DISTRIBUTED 3D DYNAMIC MESH CODING

M. Oguz Bici and Gozde Bozdagi Akar

Middle East Technical University
Electrical and Electronics Engineering Department
06531, Ankara, Turkey
{mobici, bozdagi}@eee.metu.edu.tr, http://mmrg.eee.metu.edu.tr

ABSTRACT

In this paper, we propose a distributed 3D dynamic mesh coding system. The system is based on Slepian and Wolf's and Wyner and Ziv's information-theoretic results. Our system extends the ideas in distributed video coding to 3D dynamic meshes with constant connectivity. The connectivity of the sequence and key frames are encoded and decoded by a conventional static mesh coder. The Wyner-Ziv frames are encoded independent of key frames but decoded jointly with decoded key frames. The joint decoding is performed by the low density parity check codes and the side information generated by linear interpolation of decoded key frames. Experimental results show that better rate-distortion performance is obtained compared to encoding each frame by a static mesh coder.

Index Terms— Wyner-Ziv, distributed source coding, dynamic meshes

1. INTRODUCTION

Distributed coding is a new paradigm based on Slepian and Wolf's and Wyner and Ziv's information-theoretic results from the 1970s [1] [2]. Especially in the field of Distributed Video Coding (DVC), considerable research has been conducted [3] [4] [5] [6]. The main motivation behind DVC is that the complex encoder/simple decoder structure of a conventional video coding scheme is replaced by simple encoder/complex decoder structure. This new structure is better suited where the computational power, memory and/or battery are scarce at the encoder, such as visual sensor networks, disposable video cameras or multi-view image acquisition systems. Another important property of distributed coding is that it provides error resilience since the frames are coded independently.

Theorems of distributed coding can be summarized as follows: Consider two statistically dependent discrete signals, X and Y, which are compressed using two independent encoders but are decoded by a joint decoder. The Slepian-Wolf Theorem on distributed source coding states that, even if the encoders are independent, the achievable rate region for probability of decoding error to approach zero is $R_X \geqslant H(X|Y)$, $R_Y \geqslant H(Y|X)$ and $R_X + R_Y \geqslant H(X,Y)$ [1]. The counterpart of this theorem for lossy source coding is Wyner and Ziv's work on source coding with side information [2]. Let X and Y be statistically dependent Gaussian random processes, and let Y be known as side information for encoding X. Wyner and Ziv showed that the conditional Rate-Mean Squared Error Distortion function for X is the same whether the side information Y is available only at the decoder, or both at the encoder and the decoder. Lossless distributed source coding is commonly referred as Slepian-Wolf coding and lossy source coding with side information at the decoder is commonly referred as Wyner-Ziv coding.

Another topic which is becoming more and more popular nowadays is 3D dynamic mesh compression. With increasing level of realism, 3D dynamic meshes become more important in areas like computer games, 3DTV, animation movies, physical simulations etc. A common representation for dynamic meshes is series of static triangular meshes, called mesh frames or simply frames. In this work, we deal with time-consistent dynamic meshes in which the connectivity of each frame stays constant.

Recently, several approaches for compression of dynamic 3D meshes have been presented. Karni and Gotsman [7] and Sattler et al. [8] represent dynamic meshes using Principal Component Analysis (PCA) to reduce the amount of data. In this mesh representation, the first approach uses linear prediction to exploit remaining temporal coherence, while in the second paper mesh segmentation is applied in order to exploit the coherence of rigid body parts. Guskov et al. [9] and Payan et al. [10] propose wavelet-based approaches for compression. While Guskov et al. apply the wavelet transform for each frame separately, exploiting later the temporal coherence between wavelet coefficients, Payan et al. apply the wavelet transform in temporal direction on vertex trajectories and use a model-based entropy coder for compression. Recently, Muller et al. [11] presented an approach which exploits the coherence of motion vectors between consecutive frames, using a clustering algorithm based on octrees. Yang et al. [12] and Ibarria and Rossignac [13] presented vertex traversal based compression algorithms. In the first paper a parallelogram-like prediction rule is applied, while in the second paper motion vector averaging is employed to exploit local inter and intra frame coherence between vertex locations. In [14], mesh connectivity is used to determine the order of compression of vertex locations within a frame. Compression is performed in a frame to frame fashion using only the last decoded frame and the partly decoded current frame for prediction. Following the predictive coding paradigm, local temporal and local spatial dependencies between vertex locations are exploited. In addition, a novel angle preserving predictor is presented. In [15], the same author improves the work in [14] using a layered predictive approach and a non-linear predictor.

In all of these 3D dynamic mesh compression studies, the aim is to reduce redundancy in the encoder as much as possible, which might yield in complex encoders. In this work, we propose a distributed coding framework for 3D dynamic meshes with constant connectivity. We encode the frames independently of each other but try to exploit redundancies for compression in the decoder. In this way, the encoder is very light in complexity. In addition, since the encoded frames do not depend on each other, possible mismatch is avoided and error resilience is achieved.

The rest of the paper is organized as follows. In Section 2, we overview the system. In Sections 3 and 4, we provide the details of encoder and decoder respectively. In Section 5, we present the experimental results and finally we conclude in Section 6.

2. SYSTEM OVERVIEW

An overview of the codec structure can be seen in Fig. 1. In the encoder, the mesh frames are split into sets. The odd indexed frames constitute the key frames and the even indexed frames constitute the Wyner-Ziv frames. The key frames are coded and decoded with a conventional static 3D mesh coder. The Wyner-Ziv frames are coded independently of key frames but decoded using the decoded key frame information. Details of encoding and decoding processes are given in the following sections. The connectivity of the mesh sequence is encoded and transmitted only once since it does not vary with respect to time.

3. ENCODER

The encoder deals with three different data types: connectivity, geometry of key frames and geometry of Wyner-Ziv frames. The connectivity of whole mesh sequence is coded with an efficient lossless coder. We use Touma and Gotsman's TG coder [16].

The geometry of key frames can be coded with any efficient 3D static mesh coders. In our work, we use the TG coder to encode the geometry data of key frames as well. The TG coder is based on vertex traversal, prediction of vertex coordinates and delta values coding. The number of bits to quantize the vertex coordinates is given as an input to the encoder.

To encode geometry of Wyner-Ziv frames, we treat each coordinate axis, x, y and z, separately. Each coordinate value of each vertex is quantized with a user specified quantization level. After the quantization, each resulting bitplane is extracted and the bits are fed to the LDPCA (Low Density Parity Check Accumulate) coder bitplane by bitplane. LDPCA coder is a rate-compatible LDPC coder proposed by [17].

An LDPCA encoder consists of an LDPC syndrome-former concatenated with an accumulator. For each bit plane, syndrome bits are created using the LDPC code and accumulated modulo 2 which produces the accumulated syndrome. The accumulated syndromes are stored in a buffer and the systematic part is discarded. If decoder requests a subset of accumulated syndromes which corresponds to a rate in LDPCA coder, the encoder sends the requested accumulated syndromes. Since the the number of bits requested by the decoder is less than the original data size, compression is achieved.

4. DECODER

The connectivity of the mesh sequence is decoded by TG decoder once at the beginning and then used for each decoded frame. Geometry of the key frames are decoded by the TG decoder as well.

In order to decode Wyner-Ziv frames, the accumulated syndromes from the encoder and the decoded key frames are used. Since the systematic output of LDPC coder is discarded in encoder, an estimation of the systematic part, or the original Wyner-Ziv frame, should be estimated in the decoder. This estimation of the original Wyner-Ziv frame is generated as side information in the decoder. The available decoded key frames are used to generate the side information. Using accumulated syndromes from the encoder and the side information, the bitplanes of the Wyner-Ziv frame are decoded. The final reconstruction of the frame is performed using the the decoded bitplanes and the side information. The details of the blocks in the decoder are presented in the following subsections.

4.1. Side Information Generation

Side information for a Wyner-Ziv frame is generated using the decoded previous and next key frames. Since the decoded connectivity information is same for all frames, we know the position of a vertex in the next and previous frames. In our work, we estimate the vertex locations in side information as the average of corresponding vertex locations in the previous and next decoded key frames.

Compression performance of the Wyner-Ziv frames are highly correlated with the quality of side information. As the side information becomes closer to original Wyner-Ziv frame, LDPC coder needs fewer accumulated syndromes to correct the errors between them. Hence a better compression rate is achieved. In this work, we employ a simple averaging method to generate side information. Any other sophisticated method which approximates original Wyner-Ziv frame would result a better compression ratio.

4.2. Slepian-Wolf Codec

The Slepian-Wolf encoder consists of the LDPC coder which is described in the previous section. The Slepian-Wolf decoder tries to recover original bits for each bitplane. The side information vertex values are quantized and the extracted bitplanes are fed to Slepian-Wolf decoder together with the accumulated syndromes from the encoder.

The LDPC decoder assumes a virtual Binary Symmetric Channel (BSC) for each bitplane. In this virtual channel, the systematic bits of a bitplane are assumed to be received as side information bitplanes. The accumulated syndromes are assumed to be received with no error. The cross-over probability value of the BSC is calculated by modeling the difference between original and side vertex values with a Laplace distribution as in [18].

Initially, the Slepian-Wolf decoder requests a small amount of accumulated syndromes from the encoder. If the LDPC decoder fails to decode successfully, it requests more accumulated syndromes and the encoder sends from the buffer until decoder decodes successfully.

4.3. Reconstruction

We use the reconstruction function proposed in [18]. After successfully decoding each bitplane, we know that a vertex coordinate value must be within its decoded quantization cell. In the reconstruction function employed, if a side information vertex coordinate value is within the corresponding quantization cell, then the reconstruction value is chosen as the side information value. On the other hand, if the side information value is outside the corresponding cell, then the reconstruction value is chosen as the closest boundary value of the cell.

5. EXPERIMENTAL RESULTS

In the experiments, we used the *Chicken crossing* sequence which is composed of 400 frames. Snapshots of several frames can be seen in Fig. 2. Each mesh frame is composed of 5664 triangles and 3030 vertices. To measure objective distortion, we use the error metric which is defined in Karni and Gotsman's work [7] as it is a widely used metric in the literature. We denote this error by *KG Error*.

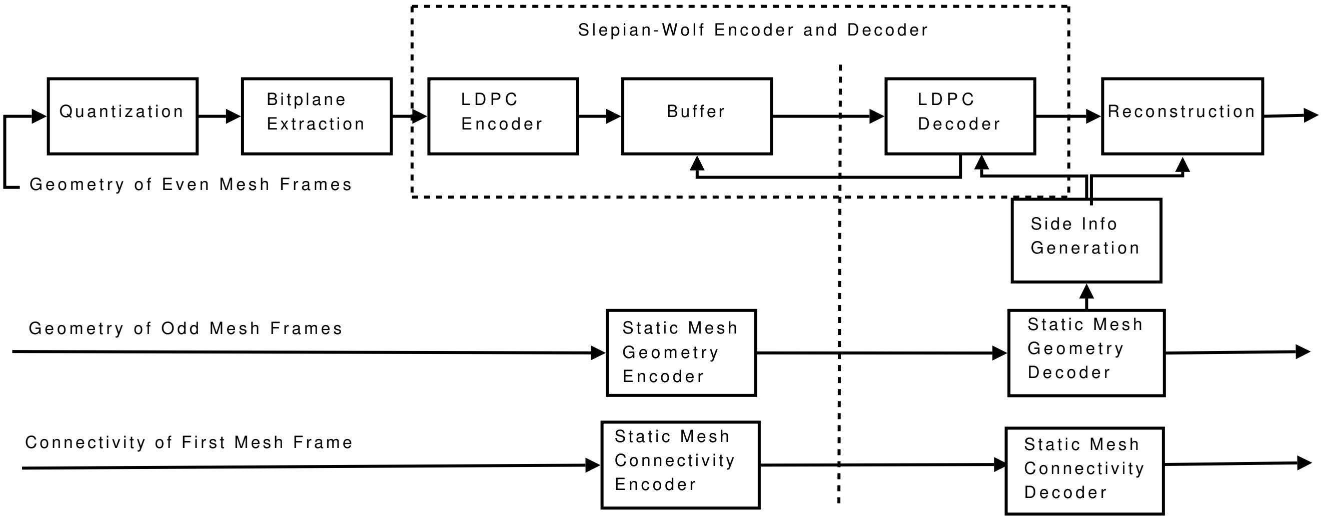

Fig. 1. The distributed 3D dynamic mesh coding encoder/decoder structure

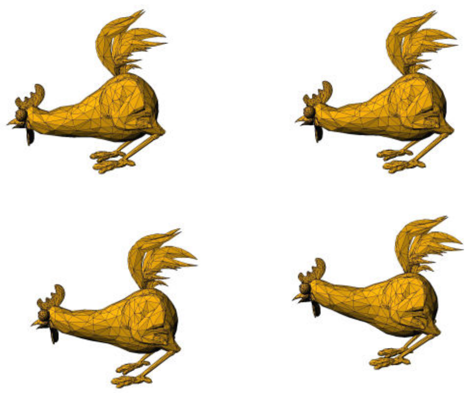

Fig. 2. Several frames of Chicken Crossing sequence.

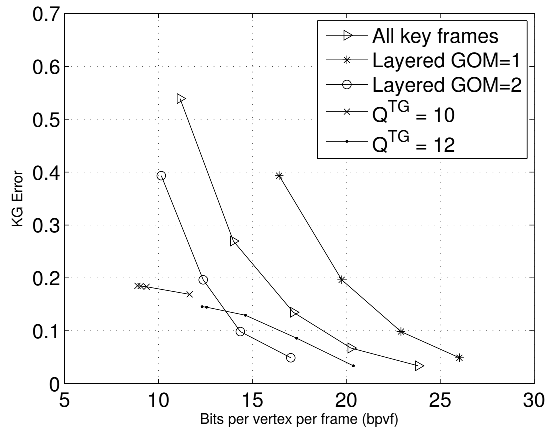

Fig. 3. Rate-Distortion curve of proposed codec and comparison with the codec coding all frames as key frames

TG coder, which compresses the key frames, initially quantizes the coordinates with a given number of bits. We denote this number of bits by Q^{TG}. In this way, bitrate can be varied by modifying Q^{TG}, which also affects Wyner-Ziv frames performance because the key frames are coded by TG coder and the side information is generated by the decoded key frames. Therefore, having a larger Q^{TG} value yields a better side information prediction and better Wyner-Ziv performance; on the other hand it spends more bits and increases the bitrate. Another way to vary bitrate of Wyner-Ziv frames is to vary the number of quantization levels before the Wyner-Ziv frames are quantized and fed to LDPC coder. We denote the number of bits for these quantization levels as Q^{WZ}.

In Fig. 3 we see the rate-distortion performance of our coder for different Q^{TG} and Q^{WZ} values, denoted as Q^{TG}*=10* and Q^{TG}*=12*. We also see the comparison with the codec that compresses each frame as a key frame with TG coder, denoted as *All key frames*. The remaining two curves in the figure are obtained from the work in [15], which reports very efficient compression performance in the literature. The two curves correspond to results obtained when Group of Meshes (GOM) is equal to 1 and 2 and denoted as *Layered GOM=1* and *Layered GOM=2*. The GOM parameter defines the interval between two key frames which are compressed without any prediction from other frames. For example GOM=1 corresponds to compressing each frame as key frame. As the GOM parameter increases, the number of predictive coded frames and consequently compression efficiency increase. We observe that the proposed codec outperforms the codecs which compress each frame as a key frame and performs similar to *Layered GOM=2*. In Fig. 4 we see the KG errors of individual frames for side information and reconstructed Wyner-Ziv frames with $Q^{WZ} = 8$ and $Q^{WZ} = 10$.

6. CONCLUSIONS AND FUTURE WORK

We propose a distributed 3D dynamic mesh coding system and presente its rate-distortion performance. We observe that better performance is achieved compared to coding each frame with a static mesh coder. We note that this is an initial work and there is room for improvements. We employ a very simple averaging method to gener-

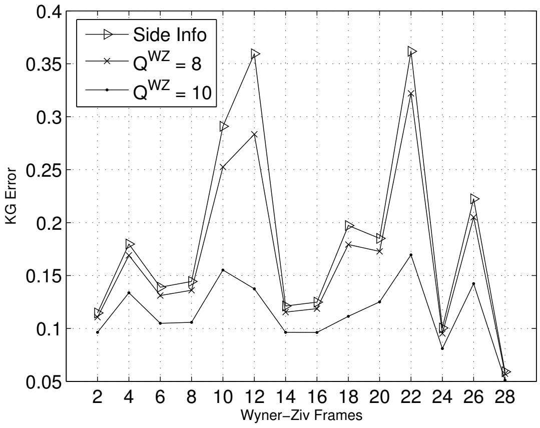

Fig. 4. KG Errors of side information and reconstructions with $Q^{WZ} = 8$ and $Q^{WZ} = 10$ for the Wyner-Ziv frames in first 29 frames.

ate side information. Employing more sophisticated estimators with better side information would result better performance. Another important point is that we exploit only temporal redundancy.

In future works, we plan to exploit redundancy in space domain as well to achieve better performance. We also plan to analyze error resilience performance of our coder and compare with error resilience of other 3D dynamic mesh coding approaches.

7. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Community's ICT programme in the context of the Seventh Framework Programme (FP7/2007-2011) under grant agreement n 216503. It is also partially supported by The Scientific and Technological Research Council of Turkey (TUBITAK). The text reflects only the authors views and the European Community or other project partners are not liable for any use that may be made of the information contained herein. The chicken character was created by Andrew Glassner, Tom McClure, Scott Benza, and Mark Van Langeveld. This short sequence of connectivity and vertex position data is distributed solely for the purpose of comparison of geometry compression techniques.

8. REFERENCES

[1] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *Information Theory, IEEE Transactions on*, vol. 19, no. 4, pp. 471–480, 1973.

[2] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *Information Theory, IEEE Transactions on*, vol. 22, no. 1, pp. 1–10, 1976.

[3] B. Girod, AM Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, 2005.

[4] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The Discover Codec: Architecture, Techniques and Evaluation," *Picture Coding Symposium, Lisbon, Portugal, November*, vol. 6, pp. 14496–10, 2007.

[5] R. Puri, A. Majumbar, P. Ishwar, and K. Ramchandran, "Distributed video coding in wireless sensor networks," *Signal Processing Magazine, IEEE*, vol. 23, no. 4, pp. 94–106, 2006.

[6] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," *Proc. Allerton Conference on Communication, Control, and Computing*, 2002.

[7] Z. Karni and C. Gotsman, "Compression of soft-body animation sequences," *Computers & Graphics*, vol. 28, no. 1, pp. 25–34, 2004.

[8] M. Sattler, R. Sarlette, and R. Klein, "Simple and efficient compression of animation sequences," *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 209–217, 2005.

[9] Igor Guskov and Andrei Khodakovsky, "Wavelet compression of parametrically coherent mesh sequences," in *SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Aire-la-Ville, Switzerland, Switzerland, 2004, pp. 183–192, Eurographics Association.

[10] F. Payan and M. Antonini, "Wavelet-based compression of 3D mesh sequences," *Proceedings of IEEE ACIDCA-ICMI2005*, 2005.

[11] K. Muller, A. Smolic, M. Kautzner, P. Eisert, and T. Wiegand, "Predictive Compression of Dynamic 3D Meshes," *Image Processing, 2005 IEEE International Conference on*, vol. 1, 2005.

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

[13] L. Ibarria and J. Rossignac, "Dynapack: space-time compression of the 3D animations of triangle meshes with fixed connectivity," *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pp. 126–135, 2003.

[14] N. Stefanoski and J. Ostermann, "Connectivity-Guided Predictive Compression of Dynamic 3D Meshes," *Image Processing, 2006 IEEE International Conference on*, pp. 2973–2976, 2006.

[15] N. Stefanoski, P. Klie, X. Liu, and J. Ostermann, "Layered Predictive Coding of Time-Consistent Dynamic 3D Meshes using a Non-Linear Predictor," *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 5, 2007.

[16] C. Touma and C. Gotsman, "Triangle mesh compression," in *Proc. Graphics Interface*, Vancouver, BC, Canada, Jun. 1998.

[17] D. Varodayan, A. Aaron, and B. Girod, "Rate-adaptive codes for distributed source coding," *Signal Processing*, vol. 86, no. 11, pp. 3123–3130, 2006.

[18] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," *Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on*, vol. 1, 2002.