PAPER

Geometry Coding for Triangular Mesh Model with Structuring Surrounding Vertices and Connectivity-Oriented Multiresolution Decomposition

Shuji WATANABE[†a)], ***Student Member and*** **Akira KAWANAKA**[†b)], ***Member***

SUMMARY In this paper, we propose a novel coding scheme for the geometry of the triangular mesh model. The geometry coding schemes can be classified into two groups: schemes with perfect reconstruction property that maintains their connectivity, and schemes without it in which the remeshing procedure is performed to change the mesh to semi-regular or regular mesh. The former schemes have good coding performance at higher coding rate, while the latter give excellent coding performance at lower coding rate. We propose a geometry coding scheme that maintains the connectivity and has a perfect reconstruction property. We apply a method that successively structures on 2-D plane the surrounding vertices obtained by expanding vertex sequences neighboring the previous layer. Non-separable component decomposition is applied, in which 2-D structured data are decomposed into four components depending on whether their location was even or odd on the horizontal and vertical axes in the 2-D plane. And a prediction and update are performed for the decomposed components. In the prediction process the predicted value is obtained from the vertices, which were not processed, neighboring the target vertex in the 3-D space. And the zero-tree coding is introduced in order to remove the redundancies between the coefficients at similar positions in different resolution levels. SFQ (Space-Frequency Quantization) is applied, which gives the optimal combination of coefficient pruning for the descendant coefficients of each tree element and a uniform quantization for each coefficient. Experiments applying the proposed method to several polygon meshes of different resolutions show that the proposed method gives a better coding performance at lower bit rate when compared to the conventional schemes.

key words: *polygonal mesh, multiresolution decomposition, 2-D structuring, geometry data coding, space-frequency quantization*

1. Introduction

3-D image models are being used increasingly in various fields, such as industrial product design, movies, video games, and the digital museum [1]. The polygonal mesh is a well known general-purpose shape model used as a representation model for representing 3-D images. The polygonal mesh is composed of two basic types of data: (a) geometry data - the coordinate values of vertices constructing the mesh, and (b) connectivity data - the set of vertex indices that represents each polygon of the mesh. The geometry data share a larger amount of the total than the connectivity data. The geometry data coding is important for efficient polygon mesh representation.

In many studies on geometry coding, polygons which were not triangles were divided into several triangles, and the triangle mesh obtained was used for the coding [2], [3]. The geometry coding schemes can be roughly classified into two categories, i.e., those with a perfect reconstruction property that performs the geometry coding maintaining the mesh connectivity [4]–[15], and those without a perfect reconstruction property that change the connectivity and geometry from the original using remeshing procedures [16]–[22]. Touma et al. [7] proposed a representative technique of the former scheme. In this scheme, the geometry data are encoded by predictive coding based on a parallelogram rule. A representative scheme of the latter category is the one using the semi-regular wavelet transform (WT) [23] proposed by Khodakovsky et al. [18]. This scheme applied the wavelet transform to the geometry data of a semi-regular mesh which has been changed from original mesh by MAPS algorithm [24]. Although this scheme is not capable of reconstructing the original mesh perfectly, it gives an excellent coding performance at lower coding rate compared to that of schemes with a perfect reconstruction property. On the other hand, the coding schemes with a perfect reconstruction property gave good coding performances at a higher coding rate.

In this paper we propose a new geometry coding scheme with a perfect reconstruction property to improve the coding performance at lower bit rates. To apply the multiresolution decomposition to the geometry data of a mesh we developed the surrounding vertex structuring which arranges the vertices of the mesh neighboring those of the processed area starting from a vertex on a 2-D plane. The surrounding vertex structuring derives the geometry data structured on the 2-D plane [15]. The 2-D structured geometry data are decomposed into four components in a non-separable manner at one decomposition level. The decomposed components are obtained by the prediction and update procedures in the same way as in the lifting wavelet transform. The prediction and update procedures are performed considering the connectivity of the mesh. The octave decomposed coefficients are derived by applying one-level decomposition to the updated component recursively. To remove the redundancy of the decomposed coefficients at similar position between the different resolution levels, zerotree coding [27]–[29] with space-frequency quantization (SFQ) [29] is introduced. In SFQ the parent-children relationships between the decomposed coefficients located

Manuscript received September 9, 2009.
Manuscript revised October 2, 2010.

[†]The authors are with the Faculty of Science and Technology, Sophia University, Tokyo, 102–8554 Japan.
a) E-mail: watanabe@akira.ee.sophia.ac.jp
b) E-mail: kawanaka@akira.ee.sophia.ac.jp

DOI: 10.1587/transinf.E94.D.886

at similar positions between different resolution levels are formed, and the optimal decision whether all descendant coefficients of each tree element are pruned (quantized to zero) or the children coefficients are quantized uniformly on the basis of the cost function treating the coordinate values of geometry as a vector is determined. To evaluate the proposed scheme some experiments using several 3D meshes with different connectivity complexity were performed. The coding performances of the proposed scheme are compared to the scheme adopted in MPEG-4, a TG coder which has been used as a benchmark of perfect reconstruction schemes, and the scheme by Gumhold et al. [10] which used a higher-order prediction.

2. 2-D Structuring Process of Surrounding Vertices

In cases where a triangle mesh has some holes, an imaginary vertex for a hole is supposed, and it is thought to be linked to the vertices forming the boundary of the hole. And the structuring procedure is applied to the triangle mesh without holes. The vertices of the mesh are structured on a 2-D plane by extracting the vertices an edge distance away from a layer subsequently starting from a vertex. Here the edge distance between two vertices is defined as the minimum number of edges on routes which are connecting the vertices. Also a valence of vertex is used as a criterion for selecting a vertex in the structuring processing. In Fig. 1 the valence of each vertex is shown as a number in the circle indicating a vertex. A vertex whose valence is a maximum among the processing candidates is chosen as the next processing object. If several vertices have the same maximum valence number, a vertex which appears ahead in the scanning of the connectivity data is selected. We call this a maximum valence vertex. The flow of the surrounding vertex structuring is described below.

Step 1:
First the maximum valence vertex in the vertices constructing the triangle mesh is selected as an initial vertex. The index of the initial vertex is substituted for an element of the vertex index table $v(0,0)$ as shown in Fig. 1 (a).

Step 2:
The vertices which are one edge distance away from the initial vertex are extracted while maintaining their neighboring relation. The vertex sequence is formed by taking the maximum valence vertex in the extracted vertices as the head of the sequence, and in addition taking the maximum valence vertex of two vertices linking the head vertex as the vertex following the head in the sequence as shown in Fig. 1 (b). When the length of the sequence is denoted by n_1, the vertex indices of the sequence are substituted for the elements of the vertex index table $v(1,k)$, $k = 0 \cdots n_1 - 1$. The vertices of this sequence are marked "processed", and construct the 1st layer of structured vertices.

Step 3:
The vertex sequence at s-th layer is extracted as vertices one edge distance away from vertices in $(s-1)$th layer. The vertex order of the sequence is determined in the same way as in step (2). When the length of sequence is denoted by n_s, the extracted vertices are substituted for the elements of the vertex index table $v(s,k)$, $k = 0 \cdots n_s - 1$. An example of substantial vertex layer extraction is shown in Fig. 1 (c).

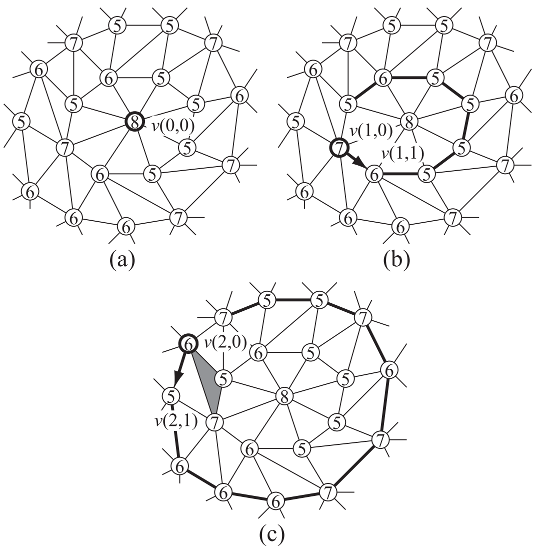


Fig. 1 An example of the surrounding acquisition process; (a) 0th layer: select the maximum valence vertex as the start vertex of the process, (b) 1st layer: select the start vertex of the 1st layer and determine the direction of the structuring of all sequences, (c) 2nd layer: select the start vertex of the 2nd layer, and the direction of the sequence is determined as same as that of the 1st layer.

Step 4:
Step (3) is repeated increasing the layer number s to $s+1$ until all vertices are marked "processed". The supposed vertices which were introduced to treat the triangle mesh without holes are removed from the vertex index table. Then, holes arisen in the vertex index table are closed by shifting the indices on the table in the direction toward the origin along the k axis.

Touma et al. [7] also trace the vertex surrounding the pre-processed region in one dimensional manner. The tracing order for the vertices is used for predicting the vertex geometry. In the proposed scheme the vertices surrounding the pre-processed region are treated as the elements of a layer which construct a line in the 2-D plane. And the surrounding vertex structuring will be used for the multiresolution decomposition of the vertex geometry of 3-D mesh.

An example of the structured vertex table is shown in Fig. 2. Geometry data of the triangular mesh are structured on a 2-D plane according to the structured vertex table $v(s,k)$. At the position (s,k) of the structured plane, the vertex index v is determined from the structured vertex table $v(s,k)$ and the structured geometry data are obtained by assigning coordinate values (x_v, y_v, z_v) of the vertex to that position. Figure 3 shows structured geometry data $\mathbf{r}(s,k) = (x(s,k), y(s,k), z(s,k))$ of the triangular mesh shown in Fig. 2.

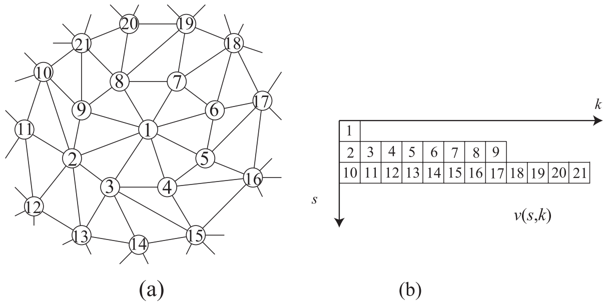


Fig. 2 A example of the triangular mesh and structured vertex index table; (a) a triangular mesh in which the number in the circles shows index of each vertex, (b) vertex index table.

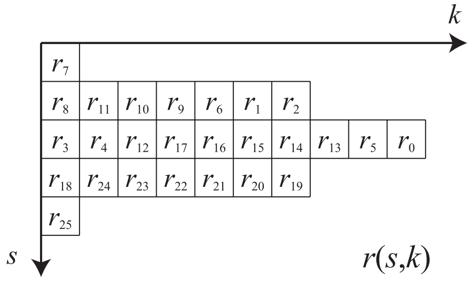

k
s
r_7
r_8 r_{11} r_{10} r_9 r_6 r_1 r_2
r_3 r_4 r_{12} r_{17} r_{16} r_{15} r_{14} r_{13} r_5 r_0
r_{18} r_{24} r_{23} r_{22} r_{21} r_{20} r_{19}
r_{25}
$r(s,k)$

Fig. 3 Structured geometry data of the triangular mesh shown in Fig. 2 (a).

3. Connectivity-Oriented Multiresolution Decomposition (CO-MRD)

In order to reduce the correlation of geometry data of vertices, a novel multiresolution decomposition is derived. Since the geometry data were structured on a 2-D plane, we can decompose the geometry data into multiresolution components by means of the wavelet transform [25], [26]. In a 2-D wavelet transform for an image, the decomposition is into four frequency bands by applying one-dimensional decomposition to it along the horizontal and vertical axes separately. The multiresolution decomposed coefficients can be obtained by repeating the one-level decomposition to the low frequency band. The separable processing is used on the assumption that the correlation of image pixel value can be represented as a product of two functions along horizontal and vertical axes respectively. Also in the lifting wavelet approach, the high frequency component is usually obtained as a prediction error of a pixel at an odd position using the pixel values on both sides of it, and the low frequency component is obtained as an updated value of a pixel at an even position with the prediction errors of the pixels on both sides of it in the one-dimensional decomposition. Since the valence of vertex which is the number of its neighbors in 3-D space is an arbitrary integer larger than 3 for a triangle mesh without holes, it is not reasonable to assume that the correlation of structured geometry data is separable in regard to the horizontal and vertical axes. So we propose a non-separable multiresolution decomposition scheme in which the geometry data are decomposed into four components depending on the positions of data in the s-k plane. And the high frequency components are obtained as a prediction error from the geometry data of vertices neighboring it in 3-D space and having not been processed. And the low frequency component is obtained as an updated value calculated with the prediction errors of the neighboring vertices.

We proposed a connectivity-oriented multiresolution decomposition (CO-MRD) method. CO-MRD decomposes coefficients with the prediction and the update process along the horizontal and vertical axes of the 2-D plane. This CO-MRD is separable component decomposition. The structured geometry $f(s,k)$ on the 2-D plane (s,k) are decomposed into four set of elements $S_{EE}^{(l)}$, $S_{EO}^{(l)}$, $S_{OE}^{(l)}$, and $S_{OO}^{(l)}$ by whether s and k of each element are even or odd. An example of the decomposition is shown in Fig. 4. Each elements in $S_{OO}^{(l)}$, $S_{EO}^{(l)}$, and $S_{OE}^{(l)}$ is transformed to the prediction error, and each element in $S_{EE}^{(l)}$ is updated with the prediction errors. The decomposition and transformation are repeated recursively. The set $S_{EE}^{(l)}$ at the resolution level l is decomposed into $S_{EE}^{(l+1)}$, $S_{OE}^{(l+1)}$, $S_{EO}^{(l+1)}$, and $S_{OO}^{(l+1)}$. The transformed coefficients at each resolution level obtained as follows.

At first the prediction error of $S_{OO}^{(l)}$, which is denoted by $g_{OO}^{(l)}(s,k)$, is obtained with the prediction value calculated from the elements of $S_{EE}^{(l)}$, $S_{OE}^{(l)}$, and $S_{EO}^{(l)}$ as shown in Fig. 5 (a).

$$\frac{g_{OO}^{(l)}(s,k)}{K} = f^{(l-1)}(2s+1,2k+1) - \frac{1}{|\,R_{OO}^{(l)}(2s+1,2k+1)\,|} \sum_{\substack{(m,n)\in \\ R_{OO}^{(l)}(2s+1,2k+1)}} f^{(l-1)}(m,n) \tag{1}$$

where $R_{OO}^{(l)}{=}S_{EE}^{(l)} \cup S_{OE}^{(l)} \cup S_{EO}^{(l)}$, $f^{(0)}(s,k) = f(s,k)$. $R_i^{(l)}(s,k)$ indicates the elements which are depart l in the edge distance from the element at (s,k) in $R_i^{(l)}$ and K is a transform parameter. Next, the prediction error of $S_{EO}^{(l)}$ denoted by $g_{EO}^{(l)}(s,k)$ is obtained from the elements in $S_{EE}^{(l)}$ and $S_{OE}^{(l)}$, and the prediction error of $S_{OE}^{(l)}$ by $g_{OE}^{(l)}(s,k)$ is obtained from $S_{EE}^{(l)}$ as shown in Figs. 5 (b) and (c).

$$\frac{g_{EO}^{(l)}(s,k)}{K} = f^{(l-1)}(2s,2k+1) - \frac{1}{|\,R_{EO}^{(l)}(2s,2k+1)\,|} \sum_{\substack{(m,n)\in \\ R_{EO}^{(l)}(2s,2k+1)}} f^{(l-1)}(m,n) \tag{2}$$

$$\frac{g_{OE}^{(l)}(s,k)}{K} = f^{(l-1)}(2s+1,2k) - \frac{1}{|\,R_{OE}^{(l)}(2s+1,2k)\,|} \sum_{\substack{(m,n)\in \\ R_{OE}^{(l)}(2s+1,2k)}} f^{(l-1)}(m,n) \tag{3}$$

where $R_{EO}^{(l)} = S_{EE}^{(l)} \cup S_{OE}^{(l)}$ and $R_{OE}^{(l)} = S_{EE}^{(l)}$. Then the updated values of $S_{EE}^{(l)}$ denoted by $f^{(l)}(s,k)$ is obtained with the prediction errors which were obtained previously.

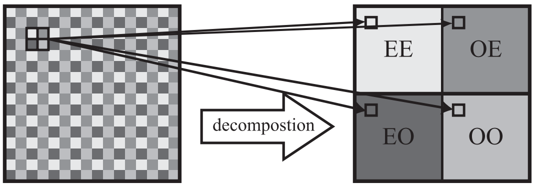


Fig. 4 The decomposition into four components: Even-Even (light gray), Odd-Even (slightly dark gray), Even-Odd (dark gray), and Odd-Odd (gray) components.

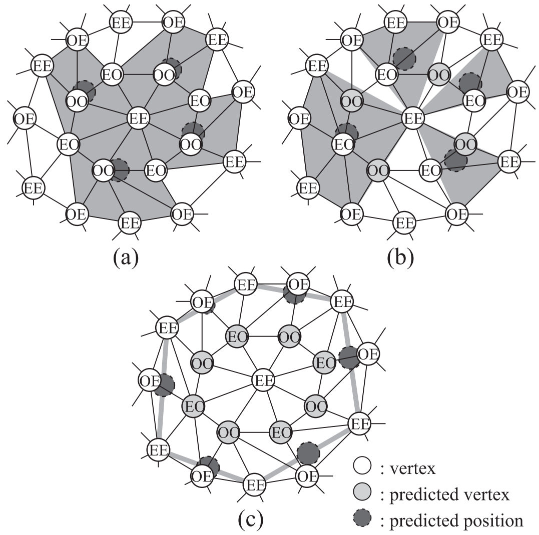


Fig. 5 An example of the prediction process; (a) prediction of the OO component: predicted by other component (EE, OE and EO), (b) prediction of the EO component: predicted by EE and OE components, (c) prediction of the OE component: predicted only by EE component.

$$\frac{f^{(l)}(s,k)}{\tilde{K}} = f^{(l-1)}(2s,2k) + \frac{1}{|\, R_{EE}^{(l)}(2s,2k)\, |}\sum_{w\in W}\sum_{\substack{(m,n)\in \\ R_{EE}^{(l)}(2s,2k)}} \frac{g_w^{(l)}(m,n)}{K} \qquad (4)$$

where $R_{EE}^{(l)} = S_{OE}^{(l)} \cup S_{EO}^{(l)} \cup S_{OO}^{(l)}$, $W = \{OE, EO, OO\}$, and $\tilde{K}$ is a transform parameter.

In the synthesis process the updated value $f^{(l-1)}(s,k)$ and the prediction errors $g_w^{(l-1)}(s,k)$, $w = OE, EO, OO$ at the resolution level (l-1) are derived from the updated value $f^{(l)}(s,k)$ and the prediction errors $g_w^{(l)}(s,k)$, $w = OE, EO, OO$ at the resolution level l. The structured geometry $f(s,k)$ is able to be obtained as $f^{(0)}(s,k)$ by repeating the synthesis process. In the decomposition and synthesis processes the extraction of the elements which are depart l in the edge distance from the element at (s,k) is performed with the connectivity of polygonal mesh.

4. Vector Space-Frequency Quantization

When structured data on a 2-D plane such as an image were wavelet transformed, there was usually redundancy in the wavelet coefficients at similar position in different resolution levels, and the zerotree coding is known to be effective in reducing this redundancy. In the zerotree coding the parent-children relationships between the coefficients at similar position in different resolution levels are constructed and the information on whether all coefficient amplitudes of the descendant of each element are smaller than a quantization level or not are encoded. Especially the space-frequency quantization which determines the optimal combination with spatial zerotree quantization and standard uniform quantization of its children node leads to a better coding performance. We introduce a zerotree coding based on SFQ [29] to reduce the redundancies in the mutiresolution decomposed coefficients at similar positions in different resolution levels. A vector SFQ which calculates the rate-distortion characteristics associated with the coordinate values of vertex as a vector is introduced.

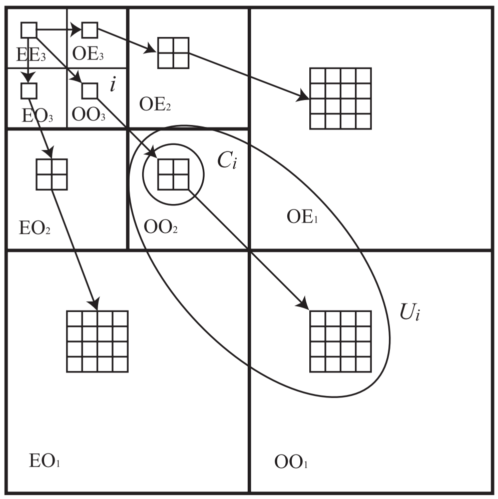


Fig. 6 The relationship between parents and children of decomposed coefficients for the space-frequency quantization.

The tree structures are defined among the transformed coefficients at similar position in different resolution levels. Each element in the lowest resolution level L is associated with three elements, and each element except for the elements in the lowest resolution level is associated with four elements at similar position as shown in Fig. 6. The associated elements of an element i are called children of the element C_i, and the set of its descendant in which its children are removed is denoted by U_i. The decision whether the descendant of an element are pruned (are quantized to zero) or not is made by an iterative processing. $Q^{(m)}$ denotes a subtree which is obtained in the iteration count m and its initial tree is set to the full tree T. The full tree T is prepared as a tree rooting the elements located in the lowest resolution $S_{EE}^{(L)}$ as shown in Fig. 6 and $Q^{(m)}$ is the optimal subtree in which the descendants of some elements were pruned as shown in Fig. 7. The optimal subtree $Q^{(m)}$ is obtained by the following algorithm:

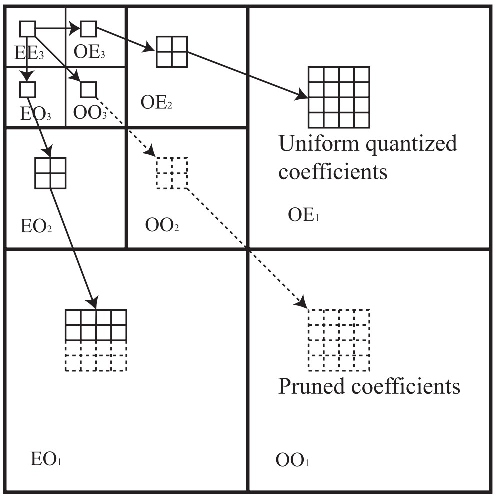


Fig. 7 An example of combination between the coefficient pruning and the uniform quantization: solid line denotes uniform quantized coefficients and dashed line denotes pruned coefficients.

ALGORITHM

Step 1:

$J^*(U_i)$ denotes the optimal Lagrange cost associated with the tree at node i. d denotes the depth of the tree. And $d_{max}^{(m)}$ denotes the maximum depth of $Q^{(m)}$. Each parameter is initialized.

$$Q^{(0)} \leftarrow T$$
$$m \leftarrow 0$$
$$J^*(U(i)) \leftarrow 0 \quad \forall i \in \text{leaf nodes of } T.$$
$$d \leftarrow d_{max}^{(m)} - 1$$

Step 2:

Calculate a probability of occurrence $p^{(m)}(\hat{\mathbf{g}}_i)$ of the quantized coefficients vector of an element i in $Q^{(m)}$. Here the original coefficients vector is denoted by $\mathbf{g}_i$ which is composed of three transformed coefficients for the coordinate values (x_i, y_i, z_i) at the element i, and $\hat{\mathbf{g}}_i$ denotes the quantized vector of $\mathbf{g}_i$.

Step 3:

The pruning map $n_i^{(m)}$ which indicates whether the descendant of each element are pruned or not is obtained.

if

$$\sum_{i \in U_i} \mathbf{g}_i^2 \leq \sum_{j \in C_i} [J_j^{(m)} + J^*(U_j)]$$

then

$$n_i^{(m)} = 0, J^*(U_i) = \sum_{i \in U_i} \mathbf{g}_i^2$$

else

$$n_i^{(m)} = 1, J^*(U_i) = \sum_{j \in C_i} [J_j^{(m)} + J^*(U_j)]$$

where

$$J_i^{(m)} = D_i^{(m)} + \lambda R_i^{(m)}$$
$$D_i^{(m)} = | \mathbf{g}_i - \hat{\mathbf{g}}_i^{(m)} |^2$$
$$R_i^{(m)} = -\log_2 (p^{(m)}(\hat{\mathbf{g}}_i))$$

Here, $J_i^{(m)}$ denotes the Lagrange cost at element i in repeat count m. $D_i^{(m)}$ and $R_i^{(m)}$ denote a distortion and a rate, respectively. $n_i^{(m)}$ denotes a binary pruning map at element i. When $n_i^{(m)}$ is 0, this means that the pruning process is performed. Otherwise, the uniform quantization process is performed. λ is the Lagrange multiplier.

Step 4:

If all nodes at every tree depth are processed, go to step 5. Else go to step 3.

$$d \leftarrow d - 1$$
$$\text{go to step3} \quad \text{if } d \geq 0.$$

Step 5:

If there are no new elements that are pruned, finish the algorithm. Otherwise, increase the repeat count m and go to step 3.

if $Q^{(m)} \neq Q^{(m+1)}$

then

$$m \leftarrow m + 1$$
$$d \leftarrow d_{max}^{(m)} - 1$$

go to step3

else

end the algorithm.

The pruning map and the quantized coefficients of the elements which were not pruned are encoded with the arithmetic coding.

5. Experiments

In order to evaluate the coding performance of the proposed scheme, we carried out some experiments using the triangular meshes “Venus”, “Feline”, “Horse”, and “Dino” shown in Fig. 8. We used a metric peak signal to noise ratio (PSNR) for the shape distortion of the reconstructed model as follows:

$$\text{PSNR} = 20 \log_{10} \frac{\text{peak}}{D_{rms}} \text{ [dB]} \qquad (5)$$

The peak is the diagonal length of the bounding box and D_{rms} is the root mean square error between the reconstructed model and the original calculated by the *Metro* [30] which is a well known tool for evaluating the distortion of reconstructed models. Figure 9 shows structured geometry data of the model “Venus” by transforming each coordinate value to image intensity. The image size is 682x296. The coordinate values in the horizontal direction tend to vary more continuously than in the vertical direction, because the coordinate values of the vertices which were located in the same layer in 3-D space were arranged in the horizontal direction.

In order to determine the optimal resolution level for the proposed scheme, we show the rate-distortion curves of

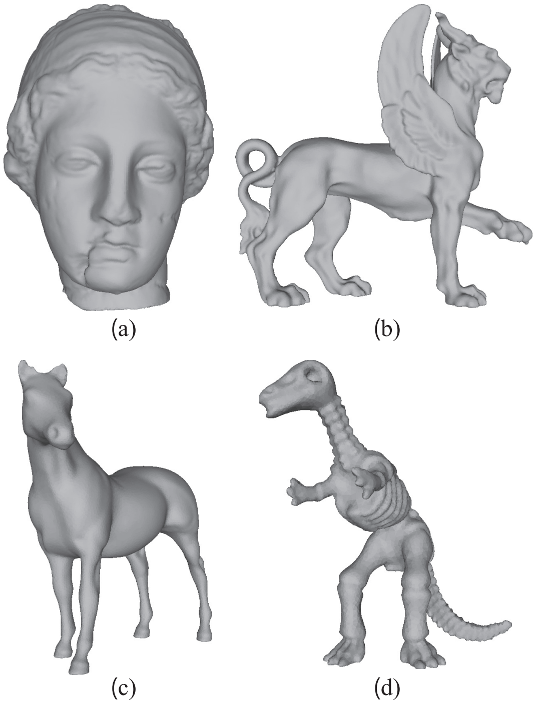


Fig. 8 Triangular meshes for experiments; (a) "Venus" (134,345 vertices, 268,686 polygons), (b) "Feline" (49,864 vertices, 99,732 polygons: non-zero genus model), (c) "Horse" (19,851 vertices, 39,698 polygons), (d) "Dino" (14,070 vertices, 28,136 polygons).

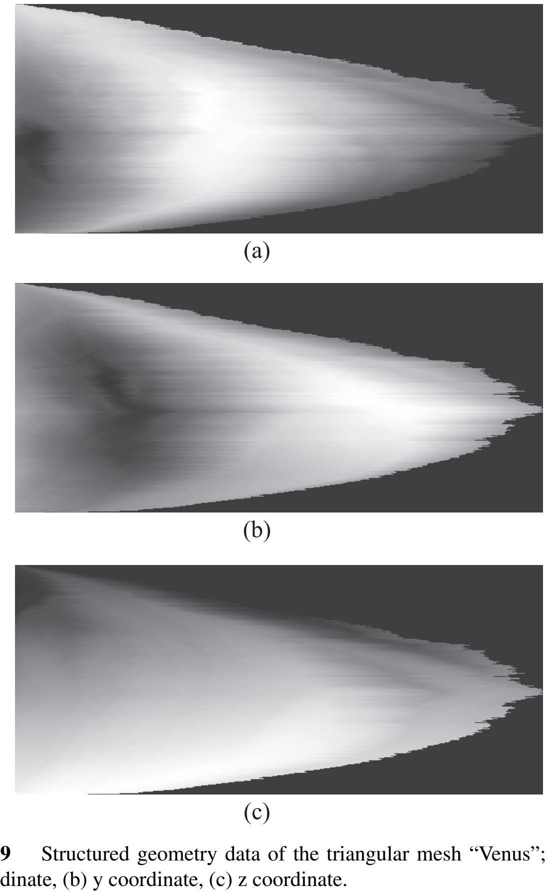


Fig. 9 Structured geometry data of the triangular mesh "Venus"; (a) x coordinate, (b) y coordinate, (c) z coordinate.

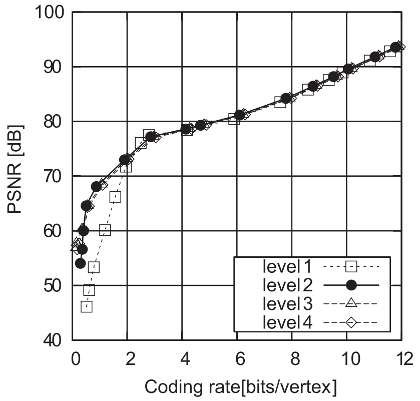


Fig. 10 Rate-distortion curves of geometry data of "Venus" when the decomposed level was changed.

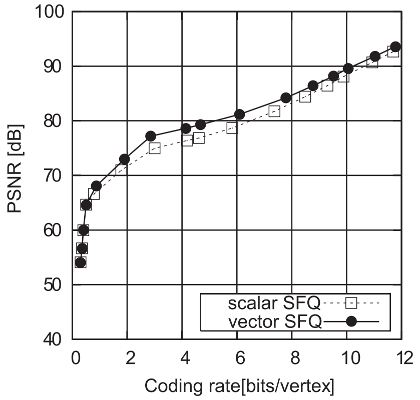
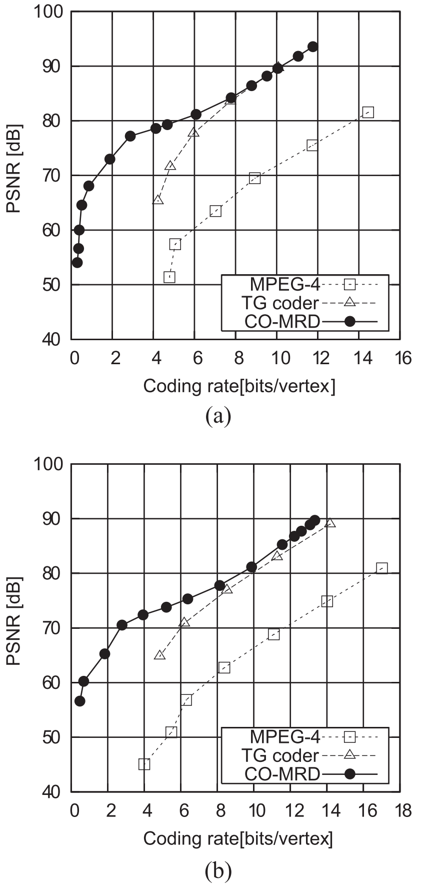


Fig. 11 Rate-distortion curves of geometry data of "Venus" in cases of treating three coordinated values as scalars and one as a vector in SFQ.

geometry data for the model "Venus" at several resolution levels in Fig. 10. Here the parameters of multiresolution decomposition K and $\tilde{K}$ are set to 1.3 and 2.0 empirically. Considering the rate-distortion curves shown in Fig. 10 and the calculation complexity, we set the optimal resolution level as two in the following experiments. Although CO-MRD requires to search the adjacent vertices for a target vertex, its computation time can be reduced by making the adjacent vertex list for each vertex before the multiresolution decomposition processing. CO-MRD for the 3-D models used in the experiments is time consumed 1.4 to 4.5 times comparing to ordinary shape-adaptive wavelet transform in case of including the processing of the adjacent vertex list formation. Next, the coding performances in case of treating the coordinate values as three scalars and a vector were compared. Figure 11 shows the coding performances in two cases. The case of treating them as a vector gives higher PSNR. Because three pruning maps are needed to be encoded in case of treating them as three scalars, it seems to become overhead compared to the vector case.

Figure 12 shows the coding performance of the proposed scheme compared to those of conventional schemes. The solid line shows the coding performance of the proposed scheme, the dotted line shows that of the scheme with



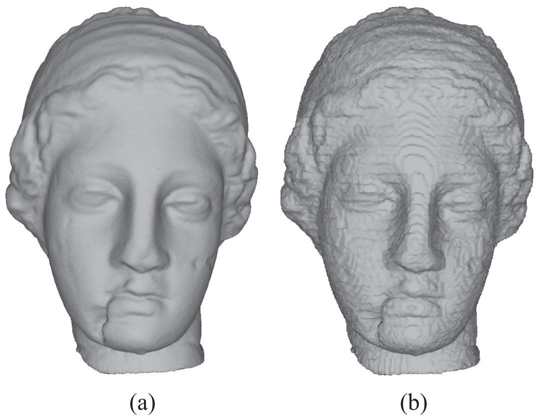
(a)

(b)

Fig. 12 Rate-distortion curves of geometry data applying the proposed scheme, the MPEG-4 and the TG coder; (a) "Venus", (b) "Feline".

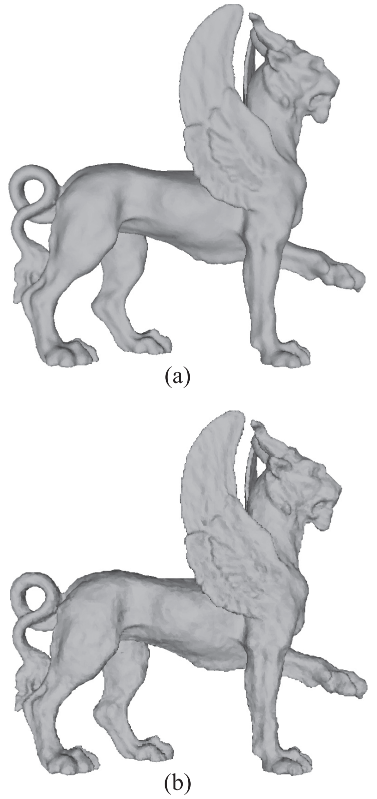
Table 1 Coding performances of the proposed scheme, TG coder and GA coder of Gumhold et al. [10] at a higher coding rate.

Model (No. of vert.)	Coding rate (bpv)		PSNR (dB)	
	TG	GA & proposed	TG & GA	proposed
Feline (49,864)	14.17	13.74	89.00	89.67
Horse (19,851)	15.16	14.26	88.49	88.86
Dino (14,070)	17.40	17.01	88.41	89.67



(a) (b)

Fig. 13 Rendering images of reconstructed model of "Venus"; (a) encoded by the proposed scheme (4.13 bpv, 78.62 dB), (b) encoded by the TG coder (4.21 bpv, 65.30 dB).



(a)

(b)

Fig. 14 Rendering images of reconstructed model of "Feline"; (a) encoded by the proposed scheme (5.11 bpv, 72.01 dB), (b) encoded by the TG coder (5.13 bpv, 64.78 dB).

the linear prediction adopted in MPEG-4 [5], [6], and the dotted line of longer length shows the TG coder [7] which is often used as a representative scheme with perfect reconstruction property. The proposed scheme improves coding performance significantly at lower coding rate among the schemes with perfect reconstruction property. Also, in order to evaluate the coding performance of the proposed scheme at a higher coding rate, it was compared with the scheme of Gumhold et al. [10] (we call it GA coder) which improved the prediction of TG coder. Table 1 shows PSNR of the proposed scheme and GA coder at the same coding rate with the coding performance of TG coder for 3-D meshes "Feline", "Horse", and "Dino". The proposed scheme gives better PSNR values at a higher coding rate. The rendered images of reconstructed 3D meshes for the proposed scheme and TG coder at approximately the same coding rate are shown in Figs. 13 and 14. The proposed scheme can be faithfully reconstructed to the original 3-D mesh.

6. Conclusions

In this paper, we proposed a novel coding scheme for the triangular mesh geometry. In order to apply the 2-D coding scheme, we structured the geometry data of the triangular mesh on the 2-D plane by the process of structuring surrounding vertices. As a coding method, we applied the connectivity-oriented multiresolution decomposition and the vector SFQ. By using the connectivity of the triangular mesh during the multiresolution decomposition, we could sufficiently consider correlations among vertices. The vector SFQ could reduce redundancies of decomposed coefficients at similar position in the different components of each level. Experimental results have shown that the proposed scheme gives better coding performances than conventional schemes. In the proposed scheme, the structuring of surrounding vertices must be performed at the decoder stage too, which leads to increasing the calculation overhead. Therefore, we should introduce the coding scheme for the connectivity data of the triangular mesh based on the structuring of surrounding vertices. Also we intend to consider influences to the reconstructed 3-D geometry, especially small features and sharp edges, from changing the coding rate of the proposed scheme.

References

[1] F. Mintzer, "Images on the path to the digital museum," Proc. Int. Conf. Image Processing, pp.555–559, Thessaloniki, Greece, 2001.

[2] J. Peng, C.-S. Kim, and C.C.J. Kuo, "Technologies for 3-D mesh compression: A survey," J. Vis. Commun. Image Represent., vol.16, pp.688–733, 2005.

[3] J.-L. Dugelay, A. Baskurt, and M. Daoudi, 3D Object Processing: Compression, Indexing and Watermarking, pp.45–86, John Wiley & Sons, 2008.

[4] M. Deering, "Geometry compression," SIGGRAPH'95, pp.13–22, 1995.

[5] G. Taubin, W.P. Horn, F. Lazarus, and J. Rossignac, "Geometry coding and VRML," Proc. IEEE, vol.86, no.6, pp.1228–1243, June 1998.

[6] G. Taubin and J. Rossignac, "Geometry compression though topological surgery," ACM Trans. Graphics, vol.17, no.2, pp.84–115, April 1998.

[7] C. Touma and C. Gotsman, "Triangle mesh compression," Proc. Graphics Interface'98, pp.26–34, J1998.

[8] J. Rossignac, "Edgebreaker: Connectivity compression for triangle meshes," IEEE Trans. Vis. Comput. Graph., vol.5, no.1, pp.47–61, 1999.

[9] P. Alliez and M. Desbrun, "Valence-driven connectivity encoding for 3D meshes," Comput. Graph. Forum, vol.20, no.3, 2001.

[10] S. Gumholda and R. Amjoun, "Higher order prediction for geometry compression," Proc. Shape Modeling International 2003, pp.59–66, 2003.

[11] M. Isenburg and J. Snoeyink, "Early-split coding for triangle mesh connectivity," Proc. Graphics Interface 2006, pp.89–97, June 2006.

[12] U. Bayazit, O. Orcay, U. Konur, and F.S. Gurgen, "Predictive vector quantization of 3-D mesh geometry by representation of vertices in local coordinate systems," J. Vis. Commun. Image R, vol.18, pp.341–353, 2007.

[13] A. Honda, K. Fukuda, and A. Kawanaka, "Permuting and lifting wavelet coding for structured 3-D geometry data with expanded nodes," Proc. Int. Conf. Image Processing 2005, vol.1, pp.761–764, 2005.

[14] A. Kawanaka and S. Watanabe, "Permuting and lifting Wavelet coding for structured geometry data of 3-D polygonal mesh," IEICE Trans. Inf. & Syst., vol.E90-D, no.9, pp.1439–1447, Sept. 2007.

[15] S. Watanabe and A. Kawanaka, "Triangular mesh geometry coding with connectivity oriented multi-resolutional decomposition based on structuring of surrounding vertices," J. ITE, vol.62, no.9, pp.1439–1442, 2008.

[16] H. Hoppe, "Progressive meshes," Proc. SIGGRAPH'96, pp.99–108, 1996.

[17] M. Garland and P. Heckbert, "Surface simplification using quadric error metrics," Proc. Computer Graphics Ann. Conf. Series, pp.209–217, Aug. 1997.

[18] A. Khodakovsky, P. Schroder, and W. Sweldens, "Progressive geometry compression," Proc. SIGGRAPH'00, pp.271–278, 2000.

[19] X. Gu, S. Gortler, and H. Hoppe, "Geometry images," Proc. SIGGRAPH'02, pp.355–361, 2002.

[20] H. Hoppe and E. Praun, "Shape compression using spherical geometry images," MINGLE 2003 Workshop, Advances in Multiresolution for Geometric Modelling, ed. N. Dodgson, M. Floater, and M. Sabin, pp.27–46, 2003.

[21] Z. WenBo, L. Bo, and Z. Hongbin, "Rate-distortion optimized progressive geometry compression," Proc. International Conference on Computer Graphics, Imaging and Visualisation (CGIV'06), pp.207–210, 2006.

[22] S.T. EL-Leithy and W.M. Sheta, "Wavelet-based geometry coding for 3D mesh using space frequency quantization," Proc. IEEE Symposium on Computers and Communications (ISCC'08), pp.1034–1039, 2008.

[23] P. Schroder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on the sphere," Proc. SIGGRAPH'95, pp.161–172, 1995.

[24] A.W.F. Lee, W. Sweedens, P. Schroder, P. Cowsar, and D. Dobkin, "MAPS: Multiresolution adaptive parameterization of surfaces," Proc. SIGGRAPH'98, pp.95–104, 1998.

[25] D.S. Taubman and M.W. Marcellin, JPEG2000: Image compression fundamentals, standards and practice, Kluwer Academic Publishers, 2002.

[26] W. Sweldens, "The lifting scheme: A custom-design construction of bi-orthogonal wavelets," Applied and Computational Harmonic Analysis, vol.3, no.2, pp.186–200, 1996.

[27] J.M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," IEEE Trans. Signal Process., vol.41, no.12, pp.3445–3462, 1993.

[28] A. Said and W.A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. Circuits Syst. Video Technol., vol.6, no.3, pp.243–250, June 1996.

[29] Z. Xiong, K. Ramchandran, and M.T. Orchard, "Space-frequency quantization for wavelet image coding," IEEE Trans. Image Process., vol.6, no.5, pp.677–693, May 1997.

[30] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," Computer Graphics Forum, vol.17, pp.167–174, June 1998.

Shuji Watanabe was born in Chiba, Japan in 1982. He entered the Department of Electrical and Electronics Engineering, Faculty of Science and Technology, Sophia University in 2002. He finished the undergraduate course in 2005 and entered soon the Master's Program in Electrical and Electronics Engineering, Graduate School of Science and Technology, Sophia University, passing through the skipped entrance examination. He received the M.S. degree in electrical and electronics engineering from Sophia University, Tokyo, Japan, in 2007. He is with the Doctor Program in Electrical and Electronics Engineering, Graduate School of Science and Technology, Sophia University. He is currently interested in image processing and 3D model data compression.

Akira Kawanaka was born in Hiroshima, Japan in 1955. He received the B.E. and Ph.D. degrees in electrical and electronics engineering from Sophia University, Tokyo, Japan, in 1977 and 1982, respectively. He was a research associate on the Faculty of Science and Technology at Sophia University in 1982, and was also a research associate at the Institute of Industrial Science, the University of Tokyo from 1983 to 1987. Since 1987, he has been with the Department of Electrical and Electronics Engineering, Sophia University, where he is currently a Professor. His research interests include multidimensional signal processing, image and 3D model data compression, and pattern recognition.