Layer based multi-view image compression

Andriy Gelman
Imperial College London
andriy.gelman04@ic.ac.uk

Pier Luigi Dragotti
Imperial College London
p.dragotti@ic.ac.uk

Vladan Velisavljević
Deutsche Telekom Labs
vladan.velisavljevic@telekom.de

ABSTRACT

We propose a compression algorithm for an array of multi-view images. First, we apply a segmentation algorithm to partition the data into coherent layers and significantly reduce the number of images required for artifact-free rendering. Then, we exploit the coherence in each layer by applying a 1D disparity compensated wavelet transform across the views followed by a 2D SA-DWT on each of the spatial subbands. Finally, the data is entropy coded using a modified version of EBCOT. Experimental results show that our coder outperforms state-of-the-art H.264/AVC at low bit-rates and intra-image JPEG-2000 over the complete range of bit-rates. Furthermore, unlike other multi-view image compression techniques, our implementation does not rely on estimating a 3D geometric model of the scene.

Keywords

Multi-view image, free-viewpoint rendering, compression, 3D wavelet, lifting.

1. INTRODUCTION

The recent development of broadband communication channels and fast processors have brought into focus multi-view image and video applications as a framework for simulating realistic, immersive and interactive environments. The related applications have already been developed within the computer graphics and gaming industry striving to the same targets. However, emerging technologies such as e-commerce, medical imaging, 3D and free-viewpoint TV, have increased the requirements not only for compression and transmission efficiency but also for the quality of reconstruction and rendering. In particular, scalable compression algorithms are essential in some communication scenarios, such as multimedia communication in mobile networks.

The key issue in these applications is handling the huge amount of data needed to achieve artifact-free rendering. This problem has been addressed in several ways, where various approximations have been applied to reduce the complexity and data size. The traditional approach in rendering, is to 'simulate' the environment using geometric entities, such as 3D models, illumination and texture maps. Although outstanding results have been achieved using specialized hardware for purely computer generated scenes, this technique has difficulties capturing the motion in complicated environments. In addition, to achieve photo-realistic results, the models must often be developed using time-consuming and expensive supervised algorithms.

An alternative approach called Image Based Rendering (IBR) has been widely researched over the last two decades [12]. The input consists of sampled light rays, which can be easily acquired using a standard hand-held camera. Unlike in the traditional time consuming 'simulating' approach, the goal of IBR is to 'estimate' the missing samples, and hence render arbitrary views. The main advantage of IBR is that the algorithm operates with complexity independent of the scene and renders photo-realistic novel views. One approach to rendering is to simply use interpolation. This problem was studied in [4], where the optimal interpolation kernel was derived using Shannon's sampling theory. It was also shown that to achieve artifact-free rendering (without using additional geometric information), the sampling rate associated with the camera spacing must be very high. This leads to an increase in the transmission and storage capacity requirements. For example, 200MB are required to store an uncompressed data set that consists of 32 by 32 images with resolution 256×256 pixels. Thus, the problem of compression of such a multi-dimensional signal is crucial.

Many 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 (densely sampled 2D array of images), a common solution is to remove inter-frame correlation and encode the residuals, which is similar to block-based video coders. A different approach is to estimate a 3D geometry and utilize it either for warping the images onto aligned view dependent texture maps [11] or to estimate dense disparity vectors [6]. Furthermore, in [6], 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 an algorithm for the compression of a set of multi-view images. The method exploits the fact that a regularly sampled static scene can be separately analyzed as a set of coherent layers at different depths. The layered representation significantly reduces the number of

Mobimedia'09, September 7-9, 2009, London, UK.

Digital Object Identifier: 10.4108/ICST.MOBIMEDIA2009.7489
http://dx.doi.org/10.4108/ICST.MOBIMEDIA2009.7489

images required to achieve artifact-free rendering and still maintains a photo-realistic reconstruction. The subsequent sub-sampled layers are individually compressed using a disparity compensated wavelet transform. Our novel algorithm outperforms the compression using JPEG-2000 over all bit-rates and H.264/AVC at low bit-rates. Furthermore, it supports both bit-rate and resolution scalability.

This paper is organized as follows. After discussing the structure of multi-view data and its layer-based representation in Section 2, we present the compression algorithm based on the layered representation in Section 3. Then, we show the obtained experimental results and compare the performance to the other related methods in Section 4. Finally, we conclude in Section 5.

2. MULTI-VIEW IMAGE REPRESENTATION

The representation of multiple view images is very complex and involves the 7D plenoptic function. For that reason, several approximations are commonly made to reduce this complexity. Here, we review the definition and general properties of the plenoptic function and the layer based representation of a 3D scene.

2.1 Plenoptic Function

The notion of the plenoptic function was introduced by Adelson and Bergen in their seminal work on the elements of early vision [2]. The plenoptic function can be empirically derived using an assumption that the space is filled with infinitesimally thin rays of light. A pencil of rays can be parameterized by a 3D point in space (V_x, V_y, V_z), where each ray within these pencils can be defined by its direction of arrival (θ, ϕ). Furthermore, two additional variables, wavelength λ and time t, are required to describe colour and dynamic scenes, respectively. Thus, the general plenoptic function is a 7D signal:

$$I = P_7 \left(V_x, V_y, V_z, \theta, \phi, t, \lambda\right). \tag{1}$$

Dealing with seven dimensions is not an easy task. Clearly, existing hardware is not capable of processing or capturing such a high dimensional signal. Common simplifications include dropping the wavelength λ and time t to analyze monochromatic (or separate RGB channels) and static scenes, respectively. These simplifications lead to a widely known representation called the Light Field [8]. Using additional assumptions that the intensity of light does not change along its path and that the viewer is confined outside a bounding box, the light rays can be parameterized by their intersection with a focal and a camera plane.

To further simplify the problem, we constrain the camera plane to be a line and model the data as a 1D array of images (also known as an Epipolar-plane Image (EPI) data set). By construction, an EPI data set is inherently highly redundant. Within each image, successive light rays likely originate from the same object and therefore contribute to the intra-frame correlation. In addition, due to the parallax, an object appears at different pixel locations within each frame. Using the model illustrated in Figure 1, the disparity associated with a specific depth can be evaluated as:

$$\Delta x = x - x' = \frac{f\left(V_x' - V_x\right)}{Z}, \tag{2}$$

where Z corresponds to the depth of the object, f is the focal length, x is the pixel coordinate and V_x is the camera location.

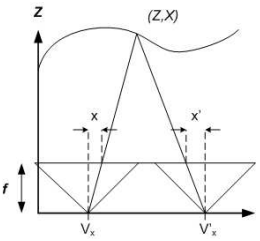


Figure 1: EPI - Horizontal parallax model used to estimate pixel disparity.

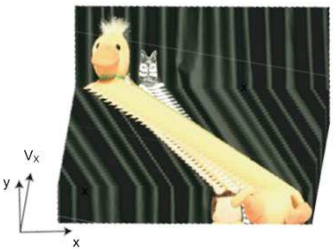


Figure 2: EPI cross-section along the (V_x, x) plane reveals that points in a 3D space are mapped onto lines.

Figure 2 illustrates a cross-section of a 1D array of images. We observe that the EPI is structured and a point in a 3D space maps onto a line in the plenoptic domain, whose gradient is inversely proportional to the depth. Therefore, given two arbitrary lines that intersect in the (V_x, x) plane, the correct occlusion ordering can be inferred using the gradient of both lines.

2.2 Layer Based Representation

To further reduce the complexity of the representation, we separate the 3D scene into layers, which are independently compressed.

The concept of the layered representation is to partition the data into coherent regions having a smaller depth variation. During novel view synthesis, each layer can be interpolated using a basis function, whose support is modified according to the layer's average depth. Therefore, using the representation, we implicitly exploit geometric information to improve the rendering quality and reduce the compression problem to a sparsely sampled data set.

However, extraction of layers from a general 3D scene is a non-trivial task. Here, we use a level-set segmentation algorithm proposed in [3], where each layer is modelled by a constant depth plane perpendicular to the camera baseline. 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.

Figure 3 shows the extracted layers from the data set in Figure 2. It can be observed, that each layer preserves the linear structure corresponding to an object location in a 3D space. In the following section, we compress the data set by exploiting the coherence within each layer.

Digital Object Identifier: 10.4108/ICST.MOBIMEDIA2009.7489
http://dx.doi.org/10.4108/ICST.MOBIMEDIA2009.7489

Figure 3: Extracted layers using a level-set segmentation algorithm proposed in [3]. The algorithm applies joint segmentation across the EPI to impose coherence throughout the data set. Each layer is modelled by a plane perpendicular to the camera baseline.

3. LAYER BASED COMPRESSION

As explained in Section 2, the constraints applied to the general plenoptic function (1) and the layer based representation reduce the problem to the compression of a sparsely sampled 3D data set, where the retained dimensions are the camera location V_x and the spatial coordinates of the pixels (x, y). To de-correlate the data, we apply to each layer a separable 3D discrete wavelet transform (DWT) that consists of the 1D disparity compensated transform across the views and the 2D shape-adaptive (SA) DWT across the spatial dimensions. However, prior to applying the 3D-DWT, we use a pre-processing algorithm to ensure the layers are spatially consistent. Finally, the data is entropy coded using a modified implementation of EBCOT [13]. The complexity of the overall algorithm is $O(N)$, where N is the total number of elements in the data set. Here, we discuss each of the coding steps in more detail.

3.1 Layer pre-processing

As illustrated in Figure 3, the segmented layers might contain occluded regions, which would degrade the compression performance generating a number of large high-pass coefficients in the inter-view transform. To solve this problem, we extrapolate the missing pixels along the EPI lines in each layer. The occluded data is computed using an average of all the non-zero pixels along each EPI line. To preserve realistic reconstruction of the occluded regions, only the pixels in the nearest layer (with the smallest depth) are shown after decoding.

3.2 Disparity compensated DWT

Here, we design new basis functions along the view dimension using disparity compensated lifting. Lifting [5], has been chosen for its reduced complexity and easy invertibility, which allows disparity compensation to be incorporated into the lifting steps.

To implement a disparity compensated Haar transform, we modify the standard equations by including a warping operator $\mathcal{W}$:

$$\mathscr{L}_e[n] = \frac{P_e[n] - \mathcal{W}\{P_o[n]\}}{2} \quad (3)$$

$$\mathscr{L}_o[n] = P_o[n] + \mathcal{W}\{\mathscr{L}_e[n]\}, \quad (4)$$

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 2D high and low-pass subbands, respectively. For simplicity, the camera sampling rate has been normalized to be an integer value. We obtain a multiresolution decomposition by re-applying the transform on the low-pass subband components $\mathscr{L}_o[n]$.

In both, (3) and (4), the warping operator is chosen to maximize the inter-image correlation. This is achieved by using a projective operation that maps one image onto the same viewpoint as its odd/even complement in the lifting step. Using (2) and the fact that the layers are modelled by a constant depth plane, we define the warping operation from viewpoint n_1 to n_2 as:

$$\mathcal{W}_{n_1 \to n_2}\{P[n_1]\}(x, y) = P[n_1](x - \Delta x(n_2 - n_1), y), \quad (5)$$

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

3.3 Shape-Adaptive 2D DWT

The extracted layers commonly contain frames with an object at a particular depth, whereas the rest is considered as a background and set to zero (or another constant). For that reason, the standard 2D DWT applied to the entire spatial domain results in many large high-pass coefficients generated by filtering across the artificial boundary between the object and the background.

To improve the coding efficiency, the SA-DWT [9] is used to encode the texture within arbitrary shaped objects. First, the contour of the object is losslessly encoded using a modified version of the Freeman code [10], then, the DWT is applied to the object domain. To reduce the influence of the object boundary, the signal is symmetrically extended whenever the wavelet filtering 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 extension, leads to critically sampled transform subbands. We note that the complete segmentation of a layer is fully defined by encoding the contour of an object in one frame and projecting it to the other images.

3.4 Quantization Entropy Coding

To encode the transform coefficients we use an implementation of EBCOT [13]. We apply a modification, where only the critically sampled transform coefficients corresponding to each layer and not the zero background are encoded.

The concept of the algorithm is to partition the data into blocks and for each one obtain an operational rate-distortion curve by losslessly encoding the data using a context-adaptive arithmetic coder. Consequently, Lagrangian multipliers are used to allocate an optimum number of bits to each block given a rate constraint. An advantage of this algorithm is that it can be easily modified to support bit-rate and resolution scalability and also a random access feature. However, notice that random access can be achieved only approximately because of the finite length of the spatial wavelet filter.

Digital Object Identifier: 10.4108/ICST.MOBIMEDIA2009.7489
http://dx.doi.org/10.4108/ICST.MOBIMEDIA2009.7489

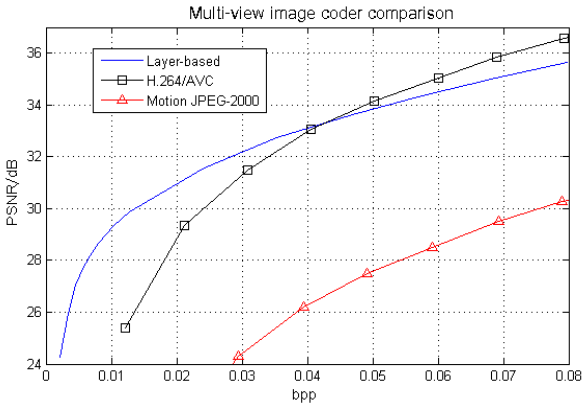


Figure 4: Rate-distortion comparison of the layer based approach with intra-image JPEG-2000 and H.264/AVC. The layer based approach outperforms JPEG-2000 at all bit-rates and H.264/AVC at low bit-rates. Here bpp and PSNR are parameters associated with the sparsely sampled representation.

4. SIMULATION RESULTS

To evaluate the performance of our layer based compression algorithm, we compare it to motion JPEG-2000 [1] and block-based state-of-the-art H.264/AVC (Main Profile, level 2.1) [7]. We use an input data set, called 'Animal Farm' [3], which consists of 32 sparsely sampled images with occlusions, having the resolution of 232×624 pixels [1]. The data is segmented into four coherent layers using [3]. A negligible overhead of 0.00087bpp is required to losslessly encode the segmentation. Figure 4 depicts a quantitative analysis of the layer based compression algorithm. In comparison to the intra-view coding using JPEG-2000, we observe a significant improvement over the complete range of bit-rates, with gains of up to 8dB. Regarding the state-of-the-art H.264/AVC, we observe a gain at low bit-rates, at an adequate quality level of 33dB.

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented a layer based compression of EPI data sets. Layers are extracted using a recently developed layer extraction algorithm [3]. The segmented layers are compressed using disparity compensated lifting across the view-dimension, followed by a spatial 2D SA-DWT transform. Furthermore, we have shown that our algorithm outperforms the intra-view JPEG-2000 across the complete range of bit-rates and H.264/AVC at low bit-rates. In the future, we aim to extend the algorithm to operate on a more general Light Field using an adaptively chosen number of layers.

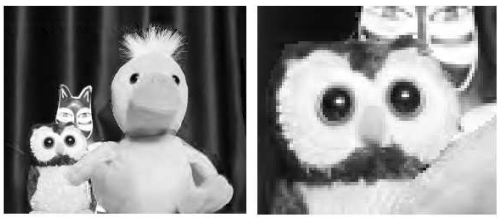
6. ACKNOWLEDGEMENTS

The authors would like to thank Jesse Berent for the layered representation of the Animal Farm data set and for his helpful comments.

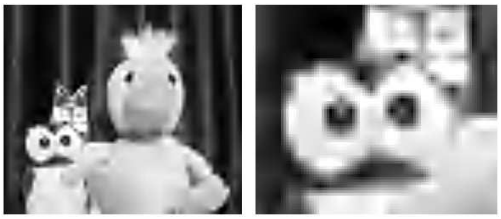
7. REFERENCES

[1] M. Adams. `JasPer-JPEG2000`. `http://www.ece.uvic.ca/ mdadams/jasper/`.

[1]Due to a lack of space, we use only one data set in the experiments.



(a) Decoded image using the layer-based approach at a rate of 0.0567bpp and a PSNR of 34.37dB



(b) Decoded image using JPEG-2000 at a rate of 0.0497bpp and a PSNR of 27.09dB

Figure 5: Qualitative coder comparison.

[2] E. Adelson and J. R. Bergen. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing*, pages 3–20. MIT Press, 1991.

[3] J. Berent and P. Dragotti. Plenoptic manifolds: Exploiting structure and coherence in multiview images. *IEEE Signal Processing Magazine*, 24(6):34–44, November 2007.

[4] J. X. Chai, X. Tong, S. Chan, and H. Shum. Plenoptic sampling. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 307–318, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

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

[6] B. Girod, C. Chang, P. Ramanathan, and X. Zhu. Light field compression using disparity-compensated lifting. *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, 4:IV–760–3 vol.4, April 2003.

[7] H.264/AVC. x264. `http://x264.nl/`.

[8] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996.

[9] S. Li and W. Li. Shape-adaptive discrete wavelet transforms for arbitrarily shaped visual object coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 10(5):725–743, Aug 2000.

[10] Y. Liu and B. Zalik. An efficient chain code with huffman coding. *Pattern Recognition*, 38(4):553 – 557, 2005.

[11] M. Magnor and B. Girod. Model-based coding of multi-viewpoint imagery. In *Proceedings SPIE Visual Communications and Image Processing VCIP-2000*, pages 14–22, 2000.

[12] H. Shum, S. Chan, and S. Kang. *Image-Based Rendering*. Springer-Verlag, 2007.

[13] D. Taubman. High performance scalable image compression with ebcot. *IEEE Transactions on Image Processing*, 9:1158–1170, 2000.

Digital Object Identifier: 10.4108/ICST.MOBIMEDIA2009.7489
http://dx.doi.org/10.4108/ICST.MOBIMEDIA2009.7489