Sequential Mesh Coding using Wave Partitioning

Tae-Wan Kim[1], Jeong-Hwan Ahn[2], Hyeok-Koo Jung[1], Yo-Sung Ho[2]

[1] Korea Electronics Technology Institute (KETI)
#455-6 MaSan-Ri JinWi-Myon, PyungTae-Si, 451-865, Korea
Phone: +82-31-610-4161, Fax: +82-31-610-4048
E-mail: {twkim, junghk}@keti.re.kr

[2] Kwangju Institute of Science and Technology (K-JIST)
1 Oryong-dong Puk-gu, Kwangju, 500-712, Korea
Phone: +82-62-970-2263, Fax: +82-62-970-2204
E-mail: {jhahn, hoyo}@kjist.ac.kr

Abstract: In this paper, we propose a sequential mesh coding algorithm using the vertex pedigree based on the wave partitioning. After a mesh model is partitioned into several small processing blocks (SPB) using wave partitioning, we obtain vertices for each SPB along circumferences defined by outer edges of the attached triangles. Once all the vertices within each circumference are arranged into one line, we can encode mesh models.

1. Introduction

In recent days, three-dimensional (3-D) synthetic VRML models have been used in various applications, such as computer animation and studio graphic design. However, such mesh representation has a problem that 3-D objects with fine details contain too excessive amount of data. Therefore, compression of mesh models is necessary to transmit 3-D models efficiently over a bandwidth-limited transmission channel, including Internet. Hoppe proposed a progressive mesh coding algorithm using mesh optimization to represent a 3-D model by a base mesh and vertex split variables; thus, the model can be shown in a progressive way using mesh simplification [1]. Taubin and Rossignac proposed a topological surgery method using vertex and triangle spanning trees to encode the connectivity and geometry data [2]. Touma and Gotsman improved coding efficiency of triangle mesh compression through traversal ordering using vertex degrees [4]. Those schemes have mainly focused on mesh compression and mesh simplification. If any bit errors occur during the transmission of 3-D model data, the reconstructed model can be severely damaged. Yan *et al.* suggested a robust coding scheme for 3-D graphic models using mesh segmentation and data partitioning, where a mesh model is partitioned and each piece is encoded and transmitted in the packet to enhance error resilience. However, coding efficiency was not considered seriously [3].In this paper, we propose a sequential mesh coding scheme using wave partitioning, aiming at both coding efficiency and error resilience. After we partition a 3-D mesh model into several pieces, we define a mother vertex and its son vertices based on their topological relationship. We will describe this relationship in detail in chapter 3. Then, we encode and decode the 3-D mesh model based on the index difference of each mother vertex between the current vertex and the next vertex.

2. Wave Partitioning

The wave partitioning is simply based on the natural wave phenomenon that one drop of water is dropped on the water surface and spreads out making circles in the lake. In the wave partitioning, a number of triangles are attached to the initial triangle along the arrow direction, as shown Fig. 1.

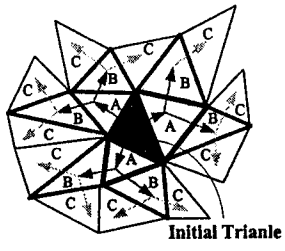


Fig. 1. SPB formation using wave partitioning

Fig. 1 shows a small processing block (SPB) that is a separate independent piece formed by attached triangles. In order to obtain a uniform size of SPB, we place two initial triangles on both ends of the model. After the whole model is divided into half, each part is partitioned into several SPBs recursively. Depending on the location of the initial triangle, SPBs are classified into circular or semi-circular types.

Fig. 2 displays a PAWN model that is partitioned into four separate parts by the wave partitioning.

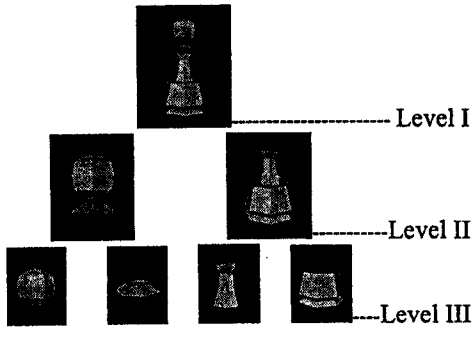


Fig. 2. Partitioned PAWN model

In order to smooth out the boundary of each separate SPB, we can apply a boundary smoothing algorithm [3]. As illustrated in Fig. 3, holes in one separate SPB can be fill up by projected triangles from its adjacent SPBs.

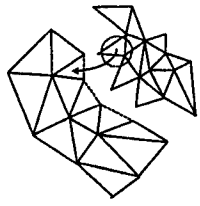
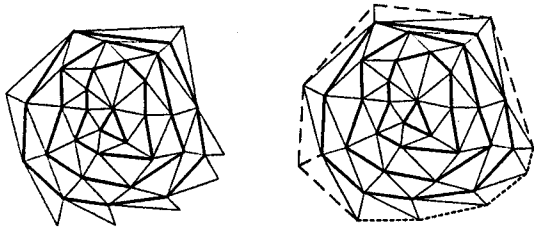


Fig. 3. Boundary smoothing

When we partition a mesh model into several SPBs by the wave partitioning, we can easily obtain circumferences from the outer edges of the attached triangles. In Fig. 4, we draw circumferences by bold lines.



(a) Before smoothing (b) After smoothing

Fig. 4. Modified boundary smoothing

Here, we define the rank of each circumference by its ordering. All processing is restricted within one rank of each independent SPB. When we apply the boundary smoothing operation on the mesh model in Fig. 4(a), the external vertex points of the SPB are connected by the short dotted lines, as illustrated in Fig. 4(b).

Since we need additional bits to represent the last unconnected rank of the SPB in our coding algorithm, we have modified the boundary smoothing operation. In order to make the last rank as a connected circle that is disjoint to its inner circle, we add long dotted lines along the outer shell of the SPB by forming additional triangles, as shown in Fig. 4(b).

3. Two Types of SPBs

As mentioned earlier, SPBs are classified into circular or semi-circular types according to the location of the initial triangle.

3.1. Circular SPB

Fig. 5 shows an example of the circular SPB. Vertices surrounding a particular vertex, except those vertices that are already included in the inner rank, are defined as son vertices (Sv) of the particular vertex, which, in turn, is called as the mother vertex (Mv) of its surrounding son vertices. In Fig. 5, v_0 is a Mv of v_3~v_5, v_1 is a Mv of v_6~v_7, and v_2 is a Mv of v_8~v_{11}. Although v_3 is connected to v_2, v_3 is not a Sv of v_2 since it is already counted as a Sv of v_0.

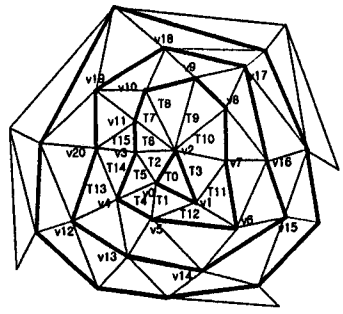


Fig. 5. Circular SPB

In Fig. 5, the closed thick line that is composed of v_0, v_1 and v_2 in the initial triangle is defined as Rank 1. The next thick line composed of v_3~v_{11} is defined as Rank 2. When we arrange all the vertices included in each rank in Fig. 5 into a row on the plane, we can form a map of the vertices, as shown in Fig. 6.

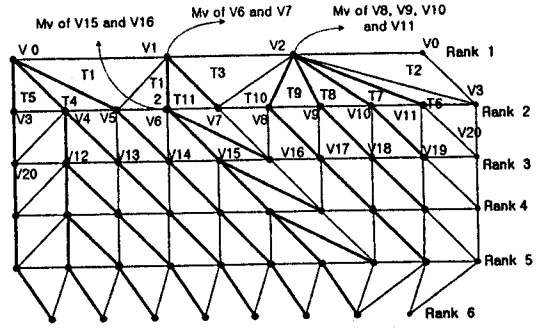


Fig. 6. Vertex arrangement of a circular SPB

In Fig. 6, each horizontal line lists vertices in the same rank and the mother-son relationship in two adjacent ranks is indicated by thick lines. Since each rank forms a closed loop in the circular SPB, the first vertex in each rank reappears at the last position of the rank.

In order to represent the topological information of the vertices, we define the index of each vertex by its distance from the first vertex in the same rank, and express the mother-son relationship by encoding the indexes of their mother vertices differentially.

In Fig. 6, v_6 in Rank 2 is the mother vertex of v_{16}. Since v_6 is the third vertex from v_3, its index is 3. Similarly, v_8 in Rank 2 is the mother vertex of v_{17} and its index is 5. Thus, we can represent the topological relationship of v_{16} and v_{17} by the index difference of their mother vertices, v_6 and v_8, i.e., 5-3=2.

3.2. Semi-circular SPB

Fig. 7 shows an example of the semi-circular SPB. As in the circular SPB, we can define the rank: Rank 1 includes v_0~v_2 and Rank 2 v_3~v_8. In Fig. 7, we indicate all ranks in the semi-circular SPB by thick lines. As shown in Fig. 7, each rank, except for the first rank, forms the semi-circular shape since there is no connecting edge between v_4 and v_5, or between v_{11} and v_{12}.

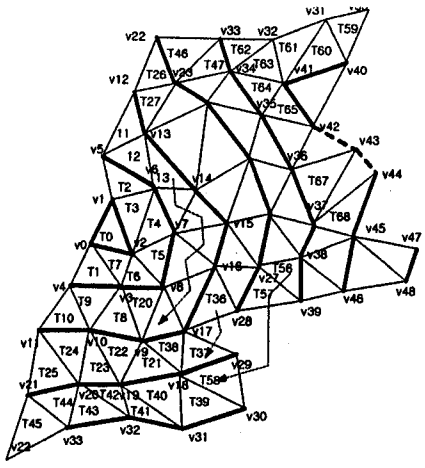


Fig. 7. Semi-circular SPB

When we arrange all the vertices included in each rank in Fig. 7 into a row on the plane, we can form a map of the vertices, as shown in Fig. 8. Since there is no connecting edge between v_4 and v_5 or between v_{11} and v_{12} in Fig. 7, we do not have a triangle containing v_4 and v_5 or v_{11} and v_{12} in Fig. 8.

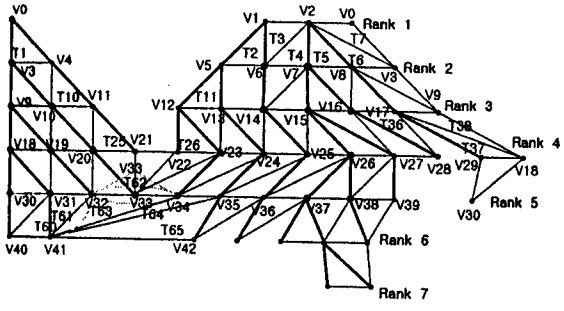


Fig. 8. Vertex arrangement of a semi-circular SPB

4. Sequential Encoding and Decoding

Fig. 9 explains our proposed encoding algorithm.

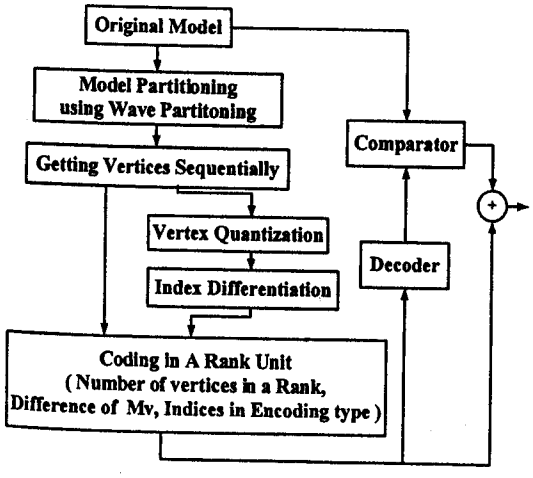


Fig. 9. Flow chart of the encoding algorithm

After the original model is partitioned, we process each SPB independently. Each SPB is encoded by the rank unit.

Following the vertex ordering defined by the rank, we classify each vertex into one of the four types: Cv is the current vertex, Nv is the next vertex, Mv_1 is the mother vertex of Cv, and Mv_2 is the mother vertex of Nv.

4.1 Geometry Information

For the geometry information, the difference between Cv and Nv in the same rank is quantized and entropy coded, and their indexes are then transmitted.

4.2 Topological Information

For the topological information, we calculate the value of $| Mv_1\text{-}Mv_2 |$ for all vertices. We classify $| Mv_1\text{-}Mv_2 |$ into three types and encode four kinds of parameter streams.

(1) For $| Mv_1\text{-}Mv_2 | = 0$, the triangle (Cv, Nv, Mv_1) is reconstructed at the decoder. For $| Mv_1\text{-}Mv_2 | = 1$, two triangles (Cv, Nv, Mv_1) and (Mv_1, Mv_2, Nv) are reconstructed at the decoder.

(2) For $| Mv_1\text{-}Mv_2 | = 2$, we can consider 28 patterns as shown in Fig. 10. At the decoder, one of 28 patterns is decoded based on the received index.

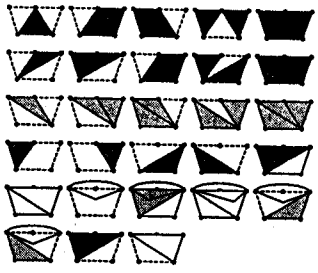


Fig. 10. 28 Patterns when $| Mv_1\text{-}Mv_2 | = 2$

(3) Fig. 11 shows the conditional triangulation operation in two different types. Table 1 indicates types and indexes for the conditional triangulation of Fig. 11.

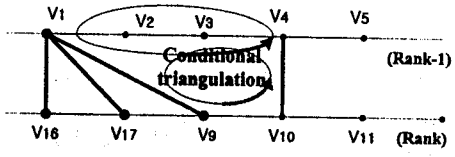


Fig. 11. Conditional triangulation

Table 1. Vertex Types and Indexes

	Indexes of each triangle
Type I	(v_9,v_1,v_2) : 0, (v_9,v_1,v_3) : 1, (v_9,v_1,v_4) : 2, (v_9,v_2,v_3) : 3, (v_9,v_2,v_4) : 4, (v_9,v_3,v_4) : 5, (v_{10},v_4,v_3) : 6, (v_{10},v_4,v_2) : 7, (v_{10}, v_4, v_1) : 8 (v_{10},v_3,v_2) : 9, (v_{10},v_3,v_1) : 10, (v_{10}, v_2, v_1) : 11 (v_9,v_{10},v_1) : 12, (v_9,v_{10},v_2) : 13, (v_9, v_{10}, v_3) : 14 (v_9,v_{10},v_4) : 15
Type II	(v_1,v_2,v_3) : 0, (v_1,v_2,v_4) : 1, (v_2,v_3,v_4) : 2, (v_1,v_3,v_4) : 3

This case can happen in the border line of the partitioned SPB. The indexes that represent two kinds of types listed in Table 1 are encoded when $|Mv_1-Mv_2| > 2$. In Type I, indexes of the triangles which are composed of 3 vertices of Cv in Rank, Nv in Rank and Mv_1~Mv_2 in (Rank-1) are transmitted. In Type II, indexes of the triangles which are composed of 3 vertices of Mv_1~Mv_2 in (Rank-1) are transmitted. When Mv of v_{41} is v_{31} and Mv of v_{42} is v_{35}, $|Mv_1-Mv_2| = 4$ in Fig. 8. Therefore, as shown in Fig. 7, there exist T_{62} that is composed of v_{32}, v_{33} and v_{34} in Rank 5. Also, there exist T_{63} that is compose of v_{32} in Rank 5, v_{34} in Rank 5 and v_{41} in Rank 4.

(4) Other parameters

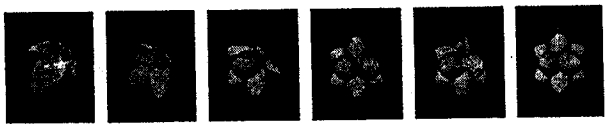
As exceptional parameters for the output of the comparator in Fig. 9, Mv of the last vertex in each rank and the number of vertices in each rank must be added.

5. Experimental Results

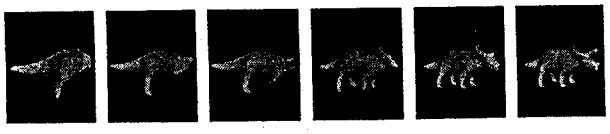
In order to evaluate the performance of the proposed scheme, we run computer simulations on several VRML models and compare the results with those of the MPEG-4 SNHC standard which is based on topological surgery [2].

Fig. 12 demonstrates the test models that are decoded sequentially by the rank unit. While the topological information in each model is decoded losslessly, the geometry information has lossy-coded by a 256-level quantizer. An arithmetic coding is also applied.

(a) PAWN



(b) SHAPE



(c) TRICERATOPS

Fig. 12. Sequential decoding

In our experiment, we have improved coding efficiency for the topological information by about one bit/vertex relative to the MPEG-4 SNHC algorithm, as summarized in Table 2. For geometrical information, however, we get lower coding efficiency than that of MPEG-4 SNHC algorithm by about 2~4 bits/vertex, as shown Table 3, because we consider only one adjacent vertex to encode the current vertex in each rank.

Table 2. Coding efficiency of topological information

	Original Model (Bytes)	Coding Efficiency (bits/vertex)	
		Proposed Algorithm	MPEG-4 SNHC
SHAPE	30,720	0.95	2.2
TRICERATOPS	33,960	3.31	4.3
BEETHOVEN	30,168	3.32	4.8

Table 3. Coding efficiency of geometrical information

	Original Model (Bytes)	Coding Efficiency (bits/vertex)	
		Proposed Algorithm	MPEG-4 SNHC
SHAPE	30,720	16.3	14.3
TRICERATOPS	33,960	11.7	10.4
BEETHOVEN	30,168	13.5	15.0

6. Conclusions

In this paper, we have proposed an efficient error-resilient coding algorithm for 3-D mesh models based on the wave partitioning by exploiting the topology and geometrical information. Since all processing is executed in rank-by-rank independently, the proposed scheme is error-resilient. We have also improved the coding efficiency over the MPEG-4 SNHC coding scheme by about 1bit/vertex for the topological information. In future, we can further improve the coding efficiency of the geometry information by exploiting several adjacent vertices.

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), and in part by the Ministry of Education (MOE) through the Brain Korea 21 (BK21) project.

References

[1] G. Taubin and J. Rossignac, "Geometry compression through topological surgery," *ACM Trans. Graphics*, vol. 17, 1998.

[2] Z. Yan, S. Kumar, J. Li and C.C. Jay Kuo, "Robust Coding of 3D Graphic Models using Mesh Segmentation and Data Partitioning," *Proc. IEEE. Int. Conf. Image Processing*, Oct. 1999.

[3] C. Touma and C. Gotsman, "Triangle Mesh Compression," *Graphics Interface'98 Conference Proc.*, pp. 26-34, 1998.

[4] M. Deering, "Geometric compression," *Computer Graphics Proc.*, pp. 13-20, Aug. 1995.