AN EFFICIENT GEOMETRY COMPRESSION METHOD FOR 3D OBJECTS IN THE SPHERICAL COORDINATE SYSTEM

Jeong-Hwan Ahn and Yo-Sung Ho

Kwangju Institute of Science and Technology
1 Oryong-Dong Puk-Gu, Kwangju, 500-712, Korea
Email : {jhahn,hoyo}@kjist.ac.kr

ABSTRACT

In this paper, we propose a new geometry coding scheme for 3D objects. The prediction error of each vertex position along a vertex spanning tree is obtained in the Cartesian coordinate system (x,y,z), and transformed into the spherical coordinate system (r,θ,ϕ). The magnitude r of the prediction error is quantized by an optimal uniform quantizer, and the corresponding pair of (θ,ϕ) is encoded according to the magnitude of the quantized value of the radius. We partition the sphere of the radius r into n^2 regions to allocate n^2 quantization points over the sphere. Experimental results of the proposed method demonstrate improved coding efficiency.

1. INTRODUCTION

In recent days, 3D objects are popularly used in various applications, such as internet services, computer graphics, and synthetic imaging systems. 3D objects are generally represented by polygonal meshes, which are defined by vertices and their associated edges.

Since geometry data is specified in the 3D space by a 3D vector of three floating-point numbers, transmission and storage requirements for such 3D objects are very demanding due to the large amount of data. Recently, several approaches for geometry compression of 3D objects have been proposed [1-7]. In the bounding box quantization (BBQ) scheme [2,3], 3D coordinate values of each vertex point are initially approximated by an evenly subdivided bounding box along its x, y and z axes, independently. Each coordinate value is then predicted by a linear combination of previous values along a vertex spanning tree, and the difference of the approximated value and its prediction value is entropy encoded. However, this scheme generates large quantization errors because the quantization step size of the bounding box is relatively large. If Δ is the step size of the evenly subdivided bounding box, the mean squared quantization error is approximately $\Delta^2/12$.

In this paper, we propose a new coding algorithm for 3D mesh representation in the spherical coordinate system (r,θ,ϕ). The magnitude r of each prediction residual error is quantized using an optimum uniform quantizer, which is designed for the χ^2 distribution of r. Once the radius r is quantized, we encode the corresponding pair of (θ,ϕ) adaptively according to the magnitude of the quantized value of r. We show that the proposed algorithm has improved coding efficiency over the BBQ approach.

2. GEOMETRY COMPRESSION

A block diagram of the proposed scheme is shown in Figure 1. As shown in Figure 1, the encoder consists of four functional parts: preprocessing, normalization and coordinate transformation, optimal uniform quantization, and entropy coding.

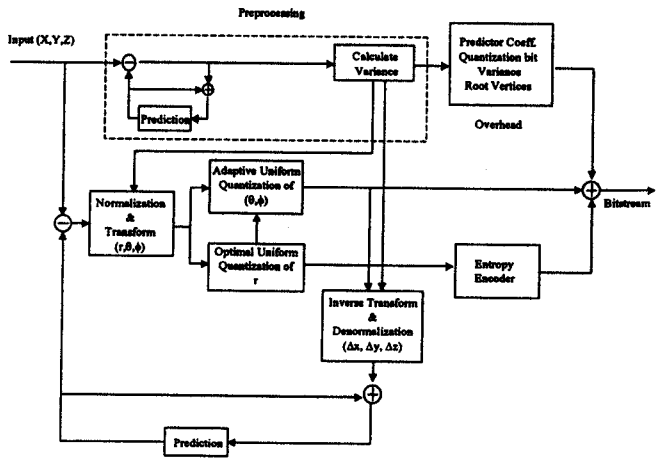


Figure 1: Block Diagram of Proposed Encoding Scheme

2.1 Linear Prediction

Since 3D models can have arbitrary shapes which form irregular polygonal meshes, it is not simple to design a universal geometry coding scheme for 3D models based on a specific data distribution. In order to exploit the strong correlation of coordinate values among neigh-

0-7803-5467-2/99/$10.00 © 1999 IEEE

boring vertex points, we take a differential coding approach, instead of encoding the absolute coordinate values of each vertex point.

We first obtain a prediction value $\hat{v}_n = (\hat{x}_n, \hat{y}_n, \hat{z}_n)$ of each vertex point $v_n = (x_n, y_n, z_n)$ based on the father-son relationship in the vertex spanning tree [3].

$$\hat{x}_n = \sum_{i=1}^{p} \lambda_{x_i} x_{n-i},\ \hat{y}_n = \sum_{i=1}^{p} \lambda_{y_i} y_{n-i},\ \hat{z}_n = \sum_{i=1}^{p} \lambda_{z_i} z_{n-i} \quad (1)$$

where p is a prediction order, and (λ_{x_i} , λ_{y_i} , λ_{z_i}) are prediction coefficients whose values are determined independently by the Levinson-Durbin algorithm [8]. We then obtain the prediction residual vector $(\Delta x_n, \Delta y_n, \Delta z_n)$.

$$\Delta x_n = x_n - \hat{x}_n,\ \Delta y_n = y_n - \hat{y}_n,\ \Delta z_n = z_n - \hat{z}_n \quad (2)$$

Since the prediction errors are highly concentrated around zero, each distribution fits well with the Laplacian distribution. In our differential coding structure, the prediction errors are modeled by the Laplacian distribution of zero mean.

Once the statistics of Δx, Δy, and Δz are observed at the preprocessing stage, we can normalize the prediction errors, Δx, Δy, and Δz, by their variances, respectively. The normalized prediction errors are regarded as the Laplacian random variables of zero mean and unit variance.

2.2 Conversion to the Spherical Coordinate System

After the normalization of all the components, each prediction error vector (Δx, Δy, Δz) is represented in the spherical coordinate system.

$$r = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \quad (3)$$

$$\theta = \tan^{-1} \Delta y / \Delta x \quad (4)$$

$$\phi = \cos^{-1} \Delta z / r \quad (5)$$

We note that each component of the residue vectors has the same Laplacian distribution with zero mean and unit variance. Therefore, the radius r has the distribution, that is shown in Figure 2.

Mathematically, the squared sum of k mutually independent Gaussian random variables has the k-th order χ^2 distribution. Since the prediction errors obtained from Eq. (2) have Laplacian distributions, their sum does not have the χ^2 distribution in a strict sense and it is difficult to treat the distribution analytically. Therefore, we model the distribution of the radius r as

$$f_r(r) = re^{-\alpha r^2},\ r \geq 0 \quad (6)$$

Experimentally, we have found that α=0.372 provides a satisfactory approximation of the actual error distribution for most 3D models that we have tested.

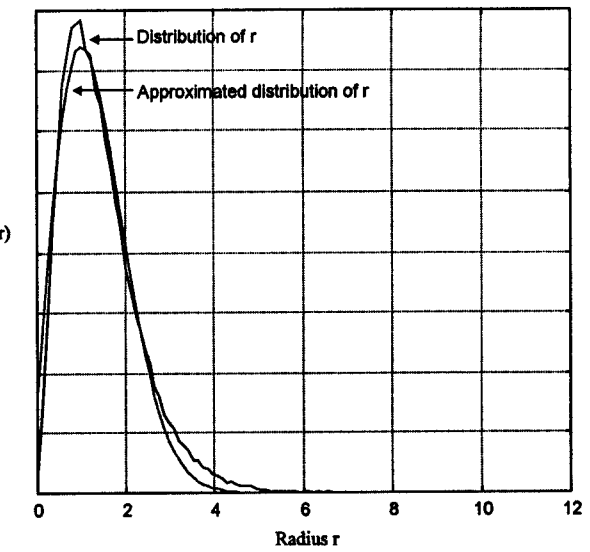


Figure 2: Distribution of The Magnitude r

2.3 Optimum Uniform Quantization of r

Based on the curve of $re^{-0.372r^2}$, we can design an optimal uniform quantizer for the magnitude r. Table 1 lists the step size of the optimum uniform quantizers for different number of output levels. Since the quantizer output values are fixed for a given number of output levels, the decoder should have the same table without any transmission of overhead.

Table 1: Step Size of Uniform Quantizers

No. of Output Levels	Step Size
2	2.9061
4	1.4123
8	0.8125
16	0.4505
32	0.2476
64	0.1349
128	0.0729
256	0.0390
512	0.0208
1024	0.0110
2048	0.0058

After the optimum uniform quantization of the magnitude of the prediction vector, the index value is encoded by a Huffman coder. Since the distribution of the magnitude is fixed by Eq. (6), we can design an optimal Huffman table uniquely. Therefore, we transmit the codeword for each quantization index, but not the Huffman table itself.

2.4 Adaptive Quantization of (θ,ϕ)

A vertex in the spherical coordinate system is parameterized by (r,θ,ϕ), where r is the distance of the vertex from the origin, θ is the angle from the x-axis, and ϕ is the angle from the z-axis. The surface of the sphere of the radius r is expanding n^2 times as the radius r is increased n times, as illustrated in Figure 3.

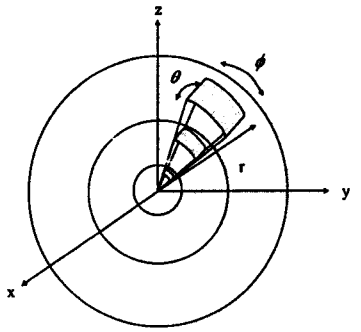


Figure 3: Relationship of (r,θ,ϕ)

If we quantize θ and ϕ uniformly for a given radius r, the area of patch, or the partition for quantization, becomes smaller near the poles, as shown Figure 4. The black points indicate quantized output values and the area of the patch is related to the quantization error.

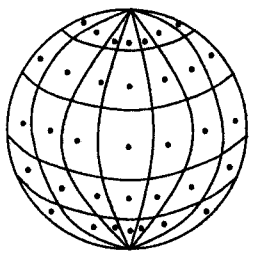


Figure 4: Uniform Quantization of θ and ϕ

Therefore, we should partition the surface of each sphere equally to yield the same quantization error. First, the surface of the sphere is divided into eight octants, as shown in Figure 5.

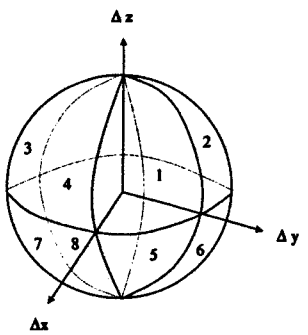
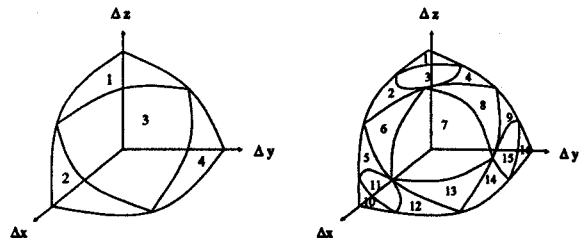


Figure 5: Partition of Eight Octants

Since we have eight octants of the sphere, three bits are used to indicate each octant. Since each octant is symmetrical, we explain the case of the first octant ($x \geq 0$, $y \geq 0$, $z \geq 0$). Each octant is divided further into four triangular-shaped patches of equal size, as shown Figure 6. The patch is subdivided recursively into the triangular-shaped smaller patches. This partitioning yields approximately equal quantization errors among all patches of the same level.



(a) Four Patches (b) Sixteen Patches

Figure 6. Subdivision Process

Once the radius r is quantized, we should encode the corresponding pair of (θ,ϕ) according to the magnitude of the quantized value of r adaptively. If we ignore the magnitude of the radius r in quantizing (θ,ϕ), the area of the patch expands n^2 times as the radius r is increased n times. Therefore, the quantization error of (θ,ϕ) increases as the radius r becomes large. In our scheme, we distribute quantization points evenly on each sphere to produce approximately equal quantization errors, irrespective of the magnitude of the residue vector r, as shown in Figure 7.

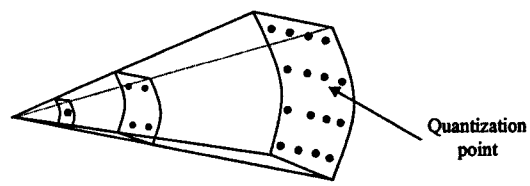


Figure 7: Allocation of Quantization Points

When the quantized value of r is small, which is a more probable case, as indicated in Figure 2, less coding bits are used to encode the pair of (θ,ϕ). When the value of r is large, which is a less probable case, more coding bits are used to encode the pair of (θ,ϕ). Since the distribution of the radius r is highly skewed toward zero, as shown in Figure 2, this type of bit allocation for (θ,ϕ) reduces the total number of coding bits for residue vectors effectively. Furthermore, coding of r and (θ,ϕ) are nicely decoupled for a sequential implementation.

2.5 Coding of Root Vertices

Usually, a 3D model is composed of multiple connected components. The first vertex of each component is

called as a root vertex, which plays a very important role as an anchor point for the entire mesh of the connected component. Any error in the root vertex position may cause the crack problem and the error can be propagated successively. Since the root vertex of each component has no preceding vertices, it cannot be coded in the same way as described before for other vertices. If the root vertex is coded as its own floating point numbers, it is not efficient. Therefore, we propose to encode root vertices using an adaptive predictor [9].

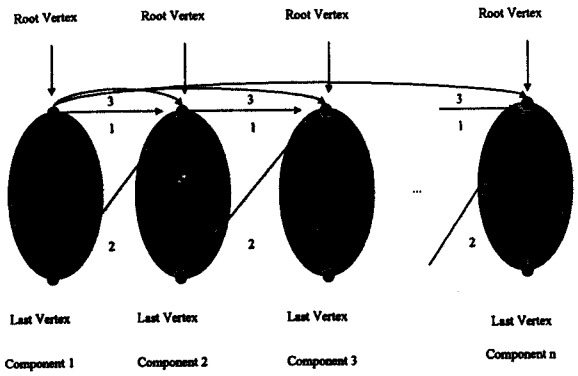


Figure 8: Adaptive Prediction for Root Vertices

As shown in Figure 8, the root vertex of a connected component can be predicted from the root vertex of the first component, the last vertex of the previous component, or the root vertex of the previous component. We can select a predictor which produces the smallest residual error. If the residual value is greater than a predetermined threshold value, the root vertex is directly coded with n bits without using any prediction. As shown in Table 2, two bits are used to indicate the prediction mode of the root vertex.

Table 2: Prediction Modes for Root Coding

Flag	Prediction Mode
00	No Prediction
01	Root Vertex of The First Component
10	Last Vertex of The Previous Component
11	Root Vertex of The Previous Component

3. EXPERIMENTAL RESULTS

We have tested the performance of the proposed algorithm with some MPEG-4 SNHC data set, such as PIETA, SKULL, HORSE, and BEETHOVEN, which are shown in Figure 9. These models are represented in the VRML format. The characteristics of the test models are summarized in Table 3.

As an object measure for evaluating the distortion between the original 3D polygonal model and the reconstructed one, we define the following error metric. Although the original model and the reconstructed model have the same connectivity information, they may have different values of vertex positions. For each vertex of the original model, the closest vertex in the reconstructed model is selected and their distance d_1 is calculated. For each vertex of the reconstructed model, the closest vertex in the original model is determined and their distance d_2 is calculated. The distortion metric is the average value of the two asymmetric values, d_1 and d_2. The error measure becomes symmetric with respect to the original and the reconstructed models.

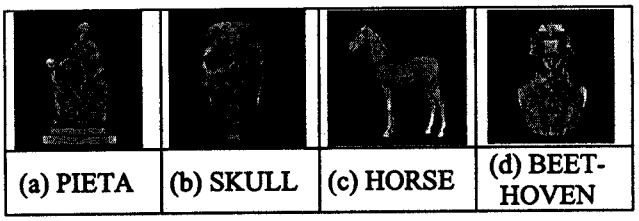


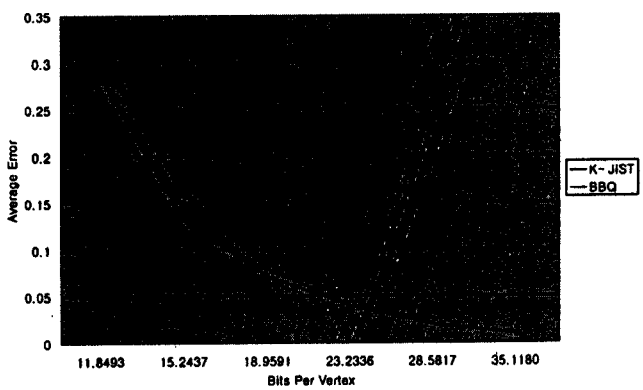
Figure 9: Test Models

Table 3. Characteristics of Test Models

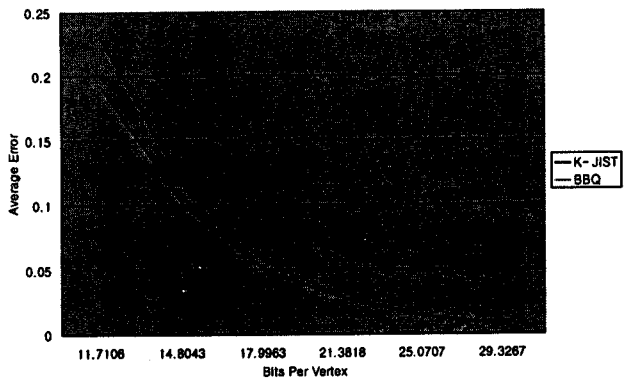
Model	No. of Vertices	No. of Faces	No. of Components
PIETA	3476	6976	1
SKULL	10952	22104	1
HORSE	11135	22258	3
BEETHOVEN	2845	2812	20

In Figure 10, we compare output performances of the proposed scheme and the BBQ scheme at various bit rates. The BBQ results are obtained with the executable file provided by IBM [2]. In Figure 10, the horizontal axis represents the bits per vertex (BPV) and the vertical axis shows the average distortion. Experimental results demonstrate that our proposed method produces the lower distortions than the BBQ scheme for the tested models.

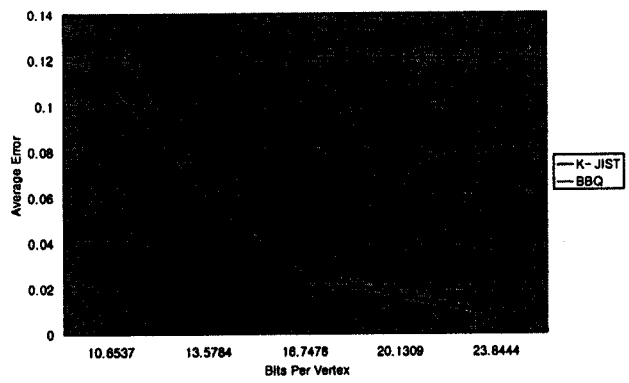
Figure 11 shows BEETHOVEN models that are reconstructed by the proposed and the BBQ schemes at the same bit rate of 8 bits per vertex (bpv).



(a) PIETA



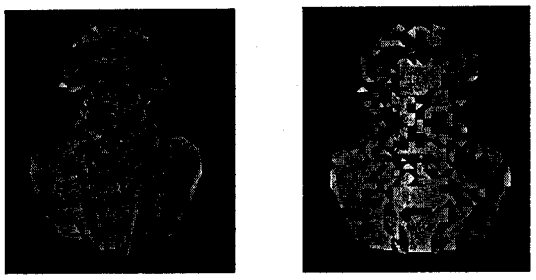
(b) SKULL



(c) HORSE

Figure 10. Performance Comparison

As we can observe in Figure 11, the visual quality of the reconstructed model by the proposed scheme is better than that of the BBQ scheme.



(a) Proposed Scheme (b) BBQ Scheme

Figure 11. Reconstructed BEETHOVEN Models (8bpv)

The BBQ scheme creates a serious problem of vertex collapsing. In general, the distribution of the original vertex positions is irregular and sometimes widely spread. If we quantize the original vertex points uniformly before the prediction operation, the quantization step size Δ is fairly large. During the quantization operation, all vertex points within the same quantization interval are represented as the same quantization output level and displayed as a single point in the reconstructed image, which generates the problem of vertex merging.

When Δ is the step size of the evenly subdivided interval of the vertex positions, the mean squared quantization error is approximately $\Delta^2/12$. The quantization error can be reduced if we apply the quantization operation on the prediction residuals, instead of the original data. Generally, the prediction errors have a narrower distribution than the original data, implying that quantization of prediction errors results in smaller distortion than quantization of the original data.

4. CONCLUSIONS

We propose a new coding scheme for 3D objects by exploiting the geometrical properties of prediction errors in the spherical coordinate system (r,θ,ϕ). We apply an optimal uniform quantizer to encode the radius r. We quantize (θ,ϕ) efficiently by allocating quantization points equally on each sphere. Our coding scheme shows improved performance over the BBQ approach.

ACKNOWLEDGMENTS

This work was supported in part by the Korea Science and Engineering Foundation (KOSEF) through the Ultra-Fast Fiber-Optic Networks (UFON) Research Center at Kwangju Institute of Science and Technology (K-JIST).

REFERENCES

[1] "SNHC Verification Model 9.0 [3D Mesh Encoding] Draft," ISO/IEC JTC1/SC29/WG11 MPEG98/N2302, Oct. 1998.

[2] G. Taubin, W.P. Horn and F.Lazarus, "The VRML Compressed Compressed Binary Format," http://www.research.ibm.com/vrml/binary.

[3] G. Taubin and J. Rossignac, "Geometry Compression through Topology Surgery," ISO/IEC JTC1/SC29/WG11 MPEG98/M3059, Feb. 1998.

[4] J. Li and C.C. Kuo, "Embedded Coding of Mesh Geometry," ISO/IEC JTC1/SC29/WG11 MPEG98/M3325, Mar. 1998.

[5] J.S. Choi, J.Y. Yang, M.H. Lee and C.T. Ahn, "Geometry Coding using PRVQ," ISO/IEC JTC1/SC29/WG11 MPEG98/M3148, Feb. 1998.

[6] Y.S. Ho and J.H. Ahn, "Adaptive Quantization Method for 3D Mesh Representation using the Spherical Coordinate System," ISO/IEC JTC1/SC29/WG11/MPEG98/M3997, Oct. 1998.

[7] M. Deering, "Geometric Compression," Proceedings of SIGGRAPH 95, pp.13-20, Aug. 1995.

[8] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, 1996.

[9] J.H. Ahn and Y.S. Ho, "Adaptive Coding of Multiple Components on 3D Model Coding," ISO/IEC JTC1/SC29/WG11 MPEG98/M4343, Dec. 1998.