Compression of Human Body Sequences Using Graph Wavelet Filter Banks

Ha Q. Nguyen*, Philip A. Chou†, and Yinpeng Chen†
*University of Illinois at Urbana-Champaign, Urbana, IL 61801
hqnguye2@illinois.edu
†Microsoft Research, Redmond, WA 98052
{pachou, yiche}@microsoft.com

***Abstract*—The next step in immersive communication beyond video from a single camera is object-based free viewpoint video, which is the capture and compression of a dynamic object such that it can be reconstructed and viewed from an arbitrary viewpoint. The moving human body is a particularly useful subclass of dynamic object for object-based free viewpoint video relevant to both telepresence and entertainment. In this paper, we compress moving human body sequences by applying recently developed Graph Wavelet Filter Banks to time-varying geometry and color signals living on a mesh representation of the human body. This model-based approach significantly outperforms state-of-the-art coding of the human body represented as ordinary depth plus color video sequences.**

I. Introduction

We are interested in immersive communication using object-based free viewpoint video. In this context, we show how a model of a human body can help with compression.

A moving human body can be represented by a sequence of 3D meshes with a *fixed* and *known* connectivity (or topology). Because the connectivity is fixed, such a sequence can be regarded as a single mesh with time-varying geometry and color, that is, as a *dynamic mesh*. And because its connectivity is known, the connectivity need not be compressed, unlike the general case [1], [2]). Only the geometry and color of the dynamic mesh need be compressed in our problem.

Various methods exist for compressing the geometry of a dynamic mesh [3]–[17]. Among these, one of the most efficient ways to decorrelate the spatial information of a mesh is to use Mesh Wavelet Transforms (MWTs), introduced by Schröder and Sweldens in [18] and subsequently developed in [10], [19]–[21]. These transforms are constructed by the *lifting scheme* [22] with mesh hierarchies generated from various types of subdivision such as Butterfly [23] and Loop [24]. Although MWTs are designed to be localized in the vertex domain, their frequency localization has not been analyzed as in the designs of classical wavelet transforms [25], [26]. Furthermore MWTs depend not only on the topology but also the geometry of the mesh. This nonuniform behavior significantly increases the computation cost as well as hinders the application of the transforms to color data. A new generation of wavelet transforms [27]–[32] has recently been developed for signals living on arbitrary graphs of which meshes are special cases. (For an introduction to the field of signal processing on graphs, interested readers are referred to [33].) Biorthogonal Graph Wavelet Filter Banks (GWFBs), introduced by Narang and Ortega in [32], are among the best designs with almost all the desired properties of a wavelet transform such as perfect reconstruction, critical sampling, compact support, and near orthogonality. In addition, they depend only on the topology of the graphs but not on the signals.

In this paper, we apply GWFBs to the compression of both the geometry and the color of a dynamic mesh representing a moving human body. For the mesh we use a skinned rig model, which can be motion compensated by transforming every frame of the mesh into a rest pose. All the parameters of the model, including animation parameters (i.e., pose at every frame) and rest pose parameters (i.e., geometry and color of the mesh in its rest pose), are estimated from depth and color image data. Multiple frames of data produce a sequence of rest pose meshes that can be considered as time-varying signals living on a graph. GWFBs are then applied to the difference of signals in consecutive frames and the coefficients are quantized and entropy coded. One of our main contributions is the use of a *quad subdivision mesh* that naturally defines a bipartite graph multiresolution, obviating the need for the arbitrary bipartite decomposition in generic GWFBs, one of their main drawbacks. A quad subdivision mesh also enjoys a vertex pattern that we exploit in *context adaptive entropy coding*.

Since the mesh sequence is estimated from color plus depth video, and alternative would be to compress the video directly, and from the decoded video estimate the mesh at the receiver (where it is needed for free viewpoint generation). Experiments show that our geometry encoder significantly outperforms such direct video encoding with a state-of-the-art video encoder (HEVC) on our data sets.

II. Preliminaries

A. Graphs and Meshes

A *graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is characterized by a set of vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$. Without loss of generality, we assume $\mathcal{V} = \{1, 2, \ldots, N\}$ for some integer N. Each element in $\mathcal{E}$ has the form (i, j) for some $i, j \in \mathcal{V}$. We consider only *undirected* graphs, in which $(i, j) \in \mathcal{E}$ implies $(j, i) \in \mathcal{E}$.

A P-sided polygon mesh $\mathcal{M} = (\boldsymbol{V}, \boldsymbol{F})$ is characterized by a *vertex matrix* $\boldsymbol{V} \in \mathbb{R}^{N\times 3}$ for which each row $\boldsymbol{v}_i^T$ is the position of a vertex, and a *face matrix* $\boldsymbol{F} \in \mathbb{N}^{K\times P}$ for which

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

each row $\boldsymbol{f}_k^T$ lists the P vertex indices of a face. The topology of a mesh induces an undirected graph $\mathcal{G}$ with vertices $\mathcal{V} = \{1, 2, \ldots, N\}$ and edges $\{(f_{pk}, f_{(p+1)k})\}_{k=\overline{1,K}, p=\overline{1,P}}$, with the understanding that $f_{(P+1)k} \overset{\Delta}{=} f_{1k}$. When $P = 4$ we say the mesh is a *quad* mesh. The mesh is *colored* if $\boldsymbol{V} \in \mathbb{R}^{N\times 6}$ and each row $(\boldsymbol{v}_i^T, \boldsymbol{c}_i^T)$ of $\boldsymbol{V}$ contains both the position and color of vertex i.

B. Skinned Rig Models

Skinned rig models have long been used to model human characters and other articulated creatures and objects in computer animation [34], [35]. A skinned rig model consists of a set of bones (sometimes called a rig) and a skin attached to the bones by weights. The bones undergo separate rigid transformations to animate the skin. Precisely, the model is characterized by a collection of parameters $(\mathcal{M}, \mathcal{B}, \mathcal{W}, \mathcal{B}(t))$. $\mathcal{M}$ is a colored mesh representing the skin of the object in a pre-defined rest pose. $\mathcal{B} = \{\boldsymbol{G}_b\}_{b=\overline{1,B}}$ is a set of B 4×4 homogeneous coordinate transformation matrices, where each 4×4 transformation matrix $\boldsymbol{G}_b$ represents the rest pose of bone b, by transforming the local coordinate system of bone b in its rest pose into some world coordinate system. $\mathcal{W} = \{w_{ib}\}_{i=\overline{1,N}, b=\overline{1,B}}$ is a set of weights, where each weight w_{ib} represents the association of vertex i (in mesh $\mathcal{M}$) with bone b. For each vertex i, the weights are non-negative and sum to one, i.e., $\sum_b w_{ib} = 1$. Finally, $\mathcal{B}(t) = \{\boldsymbol{G}_b(t)\}_{b=\overline{1,B}}$ is another set of homogeneous coordinate transformation matrices, where each transformation matrix $\boldsymbol{G}_b(t)$ represents the pose of bone b at time t, by transforming the local coordinate system of bone b at time t into the world coordinate system. The parameters $(\mathcal{M}, \mathcal{B}, \mathcal{W})$ are time-invariant, or static, while the parameters $\mathcal{B}(t)$ are time-varying, or dynamic. The dynamic parameters are used to animate the model and are known as *animation parameters*. Together, the model parameters determine the position of each vertex of the mesh at time t as

$$\bar{\boldsymbol{v}}_i(t) = \sum_{b=1}^{B} w_{ib} \boldsymbol{G}_b(t) \boldsymbol{G}_b^{-1} \bar{\boldsymbol{v}}_i, \qquad (1)$$

where $\bar{\boldsymbol{v}} = (\boldsymbol{v}^T, 1)^T$ denotes the 3D position $\boldsymbol{v}$ in homogeneous coordinates. The color $\boldsymbol{c}_i$ of each vertex remains constant.

We estimate all parameters, $(\mathcal{M}, \mathcal{B}, \mathcal{W}, \mathcal{B}(t))$, from a sequence of RGBD images $\{Y(t)\}$ of real human subjects captured with a Microsoft Kinect color plus depth camera. Further, our estimation is *online* in the sense that the parameter estimates are updated after every frame $t = 1, 2, \ldots$. Thus after every frame $t = 1, 2, \ldots$ we obtain estimates $(\mathcal{M}^{(t)}, \mathcal{B}^{(t)}, \mathcal{W}^{(t)})$ of the static parameters as well as estimates $\mathcal{B}^{(t)}(t)$ of the animation parameters. For the purposes of this paper, other details of the estimation technique are unimportant. It suffices to know that the estimates $(\mathcal{M}^{(t)}, \mathcal{B}^{(t)}, \mathcal{W}^{(t)}, \mathcal{B}^{(t)}(t))$ must be quantized, entropy coded, and transmitted, so that the receiver can reproduce the position and color of each vertex at each time t using the synthesis model (1). In this paper, we focus on the transmission of mesh estimates $\mathcal{M}^{(t)}$, specifically the position and color estimates $\boldsymbol{f}_i^{(t)} = (\boldsymbol{v}_i^{(t)}, \boldsymbol{c}_i^{(t)})$ for each vertex i at each frame t, as illustrated in Fig. 1. While it would be possible to stop updating the model after the first frame, the resulting animation would be less faithful to the actual geometry and color over time.

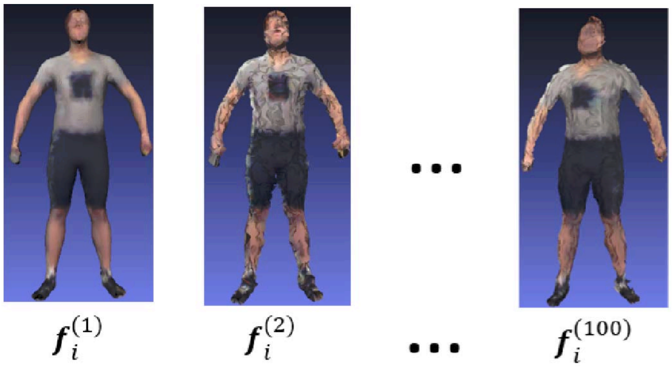


Fig. 1: Example of a sequence of estimated XYZ-RGB signals on a graph induced by a mesh topology. The meshes are displayed only in frontal view.

C. Graph Wavelet Filter Banks

A GWFB provides a vertex-frequency localized transform for graph-indexed signals. Its advantages include critical sampling, perfect reconstruction, and compact support in the vertex domain (hence computational efficiency) [32, Table I]. Like a classical discrete wavelet transform [26], a GWFB is built on a two-channel filter bank as shown in Fig. 2. The downsampling operators $\downarrow \beta_{\mathrm{L}}$ and $\downarrow \beta_{\mathrm{H}}$ respectively keep only the values at lowpass and highpass vertices, defined as the two independent sets of the original bipartite graph on which the signal $\boldsymbol{f}$ is indexed. The four filters are first designed in the graph spectral domain[1] similarly to the maximally-flat design of Cohen-Daubechies-Feauveau [38] and then are inverted into the vertex domain.

The above design is applicable only to bipartite graphs. If the graph is not bipartite we need to decompose it into a sequence of bipartite subgraphs and then apply a filter bank separably to each subgraph. A sequence of κ bipartite subgraphs can be found by coloring the vertices with up to 2^κ

[1] obtained by diagonalizing the graph Laplacian. See [33], [36], [37] for further discussions on the frequency notion of graph signals.

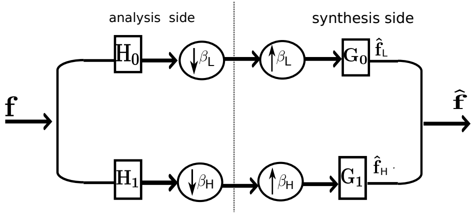


Fig. 2: Block diagram of a two-channel wavelet filter bank on a bipartite graph. Reproduced from [32, Fig. 1].

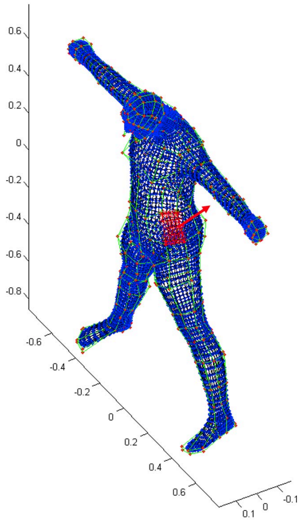


Fig. 3: Subdivision mesh (blue dots connected by black links) generated from a base quad mesh (red dots connected by green links) after 2 levels of subdivion.

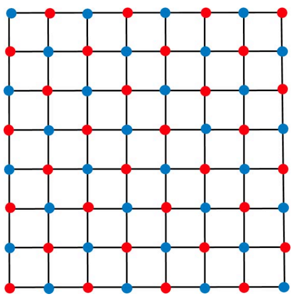


Fig. 4: Regular graph.

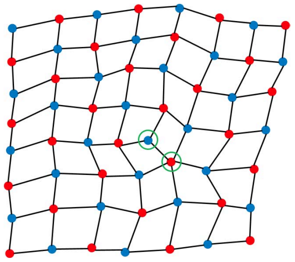


Fig. 5: Irregular graph with irregular vertices being circled.

colors such that no two neighboring vertices have the same color, and then for $k = 1 \ldots, \kappa$ splitting the vertices into two independent sets according to the kth bit of the color index [31].

A multiresolution transform can be obtained by iteratively applying the two-channel filter bank to the lowpass vertices — assuming some rule for reconnecting the lowpass vertices after downsampling. Finding a natural way to reconnect the lowpass vertices is a challenge for GWFBs on general graphs.

III. Proposed Mesh Encoder

In this section we propose a method for coding the sequence of geometry and color estimates $XYZ(t) = [\boldsymbol{v}_i^{(t)}]$ and $YUV(t) = [\boldsymbol{c}_i^{(t)}]$ that come out of our human body model estimator. The sequences of estimates of other parameters, including the animation parameters $\mathcal{B}(t) = [\boldsymbol{G}_b(t)]$, are assumed to be coded separately.

In a traditional video encoder, frame $X(t)$ at time t is predicted by a motion compensated version $\tilde{X}(t)$ of a previous reference frame, and the residual $E(t) = X(t) - \tilde{X}(t)$ is transformed, quantized, and entropy coded. The quantized residual $\hat{E}(t)$ is added to $\tilde{X}(t)$ to obtain the decoded frame $\hat{X}(t) = \hat{E}(t) + \tilde{X}(t)$. The motion estimation (ME) is performed on $\hat{X}(t)$ (i.e., inside the loop) and the motion vectors are coded separately.

We propose to use an analogous structure to code the sequences $XYZ(t)$ and $YUV(t)$, but without motion estimation and compensation in the loop. Motion estimation and compensation are provided outside the loop by estimating $\mathcal{B}(t)$ as described in Sec. II-B and using it to normalize $XYZ(t)$ to the rest pose. All geometry and color components are coded independently. Of course, the signals $XYZ(t)$ and $YUV(t)$ no longer live on a regular grid, as in Fig. 4, but rather on an irregular graph, as in Fig. 5. Thus the important part of our proposal is how to adapt the transform and the entropy coder to the graph structure. These are covered in the following two subsections.

A. Multiresolution GWFBs

The time-invariant graph of human bodies, on which the GWFBs are applied, is first created by applying a mesh subdivision (such as Catmull-Clark subdivision [39]) on a base quad mesh as shown in Fig. 3. We claim that any quad subdivision mesh provides a natural sequence of bipartite graphs for multiresolution GWFBs, as illustrated in Fig. 6 for $L = 2$ levels of subdivision. In general, let $\mathcal{G}_{2\ell}$ be the graph associated with a level-ℓ quad subdivision mesh with vertex set $\bar{\mathcal{V}}_{2\ell}$, for $\ell \geq 0$. The next level of subdivision divides each quad into a 2×2 grid. Therefore, the vertex set $\bar{\mathcal{V}}_{2(\ell+1)}$ is simply a union of $\bar{\mathcal{V}}_{2\ell}$ with two sets of new vertices $\mathcal{V}_{2\ell+1}$ and $\mathcal{V}_{2\ell+2}$, where $\mathcal{V}_{2\ell+1}$ contains all the *face points* that do not connect to any vertices in $\bar{\mathcal{V}}_{2\ell}$, and $\mathcal{V}_{2\ell+2}$ contains all the *edge points* that connect only to vertices in either $\bar{\mathcal{V}}_{2\ell}$ or $\mathcal{V}_{2\ell+1}$. This implies that $\mathcal{G}_{2(\ell+1)}$ is a bipartite graph with two independent sets $\mathcal{V}_{2\ell+2}$ and $\bar{\mathcal{V}}_{2\ell+1} \triangleq \bar{\mathcal{V}}_{2\ell} \cup \mathcal{V}_{2\ell+1}$. Let $\mathcal{G}_{2\ell+1}$ be the graph with vertices $\bar{\mathcal{V}}_{2\ell+1}$ and edges connecting every four vertices of a face in $\bar{\mathcal{V}}_{2\ell}$ to the corresponding face vertex in $\mathcal{V}_{2\ell+1}$. This construction implies that $\mathcal{G}_{2\ell+1}$ is a also a bipartite graph with two independent sets $\mathcal{V}_{2\ell+1}$ and $\bar{\mathcal{V}}_{2\ell}$. In this way we have constructed a multiresolution sequence of bipartite graphs,

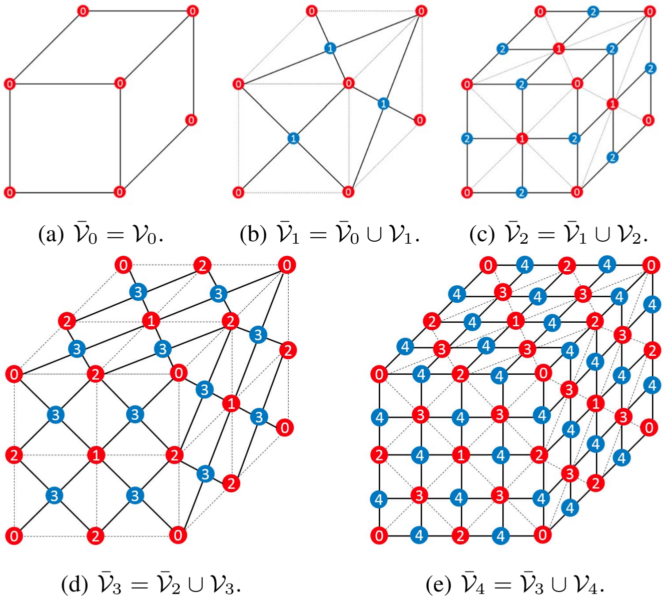
$$\mathcal{G}_0 \subset \mathcal{G}_1 \subset \cdots \subset \mathcal{G}_{2L-2} \subset \mathcal{G}_{2L-1} \subset \mathcal{G}_{2L},$$

where $\mathcal{G}_{\ell-1} \subset \mathcal{G}_\ell$ means $\mathcal{V}_{\ell-1}$ is an independent subset of $\mathcal{G}_\ell$, suitable for multiresolution GWFBs.

B. Context Adaptive Entropy Coder

After the geometry and color components are transformed using a multiresolution GWFB, the coefficients are independently quantized using a uniform scalar mid-rise quantizer with step size Δ. That is, a coefficient x is quantized to index $k_x = Q(x) = \lfloor x/\Delta \rfloor$, while an index k is dequantized to $\hat{x}_k = Q^{-1}(k) = \Delta \cdot (k + 1/2)$. The indices are entropy coded using a large-alphabet arithmetic coder assuming a Laplacian distribution on x. As a baseline, the parameter of the distribution, λ_i, is constant for all coefficients x_i in subband $\mathcal{V}_n$. However, the parameter can also be adapted based on the context of x_i, as follows. Let $\mathcal{N}_i$ be the set of neighbors of vertex i in $\bar{\mathcal{V}}_n$. Let $\boldsymbol{p}_n$ denote the pattern of neighbors of vertices in subband n, meaning each vertex in $\mathcal{V}_n$ is connected to $p_{n,1}$ vertices in $\mathcal{V}_{n-1}$, and $p_{n,2}$ vertices in $\mathcal{V}_{n-2}$, and so on. In particular, from Fig. 6 we can define $\boldsymbol{p}_n$ as

$$\boldsymbol{p}_n = \begin{cases} 4, & n = 1 \\ [2,\ 2]^T, & n = 2 \\ [2,\ 1,\ 1], & n = 3 \\ [2,\ 1], & n \geq 4 \end{cases} \qquad (2)$$



(a) $\bar{\mathcal{V}}_0 = \mathcal{V}_0$. (b) $\bar{\mathcal{V}}_1 = \bar{\mathcal{V}}_0 \cup \mathcal{V}_1$. (c) $\bar{\mathcal{V}}_2 = \bar{\mathcal{V}}_1 \cup \mathcal{V}_2$.

(d) $\bar{\mathcal{V}}_3 = \bar{\mathcal{V}}_2 \cup \mathcal{V}_3$. (e) $\bar{\mathcal{V}}_4 = \bar{\mathcal{V}}_3 \cup \mathcal{V}_4$.

Fig. 6: Construction of multiresolution bipartite graphs.

Let also $\boldsymbol{\theta}_n$ be a meta-parameter vector in $\mathbb{R}^{|\boldsymbol{p}_n|+1}$. We linearly predict the parameter λ_i from its neighbors in $\mathcal{N}_i$ by

$$\lambda_i = \boldsymbol{a}_i^T \boldsymbol{\theta}_n \triangleq [1, \boldsymbol{b}_i^T] \cdot \boldsymbol{\theta}_n, \tag{3}$$

where, for $j = 1, 2, \dots, |\boldsymbol{p}_n|$,

$$b_{i,j} = \frac{p_{n,j}}{\sum_{k \in \mathcal{N}_i \cap \mathcal{V}_{n-j}} |x_k|}. \tag{4}$$

The total code length of subband n is now a function of $\boldsymbol{\theta}_n$

$$\ell(\boldsymbol{\theta}_n) = -\sum_{i \in \mathcal{V}_n} \log\left(e^{-|k_{x_i}|\Delta \boldsymbol{a}_i^T \boldsymbol{\theta}_n} - e^{-(|k_{x_i}|+1)\Delta \boldsymbol{a}_i^T \boldsymbol{\theta}_n}\right).$$

The parameter vector $\boldsymbol{\theta}_n^*$ that minimizes $\ell(\boldsymbol{\theta}_n)$ can be found by convex programming.

IV. Experimental Results

We evaluate our geometry and color encoder on the 100-frame mesh sequence depicted in Fig. 1. The distortion-rate performance of the geometry encoder is shown in Fig. 7. (We do not show the performance of the color encoder for lack of space, but the results are similar.) We turn on various coding tools in sequence, starting with I-frame encoding, which independently codes each component of each pixel using an average parameter $\bar{\lambda}$ for entropy coding. Turning on Temporal Prediction (TP) codes the residual using the past frame as a predictor. Turning on Motion Compensation (MC) normalizes every frame to the rest pose (outside of the loop). Turning on the GWFBs transforms the residual before quantization, using the `GraphBior`(3,3) filters from [32] on our multiresolution structure. Turning on Context Adaptive Entropy Coding (CAEC) uses the per-pixel adaptive parameter λ_i for entropy coding. Also in Fig. 7 we plot the rate-distortion performance of the traditional video coding approach using state-of-the-art video codec HEVC (version 3.2), by inverting the order of model estimation and encoding, as illustrated in Fig. 8. Having the advantage of a model, our scheme outperforms HEVC, even with only a single camera. Moreover, our approach scales well when multiple cameras are used to estimate the model.

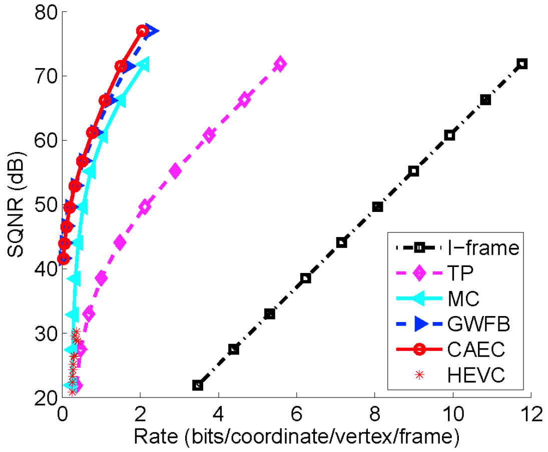
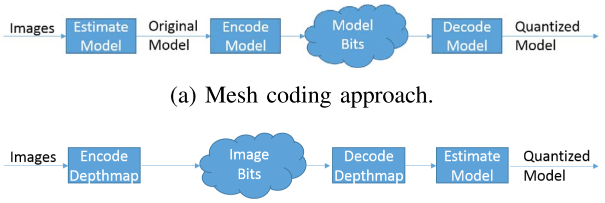


Fig. 7: Rate-distortion performance of geometry encoder. SQNR is the ratio of total signal power to total quantization error, where the totals are taken over x, y, and z components.



(a) Mesh coding approach.

(b) Video coding approach.

Fig. 8: Two different coding schemes. The mesh coding approach transmits the coded model whereas the video coding approach transmits the coded depth maps and color images.

V. Conclusion

The recent availability of low-cost high-quality color plus depth cameras is enabling practical object-based free viewpoint video for immersive communication. In this paper we have shown that a particularly effective way to compress dynamic 3D objects for this purpose is to model the object using a dynamic colored mesh based on a quad subdivision and to couple this with near-orthogonal GWFBs and CAEC, for both geometry and color. The quad subdivision structure solves a number of problems that both GWFBs and CAEC have on arbitrary graphs. We believe this will be the major paradigm for dynamic mesh compression going forward.

Acknowledgments

The authors thank Sunil Narang for his GraphBior wavelet toolbox, Josiah Manson for creating our base quad subdivision mesh, and Minh Do for valuable discussions.

References

[1] J. Peng, C.-S. Kim, and C.-C. J. Kuo, "Technologies for 3D mesh compression: A survey," *J. Vis. Commun. Image R.*, vol. 16, no. 6, pp. 688–733, Dec. 2005.

[2] J. Rossignac, "Edgebreaker: Connectivity compression for triangle meshes," *IEEE Trans. Vis. Comput. Graphics*, vol. 5, no. 1, pp. 47–61, Jan.-Mar. 1999.

[3] J. E. Lengyel, "Compression of time-dependent geometry," in *Proc. 1999 ACM Symp. Interactive 3D Graphics (I3D '99)*, 1999, pp. 89–95.

[4] J.-H. Ahn, C.-S. Kim, C.-C. Kuo, and Y.-S. Ho, "Motion-compensated compression of 3D animation models," *IEE Electron. Lett.*, vol. 37, no. 24, pp. 1445–1446, Nov. 2001.

[5] J.-H. Yang, C.-S. Kim, and S.-U. Lee, "Compression of 3D triangle mesh sequences based on vertex-wise motion vector prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 12, pp. 1178–1184, Feb. 2002.

[6] L. Ibarria and J. R. Georgia, "Dynapack: space-time compression of the 3D animations of triangle meshes with fixed connectivity," in *Proc. 2003 ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA '03)*, 2003, pp. 126–135.

[7] M. Alexa and W. Müller, "Representing animations by principal components," *Comput. Graph. Forum*, vol. 19, no. 3, pp. 411–418, Sep. 2000.

[8] H. M. Briceño, P. V. Sander, L. McMillan, S. Gortler, and H. Hoppe, "Geometry videos: a new representation for 3D animations," in *Proc. 2003 ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA '03)*, 2003, pp. 136–146.

[9] J. Zhang and C. Owen, "Octree-based animated geometry compression," in *Proc. Data Compression Conf. (DCC 2004)*, Mar. 2004, pp. 508–517.

[10] I. Guskov and A. Khodakovsky, "Wavelet compression of parametrically coherent mesh sequences," in *Proc. 2004 ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA '04)*, 2004, pp. 183–192.

[11] F. Payan and M. Antonini, "Wavelet-based compression of 3D mesh sequences," in *Proc. IEEE ACIDCA-ICMI2005*, Tozeur, Tunisia, 2005.

[12] K. Müller, A. Smolic, M. Kautzner, and P. Eisert, "Predictive compression of dynamic 3D meshes," in *Proc. 2005 IEEE Int. Conf. Image Process. (ICIP 2005)*, Sep. 2005, pp. 1–4.

[13] K. Müller, A. Smolic, M. Kautzner, and T. Wiegand, "Rate-distortion optimization in dynamic mesh compression," in *Proc. 2006 IEEE Int. Conf. Image Process. (ICIP 2006)*, Oct. 2006, pp. 533–536.

[14] N. Stefanoski and J. Ostermann, "Connectivity-guided predictive compression of dynamic 3D meshes," in *Proc. 2006 IEEE Int. Conf. Image Process. (ICIP 2006)*, Oct. 2006, pp. 2973–2976.

[15] L. Váša and V. Skala, "CODDYAC: Connectivity driven dynamic mesh compression," in *Proc. 2007 IEEE 3DTV Conf.*, May 2007, pp. 1–4.

[16] J. W. Cho, S. Valette, J. H. Park, H. Y. Jung, and R. Prost, "3D mesh sequence compression using wavelet-based multi-resolution analysis," *Appl. Math. Comput.*, vol. 216, no. 2, pp. 410–425, Mar. 2010.

[17] K. Jafari and F. Dupont, "Compression of 3D mesh sequences based on an adaptive 3D wavelet transformation," in *Proc. SPIE Three-Dimensional Image Process. Appl.*, Jan. 2010, pp. 1–6.

[18] P. Schröder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on a sphere," in *Proc. 22nd Ann. Conf. Comput. Graphics Interactive Techniques (SIGGRAPH'95)*, 1995, pp. 161–172.

[19] I. Guskov, W. Sweldens, and P. Schröder, "Multiresolution signal processing for meshes," in *Proc. 26th Ann. Conf. Comput. Graphics Interactive Techniques (SIGGRAPH'99)*, 1999, pp. 325–334.

[20] A. Khodakovsky, P. Schröder, and W. Sweldens, "Progressive geometry compression," in *Proc. 27th Ann. Conf. Comput. Graphics Interactive Techniques (SIGGRAPH'00)*, 2000, pp. 271–278.

[21] A. Khodakovsky and I. Guskov, "Compression of normal meshes," in *Geometric Modeling for Scientific Visualization*. Springer-Verlag, 2002, pp. 189–206.

[22] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, Apr. 1996.

[23] N. Dyn, D. Levin, and J. A. Gregory, "A butterfly subdivision scheme for surface interpolation with tension control," *ACM Trans. Graphics*, vol. 9, no. 2, pp. 160–169, Apr. 1990.

[24] C. Loop, "Smooth subdivision surfaces based on triangles," Master's thesis, University of Utah, Department of Mathematics, 1987.

[25] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia: Society for Industrial and Applied Mathematics, 1992.

[26] M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*. Prentice Hall, 1995.

[27] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *Proc. 2003 IEEE INFOCOM*, vol. 3, Mar. 2003, pp. 1848–1857.

[28] R. R. Coifman and M. Maggioni, "Diffusion wavelets," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 53–94, Jul. 2006.

[29] W. Wang and K. Ramchandran, "Random multiresolution representations for arbitrary sensor network graphs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2006)*, vol. 4, May 2006, pp. 161–164.

[30] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, Mar. 2011.

[31] S. K. Narang and A. Ortega, "Perfect reconstruction two-channel wavelet filter banks for graph structured data," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2786–2799, Jun. 2012.

[32] ——, "Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4673–4685, Oct. 2013.

[33] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[34] J. P. Lewis, M. Cordner, and N. Fong, "Pose space deformations: A unified approach to shape interpolation and skeleton-driven deformation," in *Proc. 27th Ann. Conf. Comput. Graphics Interactive Techniques (SIGGRAPH'00)*, New Orleans, LA, Jul. 2000, pp. 165–172.

[35] E. Allen, K. L. Murdoch, J. Fong, and A. G. Sidwell, *Body Language: Advanced 3D Character Rigging*. Indianapolis, IN: Wiley, 2008.

[36] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[37] A. Agaskar and Y. M. Lu, "A spectral graph uncertainty principle," *IEEE Trans. Inform. Theory*, vol. 59, no. 7, pp. 4338–4356, Jul. 2013.

[38] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 45, no. 5, pp. 485–560, Jun. 1992.

[39] E. Catmull and J. Clark, "Recursively generated b-spline surfaces on arbitrary topological meshes," *Comput. Aided Design*, vol. 10, no. 6, pp. 350–355, Nov. 1978.