MULTIVIEW IMAGE COMPRESSION USING A LAYER-BASED REPRESENTATION

Andriy Gelman[1]*, Pier Luigi Dragotti*[1]*, Vladan Velisavljević*[2]

[1]Communications and Signal Processing Group, Imperial College, London, UK
[2]Deutsche Telekom Laboratories, Technische Universität Berlin, Germany

ABSTRACT

We propose a novel compression method for multiview images. The algorithm exploits the layer-based representation, which partitions the data set into planar layers characterized by a constant depth value. For efficient compression, the partitioned data is de-correlated using the separable three-dimensional wavelet transform across the viewpoint and spatial dimensions. The transform is modified to efficiently deal with occlusions and disparity variations for different depths. The generated transform coefficients are entropy coded. Our coding method is capable of outperforming the state-of-the-art algorithms, like H.264/AVC, for different data sets.

Index Terms— Image coding, Wavelet transforms

1. INTRODUCTION

In recent years, multiview image and video processing has become an active research area with significant applications in the gaming industry, medical imaging, three-dimensional and free viewpoint TV. The common goal in these fields is to provide an interactive simulation of the real world with photorealistic rendering of the visual information.

The typical multiview setup assumes a set of synchronized or unsynchronized cameras that capture the same scene from different viewpoints. The main challenge is how to process a huge amount of the acquired data and, at the same time, to achieve artifact-free rendering. Several solutions have already been proposed, where different approximations have been applied to reduce the data complexity and size. The popular approach is to 'simulate' the environment using complex geometric, illumination and motion models [1]. Although these methods have been successful for computer generated scenes, they have faced difficulties to efficiently represent complicated natural environments.

In contrast to geometric modeling, Image Based Rendering (IBR) [2] has been proposed to avoid the complex modeling process and to render novel viewpoints directly from the acquired data, while preserving photorealistic rendering even in case of complex scenes. However, IBR requires a high sampling density across the viewpoints, which leads to a large number of captured images or video sequences. Therefore, to store or transmit this data, an efficient compression algorithm is essential.

Many compression algorithms with variations in complexity, efficiency, scalability and random access have been proposed. These properties are in general influenced by the type of 3D representation used during novel view synthesis. For instance, in light field compression [3], a common solution is to remove inter-frame correlation and encode the residuals, which is similar to block-based video coders. An alternative approach is to estimate the scene geometry and utilize it either for warping the images onto aligned view dependent texture maps [4] or to estimate dense disparity vectors [5]. Furthermore, in [5], the authors use a lifting implementation of the inter-view wavelet transform to maintain invertibility and inherently provide a framework to construct scalable bit-streams.

In this paper, we propose a novel compression algorithm for an array of multiview images. The method exploits the segmentation of regularly sampled static scenes into a set of coherent layers at different depths [6]. The set of layers is compressed using a 3D Discrete Wavelet Transform (DWT) followed by entropy coding. The proposed algorithm supports both bit-rate and resolution scalability. To evaluate the performance of our codec, we compare it to H.264/AVC and we show that our novel compression scheme outperforms the state-of-the-art codec when encoding natural scenes.

This paper is organized as follows. We discuss the structure and redundancy of multiview images and review the layer-based representation in Section 2. Then, in Section 3, we present the novel compression algorithm that exploits the layer-based representation. We show the experimental results and analyze the performance of our codec in Section 4. Finally, we conclude in Section 5.

2. MULTIVIEW IMAGE REPRESENTATION

In this section, we analyze the redundancy of the data in the Epipolar Plane Image (EPI) representation of multiview images [7]. Then, we briefly review the layer-based representation proposed in [6].

2.1. Multiview image data structure and redundancy

Although a huge amount of multiview data is required to achieve artifact-free rendering, this data set is highly correlated and redundant. Within each image, neighboring light rays are likely to originate from the same object and, therefore, they contribute to the intra-frame correlation. In addition, due to the parallax, an object appears at different pixel locations x and x' seen from different viewpoint coordinates (frames) V_x and V_x', thus contributing to the intra-frame correlation (see Fig. 1). Assuming the scene is Lambertian and has no occlusions, this shift in pixel locations (*disparity*) $\Delta x = |x - x'|$ can be represented as a function of the corresponding viewpoint coordinates, depth Z of the object and focal length f, that is,

$$\Delta x = \frac{f\,(V_x' - V_x)}{Z}. \tag{1}$$

The obtained relation between the viewpoint and spatial coordinates is commonly illustrated as a set of EPI lines. An example of the EPI lines is shown in Fig. 2(a), where the pixels in the 3D space are projected onto lines with slopes proportional to the depth. Notice that such a set of EPI lines is highly correlated.

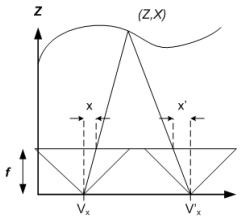
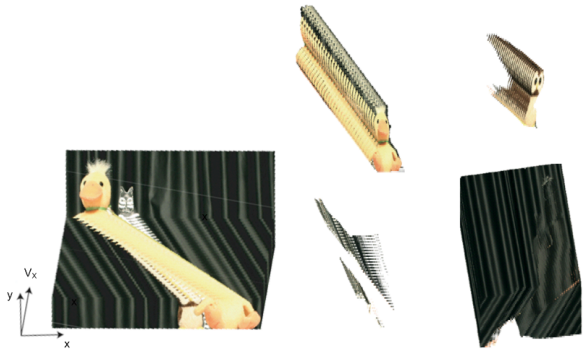


Fig. 1. Horizontal parallax model used to estimate the pixel disparity.



(a) Multiview image cross-section. The data can be analyzed as a set of EPI lines with varying gradients.

(b) Extracted layers

Fig. 2. Multiview image data set cross-section and the extracted layers.

2.2. Layer-Based Representation

To reduce the data complexity, we separate the 3D scene into layers, where each layer is modeled by a constant depth plane. The goal of the representation is to partition the data into coherent layers, which have a smaller depth variation than the original scene.

Extraction of layers from a general 3D scene is a non-trivial task. Here, we use a variation of the level-set segmentation algorithm which was proposed in [6]. An advantage of this unsupervised algorithm is that it can be extended to an arbitrary number of dimensions. Furthermore, using a semi-parametric methodology, the algorithm efficiently handles occlusions, which is an important property for the subsequent compression algorithm.

Fig. 2(b) illustrates the extracted layers from the data set in Fig. 2(a). It can be observed that each layer preserves the linear structure corresponding to an object location in a 3D space.

3. LAYER-BASED COMPRESSION

Our novel compression algorithm consists of several steps. First, the calibrated images captured at an array of viewpoints are partitioned into correlated layers using a level-set algorithm, as explained in Section 2.2. To ensure the spatial consistency of the extracted layers, the occluded pixels are extrapolated along the Epipolar Plane Image (EPI) lines in a pre-processing step. Then, each layer is separately de-correlated across the viewpoint dimension using a 1D disparity-compensated DWT. The low-pass transform coefficients that originate from different layers are grouped together in a merging step to exploit the spatial correlation more efficiently. Finally, the data is further de-correlated across the spatial dimensions using the 2D shape-adaptive (SA) DWT. The resulting transform coefficients are entropy coded using a modified implementation of EBCOT [9], where the bit allocation is obtained by a greedy optimization process. Each of these algorithm steps is described next in more detail.

3.1. Disparity-compensated 1D DWT

The extracted layers using the segmentation algorithm from [6] might contain occluded regions, as illustrated in Fig. 2(b). Such spatially inconsistent EPI lines would severely affect the compression performance generating a number of large magnitude high-pass coefficients after the transform across the viewpoint dimension. For that reason, the extracted layers are first pre-processed so that the missing pixels due to the occlusion are extrapolated along the EPI lines. The extrapolation is implemented using an average value of all the non-zero pixels along each EPI line. Notice that such a procedure increases the total number of pixels, thus, resulting in an overcomplete representation. However, to achieve a correct reconstruction, only the non-occluded EPI lines with the smallest corresponding depth (the lowest slope of the EPI lines) are used, which is realistic in case the layers are not transparent. Fig. 3 illustrates an example of an originally extracted layer and its extrapolated counterpart. Notice that the extrapolated layers are spatially consistent across the viewpoint dimension.

To apply the DWT across the viewpoint dimension, we design new adaptive basis functions using disparity compensated lifting. The lifting scheme [8] has been chosen for its reduced complexity and easy invertibility and it allows disparity compensation to be efficiently incorporated into the transform steps. To implement a disparity compensated Haar transform, we modify the standard equations by including a warping operator $\mathcal{W}$:

$$\mathscr{L}_e[n] = \frac{P_e[n] - \mathcal{W}\{P_o[n]\}}{2} \tag{2}$$

$$\mathscr{L}_o[n] = P_o[n] + \mathcal{W}\{\mathscr{L}_e[n]\}, \tag{3}$$

where, $P_o[n]$ and $P_e[n]$ represent 2D images with spatial coordinates (x, y) located at odd $(2n+1)$ and even $(2n)$ camera locations, respectively. Following the implementation, $\mathscr{L}_e[n]$ and $\mathscr{L}_o[n]$ contain the 2D high and low-pass subbands, respectively. A multiresolution decomposition is obtained by iteration of the transform on the low-pass subband component $\mathscr{L}_o[n]$.

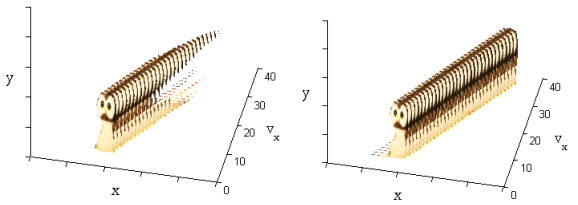
In both (2) and (3), the warping operator $\mathcal{W}$ is chosen so that the correlation within the layers across the viewpoint dimension is maximally exploited. This is achieved by using a projection that maps an image onto the same viewpoint as its odd/even complement in the lifting step. Using (1) and the fact that the layers are modeled as planes with constant depths, the warping from viewpoint n_1 to n_2 is defined as:

$$\mathcal{W}_{n_1 \to n_2}\{P[n_1]\}(x, y) = P[n_1](x - \Delta x(n_2 - n_1), y), \tag{4}$$

where Δx is the disparity between the consecutive images within a layer.

3.2. Shape-adaptive 2D DWT

To improve the de-correlation efficiency of the spatial transform, the low-pass subbands from each layer are grouped together into a single image and they are further jointly processed. Notice that, due to the pre-processing extrapolation of the EPI lines (as explained in Section 3.1), the low-pass subbands contain more pixels than in the original data set. However, such an overcomplete representation does not



(a) Original owl layer (b) Owl layer with pre-processing

Fig. 3. Extrapolation of the extracted layers. (a) Extracted layers might have discontinuities in the EPI lines due to the occlusion, which is not efficiently captured by the DWT across the viewpoint dimension. (b) The values in the EPI lines are extrapolated using the mean of the non-occluded pixels.

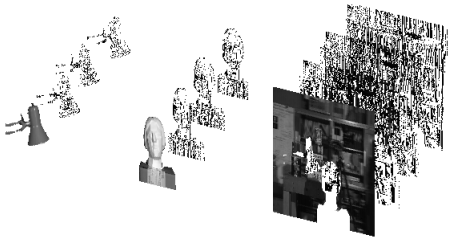
affect the compression performance because of a high correlation between the added pixels and their neighbors. Moreover, since the segmentation data is known from the layer extraction process, no additional overhead information about the shape and position of the extrapolated pixels is required to be transmitted. A comparison between the original and recombined layers is illustrated in Fig. 4.

The extracted layers are commonly bounded by an irregular (non-rectangular) shape. For that reason, the standard 2D DWT applied to the entire spatial domain is inefficient because of a boundary effect, that is, many large high-pass coefficients are generated by filtering across this artificial boundary. To improve the coding efficiency, the SA-DWT [9] is used to encode the texture of the layers within arbitrary shaped objects. The boundary of the grouped low-pass layer components includes all the subbands, whereas, the high-pass components are processed separately. First, the contour of the layers is losslessly encoded using a modified version of the Freeman code [10]. Then, the DWT is applied within the layer bounds so that the texture image is symmetrically extended whenever the wavelet is crossing the contour. The DWT is built as a separable transform with linear-phase symmetric wavelet filters (9/7 or 5/3), which, together with the symmetric signal extensions, leads to critically sampled transform subbands. Notice that the complete segmentation of a layer is fully defined by encoding the contour of an object in one frame and warping it to the other viewpoints. Finally, the transform coefficients obtained by the SA-DWT are partitioned into blocks and arithmetically coded using a variation of EBCOT [11].

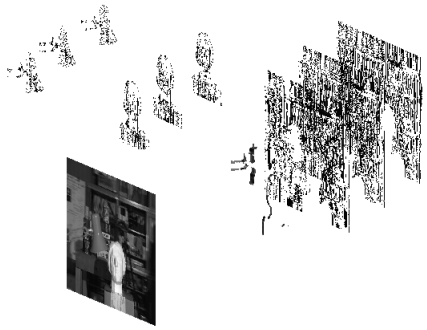
3.3. Layer merging based on rate-distortion performance

Since the original layer segmentation method implemented as in [6] is not optimized in the rate-distortion sense, it can produce layers with small size that are expensive to encode (that is, they require too many bits per pixel). To eliminate such layers and to improve the rate-distortion performance, we propose a greedy algorithm to merge the neighbor layers whenever the corresponding rate-distortion performance can be improved.

Given the resulting layer segmentation from Section 3.1, the algorithm searches for two layers with the minimal distance in depths. Denote such layers as l_1 and l_2 and the corresponding rates and distortions obtained by a separate encoding of these layers as R_1, R_2, D_1 and D_2, respectively. Furthermore, denote as $l_{1,2}$ the layer obtained by merging l_1 and l_2. The resulting rate and distortion associated to encoding the merged layer $l_{1,2}$ are $R_{1,2}$ and $D_{1,2}$. The algorithm chooses to encode these layers either separately or jointly (by merging), so that the chosen solution has a smaller associated



(a) Tsukuba Layers after applying the DWT



(b) Tsukuba Layers after recombining the lowpass component of the DWT

Fig. 4. A comparison of the separated and recombined low-pass components of the layers from the test data set Tsukuba. Notice that the recombined components resemble a downsampled version of the original signal.

Lagrangian cost $D + \lambda R$. Thus, if

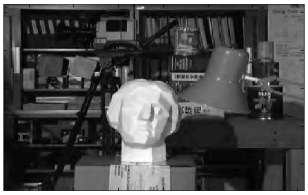
$$D_{1,2} + \lambda R_{1,2} < (D_1 + D_2) + \lambda(R_1 + R_2), \qquad (5)$$

then the layers are merged and encoded jointly. Otherwise, they are retained and encoded separately. Notice that the Lagrangian multiplier λ determines the weight of the bit-rate in the compression performance and it is preselected. This process continues in an iterative approach until either the Lagrangian cost increases or all of the layers are merged into one.

4. SIMULATION RESULTS AND ALGORITHM ANALYSIS

To evaluate the performance of our proposed compression algorithms, we compare it to the state-of-the-art H.264/AVC (Main Profile, level 2.1) [12]. We use three different input data sequences, called 'Animal Farm' $(235 \times 625 \times 16)$ [6], 'Tsukuba' $(284 \times 382 \times 4)$ and 'Teddy' $(375 \times 411 \times 4)$, both from [13]. The data sets vary in scene and texture complexity. Teddy has a wide range of disparities, whereas Tsukuba and Animal Farm can be well approximated using a small number of depth planes. A comparison of our codec against H.264/AVC is illustrated in Fig. 5 and 6. We note that our proposed algorithm outperforms H.264/AVC when encoding the Animal farm data at all bit-rates. Here, the layer based representation efficiently captures the 3D scene and the segmentation is accurate. Regarding the Tsukuba data set, our algorithm shows gains at low-rates up to 1.28dB. At higher bit-rates the performance of both codecs is very similar. When encoding the Teddy data set, the performance of our algorithm is comparable to H.264/AVC. In this case, the layer-based representation does not capture the complexity of the scene. In the future, we aim to improve these

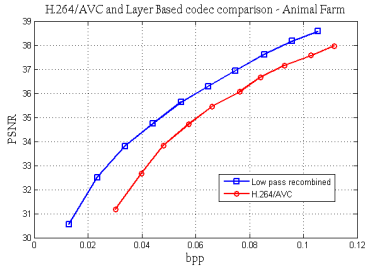
(a) Animal Farm - PSNR = 33.54dB at 0.032bpp



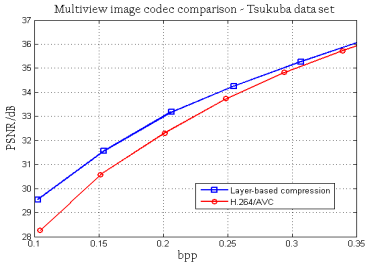
(b) Tsukuba - PSNR = 33.17dB at 0.207bpp

(c) Teddy PSNR = 33.2dB at 0.235bpp

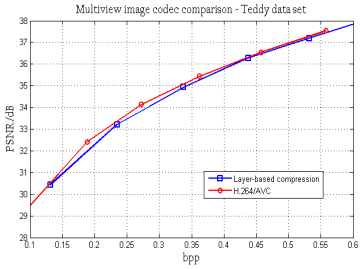
Fig. 5. Qualitative coder analysis.



(a) Animal Farm



(b) Tsukuba



(c) Teddy

Fig. 6. Quantitative coder comparison.

results by modifying the objective function in the layer extraction algorithm.

5. CONCLUSION

We presented a novel multiview still image compression algorithm. The algorithm is based on the layer-based representation, which partitions the data into correlated layers modeled by a constant depth plane perpendicular to the camera baseline. To encode the data, we apply a separable 3D-DWT as a combination of the 1D-DWT across the viewpoint dimension and the SA-DWT across the spatial dimensions followed by entropy coding. The algorithm contains two pre-processing stages to improve the compression efficiency. First, the occluded pixels are extrapolated along the EPI lines prior to applying the inter-view transform. Second, low-pass components of the inter-view transform are recombined into a new subband frame to efficiently exploit the inter-layer correlation. Moreover, the extracted layers can be merged and processed jointly whenever such a procedure improves the rate-distortion performance. The experimental results demonstrate that our algorithm is competitive or even outperforms H.264/AVC. The future work includes extending the algorithm to operate on a light field and an additional time dimension. In addition, we aim to improve the layer-extraction algorithm to improve compression performance in complicated scenes.

6. REFERENCES

[1] 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.

[2] C. Zhang and T. Chen, "A survey on image-based rendering - representation, sampling and compression," in *Technical Report AMP 03-03*, 2003.

[3] M. Magnor and B. Girod, "Data compression for light-field rendering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 3, pp. 338–343, Apr 2000.

[4] M. Magnor and B. Girod, "Model-based coding of multi-viewpoint imagery," in *Proceedings SPIE Visual Communications and Image Processing*, 2000, pp. 14–22.

[5] B. Girod, C.L. Chang, P. Ramanathan, and X. Zhu, "Light field compression using disparity-compensated lifting," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. IV–760–3 vol.4, April 2003.

[6] J. Berent and P.L. Dragotti, "Plenoptic manifolds: Exploiting structure and coherence in multiview images," *IEEE Signal Processing Magazine*, vol. 24, no. 6, pp. 34–44, November 2007.

[7] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[8] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl*, vol. 4, pp. 247–269, 1998.

[9] S. Li and W. Li, "Shape-adaptive discrete wavelet transforms for arbitrarily shaped visual object coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 5, pp. 725–743, Aug 2000.

[10] Y.K. Liu and B. Zalik, "An efficient chain code with huffman coding," *in Pattern Recognition*, vol. 38, no. 4, pp. 553 – 557, 2005.

[11] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Transactions on Image Processing*, vol. 9, pp. 1158–1170, 2000.

[12] Video Coding Algorithm, "H.264/AVC," http://x264.nl/.

[13] D. Scharstein and R. Szeliski, "Middlebury data sets," vision.middlebury.edu/stereo/.