MULTISCALE RECURRENT PATTERN MATCHING APPROACH FOR DEPTH MAP CODING

Danillo B. Graziosi[1,3]*, Nuno M. M. Rodrigues*[1,2]*, Carla L. Pagliari*[4]*,*
Eduardo A. B. da Silva[3]*, Sérgio M. M. de Faria*[1,2]*, Marcelo M. Perez*[5]*, Murilo B. de Carvalho*[6]

[1]Instituto de Telecomunicações;[2]ESTG, Instituto Politécnico de Leiria, Portugal;
[3]PEE/COPPE/DEL/EE, Univ. Fed. do Rio de Janeiro;[4]DEE, Instituto Militar de Engenharia;
[5]Brazilian Army Technological Center;[6]TET, Univ. Fed. Fluminense, Brazil.

e-mails: danillo@lps.ufrj.br, nuno.rodrigues@co.it.pt, carla@ime.eb.br,
eduardo@lps.ufrj.br, sergio.faria@co.it.pt, perez@ctex.eb.br, murilo@telecom.uff.br.

ABSTRACT

In this article we propose to compress depth maps using a coding scheme based on multiscale recurrent pattern matching and evaluate its impact on depth image based rendering (DIBR).

Depth maps are usually converted into gray scale images and compressed like a conventional luminance signal. However, using traditional transform-based encoders to compress depth maps may result in undesired artifacts at sharp edges due to the quantization of high frequency coefficients. The Multidimensional Multiscale Parser (MMP) is a pattern matching-based encoder, that is able to preserve and efficiently encode high frequency patterns, such as edge information. This ability is critical for encoding depth map images.

Experimental results for encoding depth maps show that MMP is much more efficient in a rate-distortion sense than standard image compression techniques such as JPEG2000 or H.264/AVC. In addition, the depth maps compressed with MMP generate reconstructed views with a higher quality than all other tested compression algorithms.

Index Terms— Depth Maps, 3D Image Coding, Recurrent Pattern Matching

1. INTRODUCTION

Creation and consumption of 3D-enabled media content has raised over the past few years. There have been many efforts in developing technology for production of 3D content, as well as displays (e.g. auto-stereoscopic or glass-based displays). This evolution enables 3D content to reach consumers not only at specialized digital cinema theaters, but also at home, on personal computers or even at mobile devices. Nevertheless, an efficient coding and transmission scheme needs to be employed, in order to make this technology practical.

Responding to the demands of this growing sector, MPEG group developed the MPEG-C Part 3 standard [1], which provides the means to compress and transmit video and auxiliary data (e.g. depth data). Using video+depth as the exchange format allows: backwards compatibility with 2D; independence regarding the display and capture technology; direct compatibility with most "2D to 3D" algorithms; good compression efficiency (low overhead); user-controlled global depth range.

In addition, the Joint Video Team of the Video Coding Experts Group (JVT) of the ITU-T and the Moving Picture Experts Group (MPEG) of ISO/IEC have recently developed the Multiview Video Coding (MVC) standard to enable these new services [2]. MVC allows the compression of multiple views of the same scene by exploiting not only the intra-view temporal redundancies, but also the statistical dependencies among the multiple camera views. Additionally, depth images can be compressed by MVC, provided they are converted into YUV 4:0:0 format.

Depth can be regarded as a conventional luminance signal. Therefore, monocular video compression standards can be used for coding this extra information. However, these encoders often produce artifacts in depth that lead to erroneous sample shifts, which combined with object edges in the interpolation process may result in strong sample scattering, severely affecting the 3D rendering process.

The Multidimensional Multiscale Parser (MMP) is a block-based image coding algorithm that combines prediction with pattern matching at multiple scales. It is not based on any frequency domain assumption such as smoothness; therefore MMP does not suffer from high frequency coarse quantization and is able to preserve and efficiently encode this information. Its first results for coding disparity and depth data were presented in [3]. Here we present further results, and investigate the effects of MMP coding artifacts in the synthesized view rendering process.

This paper is organized as follows. Section 2 reviews some algorithms used to encode depth data. In Section 3 MMP algorithm will be briefly presented, and the features that make it efficient for encoding depth maps will be analyzed. Section 4 deals with the reconstruction techniques that were used to synthesize the virtual view from the base view and its depth information. Experimental results are given in section 5 and section 6 concludes the paper.

2. ENCODING OF DEPTH DATA

Depth map images are usually piecewise-smooth images, where sharp edge information indicates the boundaries of objects at different viewing distances [4]. Since the smoothness assumption holds for parts of the image, they can be efficiently encoded using the

This work has been supported by Fundacão Para a Ciência e Tecnologia (FCT), under project PTDC/EEA-TEL/099387/2008, the Brazilian Funding Agency FINEP and the Brazilian Army Technological Center.

transform-quantize-encode paradigm. One common approach is to use efficient video compression algorithms such as H.264/AVC, that exploits temporal and spatial correlation, or even H.264/AVC for MVC [2], that exploits also inter-view correlation. However, both schemes may suffer from coding artifacts that affect the 3D rendering.

In [5] a platelet-based coding algorithm is proposed. The algorithm employs a quad-tree decomposition of the image, and approximates each block segment by a piecewise-linear function. Due to its ability to preserve sharp object boundaries and edges, it presents a high rendering quality, but still its does not have an efficient rate-distortion performance, when compared to H.264/AVC.

A common way to avoid edge artifacts and still profit from transform-based encoder is to use a Region of Interest (ROI), and code the depth map by parts. In [6], JPEG2000 is used, together with a ROI that is determined based on the edge information and encoded separately, using different bit planes. However, this method is not suited for scenes where there are many objects in different depths.

Other proposals for depth map compression are based on different encoding tools, such as the combination of non-uniform sampling and adaptive meshing of images as in [7], compressive sensing [8], or even 3D DCT [9].

3. THE MMP ALGORITHM

MMP is a block-based encoder [10], that divides the input image into $N \times N$ blocks, performing intra prediction for each block. Prediction modes are based on the same modes used by the H.264/AVC standard. Expansions and contractions of vectors from a dictionary are used to perform approximate matching to the residual data. This is where the term "multiscale" is derived from. If there is no satisfactory match for the residual block, it can be segmented in either vertical or horizontal direction (a flag is sent to indicate the block segmentation). A prediction step is performed for each block segment, followed by an approximate matching of the corresponding residue pixels.

Concatenations of the encoded block segments are added to the dictionary, so the dictionary grows with patterns from the input signal, in a Lempel-Ziv fashion. By learning the codewords of the dictionary from the image itself, MMP adapts itself to the image's characteristics, avoiding common artifacts from transform-based encoders, such as ringing and edge smoothness.

The best choice of block segmentation, prediction mode and codeword index from the dictionary, used in order to approximate the residual block, is chosen based on a Lagrangian cost, given by

$$J = D + \lambda R \tag{1}$$

where D stands for the sum of squared errors of the block residue coding error, and R stands for the rate necessary to perform the encoding of codewords, prediction modes and segmentation flags.

All the elements (flags, prediction modes and dictionary indexes) are encoded using an adaptive arithmetic encoder. A greedy approach is used to find the encoding options that minimize the cost for each block. If an initial 1×1 dictionary having all possible amplitudes (e.g. -255 to 255) is used, the same algorithm can be employed to compress an image in the whole range of quality levels, from lossy to lossless.

More details on efficient dictionary adaptation schemes for the MMP can be found in [10]. Recent results in [11] show that MMP outperforms state-of-the-art image compressors such as JPEG2000 and H.264/AVC for both smooth and non-smooth images.

MMP shares some similarities with the recently proposed platelet-based depth map coding [5], also being able to preserve edges through a flexible segmentation and therefore suited for depth map encoding. Once a pattern, say a sharp edge, is sliced into several uniform pieces, it can be efficiently encoded by codewords of the dictionary. The new encoded pattern is incorporated in the dictionary through adaptation process. Since an edge usually spans over several blocks, the recently added edge pattern may be used to efficiently encode future segments, without further recurring to expensive block segmentation, while still being able to represent the sharp edge with high fidelity. This accounts for most of MMP's efficiency in encoding the sharp edges of depth map images. In medium to higher rates, MMP is able to efficiently compress the image, and allows its dictionary to grow at a reasonable size with various patterns, that will then encode the image in an efficient manner.

4. DEPTH IMAGE BASED RENDERING

DIBR is the process of synthesizing an arbitrary "virtual" view from reference images and associated per-pixel depth information. With a strong calibrated system, where all the camera parameters (position, focal distance, etc.) are known, the mapping of the 3D information into 2D can be done by applying camera projections. With the help of depth data, a captured image from one camera can be projected into the 3D world, and then reprojected into another camera viewing point, a process also called 3D image warping [12]. Notice that this rendering process requires the acquisition of reliable depth data, which is an ill-posed problem, due to difficulties such as change of illumination across views, specular surfaces or occlusions [12].

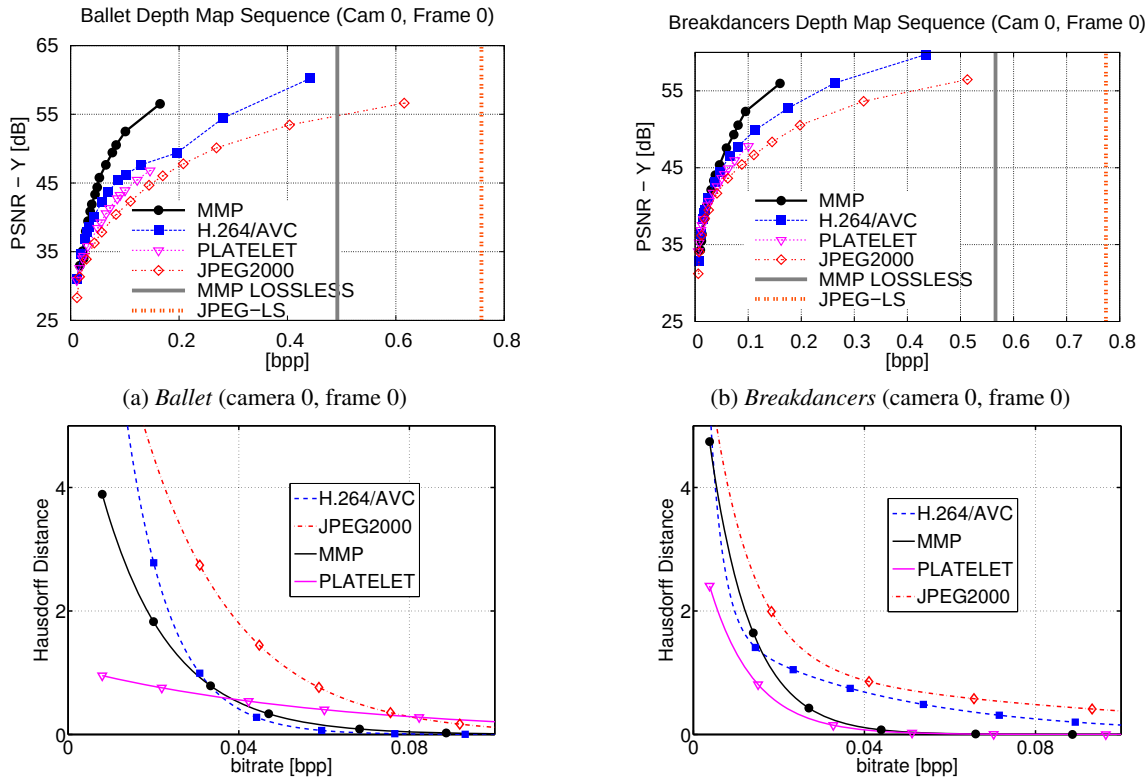
In this paper we implemented a rendering algorithm based on the work presented by Zinger et al [13]. Forward warping is performed for both texture and depth. A median filter is applied to the warped depth image, and the modified positions are inverse-warped with the altered depth value, in order to remove cracks and holes due to sampling. Ghost contours are removed by omitting warping of edges at high discontinuities and the occluded regions are filled by an inpainting algorithm that explicitly uses depth information. Figure 2(a) shows the reconstructed view of camera 4 for the *Ballet*, using the uncompressed texture and depth data from cameras 3 and 5. PSNR values are obtained between the synthesized view and the real texture from camera 4.

5. EXPERIMENTAL RESULTS

The test data set consists of the first frame of the sequences *Ballet* and *Breakdancers*, generated and distributed by Interactive Visual Group at Microsoft Research[1].

Depth images are converted into YUV 4:0:0 format video signals to be compressed by H.264/AVC-Intra. High-profile, level 4.0, INTRA mode from JM 16.2 software is used to code the depth data. For JPEG2000 encoder, the Kakadu software was used. For the experiments, depth images were compressed at different qualities. Due to MMP's capability of going from lossy to lossless, MMP's results for lossless encoding are also presented, and compared with lossless coding standard JPEG-LS [14]. This provides us an attainable bound of the compression capability of our algorithm. Source code and more resources for the MMP software are available online [15].

[1] http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload/



(a) *Ballet* (camera 0, frame 0)

(b) *Breakdancers* (camera 0, frame 0)

(c) *Ballet* (average over all cameras, frame 0)

(d) *Breakdancers* (average over all cameras, frame 0)

Fig. 1. R-D curves and Symmetrical Hausdorff distance for depth map coding.

Figures 1(a-b) show rate $\times$ PSNR results of encoded depth maps, including the platelet-based encoder [5][2]. Since PSNR values can be deceiving and do not always reflect how the algorithm will reconstruct the image [5, 7], we also present results with the Hausdorff Distance in Figures 1(c-d), a more appropriate measurement for depth map evaluation according to [5]. Lower values indicate less geometry distortion between the compressed and uncompressed depth map.

We can see that, in a rate distortion sense, MMP outperforms all the tested encoders, reaching more than 5dB at higher rates. Moreover, MMP can also perform lossless compression, and produces much better results than standard image lossless compression algorithm JPEG-LS [14]. However, the Hausdorff Distance indicates a weakness of our dictionary-based approach: the relatively bad performance at low bitrates. At such rates MMP is not able to grow its dictionary with various patterns, therefore suffering from artifacts such as blockiness. At medium to higher rates, the algorithm is able to grow its dictionary sufficiently, incorporating edges that will be crucial for the preservation and efficient coding of the edge information. At such edges, depth maps encoded by MMP have few geometry distortions, similar to the Platelet encoder, as indicated by the Hausdorff Distance of the encoded depth maps.

The quality of the rendered view created from cameras 3 and 5, with encoded depth maps, is compared to the real texture from camera 4. Objective measurements are performed by showing the SSIM (Structural SIMilarity) [16] values between the reference view and the rendered views with encoded depth maps. Since depth map coding may affect structures on the synthesized image, SSIM values give a more appropriate measurement of the resulting rendering quality. Figures 2(b-c) show that MMP have high values of SSIM for medium to high rates, due to its edge preservation feature, better preserving the reconstructed image structures. The same conclusion can be drawn by looking at Figures 2 (d-h), that show details of reconstructed image generated by each encoder. To be thorough with the reconstruction evaluation, we also provide the PSNR values in the caption of Figures 2 (d-h).

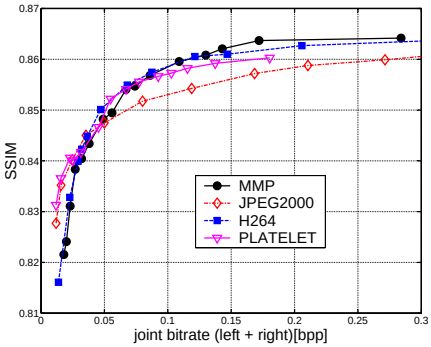
6. CONCLUSION

MMP has proved to be an effective tool for coding depth data. It is very efficient for middle to high rates, allowing also an acceptable reconstruction. It is also a flexible tool, ready to be used for texture and depth map coding, from lossy to lossless. However, for low bitrates it presents some reconstruction problems due to blockiness. Nevertheless, we advocate that most of the tested algorithms severely compromise the reconstruction at such rates, and that the image synthesis procedure is more effective encoding depth data at higher bitrates.

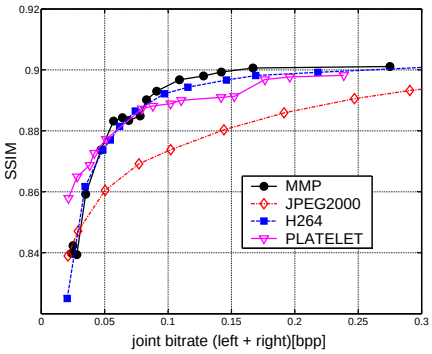
It is interesting to point out that for MMP, while the encoder suffers from high computational requirements due to dictionary search routine, the decoder can be made very simple, since block matching avoids the use of floating point operations. An interesting field of research is the reduction of the encoder's complexity through parallel computation and the use of GPU's. For future work, we will conceive a full encoding system for 3D images, where MMP is used to code the depth maps and the texture views. Due to its universal character, MMP is ready to be used for base view and depth map coding, and at all bit rates, from lossy to lossless.

[2]http://vca.ele.tue.nl/demos/mvc/PlateletDepthCoding.tgz

(a) Rendered frame with uncompressed depth data



(b) RD curves for reconstruction of the *Breakdancers* sequence (Cam4, Frame 0)



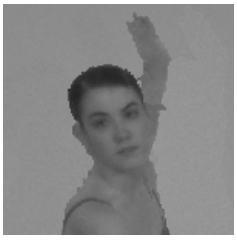
(c) RD curves for reconstruction of the *Ballet* sequence (Cam4, Frame 0)

(d) Original (32.945 dB)

(e) JPEG2000 (31.009 dB)

(f) H.264 (32.440 dB)

(g) PLATELET (32.693 dB)



(h) MMP (32.761 dB)

Fig. 2. Reconstruction of the first frame of the 4th camera of *Ballet* sequence, using the original views of the 3rd and 5th cameras and respective encoded depth maps. Figures (e)-(h) show details of the reconstruction with each depth map encoded at 0.075bpp.

7. REFERENCES

[1] Philips Applied Technologies, "MPEG-C part 3: Enabling the introduction of video plus depth contents," 2008, Suresnes, France.

[2] ITU-T,ISO/IEC JTC1, "Advanced video coding for generic audio-visual services," ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG4-AVC), Version 11: 2009.

[3] D. B. Graziosi, N. M. M. Rodrigues, C. L. Pagliari, S. M. M. de Faria, E. A. B. da Silva, and M. B. de Carvalho, "Compressing depth maps using multiscale recurrent pattern image coding," *Electronics Letters*, vol. 46, no. 5, pp. 340–341, 2010.

[4] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47(1/2/3), pp. 7–42, April-June 2002.

[5] P. Merkle, Y. Morvan, A. Smolic, D. Farin, K. Mueller, P. H. N. de With, and T. Wiegand, "The effects of multiview depth video compression on multiview rendering," *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 73–88, 2009.

[6] R. Krishnamurthy, B. Chai, H. Tao, and S. Sethuraman, "Compression and transmission of depth maps for image-based rendering," in *IEEE International Conference on Image Processing*, 2001, pp. 828–831.

[7] M. Sarkis, W. Zia, and K. Diepold, "Fast depth map compression and meshing with compressed tritree," in *The Ninth Asian Conference on Computer Vision (ACCV)*, September 2009.

[8] M. Sarkis and K. Diepold, "Depth map compression via compressed sensing," in *IEEE International Conference on Image Processing, 2009*, Nov 2009.

[9] M. Zamarin, S. Milani, P. Zanuttigh, and G.M. Cortelazzo, "A novel multi-view image coding scheme based on view-warping and 3D-DCT," *J. Vis. Commun. Image*, vol. 21, no. 8, pp. 462–473, 2010.

[10] N. M. M. Rodrigues, E. A. B. da Silva, M. B. de Carvalho, S. M. M. de Faria, and Vitor M. M. Silva, "On dictionary adaptation for recurrent pattern image coding," *Image Processing, IEEE Transactions on*, vol. 17, no. 9, pp. 1640–1653, September 2008.

[11] D. B. Graziosi, N. M. M. Rodrigues, E. A. B. da Silva, S. M. M. de Faria, and M. B. de Carvalho, "Improving multiscale recurrent pattern image coding with least-squares prediction," in *IEEE International Conference on Image Processing, 2009*, Nov 2009.

[12] Y. Morvan, *Acquisition, compression and rendering of depth and texture for multi-view video*, Ph.D. thesis, Eindhoven University of Technology, 2009.

[13] S. Zinger, L. Do, and P.H.N. de With, "Free-viewpoint depth image based rendering," *J. Vis. Commun. Image*, vol. 21, no. 8, pp. 533–541, 2009.

[14] Marcelo J. Weinberger, Gadiel Seroussi, and Guillermo Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Trans. Image Processing*, vol. 9, pp. 1309–1324, 2000.

[15] MMP Project, "http://www.lps.ufrj.br/profs/eduardo/mmp/," .

[16] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli, "Image quality assessment: from error visibility to structural similarity.," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.