

EDGE AND MOTION-ADAPTIVE MEDIAN FILTERING FOR MULTI-VIEW DEPTH MAP ENHANCEMENT

Erhan Ekmekcioglu¹, Vladan Velisavljević², Stewart T. Worrall¹

¹Centre for Comm. Systems Research, University of Surrey, Guildford, UK

²Deutsche Telekom Laboratories, Technische Universität Berlin, Germany

ABSTRACT

We present a novel multi-view depth map enhancement method deployed as a post-processing of initially estimated depth maps, which are incoherent in the temporal and inter-view dimensions. The proposed method is based on edge and motion-adaptive median filtering and allows for an improved quality of virtual view synthesis. To enforce the spatial, temporal and inter-view coherence in the multi-view depth maps, the median filtering is applied to 4-dimensional windows that consist of the spatially neighbor depth map values taken at different viewpoints and time instants. These windows have locally adaptive shapes in a presence of edges or motion to preserve sharpness and realistic rendering. We show that our enhancement method leads to a reduction of a coding bit-rate required for representation of the depth maps and also to a gain in the quality of synthesized views at an arbitrary virtual viewpoint. At the same time, the method carries a low additional computational complexity.

Index Terms— Depth estimation, Free viewpoint video, Multi-view coding, Multi-view depth map.

1. INTRODUCTION

The multi-view video signals are typically captured by an array of synchronized cameras, which cover the same scene from different viewpoints. Such a system enables applications in free-viewpoint video (FVV), where the user freely chooses a viewpoint in the scene.

A recent research in the area of FVV brought into focus new technological challenges, like recording, encoding and decoding the FVV signals [1]. The key issue is how to deal with a large amount of data required for representation or realistic approximation of a scene within the FVV. The current commercial broadcasting systems are not capable of processing and transmitting the video sequences recorded at all possible views. For that reason, several different coding approaches have been recently proposed to improve the coding efficiency.

In mesh-based FVV coding [2], the underlying 3D geometry of the scene is encoded and sent to the user along with the corresponding photometric properties. This method significantly reduces the amount of data needed for transmission, but it also requires a precise knowledge of the 3D geometry in the scene, which is still difficult to acquire in a general case. By contrast, in the depth map-based FVV coding [3], depth map sequences are joint to color texture sequences in each chosen viewpoint to locally approximate the light-field function (see Fig. 1). Such a representation does not need a precise geometric knowledge of the scene and, thus, the required number of

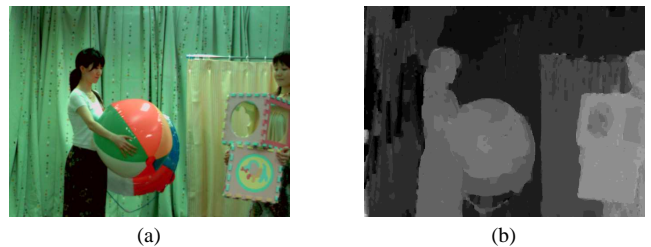


Fig. 1. An example of a depth map estimated from one frame of the sequence Akko. (a) A frame from the original color texture sequence Akko. (b) The corresponding depth map.

views (and, consequently, cameras) is reduced. Moreover, arbitrary views can be generated on demand at the decoder using the image-based rendering techniques [4].

A similar approach has been adopted in [5], where authors propose to use multi-view coding (MVC) tools developed under the standardization group in JVT for the compression of both the multiple viewpoint color texture videos and associated per-pixel depth map sequences. These two types of sequences are encoded separately using the MVC method based on hierarchical B-prediction and H.264/AVC that exploits the intra-frame spatial, inter-frame temporal and inter-view correlation in the multi-view signal [1, 6]. However, the depth map sequences differ from the color texture sequences, having large regions of smoothly changing or even constant gray levels. Thus, applying the same MVC method to both types of sequences might result in a suboptimal coding performance. This shortcoming has already been addressed and the depth maps have been encoded using modified methods based on wedgelets and platelets [7] or on shape-adaptive wavelets [8]. Moreover, the latter method has also exploited a correlation between location of edges in the depth map and color texture sequences.

Our goal is to post-process the estimated multi-view depth maps, which are incoherent in the temporal and inter-view dimensions (e.g., the depth maps obtained by the method in [9]), so that both the resulting coding efficiency and the quality of image-based rendering at decoder are improved. We propose a novel enhancement method based on an edge and motion-adaptive median filtering of the multi-view depth maps across the spatial, temporal and inter-view dimensions. Owing to the enhancement, a simple and quick initial depth estimation can still result in a good rendering at decoder in terms of both numeric and visual quality of the synthesized views. Moreover, our method carries only a low additional computational complexity.

The paper is organized as follows. In Section 2, we review depth map estimation. Then, we explain the principles of the novel enhancement method in Section 3. We present the results in Section 4 and, finally, we conclude in Section 5.

This work was in part developed within VISNET II, a European Network of Excellence (<http://www.visnetoe.org>), funded under the European Commission IST FP6 programme.

2. REVIEW OF DEPTH MAP ESTIMATION

Depth map sequences can be either estimated from the multi-view color texture sequences or directly measured in the scene by range scanners or specialized cameras. Here, we revisit the first approach that attracted lot of research attention and resulted in a number of methods with different accuracy and complexity.

In [10], the depth maps are estimated by a multi-stage segmentation method. An initial color-based segmentation is followed by a coarse disparity assignment obtained by exploiting the inter-view correlation and, finally, the resulting disparity space is smoothed to improve the quality of view synthesis. In [11], the coarse depth maps obtained by segmentation are refined using an iterative process based on belief propagation. Furthermore, in [12], depth maps are estimated by a stereo matching method and, then, regularized in a filtering step. However, even though these methods are efficient, they are computationally complex and require strong processors for a real-time implementation.

Another depth estimation method proposed in [9] (and also used as a reference approach in the FVV standardization [13]) is based on stereo matching. The method uses a graph cuts algorithm for energy minimization during a bi-directional disparity search¹ and applies a regularization to a spatial neighborhood of pixels. The main advantage of this method is a low resulting computational complexity as compared to the previous depth map estimation methods. However, the generated depth maps can be inconsistent across views and time, since the inter-view and temporal coherence is not fully exploited. A lack of inter-view coherence is caused by the estimation using the color texture frames only at two neighbor viewpoints. Similarly, a lack of the temporal coherence is caused by the fact that each depth map frame is generated independently across time, using only the color texture frames at the same time instant. As a consequence, the depth maps can be locally erroneous with spot noise in some regions. This affects both the coding performance (by reducing the prediction efficiency in the coder) and the quality of view synthesis (by producing visually annoying distorted object boundaries or jitters in time). An improvement of the temporal coherence was proposed in [14], where the authors added a depth-index dependant temporal regularization factor in the energy function definition. However, this approach is not adaptive to the local motion activity in the scene and, thus, the temporal consistency is not improved in case of pixels that are occluded at some time instants.

Our depth map enhancement method exploits the algorithm in [9] and provides a solution to the shortcomings by post-processing the depth maps using an edge and motion-adaptive multi-dimensional median filtering. The novel method results in an improved coding performance and the quality of view synthesis, as explained in the sequel.

3. DEPTH MAP ENHANCEMENT

The proposed enhancement method of the initially estimated multi-view depth maps consists of three stages: 1) warping all viewpoints to the central viewpoint, 2) applications of the edge and motion-adaptive median filtering to the warped viewpoints to produce consistent depth maps and 3) inverse view warping of the filtered depth map to all initial viewpoints. We explain each stage in detail.

3.1. View warping

The multi-view per-pixel depth map sequences $d(x, y, t, n)$ are estimated using the algorithm from [9] at each viewpoint $n = 1, \dots, N$,

¹At the left and right directions across the neighbor viewpoints.

time instant t and spatial location (x, y) . Since the goal of our method is to enforce the spatial, temporal and inter-view coherence in the depth map sequences, they have to be warped to the same viewpoint to ensure spatial alignment.

First, assuming $0 \leq d(x, y, t, n) \leq 255$, the depth maps are transformed to the real-world depths $D(x, y, t, n)$ by

$$D(x, y, t, n) = \left(\frac{d(x, y, t, n)}{255} \cdot (z_{near}^{-1} - z_{far}^{-1}) + z_{far}^{-1} \right)^{-1}, \quad (1)$$

where z_{near} and z_{far} are the smallest and the largest depths in the scene, respectively.

Then, these depths are used to obtain the corresponding three-dimensional coordinates as

$$(u, v, w) = R(n) \cdot A^{-1}(n) \cdot (x, y, 1) \cdot D(x, y, t, n) + T(n),$$

where the functions $A(n)$, $R(n)$ and $T(n)$ represent the intrinsic camera parameters, rotation and translation, respectively, at the n th viewpoint. The three-dimensional coordinates are further warped to the same n_0 th viewpoint using

$$(u', v', w') = A(n_0) \cdot R^{-1}(n_0) \cdot \{(u, v, w) - T(n_0)\}$$

to ensure spatial alignment of the depth maps estimated at different viewpoints. Finally, the warped coordinates (x', y') are expressed in a homogenous two-dimensional form as $x' = u'/w'$ and $y' = v'/w'$. The corresponding warped depth map is denoted as

$$d'_{n_0}(x, y, t, n) = d(x', y', t, n). \quad (2)$$

3.2. Adaptive median filtering

The depth maps $d(x, y, t, n)$ are first warped using (2) to the central viewpoint $n_0 = \lceil N/2 \rceil$. Then, the warped depth map values $d'_{n_0}(x, y, t, n)$ are transformed to real-world depths from the viewpoint n_0 based on the transform in (1). These resulting depth values are denoted as $D'_{n_0}(x, y, t, n)$.

To exploit spatial, temporal and inter-view coherence, median filtering is applied to $D'_{n_0}(x, y, t, n)$ within a 4-dimensional window \mathcal{S} . The shape of the window \mathcal{S} is adaptive to edges and motion in the multi-view sequences to prevent a visually annoying distortion in a synthesized view.

This adaptation is based on two parameters that measure a presence of edges and motion, respectively: 1) local variance $V(x, y, t)$ of the depth values and 2) local mean $m(x, y, t)$ of the absolute difference between the luminance components of the two consecutive color texture frames. Both parameters are computed in a 5×5 spatial neighborhood $\mathcal{N}_{t, n_0}(x, y)$ taken at the time instant t and viewpoint n_0 and centered around the spatial location (x, y) , that is, $\mathcal{N}_{t, n_0}(x, y) = \{(i, j, t, n_0)\}$, for $|i - x| \leq 2$ and $|j - y| \leq 2$. Hence, the first parameter is computed from the transformed depth values $D'_{n_0}(x, y, t, n_0)$ as

$$V(x, y, t) = \text{var}[D'_{n_0}(\mathcal{N}_{t, n_0}(x, y))].$$

The second parameter is computed from the luminance components $c(x, y, t, n_0)$ of the color texture sequences at two consecutive time instants $t - 1$ and t taken at the n_0 th viewpoint, that is,

$$m(x, y, t) = \text{mean}[|c(\mathcal{N}_{t, n_0}(x, y)) - c(\mathcal{N}_{t-1, n_0}(x, y))|].$$

The two parameters are compared to thresholds T_v and T_m , respectively. If no edge is detected in the neighborhood (that is, if $V(x, y, t) \leq T_v$), then the spatial coherence is exploited and the

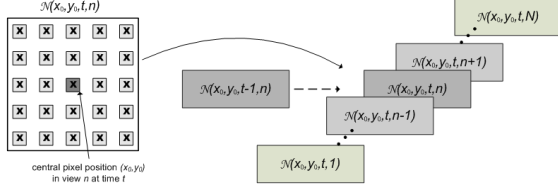


Fig. 2. If no edge or motion is detected in the multi-view video sequence, median filtering is applied locally to a window \mathcal{S} composed of a 5×5 spatial neighborhood, the current t and previous $t - 1$ time instants and all views $n = 1, \dots, N$. In this way, the spatial, temporal and inter-view coherence is exploited. However, if an edge and/or motion are detected, the window does not include the spatial and/or temporal neighborhood, respectively, to prevent distortions in a synthesized view.

window includes the 5×5 neighborhoods across all views, that is, $\mathcal{S}_v = \cup_{n=1}^N \mathcal{N}_{t,n}(x, y)$. Otherwise ($V(x, y, t) > T_v$), only the current location (x, y) is considered and $\mathcal{S}_v = \cup_{n=1}^N (x, y, t, n)$.

Furthermore, if no motion is detected ($m(x, y, t) \leq T_m$), then the temporal coherence is enforced by including the locations at the previous time instant $(t - 1)$ in the window and, thus, $\mathcal{S}(x, y, t) = \mathcal{S}_v \cup \mathcal{N}_{t-1, n_0}(x, y)$. Otherwise, $\mathcal{S}(x, y, t) = \mathcal{S}_v$. An example of the window for $V \leq T_v$ and $m \leq T_m$ is shown in Fig. 2.

Finally, the resulting depth values are obtained by median filtering applied to the adaptive window \mathcal{S} as

$$D'_{n_0}(x, y, t, n) = \text{median}[D'_{n_0}(\mathcal{S}(x, y, t))], \quad (3)$$

for all $n = 1, \dots, N$. Notice that the inter-view coherence is exploited by using the same resulting real-world depth value $D'_{n_0}(x, y, t, n)$ for all views $n = 1, \dots, N$.

3.3. Inverse view warping

The resulting depth map sequence $D'_{n_0}(x, y, t, n)$ is first transformed back to $d'_{n_0}(x, y, t, n)$ using the inversion of (1) and, then, inverse warped to the original viewpoints. This inversion is implemented following the opposite order of steps from Section 3.1.

3.4. Occluded pixels

Notice that the depth values for the pixels visible in one of the viewpoints, but occluded in the central viewpoint $\lceil N/2 \rceil$, are not computed during the median filtering. To improve also the consistency of these values, the three stages explained in Sections 3.1-3.3 are iterated for these pixels with the target warping viewpoint $n_0 = \lceil N/2 \rceil \pm 1, \lceil N/2 \rceil \pm 2, \dots$ instead of $\lceil N/2 \rceil$. This iteration continues until either all depth values are processed or all possible n_0 are exhausted. Notice also that this additional procedure does not carry a significant overhead complexity because of a small number of occluded pixels. Moreover, these pixels are likely located in a neighborhood of edges and, thus, require a smaller size of the window in the spatial and/or temporal dimension.

Fig. 3 shows an example of depth maps processed by median filtering. The four frames are sampled at two successive viewpoints and time instants and processed by the proposed algorithm. The spatial, temporal and inter-view coherence is apparently improved after filtering.

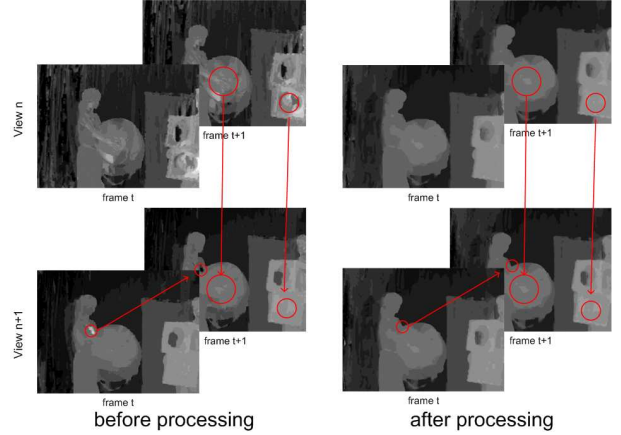


Fig. 3. An example of the effect of median filtering on the depth maps. Four frames are shown for two successive viewpoints and time instants. Notice the improvement of the spatial, temporal and inter-view consistency in the processed depth maps.

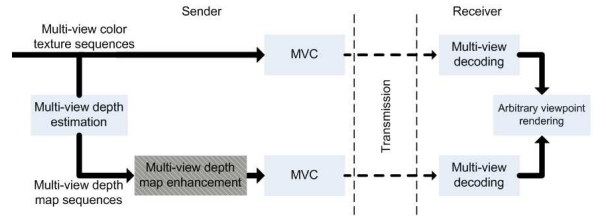


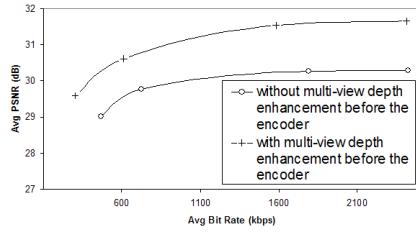
Fig. 4. A common FVV scenario: the input multi-view color texture sequences are used for depth map estimation and, then, both signals are encoded using MVC and sent to the receiver. Our enhancement method is implemented as a post-processing of the estimated depth-maps.

4. RESULTS

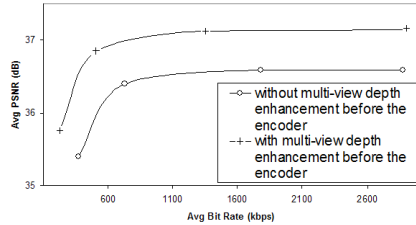
A common FVV scenario comprises a depth map estimation from the input signal and two MVCs applied to both multi-view depth maps and color texture sequences, as shown in Fig. 4. Our enhancement method is integrated in this scenario as a post-processing of the estimated multi-view depth maps implemented at the encoder side. Notice that this step does not reduce the depth resolution, since it is implemented in the real-world depth domain and quantized back to the same set of quantization values. After the post-processing, both sequences are encoded using the Joint Multi-view Video Model (JMVM), version 6.0, developed within the Joint Video Team. A hierarchical B-prediction across time and view is used and the temporal Group of Pictures (GOP) size is set to 8. The MVC blocks are tuned in such a way that the encoding of depth maps spends between 20% and 40% of the total bit-rate sent to the receiver.

The experiments are made with two test multi-view videos (also used in [9]): 1) Akko with 5 consecutive viewpoints (#26 - #30) and 2) Rena with 7 consecutive viewpoints (#41 - #47). The cameras are arranged on a line and the sequences are rectified. The encoding is performed at 4 different quantization points for both the multi-view color texture and multi-view depth map sequences.

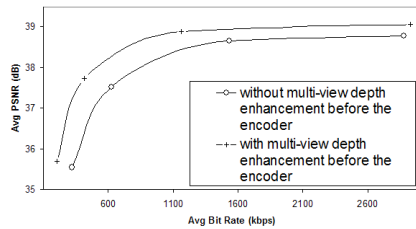
To compare the quality of the view synthesis using the original and enhanced depth maps, we estimate the color texture sequence at the view #28 in case of Akko and at the views #43 and #45 in case of Rena using the neighbor color texture and depth map sequences. The thresholds are chosen as $T_v = 5000$ and $T_m = 50$. The quality of the synthesized sequences is measured in terms of PSNR with the original sequences at the same viewpoints as reference signals. The



(a) The view #28 of Akko



(b) The view #43 of Rena



(c) The view #45 of Rena

Fig. 5. The color texture sequences at the viewpoint #28 of Akko and the viewpoints #43 and #45 of Rena are synthesized using the neighbor color texture and depth map sequences. The quality of the synthesis with and without the depth map enhancement applied before multi-view depth map encoding is compared in terms of PSNR for a wide interval of the total bit-rates. The novel method outperforms the previous method without the enhancement.

PSNR values are plotted in Fig. 5 for these 3 views and for a wide interval of total coding bit-rates. Notice that the novel enhancement method allows for both an improved quality of view synthesis at the same bit-rate and a reduced bit-rate for the same quality. The gain is significant achieving more than 1dB in case of Akko and around 0.5dB in case of Rena. The visual quality of the synthesized views is also improved, as shown in Fig. 6. The annoying artifacts that are especially noticeable around edges and occluded pixels are suppressed and the synthesized sequences show a higher consistency across space and time.

5. CONCLUSIONS

We propose a novel enhancement method for the multi-view depth maps based on a locally edge and motion-adaptive median filtering. The method allows for an improvement of the coding performance in a common FVV scenario and provides a better quality of synthesized color texture sequences at arbitrary virtual viewpoints. Furthermore, the additional computational complexity added by the method is retained low.

6. REFERENCES

[1] P. Merkle, A. Smolic, K. Mueller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

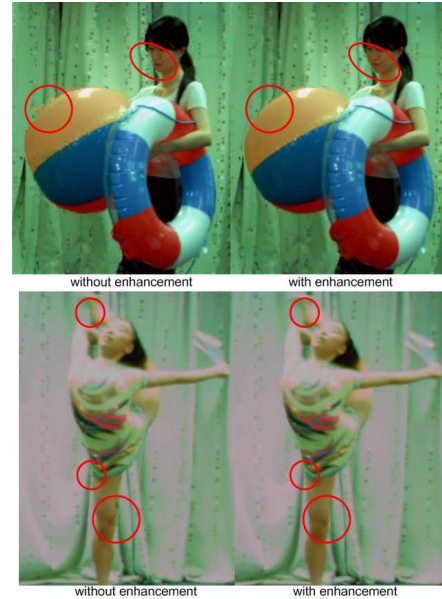


Fig. 6. An example of synthesized frames of Akko and Rena without and with the enhancement of depth maps. The artifacts that appear in the synthesized frames without the enhancement are significantly suppressed.

- [2] K. Mueller, A. Smolic, M. Kautzner, P. Eisert, and T. Wiegand, "Rate-distortion-optimized predictive compression of dynamic 3-D mesh sequences," *Signal Proc.: Image Comm.*, vol. 21, no. 9, pp. 812–828, 2007.
- [3] E. Martinian, A. Behrens, J. Xin, A. Vetro, and H. Sun, "Extensions of H.264/AVC for multiview video compression," in *IEEE Int. Conf. on Image Proc.*, Atlanta, GA, 2006.
- [4] H.-Y. Shum, S.-C. Chan, and S. B. Kang, *Image-Based Rendering*, Springer-Verlag, 2007.
- [5] P. Merkle, A. Smolic, K. Mueller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *IEEE Int. Conf. on Image Proc.*, San Antonio, TX, 2007.
- [6] K. Mueller, P. Merkle, H. Schwarz, T. Hinz, A. Smolic, and T. Wiegand, "Multi-view video coding based on H.264/AVC using hierarchical B-frames," in *Picture Coding Symposium*, Beijing, China, Apr. 2006.
- [7] Y. Morvan, D. Farin, and P. H. N. de With, "Depth-image compression based on an R-D optimized quadtree decomposition for the transmission of multiview images," in *IEEE Int. Conf. on Image Proc.*, San Antonio, TX, 2007.
- [8] M. Maitre and M. N. Do, "Joint encoding of the depth image based representation using shape-adaptive wavelets," in *IEEE Int. Conf. on Image Proc.*, San Diego, CA, 2008.
- [9] M. Tanimoto, T. Fujii, and K. Suzuki, "Multi-view depth map of Rena and Akko & Kayo," *MPEG Doc. M14888*, Oct. 2007.
- [10] L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *ACM SIGGRAPH*, New York, NY, 2004.
- [11] S. Lee, K. Oh, and Y. Ho, "Segment-based multi-view depth map estimation using belief propagation from dense multi-view video," in *IEEE 3D-TV Conf.*, Istanbul, Turkey, 2008.
- [12] C. Cigla, X. Zabulis, and A. Alatan, "Region-based dense depth extraction from multi-view video," in *IEEE Int. Conf. on Image Proc.*, San Antonio, TX, 2007.
- [13] ISO/IEC JTC1/SC29/WG11, "Description of exploration experiments in 3D video coding," *MPEG Doc. N9991*, July 2008.
- [14] S. Lee and Y. Ho, "Enhancement of temporal consistency for multi-view depth map estimation," *MPEG Doc. M15594*, July 2008.