		SSIM	
	no color correction	**color correction**	**no color correction**	**color correction**
BBB Butterfly	32.30	32.30	0.959	0.959
BBB Flowers	24.62	24.62	0.854	0.854
Poznan_Blocks	**23.79**	23.78	0.760	**0.761**
Poznan_Blocks2	29.07	**29.59**	0.842	**0.843**
Poznan_Fencing2	28.36	**28.98**	0.858	**0.861**
Poznan_Service2	23.44	**23.75**	0.783	**0.786**
Ballet	28.68	**29.05**	0.822	0.822
Breakdancers	30.89	**31.08**	0.814	0.814
Soccer Arc	20.88	**20.98**	0.739	**0.743**
Soccer Linear	32.84	32.84	0.883	0.883
Poznan_Carpark	31.81	31.81	0.915	0.915
Poznan_Street	33.23	33.23	0.920	0.920

In Fig. 6 the subjective quality gain affected by using proposed color correction method is presented.

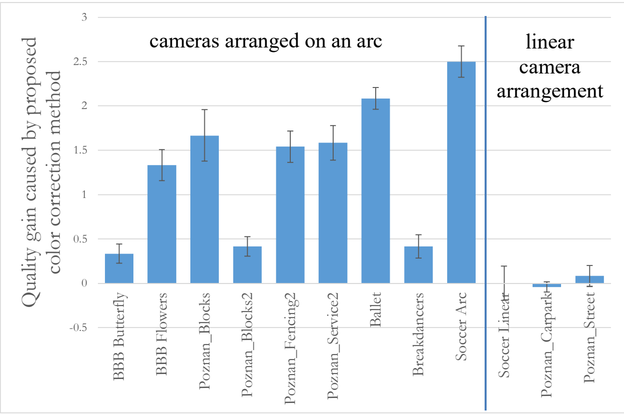

Figure 6. Subjective quality gain caused by proposed color correction (significance level: 5%)

For 9 of 12 test sequences (all the sequences captured by systems with cameras placed on an arc), the quality gain caused by usage of proposed color correction method is statistically significant (significance level: 5%). For all the sequences captured by linear multicamera systems there was no quality decrease.

Removal of the color artifacts is presented in Figs. 7 – 10.

A B

Figure 7. Fragment of the synthesized view before (A) and after (B) color correction; sequence *BBB Flowers*

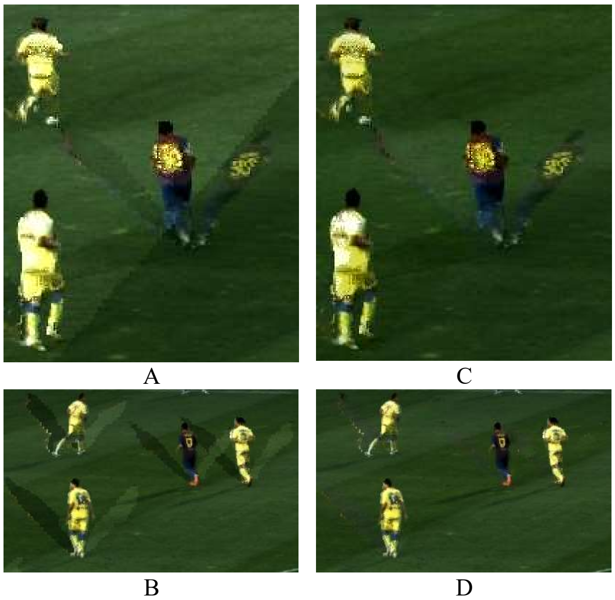

Figure 8. Fragments of the synthesized view before (A, B) and after (C, D) color correction; sequence *Soccer Arc*

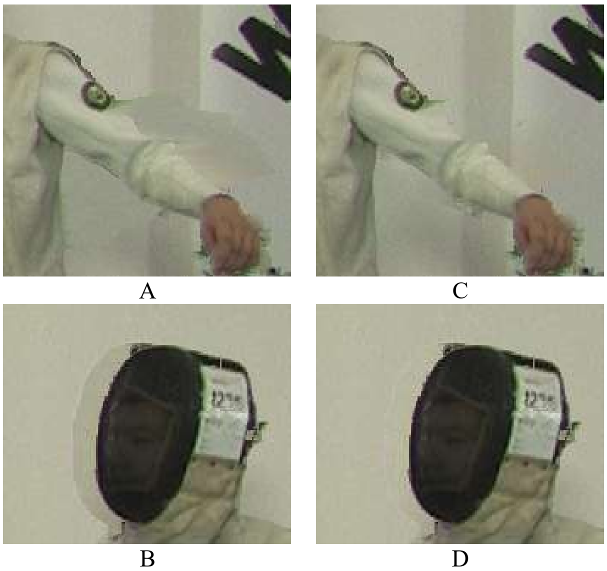

Figure 9. Fragments of the synthesized view before (A, B) and after (C, D) color correction; sequence *Poznan_Fencing2*

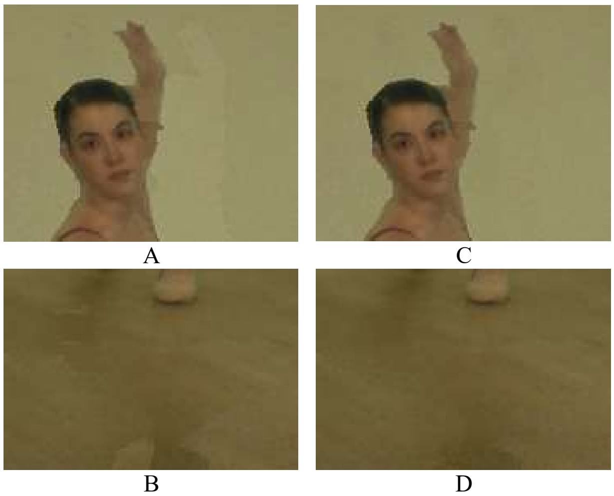

Figure 10. Fragments of the synthesized view before (A, B) and after (C, D) color correction; sequence *Ballet*

5. CONCLUSIONS

An efficient method of color correction was presented in the paper. Proposed technique adjusts local color characteristics of objects captured by two real cameras in order to eliminate color artifacts in synthesized virtual view.

Proposed method allows to obtain objectively and subjectively better virtual views for sequences captured by sparse multicamera systems.

6. ACKNOWLEDGEMENT

The presented work has been funded by the Polish Ministry of Science and Higher Education for the status activity consisting of research and development and associated tasks supporting development of young scientists and doctoral students in 2018 in Chair of Multimedia Telecommunications and Microelectronics.

7. REFERENCES

[1] O. Stankiewicz, et. al., "A free-viewpoint television system for horizontal virtual navigation", *IEEE Tr. on Mult*, 2018.

[2] M. Tanimoto, M. P. Tehrani, T. Fujii, T. Yendo "FTV for 3-D spatial communication", *Proc. IEEE*, vol. 100, no. 4, pp. 905-917, 2012.

[3] G. Lafruit, et. al, "New visual coding exploration in MPEG: Super-MultiView and Free Navigation in Free viewpoint TV", *IST Electronic Imaging, Stereoscopic Displays and Applications XXVII*, San Francisco, USA, 2016.

[4] M. Domański, M. Gotfryd, K. Wegner, "View synthesis for multiview video transmission", *The Int. Conf. Image Proc., Computer Vision, and Pattern Recognition*, 2009.

[5] L. Du-Hsiu, H. Hsueh-Ming, L. Yu-Lun, "Virtual view synthesis using backward depth warping algorithm", *Picture Coding Symposium 2013*, San Jose, USA, 8-11.12.2013.

[6] J. Jin, A. Wang, Y. Zhao, C. Lin, B. Zeng, "Region-aware 3D warping for DIBR", *IEEE Transactions on Multimedia*, vol. 18, no. 6, pp. 953-966, 2016.

[7] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, M. Tanimoto, "View generation with 3D warping using depth information for FTV", *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 65-72, 01.2009.

[8] S. Rogmans, J. Lu, P. Bekaert, G. Lafruit, "Real-time stereo-based view synthesis algorithms: a unified framework and evaluation on commodity GPUs", *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 49-64, 01.2009.

[9] M. Tanimoto, T. Fujii, K. Suzuki, "View synthesis Algorithm in View Synthesis Reference Software 2.0 (VSRS2.0)", *ISO/IEC JTC1/SC29/WG11 MPEG2009, M16090*, Lausanne, Switzerland, 26-28.01.2009.

[10] S.J. Koppal, "Lambertian reflectance", in: K. Ikeuchi (ed.) *Encyclopedia of Comp. Vision, Springer*, pp. 441-443, 2014.

[11] W. Xu, J. Mulligan, "Performance evaluation of color correction approaches for automatic multi-view image and video stitching*", IEEE Conference on Comp. Vision and Pattern Recognition, CVPR '10*, San Francisco, USA, 13-18.06.2010.

[12] E. Reinhard, M. Adhikhmin, B. Gooch, P. Shirley, "Color transfer between images", *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 31-41, 09-10.2001.

[13] U. Fecker, M. Barkovsky, A. Kaup, "Histogram-based prefiltering for luminance and chrominance compensation of multiview video", *IEEE Tr. on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1258-1267, 09.2008.

[14] K. Yamamoto, M. Kitahira, H. Kimata, T. Yendo, "Multiview video coding using view interpolation and color correction", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 1436-1449, 2007.

[15] S.A. Fezza, M.C. Larabi, K.M. Faraoun, "Feature-based color correction of multiview video for coding and rendering enhancement", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, pp. 1486-1498, 2014.

[16] S. Lu, B. Ceulemans, A. Munteanu, P. Schelkens, "Spatio-temporally consistent color and structure optimization for multiview video color correction", *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 577-590, 2015.

[17] F. Shao, G. Jiang, M. Yu, "New color correction method of multi-view images for view rendering in free-viewpoint television", *WSEAS Tr. on Computers*, vol. 7, pp. 569-578, 2008.

[18] S. Ye, S.P. Lu, A. Munteanu, "Color correction for large-baseline multiview video, Signal Processing: Image Communication", vol. 53, pp. 40-50, 04.2017.

[19] A. Dziembowski, et. al, "Multiview Synthesis – improved view synthesis for virtual navigation", *Picture Coding Symposium*, 2016.

[20] T. Senoh, et.al, "View Synthesis Reference Software (VSRS) 4.2 with improved inpainting and hole filling", *ISO/IEC JTC1/SC29/WG11 MPEG2013, M40657*, Hobart, Australia, 3-7.04.2017.

[21] P.T. Kovacs, A. Fekete, K. Lackner, V.K. Adhikarla, A. Zare, T. Balogh, "[FTV AHG] Big Buck Bunny light-field test sequences", *ISO/IEC JTC1/SC29/WG11 MPEG2013, M36500*, Warsaw, Poland, 22-26.06.2015

[22] M. Domański, et. al, "Poznan Blocks - a multiview video test sequence and camera parameters for free viewpoint television", *ISO/IEC JTC1/SC29/WG11 MPEG2014, M32243*, San Jose, United States, 13-17.01.2014.

[23] M. Domański, A. Dziembowski, A. Grzelka, A. Łuczak, D. Mieloch, O. Stankiewicz, K. Wegner, "Multiview test video sequences for free navigation exploration obtained using pairs of cameras", *ISO/IEC JTC1/SC29/WG11 MPEG2016, M38247*, Geneva, Switzerland, 30.05-3.06.2016.

[24] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, "High-quality video view interpolation using a layered representation", *ACM Transactions on Graphics*, vol. 3, no. 23, pp. 600-608, 08.2004.

[25] P. Goorts, "Real-time adaptive plane sweeping for free viewpoint navigation in soccer scenes", *doctoral dissertation*, Hasselt University, Belgium, pp. 175-186, 16.06.2014.

[26] M. Domański, et. al, "Poznań multiview video test sequences and camera parameters", *ISO/IEC JTC1/SC29/WG11 MPEG2009, M17050*, Xi'an, China, 26-30.10.2009.

[27] ITU-T, "Subjective video quality assessment methods for multimedia applications", *Rec. ITU-T P.910*, 04.2008.

[28] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", *IEEE Image Proc.*, vol. 13, no. 4, pp. 600-612, 04.2004.