DEPTH MAPS CODING WITH ELASTIC CONTOURS AND 3D SURFACE PREDICTION

Marco Calemme[1], Pietro Zanuttigh[2], Simone Milani[2], Marco Cagnazzo[1], Beatrice Pesquet-Popescu[1]

[1] Télécom Paristech, France [2] University of Padova, Italy

ABSTRACT

Depth maps are typically made of smooth regions separated by sharp edges. Following this rationale this paper presents a novel coding scheme where the depth data is represented by a set of contours defining the various regions together with a compact representation of the values inside each region. A novel coding scheme based on elastic curves allows to compactly represent the contours exploiting also the temporal consistency between them in different frames. A 3D surface prediction algorithm is then exploited in order to obtain an accurate estimation of the depth field from the contours and a subsampled version of the data. Finally, an ad-hoc coding strategy for the low resolution data and the prediction residuals is presented. Experimental results prove how the proposed approach is able to obtain a very high coding efficiency outperforming the HEVC coder at medium-low bitrates.

Index Terms— Depth Maps, Contour Coding, Elastic Deformation, Segmentation, HEVC

1. INTRODUCTION

Depth maps can be represented by means of grayscale images and compressed with standard image compression tools but better results can be obtained by exploiting ad-hoc approaches tailored to the specific properties of these representations and in particular the fact that depth maps are usually made by a set of smooth surfaces separated by sharp edges. The proposed compression scheme follows this rationale and is based on the idea of representing the depth field as a set of contours defining the various regions together with a compact representation of the depth field inside each region. This allows to preserve the sharp edges between the various surfaces, a relevant property since the positive impact of contour-preserving compression techniques on synthesized images, obtained through DIBR approaches [1], has been recently confirmed by means of subjective tests [2].

Segmentation information has been widely used for depth compression but the coding of the segments shape is critical and most available approaches are not able to propagate the information from one frame to the following. An efficient contour coding technique has recently been proposed in [3]: it uses elastic deformation of curves in order to exploit the temporal or inter-view redundancy of object contours and code more efficiently the contour information to be used for depth compression. After the main discontinuities have been captured by the contour description, the depth field inside each region is smooth and can be efficiently predicted from just a small number of samples. For this task we exploited an approach derived from [4].

The general workflow of the proposed approach is shown in Fig. 1. The first step is the segmentation of the depth maps in order to extract the main objects and structures in the scene. The segment contours are then compressed using the approach described in [3] and the average depth value in each segment is stored. The depth maps and the segmentation data are also subsampled and the difference between the subsampled representation and the segment averages is compressed using a differential prediction scheme followed by the HEVC coder. Then a surface prediction algorithm derived from [4] is used to predict the input depth maps from the subsampled data and the contours. Finally the residuals between the prediction and the input depth map are lossy compressed using the HEVC coder.

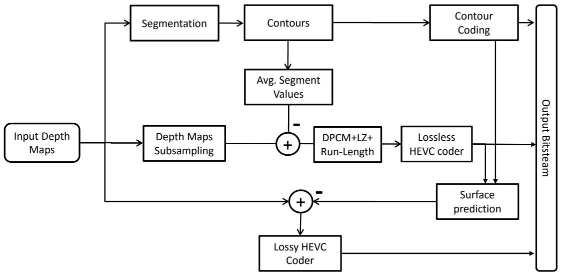


Fig. 1. Overview of the proposed approach.

2. RELATED WORK

Many depth coding approaches exploits the key idea that depth maps are made of smooth regions divided by sharp edges. A possible solution is to exploit a segmentation of the depth map in order to decompose it into a set of homogeneous regions that can be represented by simple models or low resolution approximations. Examples of approaches belonging to this family are [5, 4, 6]. The approach of [5] adopts a segmentation of the depth image followed by region-based coding scheme, while [4] exploits segmentation followed by a linear prediction scheme and finally [6] exploits a scalable decomposition into progressive silhouettes.

Another possibility is to exploit adaptive interpolation schemes able to correctly handle edge regions, for example the approach of [7] subsamples the depth information and then reconstructs it with an adaptive interpolating filter. The approach of [8] exploits platelets with support areas adapted to the object boundaries. Wavelet-based schemes have also been used, the solution of [9] uses an edge-preserving lifting scheme while the approach of [10] exploits a breakpoint field representing crack edges in order to avoid filtering across the edges in the wavelet reconstruction. The idea of exploiting crack edges is used also in [11].

Notice that these families of approaches need a representation of the segmentation or edge information that can have a relevant impact on the total bitrate, for this task efficient contour coding schemes are needed. Typical contour coding techniques rely on chain coding and differential chain coding [12], or on polygon approximation [13]. Even if they are quite effective, these techniques do not exploit the temporal redundancy of contours of the same objects in different time instants or views.

Color data can also been used to assist the compression of depth information as in [14], [15], [16] and [17]. The coding of depth videos has also been considered, e.g., in [18] the idea that pixels with similar depth have similar motion is exploited. In addition, the approach in [19] performs an object classification on the scene and adapts the depth coding strategy according to the motion characteristics.

3. CONTOUR CODING

Srivastava *et al.*[20] introduced a framework to model a continuous evolution of elastic deformations between two reference curves. According to the interpretation of the elastic metric, it is relatively easy to compute the geodesic between two curves: it consists in a continuous set of deformations that transform one curve into another with a minimum amount of stretching and bending, and independently from their absolute position, scale, rotation and parametrization. The referred technique thus interpolates between shapes and makes the intermediary curves retain the global shape structure and the important local features such as corners and bends. An example of the geodesic connecting two curves is shown in Fig. 2. We show in black two contours extracted from a MVD sequence, corresponding to views 1 and 8. The curves in red are the contours extracted from the intermediate views, while in dashed blue we show a sampling of the elastic geodesic computed between the two extreme curves. The elastic deformations along the geodesic reproduce very well the deformations related to a change of viewpoint or a temporal evolution of an object in a sequence, given the initial and final shapes.

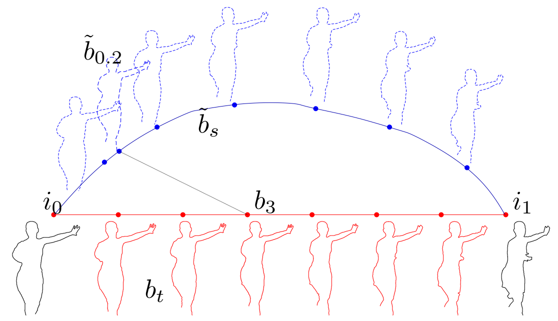


Fig. 2. Geodesic path of elastic deformations $\tilde{b}_s$ from the curve i_0 to i_1 (in dashed blue lines). b_3 is one of the contours b_t extracted from the intermediate frames between the two reference ones, a good matching elastic curve $\tilde{b}_{0.2}$ along the path is highlighted.

The lossless coding of the contour is performed through an arithmetic coder, and the input symbol probability distribution is modified on the fly according to the context provided by the elastic prediction [3].

Supposing that the encoder and the decoder share a representation of the initial and final shape, they can reproduce exactly the same geodesic path between them. Then, the decoder will use a suitable point of the geodesic, *i.e.* one of the dashed curves in Fig. 2, as context [21] to encode an intermediary contour (one of the solid curves in the same figure). The encoder will only have to send a value in $[0, 1]$ to let the decoder identify this curve. Moreover, accurate probability values are obtained "following" the elastic prediction, so the decoder needs also a correspondence between the points of the curve b and its elastic prediction $\widetilde{b}$ to relate each point of b to a portion of $\widetilde{b}$. An example of the result of the association of the two curves is given in Fig. 3.

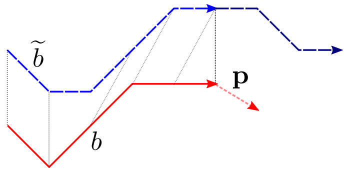


Fig. 3. Extracts from the curves b (red) and $\widetilde{b}$ (dashed blue). The correspondences between the two curves are indicated with thin dotted black lines. The vector $\mathbf{p}$ represents the most probable future direction.

As we compute the most probable future direction $\mathbf{p}$, a distribution is defined for each symbol using the von Mises statistical model, and an efficient representation of the curve is obtained using these distributions in conjuction with an arithmetic coder.

4. DEPTH CODING

After the segmentation and the coding of the segment contours, the next step is the subsampling of depth data. The depth map is simply subsampled according to a regular grid (see Fig. 4c) and the average depth values in the segments are subtracted from the subsampled data. The sampling factor Δ is selected according to the resolution and the amount of detail of the depth data. In the experimental set-up presented in Section 5, sampling grids of 16×16 or 32×32 were used. The subsampled data is basically a low resolution depth map of size (W/Δ) by (H/Δ), where W and H are the dimensions of the original depth map. This information is compressed in lossless way in order to prevent blurring on the edges and averaging between neighboring samples of different segments, which would make useless the benefits of the segmentation step. In order to obtain a high coding gain, the samples of the first depth image (Intra depth frame) are first scanned on a raster order and converted into a sequence of couples (r, l) according to a run-length coding strategy. Then, both runs and lengths are coded using a Lempel-Ziv 78 algorithm, which permits obtaining good coding gains with a limited complexity. For the following frames, a DPCM strategy is tailored in order to exploit temporal redundancy: depth samples are first predicted from the previous frames (zero-motion prediction), and the prediction difference is coded like depth samples of the Intra depth frame.

Although technical literature provides many other examples of more effective coding strategies, the very low resolution of the input images permits obtaining very small file sizes.

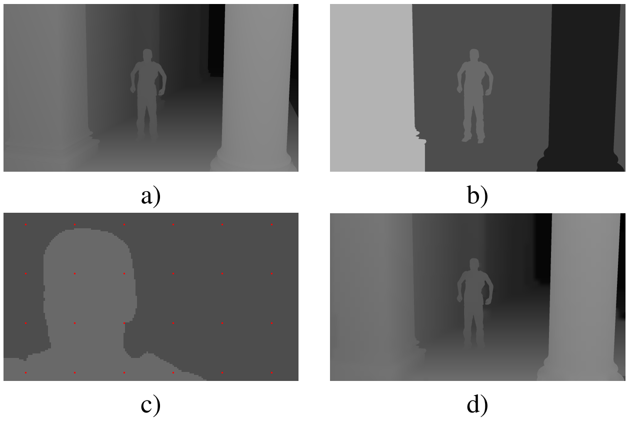


Fig. 4. a) Sample depth map from the Dancers sequence; b) segmented depth map; c) detail of the segmented depth map with the subsampling grid (red dots represent the position of the grid samples); d) depth map predicted from the contour and the low resolution samples.

The high resolution contour information and the low resolution depth data can be used to produce a very accurate prediction of the input high resolution depth map. For this step we used an approach derived from [4] that here we briefly recall. The key idea is that the depth map is made of a set of smooth surfaces represented by the segmented regions. For each region a set of samples is available, i.e., the subsampled depth map pixels belonging to that region. Each pixel p_i of the high resolution depth map is so surrounded by 4 samples of the grid (see Fig. 5) and the idea is to predict it by interpolating only the grid samples belonging to the same region. If all the 4 samples belong to the same region of p_i, the estimation of p_i is simply given by the bilinear interpolation of the 4 samples. When instead the sample is close to the contour of one of the segmented regions, some of the 4 samples could belong to different regions and the value of p_i is estimated by interpolating only the values of the grid samples that belong to the same region. Up to the symmetry and excluding the trivial case of samples that correspond to grid points, there are 5 possible cases (see Fig. 5).

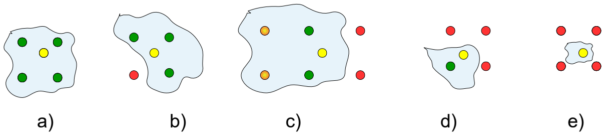


Fig. 5. Grid samples: the 5 possible configurations. The unknown yellow pixel is estimated by using only the green pixels (plus the orange ones in case c).

a) If all the 4 samples are inside the same region p_i is simply given by the bilinear interpolation of the 4 samples.

b) If p_i is surrounded by 3 samples belonging to the same region it is estimated by assuming that it lies on the plane defined by the 3 points.

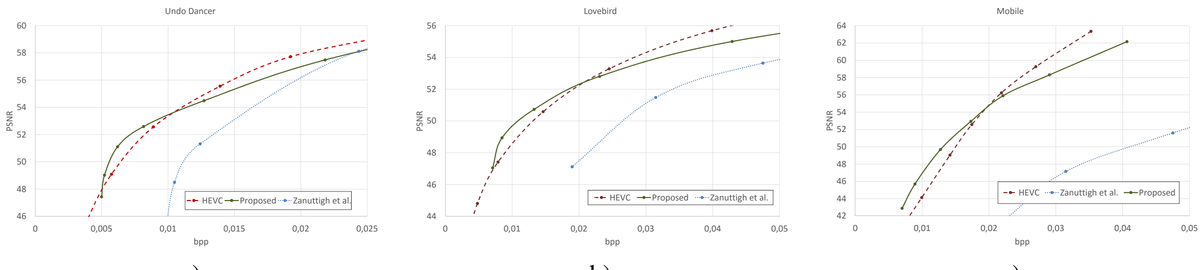
c) If p_i is surrounded by 2 neighbors in the region and 2 outside (e.g., when it is close to an edge) the two external points are predicted by assuming that each of them lies on the line passing through the closest available point and the symmetrical point with respect to the available one (see Fig. 5c). Then the pixel to be predicted is computed by bilinear interpolation.

d) If p_i has just one neighbor in the same region the value of this sample is taken as estimate.

e) If p_i has no neighbors the average depth value of the region is used.

This approach allows to obtain a very accurate prediction of the input depth map, with only some small artifacts typically on the edges not captured by the segmentation.

Finally the residual difference between the estimated and actual depth map is computed and lossy compressed using the HEVC coding engine. In this case, the main RExt profile is used, with disabled lossless mode and rate-distortion optimization enabled.



a) b) c)

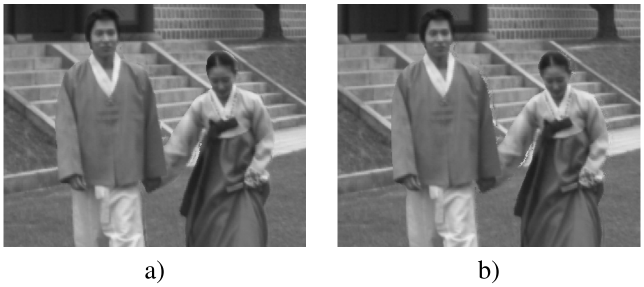
Fig. 6. Comparison of the performances of the proposed approach with the HEVC coder and with the method of Zanuttigh et al. [4]: a) *Undo Dancer sequence*, b) *Lovebird* sequence, c) *Mobile* sequence.

5. EXPERIMENTAL RESULTS

In order to evaluate the performances of the proposed approach we used three different sequences with different resolution and properties. The first is the *undo dancer* sequence, an high resolution (1920×1088) synthetic scene, the second is the *lovebird* sequence, a real world sequence with a resolution of 1024×768 and the third is the *mobile* sequence, another synthetic scene with resolution 720×480. We compared the proposed approach with the HEVC standard video coder and with the segmentation-based depth coding approach of [4].

Fig. 6a shows the performance of the proposed and competing approaches on the *undo dancer* sequence. There is a large performance gain with respect to [4] at all bitrates. The two approaches share the idea of exploiting segmentation and low resolution approximation, but the contour coding strategy of this work is more efficient than the arithmetic coder of [4] and the proposed low resolution samples and residual compression strategies largely outperform the JPEG2000 based coding used in that work. The comparison with HEVC is more though: at low and medium bitrates (up to $0.01bpp$) the proposed approach is able to outperform HEVC thanks to the efficient representation of the contours and of the very low information content of the residuals. The performance gain reaches around $2db$ at around $0.006bpp$. Notice how the contours remain sharp and not blurred on the whole bitrate range while HEVC achieves this result only at high bitrates. Fig. 6b shows the results for the *lovebird* sequence: even if the resolution is different and this is a real world scene and not a synthetic one, the results are very similar to the previous one, with the proposed approach able to outperform [4] at all bitrates and HEVC for bitrates up to $0.02bpp$ corresponding to around $53db$. The *mobile* sequence (Fig. 6c) is more challenging for our approach since the edges are already blurred in the input depth maps. This reduces the effectiveness of coding strategies based on the assumption of sharp edges between the various regions like the proposed one and [4] (that on this sequence has very poor performances). However our approach is still able to outperform HEVC at low bitrates.

The edge preserving capabilities of the proposed approach are particularly evident when the depth is used for view warping and interpolation. Fig. 7 shows a detail from a synthesized view from two frames of the *lovebird* sequence. Depth data compressed at around $0.008bpp$ with both our approach and HEVC have been used. From the figure it is clear how depth compressed with the proposed approach leads to a better interpolation, in particular notice how the artifacts close to the people edges are much smaller. Other examples of interpolation are contained in the additional material.



a) b)

Fig. 7. Detail of an interpolated view from the *lovebird* sequence: a) using depth compressed with our approach; b) using depth compressed with HEVC.

6. CONCLUSION

In this paper we proposed a novel depth coding scheme. The proposed scheme is able to exploit the temporal redundancy through an efficient contour coding scheme based on elastic curves. The depth field inside each region is efficiently compressed by exploiting a surface interpolation scheme and efficient coding schemes for both the low resolution depth and the prediction residual. Experimental results demonstrated that the proposed approach is able to properly preserve the edges even at low bitrates, a property very useful for warping applications and that it is very effective at medium-low bitrates, where it outperforms the HEVC coder.

Further research will be devoted to the replacement of the segmentation masks with edge-based schemes able to work also with partial and interrupted edges and to a better exploitation of the temporal redundancy.

7. REFERENCES

[1] C. Fehn, “Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV,” in *Electronic Imaging 2004*. International Society for Optics and Photonics, 2004, pp. 93–104.

[2] M. Calemme, M. Cagnazzo, and B. Pesquet-Popescu, “Contour-based depth coding: a subjective quality assessment study,” in *Proceedings of 2015 IEEE International Symposium on Multimedia (ISM)*, 2015.

[3] M. Calemme, M. Cagnazzo, and B. Pesquet-Popescu, “Lossless contour coding using elastic curves in multiview video plus depth,” *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014.

[4] P. Zanuttigh and G.M. Cortelazzo, “Compression of depth information for 3d rendering,” in *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2009*. IEEE, 2009, pp. 1–4.

[5] B. Zhu, G. Jiang, Y. Zhang, Z. Peng, and M. Yu, “View synthesis oriented depth map coding algorithm,” in *Proc. of APCIP 2009*, 2009, vol. 2, pp. 104 –107.

[6] S. Milani and G. Calvagno, “A Depth Image Coder Based on Progressive Silhouettes,” *Signal Processing Letters, IEEE*, vol. 17, no. 8, pp. 711–714, 2010.

[7] K.-J. Oh, S. Yea, A. Vetro, and Y.-S. Ho, “Depth reconstruction filter and down/up sampling for depth coding in 3-d video,” *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 747–750, Sept. 2009.

[8] Y. Morvan, D. Farin, and P. de With, “Depth-Image Compression based on an R-D Optimized Quadtree Decomposition for the Transmission of Multiview Images,” in *Proc. of IEEE ICIP 2007*, 2007.

[9] A. Sanchez, G. Shen, and A. Ortega, “Edge-preserving depth-map coding using graph-based wavelets,” in *Proc. of* 43^{rd} *Asilomar Conference 2009*, Pacific Grove, CA, USA, Nov. 2009, pp. 578 –582.

[10] R. Mathew, D. Taubman, and P. Zanuttigh, “Scalable coding of depth maps with r-d optimized embedding,” *Image Processing, IEEE Transactions on*, vol. 22, no. 5, pp. 1982–1995, 2013.

[11] I. Tabus, I. Schiopu, and J. Astola, “Context coding of depth map images under the piecewise-constant image model representation,” *Image Processing, IEEE Transactions on*, vol. 22, no. 11, pp. 4195–4210, 2013.

[12] A. Katsaggelos, L.P. Kondi, F.W. Meier, J. Ostermann, and G.M. Schuster, “MPEG-4 and rate-distortion-based shape-coding techniques,” vol. 86, no. 6, pp. 1126–1154, June 1998.

[13] J.I. Kim, A.C. Bovik, and B.L. Evans, “Generalized predictive binary shape coding using polygon approximation,” vol. 15, no. 7-8, pp. 643–663, May 2000.

[14] E.-C. Forster, T. Lowe, S. Wenger, and M. Magnor, “Rgb-guided depth map compression via compressed sensing and sparse coding,” in *Proc. of PCS 2015*, 2015.

[15] S. Milani, P. Zanuttigh, M. Zamarin, and S. Forchhammer, “Efficient depth map compression exploiting segmented color data,” in *Proc. of IEEE ICME 2011*, 2011, pp. 1–6.

[16] S. Milani and G. Calvagno, “3D Video Coding via Motion Compensation of Superpixels,” in *Proc. of EUSIPCO 2011*, Aug. 29 – Sept. 2, 2011, pp. 1899 – 1903.

[17] M. Georgiev, E. Belyaev, and A. Gotchev, “Depth map compression using color-driven isotropic segmentation and regularised reconstruction,” in *Proc. of DCC 20155*, 2015, pp. 153–162.

[18] I. Daribo, G. Cheung, and D. Florencio, “Arithmetic edge coding for arbitrarily shaped sub-block motion prediction in depth video compression,” Orlando, FL, USA, Sept. 2012, IEEE, pp. 1541 – 1544.

[19] S. Milani and G. Calvagno, “A cognitive approach for effective coding and transmission of 3D video,” *ACM Trans. Multimedia Comput. Commun. Appl. (TOMCCAP*, vol. 7S, no. 1, pp. 23:1–23:21, Nov. 2011.

[20] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn, “Shape analysis of elastic curves in euclidean spaces,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1415–1428, Sept. 2010.

[21] M. Cagnazzo, M. Antonini, and M. Barlaud, “Mutual information-based context quantization,” *Signal Processing: Image Communication*, vol. 25, no. 1, pp. 64–74, Jan. 2010.