# AN EFFICIENT PROGRESSIVE ENCODING TECHNIQUE FOR BINARY VOXEL MODEL USING PYRAMIDAL APPROACH

*Musik Kwon†, Kyoung Mu Lee‡, and Sang-Uk Lee†*

† M. Kwon and S. U. Lee are with the Signal Processing Lab., School of Electrical Engineering,
Seoul National University, 151-742, Seoul, Korea
(E-mail: musik@diehard.snu.ac.kr, sanguk@sting.snu.ac.kr).
‡K. M. Lee is with the Dept. of Electrical and Electronics Engineering,
Hong-Ik University, Seoul, 121-791, Korea (E-mail: kmlee@wow.hongik.ac.kr).

**ABSTRACT**

In this paper, we propose a progressive encoding method for geometric information of 3D object, which is represented with binary voxels. By the pyramidal decomposition, the proposed algorithm first creates multi-resolution models for the 3D object. Then, each resolution model is predicted from its lower resolution model, and the prediction errors are encoded with an arithmetic coding method. To yield high compression ratio, the lower resolution model is partitioned into the inside, boundary, and outside regions. This partitioning method significantly reduces the amount of data to be encoded, since the prediction errors are concentrated near the boundary region in general. Moreover, the local concavity of each boundary voxel is used as the context for the arithmetic coding to further increase the compression efficiency. It is demonstrated by intensive simulation results that the proposed algorithm provides better coding gain than the conventional mesh-based algorithm, especially for natural objects such as human body, terrain, and animal.

## 1. INTRODUCTION

During the past decade, the surface graphics has been investigated with amazing speed, while volume graphics has been employed only in specific fields, such as medical applications, which should represent the internal information of objects as well as the surface information. However, the recent progress in volume graphics makes it possible to enhance its applications to the general 3D objects [1]. Owing to the preferable property, $i.e.$, the volumetric data yields a uniform grid structure, it has powerful potential that the established signal processing techniques, such as the transform and the filtering, can be easily applied, simply by extending the dimension. Furthermore, the current advanced hardware and augmented memory enable fast processing for visualizing volume data[3], which has been one of main bottlenecks in the field of volume graphics for a long time.

In this paper, we propose an efficient encoding method for the progressive transmission of 3D volume data. The progressive method can not only cope with variable bandwidth requirements for many networking users, but also offer multi-resolution models. The multi-resolution representation enables easier manipulation of 3D data in the interactive virtual reality applications, such as the walk-through and fly-trough scenarios, by changing the level of detail for each object according to its distance from viewer. For realizing the progressiveness, we restructure the existing pyramidal encoding methods for 2D images[4].

## 2. 3D LAPLACIAN PYRAMID OF BINARY VOXEL MODEL

A binary voxel model $\mathbf{G_0}$ can be represented as a set of binary numbers defined on integer grid points in 3D space, given by

$$\mathbf{G_0} = \{G_0(x,y,z) = 0 \text{ or } 1 : x = 0,1,\cdots,L-1; \\ y = 0,1,\cdots,M-1;\ z = 0,1,\cdots,N-1\}. \quad (1)$$

The "1" voxels represent the inside region of 3D object, while the "0" voxels represent the outside region of the object, $i.e.$, background region, and $L \times M \times N$ is the resolution for the voxel model. For the sake of convenience, we assume that $L = p \times 2^K, M = q \times 2^K, N = r \times 2^K$, where $p, q, r, K$ are positive integers.

In this paper, the voxel model is obtained by voxelizing or 3D scan-converting triangular mesh model[1]. But, the voxel model, obtained from the voxelization of mesh model, represents only the surface region of 3D object, whose inner part is empty. Therefore, the inside filling operation of mathematical morphology is accomplished onto the surface voxel model, yielding the inside-filled voxel model. Fig. 1 shows examples of binary voxel models. Fig. 1(a) is the 'Bunny' triangular mesh model. Fig. 1(b) is the voxel model with $64 \times 64 \times 64$ resolution, and Fig. 1(c) is the voxel model with $512 \times 512 \times 512$ resolution. Fig. 1(d) is a resultant model rendered by shading each voxel with a normal vector, which is calculated by the planar approximation of $7 \times 7 \times 7$ adjacent voxels around the center voxel.

The 3D Gaussian pyramid of binary voxel model is obtained by recursive decimations of the input model. Let $\mathbf{G_k}$ denote the reduced model after the $k$-th decimation, given by

$$\mathbf{G_k} = \{G_k(x,y,z) : x = 0,1,\cdots,L/2^k-1; \\ y = 0,1,\cdots,M/2^k-1;\ z = 0,1,\cdots,N/2^k-1\} \quad (2)$$

where $G_k(x,y,z) = G_{k-1}(2x,2y,2z)$.

Let $\mathbf{U_k}$ denote the upsampled voxel model of $\mathbf{G_{k+1}}$, then the region in $\mathbf{U_k}$, whose voxel values are determined by $\mathbf{G_{k+1}}$, is referred to as the deterministic region $\mathbf{DR_k}$.

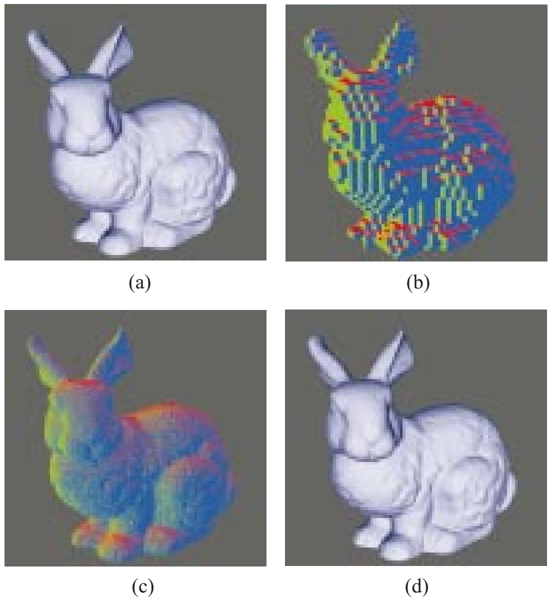


**Fig. 1**. Mesh model and binary voxel model for Bunny.(a) original mesh model Bunny, (b) voxel model Bunny64 which scan-converted with $64 \times 64 \times 64$ resolution,(c) voxel model Bunny512 which scan-converted with $512 \times 512 \times 512$ resolution, (d) normal shading for Bunny512.

After upsampling, proper voxel values should be assigned on the zero-padded region $\mathbf{DR_k^c}$. During this assignment, we should carefully estimate whether each voxel on $\mathbf{D_k^c}$ is likely to be placed inside the 3D object or outside. To this end, we classify $\mathbf{DR_k^c}$ into three parts, depending on the estimation value and probabilistic certainty. The first part is denoted by $\mathbf{I_k^0}$, whose estimation values are "0", since the voxels are considered as the outer part of the object. The second one is denoted by $\mathbf{I_k^1}$, whose estimation values are "1" since the voxels are considered as the inner part of the object. The remaining part, $\mathbf{DR_k^c} - \mathbf{I_k^0} - \mathbf{I_k^1}$, is the boundary region, where it is difficult to estimate the voxel values since they may be located inside or outside of the object. For the sake of computational simplicity, we set the estimated voxel values of the whole remaining part to "0". Though the remaining region is uniformly estimated as value "0" in this procedure, the voxels will be efficiently classified by their convexness just before the arithmetic coding. The $\mathbf{I^0}$ is a part in $\mathbf{D_k^c}$ where none of $3 \times 3 \times 3$ neighbor voxels belongs to deterministically inside parts in $\mathbf{D_k}$, and $\mathbf{I^1}$ is union of all the subsets of Fig.2 ,$i.e.$, $\mathbf{I_k^{1X}}$, $\mathbf{I_k^{1Y}}$, $\mathbf{I_k^{1Z}}$, $\mathbf{I_k^{1XY}}$, $\mathbf{I_k^{1YZ}}$, $\mathbf{I_k^{1ZX}}$, and $\mathbf{I_k^{1XYZ}}$.

Then, the resultant voxel model $\mathbf{E_k}$, after upsampling and estimation, can be written as

$$\mathbf{E_k} = \{E_k(x,y,z); x = 0, \cdots, L/2^k - 1; \\ y = 0, \cdots, M/2^k - 1; z = 0, \cdots, N/2^k - 1\}, \qquad (3)$$

where,

$$E_k(x,y,z) = \begin{cases} G_{k+1}(x/2, y/2, z/2) & \text{if } (x,y,z) \in \mathbf{D_k}, \\ 1 & \text{if } (x,y,z) \in \mathbf{I_k^1}, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, we obtain the difference model between $\mathbf{G_k}$ and $\mathbf{E_k}$, denoted by $\mathbf{L_k}$, whose spatial redundancy is effectively removed. As a result, most "1" voxels resides near the boundary region. The Laplacian voxel model is defined as,

$$\mathbf{L_k} = \begin{cases} \mathbf{G_K} & \text{if } k = K, \\ \mathbf{G_k} - \mathbf{E_k} & \text{if } k = 0, 1, 2, \cdots, K-1. \end{cases} \qquad (4)$$

The set of these voxel models $\{\mathbf{L_k}, k = 0, 1, \cdots, K\}$ forms the Laplacian pyramid.

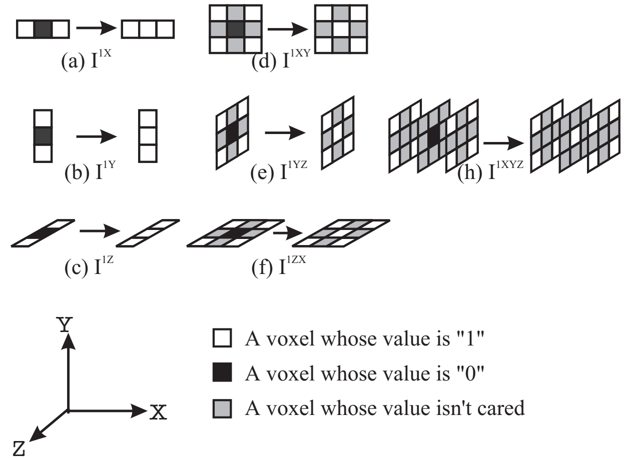


**Fig. 2**. The cases to be assigned with "1" voxel during estimation step.

## 3. SET PARTITIONING

In the previous section, the whole region of the $k$-th reduced voxel model $\mathbf{G_k}$ is implicitly partitioned into three regions as follows:

- Deterministic region $\mathbf{DR_k}$ : The region whose voxel values are deterministically decided from the coarse model $\mathbf{G_{k+1}}$,
- Almost certain region $\mathbf{AR_k} = \mathbf{I_k^0} \cup \mathbf{I_k^1}$ : The region whose voxel values are almost certainly estimated from the neighboring voxel values,
- Uncertain region $\mathbf{UR_k} = \mathbf{DR_k^c} - \mathbf{I_k^0} - \mathbf{I_k^1}$ : The region whose voxel values are uncertain to estimate.

Consequently, most "1" voxels of the Laplacian model $\mathbf{L_k}$ belong to the uncertain region $\mathbf{UR_k}$, which corresponds to the boundary of the 3D object.

Fig. 3 illustrates an example of $\mathbf{G_{k+1}}$, $\mathbf{G_k}$ and $\mathbf{L_k}$ for simplified 2D case. When the model includes uneven or locally sharp areas in the surface as depicted in the circles of Fig. 3(a), the difference voxels may occur within the almost certain region as shown in the circles of Fig. 3(b). This incorrectly estimated area within $\mathbf{AR_k}$ is defined as the exceptional region, given by

$$\mathbf{XR_k} = \{(x,y,z) : (x,y,z) \in \mathbf{AR_k}, \\ G_k(x,y,z) \neq E_k(x,y,z)\}. \qquad (5)$$

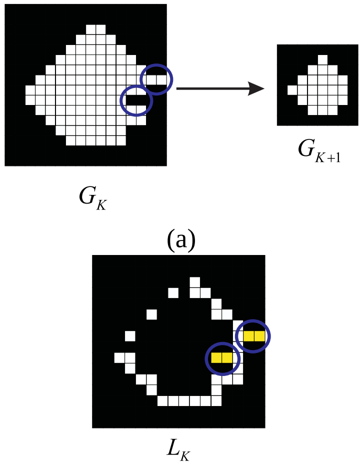


**Fig. 3**. An example of reduced procedure by decimation for simplified 2D case (a) Decimation (b) Interpolation ; upsampling and estimation (c)The resultant Laplacian model.

## 4. ENCODING PROCEDURE

For the purpose of progressive transmission from the coarsest level $\mathbf{G_K}$ to the finer resolutional level $\mathbf{G_k}$, we encode $\mathbf{L_K}, \mathbf{L_{K-1}}, \cdots, \mathbf{L_k}$ sequentially. First, since the coarsest level $\mathbf{L_K}$ is just $\mathbf{G_K}$, we encode $\mathbf{L_K}$ differently from the finer level. On the other hand, $\mathbf{L_k}(k \neq K)$ is first split into two subsets $\mathbf{UR_k}$ and $\mathbf{XR_k}$. Thus, we apply two different encoding methods, which are appropriate to the subsets, respectively. In this section, we describe the encoding method for these three cases. For the coarsest level $\mathbf{L_K}$, the encoder employs a differential pulse code modulation (DPCM) predictor, followed by the arithmetic coder. The prediction is performed using the preceding voxel in the raster scan order, so that the predictive error is the difference between the present voxel and the preceding voxel. After the prediction, the error value is encoded by the arithmetic coder.

For the uncertain region in $\mathbf{L_k}$, we need not encode the location of $\mathbf{UR_k}$. But, we encode the voxels for the uncertain region in a sequential way. But, the estimated model $\mathbf{E_k}$ contains the locally-convex or locally-concave regions on the surface. Considerable portion of the locally-convex or concave region are generated, due to the property of the interpolation where "0" values are assigned to the voxels in the uncertain region. This indicates that $\mathbf{E_k}$ is more wrinkle than the original model $\mathbf{G_k}$. Therefore, if the decoder uses the local concavity as the context for the arithmetic coding, further coding gain can be achieved. The local concavity $\mathcal{LC}$ of a voxel $(x, y, z)$ in $\mathbf{UR_k}$ is defined as the sum of the values of $3 \times 3 \times 3$ neighboring voxels, given by

$$\mathcal{LC} = \frac{1}{N} \sum_{i=-1}^{1} \sum_{j=-1}^{1} \sum_{k=-1}^{1} V_k(x+i, y+j, z+k), \qquad (6)$$

where

$$V(x,y,z) = \begin{cases} L_k(x,y,z) & \text{if } V_k(x,y,z) \text{ encoded,} \\ E_k(x,y,z) & \text{otherwise.} \end{cases} \qquad (7)$$

On the assumption of smooth surface, the voxel value is likely to be "1", as $\mathcal{LC}$ approaches "1". If not, $i.e.$, the real voxel value is "0", there happens the locally concave region in the surface, which is generally unusual case. On the contrary, the small value of $\mathcal{LC}$ indicates that the voxel value is "0" with high probability. Similar to the encoding of $\mathbf{L_K}$, the voxel values of $\mathbf{UR_k}$ are encoded in the raster scan order. But, the arithmetic coding employs the context in (6) to further improve the coding gain. Notice that the context is only based on the already encoded voxels, thus the decoder can use the same context with the encoder.

The exceptional region $\mathbf{XR_k}$ is the incorrectly estimated parts among the $\mathbf{AR_k}$. The positions of the voxels are DPCM-predicted in the raster scan order, and the prediction errors are encoded with the arithmetic coder

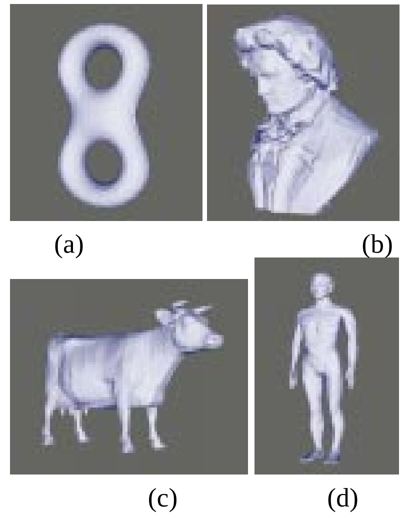


**Fig. 4**. An example of models used in the experiments (a) Eight (b) Beethoven (c) Cow (d) Man.

## 5. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed algorithm, experiments are carried out on several voxel models, which are scan-converted from mesh models. The mesh models are illustrated in Fig. 4. In the experiments, the resolution of scan-converting is fixed to $256 \times 256 \times 256$. For the objective measurement, the distortion between original model and reconstruction model is defined as

$$D(\mathbf{G_0}, \hat{\mathbf{G_k}}) = \sum_{x=0}^{L-1} \sum_{y=0}^{M-1} \sum_{z=0}^{N-1} \frac{1}{L \times M \times N} (G_0(x,y,z) - \hat{G_k}(x,y,z))^3 \qquad (8)$$

where $\hat{\mathbf{G_k}}$ is the 0th order interpolated version of $\mathbf{G_k}$ with size of $L \times M \times N$. Since the original model and the reconstructed model differ in the resolutions, we need to arrange the resolution to the same level each other. By the measurement, we evaluate the rate-distortions for several models which are shown in Fig.5. The decoded pyramidal levels are illustrated in Fig.6, together with the measured distortion and the allocated bitrates.

Since the proposed algorithm provides both lossy and lossless compression, we can compare with Taubin's lossless mesh model compression method[5]. In the Taubin's algorithm, the bounding box and vertex coordinate for triangular mesh are pre-quantized before lossless encoding. In

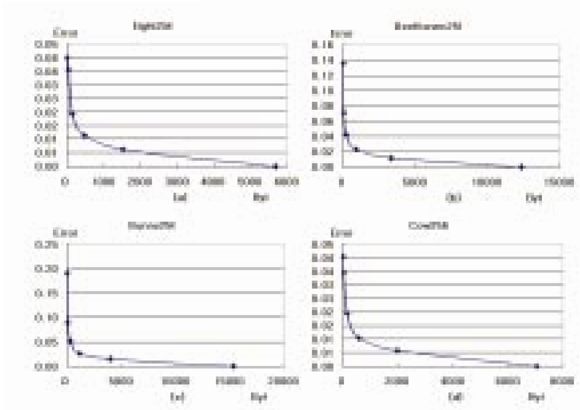

**Fig. 5**. The rate-distortion for several voxel models using the proposed algorithm.

this case, if the scan-converted resolution of voxel model is same as the quantized resolution of mesh model, it is argued that the errors for both models are placed within equivalent bound. Under this assumption, the allocated bitrates for lossless encoding of each model are provided in the Table 1. From the results, it is observed that proposed algorithm yields much higher coding gain than Taubin's method. But, for the model having simple meshes, it is expected that the Taubin's algorithm yield much higher coding gain than the proposed algorithm. However, because the majority of natural objects such as human body, animals, and terrains, have smooth and curved surface, it is strongly believed that the proposed algorithm could provide various and practical applications in real world.

**Table 1**. The comparison of proposed algorithm with Taubin's mesh compression algorithm

| Mesh model | Bunny(Q=256) | Man(Q=256) |
|---|---|---|
| Compressed file size(bytes) | 52,147 | 25,468 |
| Voxel model | Bunny256 | Man256 |
| Compressed file size(bytes) | 20,278 | 4,926 |
| Ratio | 2.57:1 | 5.17:1 |

## 6. CONCLUSION

In this paper, we proposed a progressive encoding method for 3D binary voxel model which is represented as 3D array and the inside region is filled with value "1". By the pyramidal decomposition, the proposed method first creates multi-resolution models of 3D object. Then, each resolution model is predicted from its lower resolution model, and the prediction errors are encoded with an arithmetic coding method. Since the prediction errors are concentrated in the boundary region of 3D object, the quantitative specification for the boundary region is needed. Thus, the whole region of voxel model is partitioned into the inside, boundary, and outside regions, depending on the estimation value and probabilistic certainty. The partitioning method significantly reduces the amount of date to be encoded.

Moreover, the local concavity of each boundary voxel is

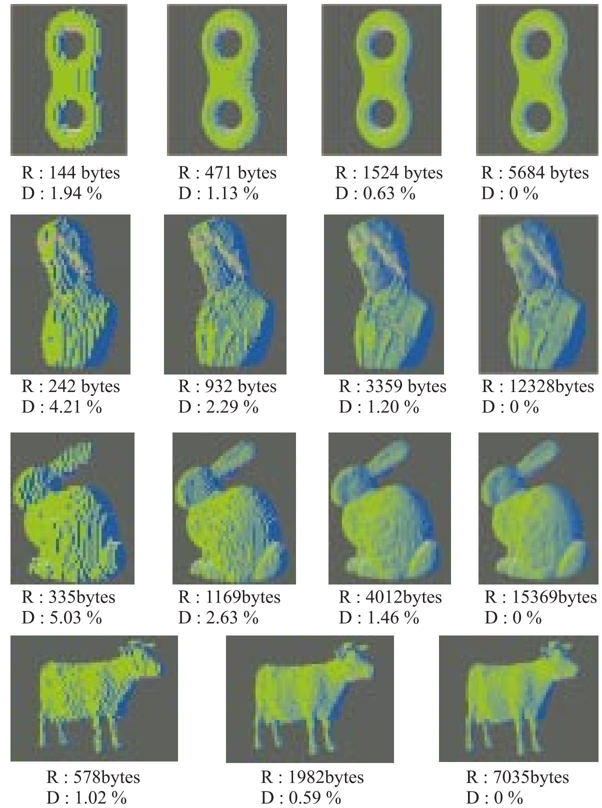


**Fig. 6**. The decoded pyramid levels for several models with the measurement distortion and the allocated bitrates.

used as the context for the arithmetic coding to further enhance the coding gain. It is demonstrated by intensive simulation results that the proposed algorithm provides better coding gain than the conventional mesh-based algorithm, especially for natural objects in the real world.

## 7. REFERENCES

[1] A. Kaufman, D. Cohen, and R. Yagel, "Volume graphics," *IEEE Computer* , vol. 26, no. 7, pp. 51-64, Jul., 1993.

[2] D. Cohen-Or and A. Kaufman, "Fundamentals of surface voxelization," *Graphical Models and Image Processing,* vol. 6, no. 6, pp. 453-461, Nov. 1995.

[3] C. Rezk-Salama et al, "Interactive Volume Rendering on Standard PC Graphics Hardware Using Multi-Texture and Multi-stage Rasterization,"*Proceedings of SIGGRAPH,* pp. 109-118, 2000.

[4] B. Aiazzi, L. Alparone, S. Baronti, F. Lotti,"Lossless Image Compression by Quantization Feedback in A Content-driven Enhanced Laplacian Pyramid,"*IEEE Trans. on Image Processing,* vol. 66, pp. 831-843, June. 1997.

[5] G. Taubin and J. Rossignac,"Geometric Compression through topological surgery,"*Research Report RC-20340, IBM Research Devision,* Jul. 1997.