A Depth Image Coder Based on Progressive Silhouettes

Simone Milani, *Member, IEEE,* Giancarlo Calvagno, *Member, IEEE*

***Abstract*—An efficient compression of depth maps proves to be a crucial element in the transmission and storage of 3D scenes. However, the peculiarities of geometry information make the traditional coding paradigms for natural images less effective for the coding of depth images. The paper presents a novel coding scheme that employs an oversegmentation of the input depth image into a huge set of small regions. These regions are then fused together according to the target number of objects that the algorithm needs to identify in the representation. This procedure is iterated more than once generating several refinement layers that permit obtaining a progressively-increasing quality in the scene. Experimental results show that in most cases the proposed approach reaches a better coding performance with respect to previous coding methods.**

***Index Terms*—depth map coding, image segmentation, H.264/AVC, scalable compression**

I. Introduction

The recent availability of low-cost and high definition 3D displays has nurtured novel 3D video consumer applications which permit the access and the fruition of 3D multimedia contents over an heterogeneous set of networks. However, an effective and timely delivery of the required information implies the adoption of efficient coding paradigms that permit maximizing the quality of 3D experience perceived by the end user for a given transmission rate. For many 3D video applications (such as 3DTV, free-viewpoint video, etc...), the transmitted data consist of a set of video texture and geometry information streams. These permit synthesizing the desired virtual camera views at different positions by mapping each pixel into a three-dimensional point and reprojecting it according to the desired viewpoint (Depth Image Based Rendering - DIBR [1]). While the texture information can be associated to a traditional video sequence, geometry information is represented by a sequence of depth maps which associate each pixel with its distance from the camera. At the receiver, the accuracy of the depth map has a significant impact on the final visual quality since the additional noise introduced by the coding operations leads to a wrong positioning of the warped points in the synthesized view. Moreover, depth images present different characteristics with respect to the texture views. As a matter of fact, video codec experts have been focusing on novel ad-hoc coding paradigms that permit compressing the depth images effectively.

Initial works on compression of depth maps employed traditional video coders which have been designed for the transmission of standard sequences. However, recent results have shown that it is possible to build more efficient solutions taking into consideration the peculiarities of geometry signals. Depth images are characterized by an alternation of smooth regions and sharp edges. The first usually denote the surface of the objects which can be flat or rounded according to the distance of the object from the camera. Edges, instead, denote the object boundaries and permit distinguishing the different elements from the background. As a matter of fact, these discontinuities make traditional video coders ineffective since most of them have been designed to deal with low-pass video signals, and their application to depth images produces the appearing of evident artifacts along image borders and depth discontinuities. This quality degradation proves to be significant whenever synthesizing novel views since the warping process maps pixels into wrong positions (see [1]). Most of the approaches proposed in literature try to identify these discontinuities and remove them to obtain a low pass signal. A first work by Krishnamurty *at al.* [2] employs an edge detector to drive the JPEG2000 coder [3] and to avoid the creation of artifacts along the edges.

This task can also be performed using Bandelets [4] and Contourlets [5]. In [6] Morvan *et al.* proposed an effective approach that combines a quad-tree decomposition of the depth image (employed to identify smooth regions among sharp edges) with a Wedgelet-based approach (used to fill the regions). Other solutions perform a segmentation of the input image [7] identifying the objects in the scene. The shapes of the objects are then progressively filled with points from a mesh-grid structure that permits a scalable reconstruction of the depth image. In this case, the effectiveness of the approach strongly relies on the accurateness of the segmentation algorithm.

This paper presents a novel depth image coding algorithm that employs oversegmentation to split the input depth image into multiple segments. These regions are then merged together according to the depth information creating an object-based quality-scalable prediction for the depth map to be coded. Section II presents the general scheme of the approach, with Section II-A describing the segmentation strategy and Section II-B presenting how different layers are created. Section III compares the proposed approach with other solutions while Section IV draws the final conclusions.

S. Milani and G. Calvagno are with the Dept. of Information Engineering, University of Padova, via Gradenigo 6/B, 35131 Padova - Italy, phone: + 39 049 827 7641, fax: +39 049 827 7699, e-mail: {simone.milani,calvagno}@dei.unipd.it

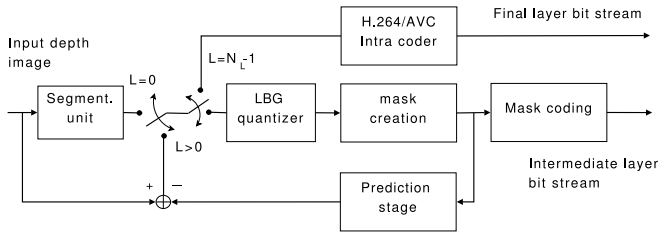


Fig. 1. Block diagram of the encoder. The parameter L denotes the layer number and N_L is the maximum number of layers.

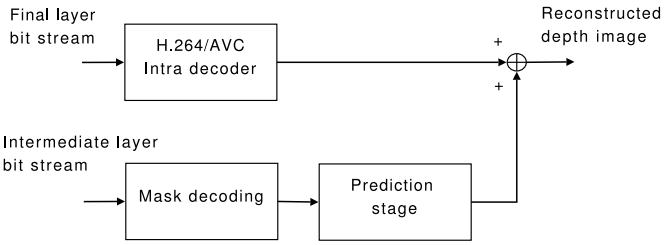


Fig. 2. Block diagram of the decoder.

II. The proposed video codec

The adopted scheme has been built on the structure of the H.264/AVC Intra coder since, among the traditional coding standards, it provides one of the best coding performance. In our approach, we considered the possibility of improving the compression gain of H.264/AVC Intra with the inclusion of a segmentation routine in the prediction strategy. The segmentation permits identifying the position of edges in the sequence and isolating smooth regions of the depth image.

Figure 1 shows the scheme of the adopted depth coder. The input depth image is oversegmented into a wide set of small regions using the algorithm in [8]. In a second step, the different regions are merged together according to the average depth value for each region and the number N_{Obj} of objects that the coding routine wants to identify in the depth map. Each object (coded by a binary mask) can be associated to an average depth value computed from the pixels included in the mask. The segmentation permits building a first prediction for the current depth image by combining the different masks with their relative average depth levels (see Fig. 2). This approximation can be refined by computing the residual signal and iterating the coding process. Additional masks are created at each iteration determining several quality layers. After a tunable number of iterations, the residual signal is coded by a standard H.264/AVC Intra coder creating the upper enhancement layer. In the following, we will present the different phases in detail.

A. Oversegmentation of the depth image and cluster fusion

Depth images are usually characterized by the alternation of smooth areas enclosed within shapes that approximate the silhouettes of the objects. As a consequence, the identification of object shapes permits improving significantly the coding performance.

As it was mentioned before, the input frame is oversegmented into a set of N_R regions $R_i \subset \mathbb{Z}^2$, $i = 0, \ldots, N_R - 1$, which cover the whole image area (see Fig. 3(b)). The configuration parameters of the segmentation routine [8] are set so that about 100 regions per image are generated. This choice has been done evaluating the performance of the segmentation algorithm on a wide set of test images. According to the results of our tests, using 100 different partitions permits cutting the input image effectively without generating an exceeding number of segments.

Each region can be associated to an average depth value $\overline{d}_i$ obtained averaging the depth values of its pixels, i.e.

$$\overline{d}_i = \frac{1}{|R_i|} \sum_{(x,y)\in R_i} d(x,y) \tag{1}$$

where $d(x,y)$ is the depth value at position (x,y) from the input depth image D and $|R_i|$ is the cardinality of the set R_i. The set of values $\overline{d}_i$ is then quantized into N_{Obj} classes using an LBG vector quantizer [9]. The initial centroids are identified partitioning the range of values $\overline{d}_i$ equally into N_{Obj} regions and computing the average values of depth pixels within these regions. After a finite number of iterations of the LBG algorithm, the coding routine assumes that the algorithm has converged to a locally-optimal solution. Each set R_i can be associated to one of the N_{Obj} classes merging them together into a common macro-region $\overline{R}_{0,k}$ $(k = 0, \ldots, N_{Obj} - 1)$. The obtained N_{Obj} regions can be characterized by $N_{Obj} - 1$ binary masks. Also in this case, depth values within the region $\overline{R}_{0,k}$ can be approximated with the average $\overline{d}'_{0,k}$ of all the included depth pixels, which provides a coarser estimate of the depth information. Since each region has been obtained from a segmentation routine, the masks usually identify the objects comprised in the scene and the background. As a matter of fact, the value $\overline{d}'_{0,k}$ permits obtaining a first flat representation of the related object. The combination of the different depth masks weighted by the averages $\overline{d}'_{0,k}$ permits approximating the input depth image with the prediction image

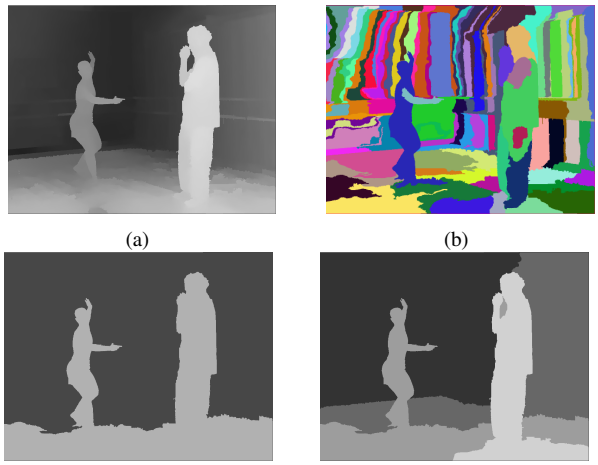
$$d_{0,p}(x,y) = \sum_{k=0}^{N_{Obj}-1} \overline{d}'_{0,k}\, \mathcal{I}\big((x,y) \in \overline{R}_{0,k}\big) \tag{2}$$

where $\mathcal{I}(\cdot)$ is the indicating function. The generated approximation will be the initial prediction for the proposed coding algorithm (see Fig. 3(c)). Masks are converted into a binary stream using a JBIG binary coder [10]. The prediction accuracy will be refined by iterating several approximation steps as reported in the following subsection.

B. Enhancement layers

After creating the initial prediction frame $D_{0,p}$ of pixels $d_{0,p}(x,y)$ (layer $L = 0$), the depth map D can be approximated more finely by additional enhancement layers, which are obtained via an iterative procedure producing a set of $N_L - 1$ incremental refinements (obtaining N_L layers).

More precisely, the residual frame $E_1 = D - D_{0,p}$ is quantized into N_{Obj} levels by an additional LBG procedure. Once again each residual pixel $e_1(x,y)$ in D_1 can be approximated by one of the N_{Obj} centroids $\overline{e}_{1,k}$ computed by the



(a) (b) (c) (d)

Fig. 3. Reconstructed images at different layers for frame 0 of `ballet` sequence ($N_{Obj} = 2$). a) original b) segmented c) layer 0 d) layer 1.

LBG routine. In this way, the residual map D_1 is partitioned into a set of N_{Obj} regions $\overline{R}_{1,k}$, where each pixel value $e_1(x,y) \in \overline{R}_{1,k}$ is replaced with $\overline{e}_{1,k}$ (see Fig. 3(d)). The original residual map E_1 can then be approximated by its quantized version $E_{1,q}$, whose pixel values are

$$e_{1,q}(x,y) = \sum_{k=0}^{N_{Obj}-1} \overline{e}_{1,k}\, \mathcal{I}\left((x,y) \in \overline{R}_{1,k}\right). \quad (3)$$

Note that also in this case the signal $E_{1,q}$ can be fully characterized by a set of $N_{Obj} - 1$ binary masks and the associated average $\overline{e}_{1,k}$. Then, the original depth image D can be approximated by the prediction $D_{1,p} = E_{1,q} + D_{0,p}$.

The accuracy of the representation can be increased by generating the residual signal $E_2 = D - D_{1,p}$ and iterating the coding process described in this subsection more than once (i.e. performing a refining LBG quantization on the residual signal). After $L-1$ iterations, the input depth can be approximated by the signal

$$D_{L,p} = D_{0,p} + \sum_{l=1}^{L} E_{l,q} = \sum_{l=1}^{L} \sum_{k=0}^{N_{Obj}-1} \sum_{x,y} \overline{e}_{l,k}\, \mathcal{I}\left((x,y) \in \overline{R}_{1,k}\right) \quad (4)$$

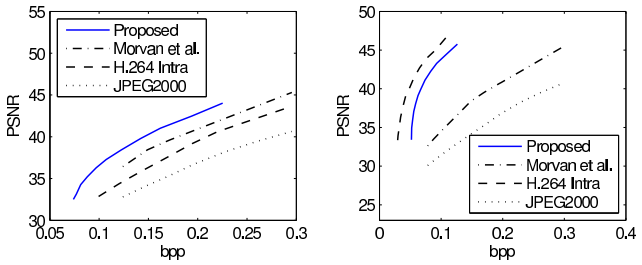
which composes all the masks to generate the final prediction at layer L.

It is possible to notice that at each iteration, the coding process generates an approximation of the original depth that permits distinguishing an increasing number of objects in the scene and models the captured volumes more and more accurately. In Fig. 3(c), the first layer permits distinguishing objects in the foreground from the background, while the second layer permits distinguishing the different foreground objects at different depths (see Fig. 3(d)). Assuming that we want to generate N_L layers, the last residual image E_{N_L-1} is coded using a standard H.264/AVC Intra coder (see Fig. 1).

The choice of the parameters N_{Obj} and N_L is driven by the minimization of the metric

$$L(N_L, N_{Obj}) = D(N_L, N_{Obj}) + \lambda \cdot R(N_L, N_{Obj}) \quad (5)$$

where $D(N_L, N_{Obj})$ is the mean square error (MSE) between the original depth and the reconstructed depth from the binary mask, and $R(N_L, N_{Obj})$ is the bit rate. The parameter λ is set to $5 \cdot 10^{-5}$.



(a) (b)

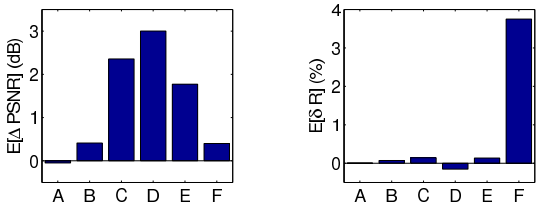
Fig. 4. PSNR vs. bpp for different codecs obtained on the depth images `teddy` (a) and on the first frame of the sequence `breakdancers` (view 0).

III. Experimental results

The proposed method was evaluated on a set of test images with different spatial resolutions available at [11], at [12] and at [13]. The proposed approach was compared with the codec by Morvan *et al.* in [6], the H.264/AVC codec [14], and the JPEG 2000 codec [3]. Figure 4(a) reports the PSNR vs. bit rate plots obtained with the different codecs for the image `teddy` from [12]. The proposed approach with $N_{Obj} = 2$ and $N_L = 2$ permits obtaining an improvement of 3.5 dB over H.264/AVC Intra codec. The residual signal at the final level is coded with different QP values varying within the range $[28, 46]$ with steps of 2. The plots also show that the proposed approach improves the coding performance of the JPEG2000 standards (up to 4.5 dB) and of the coding scheme by Morvan *et al.*, which characterizes the shapes of objects via a quad-tree coding procedure and approximates the surface of the objects via several modeling functions. Plots in Fig. 4(a) show that at 0.15 bpp the progressive-shape coder obtains a PSNR value approximately 1.7 dB higher than that of the quadtree-based coder.

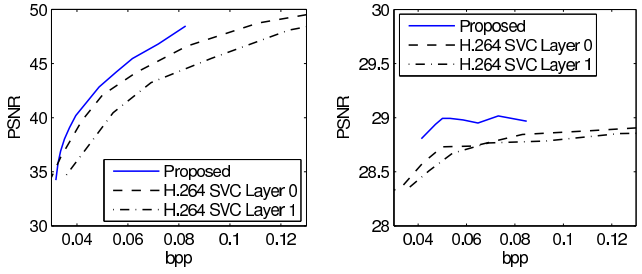
Figure 4(b) reports the PSNR value vs. the bit rate for the first frame of view 0 in the sequence `breakdancers` (available at [11]). The plots show that the best performance is provided by the H.264/AVC Intra coder, and the performance of the proposed solution proves to be close to it. Moreover, the proposed coder permits obtaining a significantly better quality with respect to the JPEG2000 coder and the quadtree-based solution in [6]. The reason for this behavior can be found in the low quality of the depth map for view 0 of the `breakdancers` sequence, which presents a significant noise level added to smooth regions and makes difficult an effective approximation via progressive masks. As for the H.264/AVC Intra codec, the possibility of predicting the input signal along different spatial directions [14] permits identifying the different edges of the coded depth map and efficiently predicting the signal along the borders of the objects.

Additional tests were devoted to analyze the performance of the algorithm on a wide set of test depth images for different quality levels. Several R-D points were generated varying the quantization parameter QP for the final residual signal with different (N_L, N_{Obj}) configurations (found via a Lagrangian optimization). Results were compared with the performance of



(a) (b)

Fig. 5. Average $E[\delta\text{PSNR}]$ (a) and $E[\Delta R]$ (b) for test sets Middlebury 2001 (A), 2003 (B), 2005 (C), 2006 (D) (see [12]), ABW (E) (see [13]), and the view 0 of the sequence `ballet` (F) with only Intra coding (see [11]).



(a) (b)

Fig. 6. PSNR vs. bpp for different codecs obtained on the depth map (a) and the warped view 3 (b) of the sequence `ballet` (coding depth frame 0 of view 4).

the H.264/AVC Intra codec generating two interpolating R-D curves (rate as function of PSNR and PSNR as function of rate) for both coders via the procedure reported in [15]. From these interpolated models, we computed the PSNR differences for a set of bit rates (using the PSNR vs. bit rate curve) and the relative rate increment for a set of PSNR values (using the bit rate vs. PSNR model) as in [16], and results are reported in Fig. 5. Note that a positive value of E[δbpp] denotes a smaller bit rate produced by the proposed coder, while a positive value of E[Δ PSNR] implies that the proposed solution obtains a better quality with respect to the H.264/AVC standard (Intra coding). The data show that the performance of the object-oriented coder permits an average PSNR increase up to 3 dB for the test set Middlebury 2005. However, different gains can be noticed depending on how well planar shapes approximate the objects in the scene (compare Middlebury 2001 and 2006 in Fig. 5(a)).

Final tests were devoted to compare the performance of the proposed strategy with the scalable extension (SVC) of the H.264/AVC Intra coder evaluating the effects of coding on the views reconstructed via a warping operation. In Fig. 6 the plots reports the PSNR vs. bpp for both the reconstructed depth map and the view 3 synthesized by warping view 4 with the reconstructed depth map. It is possible to notice that the PSNR of the warped view is higher for the proposed solution with respect to the other approaches despite the H.264 SVC coder permits obtaining a more flexible quality characterization. As for the computational complexity, the solution based on progressive silhouettes implies a higher computational load because of the cost of the segmentation routine and the LBG strategy. As an example, experimental results on the image `teddy` have shown that the overall complexity of the coding process is 2.5 times higher than the complexity of H.264/AVC Intra coding (measured via the coding time), while the decoding time is approximately the same of H.264 since the computational load for the JBIG decoder is limited. However, in case segmentation masks are already available at the decoder (for depth map estimation), the computational complexity ratio can be reduced to 1.4 with respect to H.264/AVC Intra.

IV. Conclusions

The paper presented a coding scheme for depth images that employs a progressive approximation of the object shapes in the scene. Elements in the scene are coded by a set of masks that progressively refines the volumes of the objects. The final residual is coded using an H.264/AVC Intra coder. This procedure permits a roughly-scalable coding of the depth information, as well as a satisfying coding performance in comparison to the previous solutions proposed in the literature. Future work will be devoted to include temporal prediction in the progressive-shape coder by performing temporally-coherent segmentation and accurate temporal shape prediction.

References

[1] C. Fehn, "3D-TV Using Depth-Image-Based Rendering (DIBR)," in *Proc. of PCS 2004*, San Francisco, CA, USA, Dec. 2004.

[2] R. Krishnamurty, B. B. Chai, H. Tao, and S. Sethuraman, "Compression and transmission of depth maps for image-based rendering," in *Proc. of IEEE ICIP 2001*, Thessaloniki, Greece, Oct. 2001, pp. 828 – 831.

[3] D. Taubman and M. W. Marcellin, *JPEG2000 Image Compression: Fundamentals, Standards and Practice*. Boston, MA, USA: Kluwer, 2002.

[4] G. Peyre and S. Mallat, "Surface compression with geometric bandelets," in *Proc. of ACM SIGGRAPH 2005*, vol. 24, Los Angeles, CA, USA, Jul. 2005, pp. 601–608.

[5] M. N. Do and M. Vetterli, "The contourlet transform: an effective multiresolution image representation," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2091 – 2106, Dec. 2005.

[6] Y. Morvan, D. Farin, and P. H. N. 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*, vol. V, San Antonio, TX, USA, Sep. 2007, pp. 105–108.

[7] P. Zanuttigh and G. M. Cortelazzo, "Compression of Depth Information for 3D Rendering," in *Proc. of 3DTV-CON 2009*, Potsdam, Germany, May 2009, pp. 1–4.

[8] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167 – 181, Sep. 2004.

[9] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA, USA: Kluwer Academic Publisher, 1991.

[10] ISO/IEC JTC 1/SC 29/WG 1 (ITU-T SG 8), "Information Technology - Coded Representation Of Picture And Audio Information - Lossy/Lossless Coding Of Bi-Level Images (JBIG)," Final Committee Draft, 1999.

[11] MSR 3D Video download. [Online]. Available: http://research.microsoft.com/en-us/um/people/sbkang/3dvideodownload

[12] Repository vision.middlebury.edu: Stereo datasets . [Online]. Available: http://vision.middlebury.edu/stereo

[13] Repository of the Range Image Segmentation project. [Online]. Available: http://marathon.csee.usf.edu/range/seg-comp/images.html

[14] T. Wiegand, "Version 3 of H.264/AVC," in 12^{th} *JVT Meeting*, Redmond, WA, USA, Jul. 2004.

[15] G. Bjontegaard, "Calculation of average psnr differences between rd-curves (vceg-m33)," in *Proc. of the* 13^{th} *ITU VCEG Meeting*, Austin, TX, USA, Apr. 2001, vCEG-M33.

[16] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 813–822, Jul. 2005.