A NEW OBJECT-BASED FRACTAL STEREO CODEC WITH QUADTREE-BASED DISPARITY OR MOTION COMPENSATION

Kamel Belloulata and Shiping Zhu

Département de génie électrique et de génie informatique, Université de Sherbrooke
2500, boulevard Université, Sherbrooke (Québec), Canada, J1K 2R1
Kamel.Belloulata@USherbrooke.ca, Shiping.Zhu@USherbrooke.ca

ABSTRACT

A new object-based fractal stereo video coding scheme with quadtree-based disparity or motion compensation is proposed in this paper. Monocular or stereo video are compressed by using fractal coding and each object is encoded independently by a prior image segmentation alpha plane which is exactly the same as in MPEG-4. The first 2 frames of right video are encoded by using the Circular Prediction Mapping (CPM) and the remaining frames are encoded by using the Non Contractive Interframe Mapping (NCIM). The CPM and NCIM methods accomplish the motion estimation/compensation of right video (reference), and according to different coding or user requirements, the spatial correlations between left and right frames can be explored by partial or full affine transformation quadtree-based disparity estimation/compensation, or simply applying CPM/NCIM on left video. The experimental results indicate that we get high compression ratio and also good PSNR.

1. INTRODUCTION

Stereo video or in general multiview video can provide plentiful information about the 3D scene. It can also be used to generate virtual synthesized intermediate views and supply the 3D feeling of the scene, so it has wide applications in many fields, such as the 3D visual communication, etc. However how to economically and efficiently transmit the huge video data should be solved at first to realize the above 3D feeling functionality.

Object-based video coding has been intensely researched in the last few years and it is also supported by the new MPEG-4 standard. It has an important advantage over standard coding: it allows manipulation of image objects without complete decoding of the stream, and then improves the coding quality and reduces bit rate. In such a scheme, a prior segmentation map alpha plane of the image into objects is known in advance. We have implemented a fractal-based image codec with this type of functionality, so it can encode each object independently [1]. However, little work has been reported on the fractal video coding technique [2], especially with the object-based functionality [3]. We want extend our new approach to stereo video coding with this type of object-based functionality, then we can benefit for permitting new functionalities at the decoder, such as allowing independent transmission/decoding of each object in the video, object/background replacement, object-based video retrieval, and especially the gain of better image quality than standard coding since the object boundaries usually coincide with intensity edges that are difficult to encode[1] We can also benefit from the inherent advantages of fractal coding, such as good compression ratio, size scaleable output to get rate scaleable decoded video. Because the decoding process is formulated in terms of (geometric) partitions and not in terms of pixels or fixed size blocks of pixels, so it is possible to perform it at any resolution. All these potential applications promote the development of new object-based fractal video codec.

2. OBJECT-BASED FRACTAL CODING

Fractal image codec performs better, in terms of compression ratio and image quality, for variable-size than for fixed-size block, using a quadtree-based image partitioning which the image is progressively sub-divided by thresholding the range-domain comparison errors [4].

We propose a true object-based fractal coding scheme, objects can be defined by alpha plane and are coded independently of each other [1]. It is obvious that some range and domain blocks are located at object boundaries thus containing pixels from two or more objects, such as foreground and background. Therefore, in order not to mix pixels from different objects within one transformation, we associate the alpha plane a label with each pixel, it means same label pixels are from one object. We modify the dissimilarity measure to account for the pixels belonging to one object only and restrict the search for matching domain blocks to the object of interest.

Let $P(x, y)$ be the image intensity of a pixel at position (x, y), and let $\{r_1,...,r_M\}$ be the set of M non-overlapping range blocks partitioning the image. Similarly, let $\{d_1,...,d_N\}$ be the set of N, possibly overlapping, domain blocks covering all the image. So $P_{r_i} = \{P(x, y) : (x, y) \in r_i\}$ and $P_{d_j} = \{P(x, y) : (x, y) \in d_j\}$. In order to encode range block r_i, a search for index j of domain block d_j and the computation of photometric parameters s_i and o_i must be executed. And in order to ensure object-by-object encoding/decoding, both range and domain blocks must be located within the same object. Therefore, there are four cases regarding to the locations of blocks r_i and d_j with respect to one object:

a) r_i and d_j are both interior blocks;

b) r_i and d_j are both boundary blocks;

c) r_i is an interior block whereas d_j is a boundary block;

d) r_i is a boundary block whereas d_j is an interior block.

In the first case, standard full-block search is executed among same object interior blocks. In the second case, let $S^n_{d_j}$ be the n-th segment in the domain block d_j, and $S^m_{r_i}$ be the m-th segment in the range block r_i. Let $\widetilde{P}_{d_j}$ be a padded version of P_{d_j} defined as following:

$$\widetilde{P}_{d_j}(x, y) = \begin{cases} P_{d_j}(x, y) & (x, y) \in S^n_{d_j} \\ v & (x, y) \notin S^n_{d_j},\ (x, y) \in S^m_{r_i} \end{cases} \quad (1)$$

Where v is a padding value, typically mean local intensity in $S^n_{d_j}$. Because only partial matching is performed, so a new dissimilarity measure is defined as following:

$$\varepsilon\left(P_{r_i},\ \widetilde{P}_{d_j},\ \omega^m_i\right) = \frac{1}{\left|S^m_{r_i}\right|} \sum_{(x,y)\in S^m_{r_i}} \left[P_{r_i}(x, y) - \omega^m_i \widetilde{P}_{d_j}(x, y)\right]^2 \quad (2)$$

Where ω^m_i denotes an affine transformation for segment $S^m_{r_i}$. The above dissimilarity measure is evaluated only at pixel positions within a single range block segment $S^m_{r_i}$. In the third case, intensity extrapolation (1) of the domain block is always needed. In the fourth case, no padding is needed. The 1+2 search is attractive computationally while at the same time achieves very good performance, in which interior range blocks are searched and matched with the same object interior domain blocks, and boundary range blocks are partially searched and matched with the same object boundary domain blocks.

3. MONOCULAR VIDEO CODING BY CPM/NCIM

We proposed a new object-based scheme for fractal video coding which is based on the hybrid circular prediction mapping (CPM) and noncontractive interframe mapping (NCIM) [5]. This new scheme provides object-based functionality for video coding based on its alpha plane. The CPM/NCIM combines fractal video coding with the well known motion estimation/compensation algorithm that exploits the high temporal correlations between adjacent frames. In CPM and NCIM, each range block is motion compensated by a domain block in the previous frame, which is of the same size as the range block. The main difference between CPM and NCIM is CPM should be contractive for the iterative decoding process, while NCIM need not to be contractive since the decoding depends on the already decoded frame and is non-iterative. To provide object-based functionality, video will be encoded object by object according to its alpha plane. Boundary blocks are divided into segments belonging to different objects. The first n frames of the video are treated as a coding group and are encoded by applying CPM, each frame is predicted block-wise from the n-circularly previous frame. The remaining frames are encoded by employing NCIM. The structure of NCIM is the same as that of an interframe mapping which forms CPM, except that there is no constrain on the contrast scaling coefficients. In such case, the domain-range mapping can be interpreted as a kind of motion compensation. In the decoder, the first n frames are reconstructed by applying CPM iteratively, they are the minimal decodable set [6] of all the frames and can be decoded without reference to other frames, therefore, only CPM affects the convergence of the total fractal mapping. The remaining frames can be reconstructed by applying NCIM to the previous reconstructed frame without requiring iteration.

4. A NEW STEREO VIDEO CODING SCHEME BY MOTION OR QUADTREE-BASED DISPARITY COMPENSATION

The most mature technique for stereo video compression is the block-based stereo coding method defined in the MPEG-2 multiview profile. With this approach, for example, the coder first compresses right view with a monocular video codec, to code the left view, each macroblock is predicted both from the right view using disparity compensation, and from the previous frame of the left view using motion compensation. Either or both are used and the prediction errors are coded depending on which gives smaller prediction error. To make use of the existing coders for monosequence, the disparity vectors can be estimated by the same way as for motion estimation. The disadvantage of this approach is that the estimated disparity for macroblock is usually discontinuous and the decoded video usually has visible artifacts because of the fixed size matching blocks, and it is also very time consuming because of its exhaustive block-matching process.

In [7], in order to synthesize a 3D image having a larger number of views from a 3D image having fewer views, a new viewpoint interpolation technique and fractal-based data compression method which is scalable to any number of views are proposed. But this scheme is only applicable to multi-view based image interpolation, stereo images are not sufficient to use this technique.

In [8], the input views firstly be aligned to make their epipolar lines become horizontal, and mesh-based disparity estimation is then applied to the aligned images. Node points are iteratively moved in the direction of minimizing the prediction errors of full frame and disparities compensation are then accomplished based on the nodal displacements mapped back to the original image coordinates. However, the computational complexity of this approach is high because of iteration, and the occlusion problem is not well considered. This scheme yields visually more smooth predicted image but gets a lower PSNR because of some occlusion regions become distorted, comparing to the exhaustive block-matching algorithm which using a fixed block size. On the contrary, even visible artifacts appear in the block matching algorithm, but it seems keep the predicated view more faithful to the original view since the PSNR is higher.

In [9], a MPEG-4 based stereoscopic video encoder is proposed which the main view is encoded using MPEG-4 encoder and the auxiliary view is encoded by joint motion and disparity compensation.

On the base of our object-based fractal monocular video coding scheme, we encode the right view video by CPM/NCIM, and we nominate the reconstructed right view video during the coding process as the reference to predict left view video. We propose to use partial ($\{x, y\}$) or full ($\{x, y, s, o\}$) affine transformation quadtree-based disparity estimation/compensation, or simply apply CPM/NCIM motion estimation/compensation on left view video. Because the searching block sizes are variable and the range to domain searching and matching are only performed within the same pre-computed domain classes in the entire image, so we can expect the quadtree scheme can supply good performances both in PSNR and also visible quality. At the mean time, much less computation time than the exhaustive searching and iteration methods.

We benefit from fractal coding by using CPM/NCIM on right view video which can supply higher compression ratio than other compression methods. After the partial or full affine transformation quadtree-based disparity estimation between left and right view images, we can only record the $\{x, y\}$ positions or the full affine transformation parameters ($\{x, y, s, o\}$) of the similar blocks of left view image relative to the reconstructed right view image, or just applying CPM/NCIM on left view video. The predicated left image can be obtained by replacing each block by its best matching block in the reference right image, thus we can get much higher compression ratio for left video than reference right video. We can also apply full affine transformation or CPM/NCIM on left video, depending on the various video contents and user demands on PSNR and bit rate.

5. EXPERIMENTAL RESULTS

To evaluate the performances of the proposed codec on stereo video, we use two stereo videos and also the alpha planes of "Robert" and "Anne_Ion" (352×288 pixel, 5 frames for right and left, 8.33 frames/second, respectively). All our experiments are based on Sun Blade 100 workstation with 500 MHz CPU.

The average compression results of "Robert" are shown in Table 1 and Table 2 for right and left view video respectively. Object 0 means foreground, object 1 means background. The encoding time is 4 minutes 41 seconds and the decoding time is 7 seconds.

Table 1 The Compression Results of "Robert (Right)"

Average Value	Object 0	Object 1	Full Frame
PSNR (dB)	32.29	36.27	34.49
Compression ratio	26.67	64.07	18.83
bit per pixel (bpp)	0.30	0.12	0.42
bit rate (kbps)	247.45	103.01	350.46

Table 2 The Compression Results of "Robert (Left)"

Average Value	Object 0	Object 1	Full Frame
PSNR (dB)	31.33	37.15	34.20
compression ratio	32.87	65.84	21.92
bit per pixel (bpp)	0.24	0.12	0.36
bit rate (kbps)	200.78	100.25	301.03

The decoded object 0, full frame of fifth right and left frames of "Robert" are shown in Fig. 1.

Right: Obj 0, 35.20 dB, 0.26 bpp   Full, 37.74 dB, 0.36 bpp

Left: Obj 0, 33.17 dB, 0.23 bpp Full, 36.08 dB, 0.33 bpp

Fig.1 The decoded object 0, full frame of fifth right and left frames of “Robert”

The average compression results of “Anne_Ion” are shown in Table 3 and Table 4 for right and left view video respectively. The encoding time is 8 minutes 41 seconds and the decoding time is 8 seconds.

Table 3 The Compression Results of “Anne_Ion (Right)”

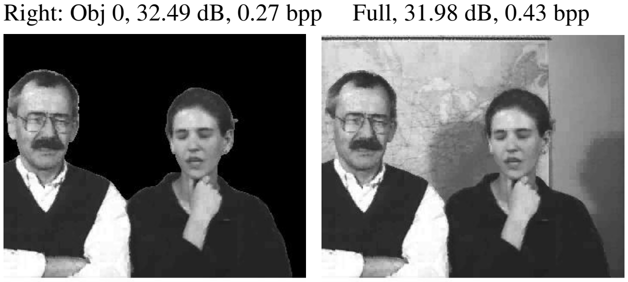
Average Value	Object 0	Object 1	Full Frame
PSNR (dB)	30.24	28.11	29.12
compression ratio	23.52	28.76	12.94
bit per pixel (bpp)	0.34	0.28	0.62
bit rate (kbps)	280.61	229.45	510.07

Table 4 The Compression Results of “Anne_Ion (Left)”

Average Value	Object 0	Object 1	Full Frame
PSNR (dB)	28.66	27.18	27.86
compression ratio	31.52	27.94	14.81
bit per pixel (bpp)	0.25	0.29	0.54
bit rate (kbps)	209.40	236.20	445.60

The decoded object 0, full frame of fifth right and left frames of “Anne_Ion” are shown in Fig. 2.

Right: Obj 0, 32.49 dB, 0.27 bpp Full, 31.98 dB, 0.43 bpp



Left: Obj 0, 30.27 dB, 0.24 bpp Full, 30.71 dB, 0.42 bpp

Fig.2 The decoded object 0, full frame of fifth right and left frames of “Anne_Ion”

Our stereo codec has the comparative performances as in [9], and with the advantages of simpler process, object-based independently coding/decoding functionalities, etc..

6. CONCLUSION

We propose a new object-based stereo video codec which employs the CPM/NCIM fractal coding scheme and partial or full affine transformation quadtree-based disparity compensation, or CPM/NCIM motion compensation on left video. Two stereo videos are tested to evaluate the performances of the proposed codec. The experimental results indicate its correctness and efficiency, and a compromise between high compression ratio and good PSNR is well obtained. Further efforts will be invested to speed up the current encoding process.

7. ACKNOWLEDGEMENTS

The authors thank FQRNT of Canada and PURAQ (MESRS et MELS/PIEQ) for the financial support. The authors also thank Dr. Sotiris Malassiotis for providing the stereo video sequences.

8. REFERENCES

[1] K. Belloulata and J. Konrad, “Fractal image compression with region-based functionality,” *IEEE Trans. Image Processing*, 11(4), pp. 351-362, 2002.

[2] Y. Fisher, T. P. Shen and D. Rogovin, “Fractal (Self-VQ) encoding of video sequences,” *Proc. SPIE:VCIP*, 2308, pp. 1359-1370, 1994.

[3] R. E. H. Franich, R. L. Lagendijk and J. Biemond, “Fractal Coding in Object-Based System”, *Proc. IEEE ICIP*, pp. 405-408, 1994.

[4] Y. Fisher, “Fractal encoding with quadtrees,” *Fractal image compression: theory and applications to digital images*, Springer-Verlag, New York, 1995.

[5] C. S. Kim, R. C. Kim and S. U. Lee, “Fractal coding of video sequence using circular prediction mapping and noncontractive interframe mapping,” *IEEE Trans. Image Processing*, 7(4), pp. 601-605, 1998.

[6] J. Domaszewicz and V. A. Vaishampayan, “Graph-theoretical analysis of the fractal transform,” *Proc. ICASSP*, pp. 2559-2562, 1995.

[7] T. Naemura and H. Harashima, “Self-similarity modeling for interpolation and extrapolation of multi-viewpoint image sets,” *Proc. ICASSP*, pp. 369-372, 1994.

[8] R. S. Wang and Y. Wang, “Multiview video sequence analysis, compression, and virtual viewpoint synthesis,” *IEEE Trans. Circuit Syst. Video Technol.*, 10(3), pp. 397-410, 2000.

[9] W.Yang and N. K. Ngi, “MPEG-4 based stereoscopic video sequences encoder,” *Proc. ICASSP*, pp. 741-744, 2004.