УДК 621.397:004.932

3D IMAGE MESH ENTROPY CODING

Samus N.S., Osharovskaya E.V.
O.S. Popov Odessa national academy of telecommunications,
1 Kovalska St., Odessa, 65029, Ukraine.
osharovskaya@mail.ru

СІТКОВЕ ЕНТРОПІЙНЕ КОДУВАННЯ 3D ЗОБРАЖЕННЯ

Самусь Н.С., Ошаровська О.В.
Одеська національна академія зв'язку імені О.С. Попова,
65029, Україна, м. Одеса, вул. Ковальська,1.
osharovskaya@mail.ru

СЕТОЧНОЕ ЭНТРОПИЙНОЕ КОДИРОВАНИЕ 3D ИЗОБРАЖЕНИЙ

Самусь Н.С., Ошаровская Е.В.
Одесская национальна академия связи имени А.С. Попова,
65029, Украина, г. Одесса, ул. Ковальская,1.
osharovskaya@mail.ru

Abstract. In this article we have introduced a compressed representation for triangular meshes. Because proximity in this vertex spanning tree often implies geometric proximity of the corresponding vertices, we can use ancestors in the tree to predict vertex positions, and thus only need to encode the difference between predicted and actual vertex positions. When vertex coordinates are quantized (i.e., truncated to the nearest number in a fixed-point representation scheme), these corrective vectors have on average smaller magnitude than absolute positions and can therefore be encoded with fewer bits. Furthermore, the corrective terms are then compressed by entropy encoding using, for example, Huffman or arithmetic coding as in the JPEG/MPEG standards.

Key words: 3D image, mesh, entropy coding.

Анотація. У цій статті надано результати стиснення для трикутних сіток. Через близькість вершин сполучного дерева, що передбачає геометричну близькість відповідних вершин, ми можемо використовувати предків в дереві для передбачення позиції вершин, і, таким чином, тільки потрібно кодувати різницю між передбаченням і реальними координатами вершин. Коли координати вершин квантуються (тобто, зрізані до найближчого числа у схемі подання з фіксованою точкою), ці коригувальні вектори мають у середньому меншу розрядність, ніж величини абсолютних координат і, отже, можуть бути закодовані з меншою кількістю бітів. Крім того, коригувальні вектори потім стискуються за допомогою ентропійного кодування, використовуючи, наприклад, кодування Хаффмана або арифметичне, як в стандартах JPEG / MPEG.

Ключові слова: 3D зображення, сітка, ентропійне кодування.

Аннотация. В этой статье представлены результаты энтропийного сжатия для треугольных сеток. Из-за близости вершин в связующем дереве, подразумевая геометрическую близость соответствующих вершин, мы можем использовать предков в дереве для предсказания позиции вершин, и, таким образом, кодировать только разность между предсказанной и реальной позицией вершин. Когда координаты вершин квантуются (т.е., усекаются до ближайшего числа в схеме представления с фиксированной точкой), эти корректирующие векторы имеют в среднем меньшую разрядность, чем величины абсолютных координат и, следовательно, могут быть закодированы с меньшим количеством битов. Кроме того, корректирующие вектора затем сжимаются с помощью энтропийного кодирования, используя, например, кодирование Хаффмана или арифметическое, как в стандартах JPEG / MPEG.

Ключевые слова: 3D изображение, сетка, энтропийное кодирование.

Research efforts on 3DTV technology have been reinforced recently, covering the whole media processing chain from the sensor of video signal to display. Different 3DTV systems based on different 3-D scene representations that integrate various types of data. Efficient coding of these data determines the success of the implementation of 3DTV. Compression of pixel-type data including stereo video, multiview video, and associated depth or disparity maps extends available principles of classical video coding. Powerful algorithms and open international standards for multiview video coding and coding of video plus depth data are available and under development. Compression of 3-D mesh models has also reached a high level. Many proposed technologies, including the latest MPEG-4 3D mesh coding (3DMC) standard, achieve high compression efficiency [1]. 3D mesh compression has been extensively researched in many studies with a focus on compression efficiency.

The compression ratio is determined mainly by the total number of runs of the vertex and triangle trees. The optimal compression is achieved by minimizing this number. The compression of both static and dynamic meshes over time has been investigated. Temporal prediction is an important mechanism to remove redundancy from animated 3-D mesh sequences.

During the last decades there has been a lot of research in the area of static mesh compression. A mesh can be simply represented as the set of vertices, edges, faces together with their incidence relationships. 3D meshes are visualization of 3D objects using vertices (geometry), edges, faces, some attributes like surface normals, texture, color, etc, and connectivity. 3D points $\{v_1, v_2,..., v_n\} \in V$ *in* $\boldsymbol{R}^3$ are called vertices of a 3D mesh. The convex hull of two vertices in $\boldsymbol{R}^3$, *conv* $\{v_n, v_m\}$ is called an *edge*. So an edge is mapped to line segment in $\boldsymbol{R}^3$ with end points at v_n and v_m. *Face* of a triangular mesh is a surface which is *conv* $\{v_n, v_m, v_k\}$. Thus, a face is mapped to a surface in $\boldsymbol{R}^3$ that is enclosed by the edges incident to the vertices v_n; v_m; v_k. A face may have no direction or its direction can be determined using the surface normals data. The additional attributes of a mesh are mostly carried by the vertices. That information can be extended along the edges and the faces using linear interpolation or other techniques. The connectivity information summarizes which mesh elements are connected to each other. Edges $\{e_1,...,e_n\} \in E$ are incident to its two end vertices. Faces $\{f_1,..., f_n\} \in F$ are surrounded by its composing edges and incident to all the vertices of its incident edges. The edges have no direction. Two types of mesh connectivity are common in mesh representations. Edge Connectivity is the list of edges in the mesh and Face Connectivity list of faces in the mesh. In a triangular mesh, since all the vertices incident to a face lie in a plane, the face also lies in a plane. In polygonal meshes the number of the vertices that are incident to the face, is four or more. So face of a polygonal mesh not necessarily lie in a plane [2]. Vertices of a mesh can be incident to any number of edges. The number of the edges that are incident to a vertex is named as the valence of the vertex [3]. The number of the edges that are incident to a face is named as the degree of a face [3]. The number of the faces incident to an edge and number of the face loops incident to a vertex, are important concepts while defining, if the mesh is manifold or non-manifold. A 2-manifold is a topological surface where every point on the surface has a neighborhood topologically equivalent to an open disk of R^{2}" [4]. If the neighborhood of a point on the surface is equivalent to a half disk than the mesh is manifold with boundary [4].

Two other important concept about meshes are shell and genus. Shell is a part of the mesh that is edge-connected. The genus of a mesh is an integer that can be derived from the number of closed curves that can be drawn on the mesh without dividing it into two or more separate pieces. It is equal to the number of handles on the mesh object [3]. a torus is genus-1 since it has one hole and sphere is genus-0 since it has no hole [3].

We can consider the incidence relationships (i.e. what faces are incident on a vertex, what edges are incident on a face, etc...) the mesh connectivity and the vertex positions the mesh geometry. Most mesh compression techniques have treated mesh geometry and the mesh connectivity separately.

In 1996, Taubin and Rossignac [5] proposed the method Topological Surgery (TS), which was the first method for lossless compression of mesh connectivity and compression of locations of mesh vertices with controllable loss. They decomposed connectivity in so called vertex and triangle spanning trees, which were encoded. Linear predictive coding was employed for compression of vertex locations. They were predicted in an order guided by connectivity using already encoded locations. Later, Taubin *et al.* [6] extended the TS approach in order to obtain a Levels of Detail representation of a compressed static mesh, providing progressive decoding from low to high resolution. They introduced the Progressive Forest Split (PFS) scheme using a forest, i.e., a set of trees, in order to describe connectivity in different resolution levels. Both, TS and PFS, are building the basis of the MPEG-4 3-D Mesh Coding standard (3DMC tools) for Single Resolution and Progressive Levels of Detail static mesh compression. Many improvements and generalizations upon the TS approach were presented later. We want to point out the Edgebreaker technique [7] of Rossignac. Method uses a finite state machine to compactly describe mesh connectivity. They apply face based vertex traversal emitting one out of five symbols each time a new vertex is visited, with each symbol describing the configuration of this vertex relative to the traversed region. Subsequently this stream of only five different symbols is entropy encoded. As we show in [8] Edgebreaker recurses on the right subtree and then the left. Edgebreaker can compress the connectivity of the mesh to near optimal rates, normally around 2 bits/vertex. Algorithm edgebreaker visited triangles in a spiral (" in depth ") triangular protocol binding tree lines and creates labels - clers; one mark for each triangle that indicates the decoding, i.e., as can be retrieved by attaching new mesh triangles to previously restored.

Developed many software visual design . Use one of them - MeshLab - to build a dog image and mesh representation of the image with the selected area of the screen dots represented in the program MeshLab (Figure 1).

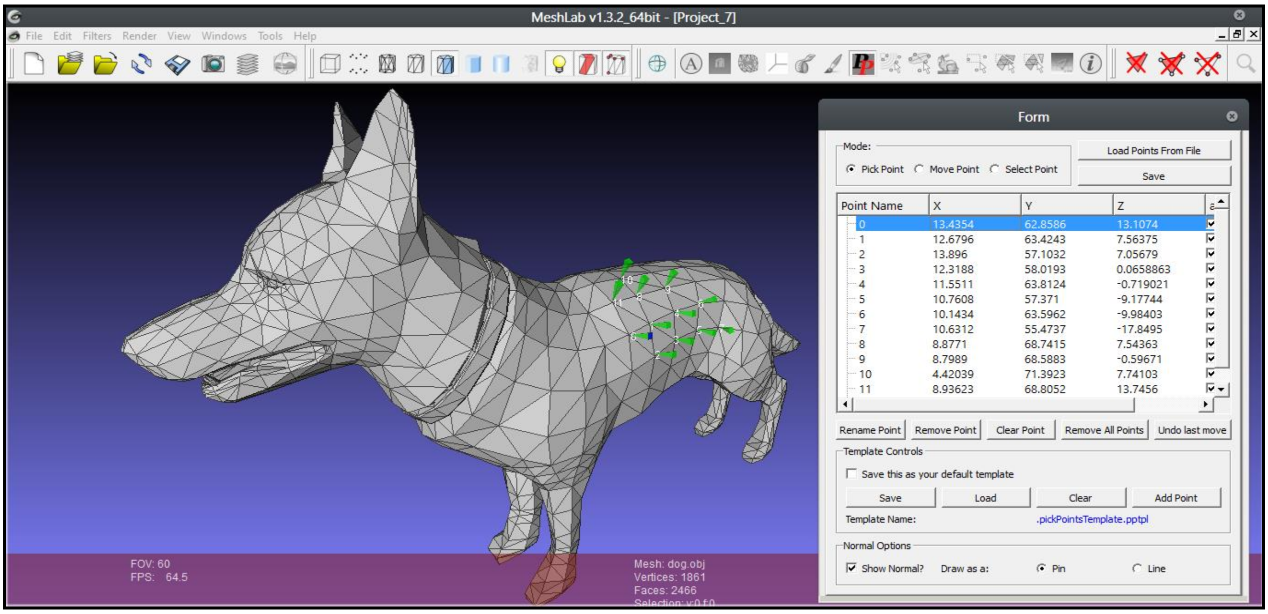

Figure 1 – Mesh representation of the image.

Then select a random station for further processing. This action can be seen in Figure 2.

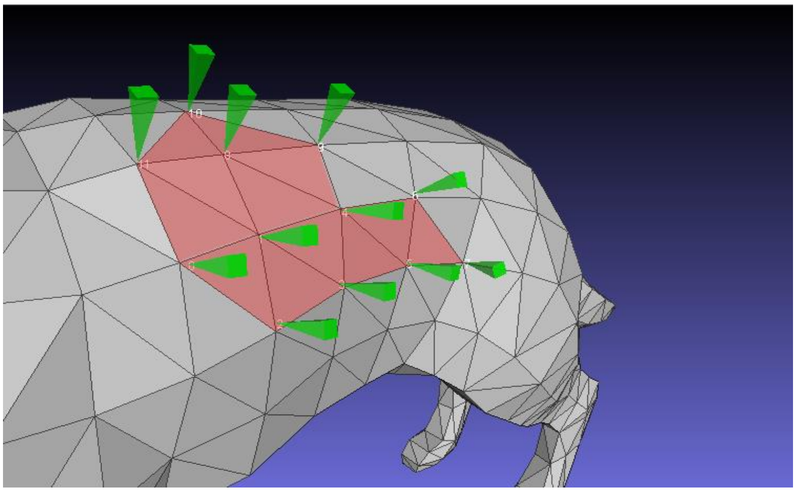

Figure 2 – The selected image area to encode using MeshLab

In the construction of any three-dimensional object using coordinate system. In general, this three-dimensional coordinate system X, Y, Z. We will produce in table 1 all values of points on the selected area.

Table 1 – Normalized values and integer values of each coordinate

№ point	coordinate *X*		Coordinate *Y*		Coordinate *Z*	
	normalized value	integer values	normalized value	integer values	normalized value	integer values
0	0,951	951	0,464	464	0,980	980
1	0,872	872	0,499	499	0,804	804
2	1	1000	0,102	102	0,788	788
3	0,834	834	0,160	160	0,567	567
4	0,753	753	0,524	524	0,542	542
5	0,669	669	0,119	119	0,274	274
6	0,604	604	0,510	510	0,249	249
7	0,655	655	0	0	0	0
8	0,470	470	0,834	834	0,804	804
9	0,462	462	0,824	824	0,546	546
10	0	0	1	1000	0,810	810
11	0,477	477	0,837	837	1	1000

Algorithm Edgebreaker visited triangles in a spiral (" in depth ") triangular protocol binding tree lines and creates labels - clers; one mark for each triangle that indicates the decoding, i.e., as can be retrieved by attaching new mesh triangles to previously restored .

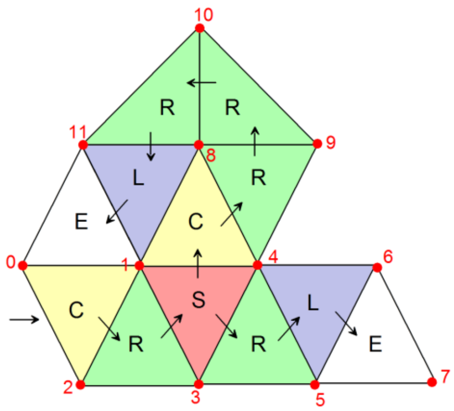

Figure 3 – The process of encoding the selected image area

Thus, we obtain a sequence for encoding – CRSRLECRRRLE. So exactly half of the triangles are of type C. For coding labels are used simple binaries. They can be obtained by using Huffman or arithmetic code, provided that the probability of triangles of type C is the same as the sum of the probabilities of all others, which in turn have the same probability (Figure 4).

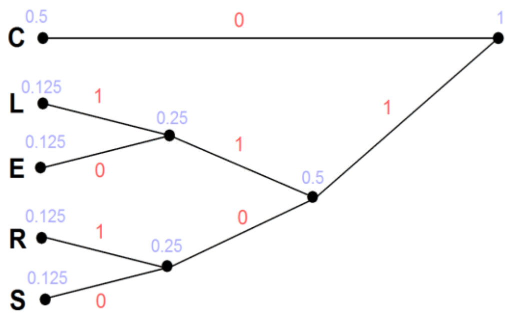

Figure 4 – Algorithm for constructing Huffman and arithmetic codes

The basic idea is to assign codes are not individual characters but for their sequence. Take the same raw data as for the Huffman code [8]: see table 2.

Table 2 – Probability for the sequence of characters

Character	Probability	Next part of the working segment:
C	0,5	0,5
L	0,125	0,625
E	0,125	0,75
R	0,125	0,875
S	0,125	1

For further coding we need the following formula :

$$High = Low_{old} + (High_{old} - Low_{old}) \cdot Range_{High(x)}; \quad (1)$$

$$Low = Low_{old} + (High_{old} - Low_{old}) \cdot Range_{Low(x)}; \quad (2)$$

where Low_{old} and $High_{old}$ respectively lower and upper bounds of the interval,

$Range_{Low(x)}$ and $Range_{High(x)}$ lower and upper limits of the encoded character.

Encode sequence CRSRLECRRRLE, and the result is formalized in the following table 3:

Table 3 – Encode sequence

characters	The result of the encoding
C	[0; 0,5)
R	[0,375; 0,4375)
S	[0,4296875; 0,4375)
R	[0,435546875; 0,4365234375)
L	[0,4360351563; 0,4361572266)
E	[0,4361114502; 0,436126709)
C	[0,4361114502; 0,4361190796)
R	[0,4361171722; 0,4361181259)
R	[0,4361178875; 0,4361180067)
R	[0,4361179769; 0,4361179918)
L	[0,4361179844; 0,4361179862)
E	[0,4361179855; 0,4361179858)

Consequently, the result of the coding will be any number in the interval [0.4361179855; 0.4361179858). Next, perform decoding. Assume that the result of coding has been chosen the left boundary of the interval , i.e. number 0.4361179855. Since the code is in the interval [0; 0.5), the first character of the message "C". For further calculations, we need the formula:

$$code = {(code - Range_{Low(x)})}/{(Range_{High(x)} - Range_{Low(x)})}, \qquad (3)$$

where '$code$ '- the current value of the code.

We perform decoding and place the results in the table below:

Table 4 – The result of decoding

Meaning «*code*»	The corresponding character
0.872235971	R
0.9778877683	S
0.8231021464	R
0.5848171711	L
0.6785373688	E
0.4282989502	C
0.8565979004	R
0.8527832031	R
0.822265625	R
0.578125	L
0.625	E

This article gives an overview of the state of the art in coding the form of 3DTV still objects by meshes. On one hand, compression of static meshes is already a well-established research area, but on other hand, only theoretic results are known. However, compressing the mesh geometry is still a difficult problem with no clear-cut solution. When the restriction of losslessness is lifted, even more techniques become possible. We have shown examples in encoding and decoding the static meshes by arithmetic code with no loss of information. We also compare our results with the performance of the Huffman coder. We found that the entropy coding of static meshes Huffman codes and arithmetic gives comparable results. For efficient compression of static and dynamic meshes the future is supposed to investigate the spectral algorithms.

REFERENCES:

1. Smolic A., Mueller K., Stefanoski N., Ostermann J., Gotchev A., Gozde B. Akar, Koz. A. and Triantafyllidis G. "Coding Algorithms of 3DTV – A Survey." *IEEE Trans. On Curcuits and Systems for Video Technology* 17.11 (2007): 1606-1621.
2. http://signal.ee.bilkent.edu.tr/Theses/kkoseThesis.pdf.
3. Peng J., Kim C.-S., Kuo C.-C. Jay "Technologies for 3D triangular mesh compression: a survey." *Journal of Visual Communication and Image Representation* 16.6 (2005): 688-733.
4. http://www.inf.tu-dresden.de/content/institutes/smt/cg/publications/paper/diss_gumhold.pdf.
5. Taubin G., Rossignac J. "Geometry compression through topological surgery." *ACM Transactions on Graphics* 17.2 (1998): 84-115.
6. Taubin G., Gueziec A., Horn W. and Lazarus F. "Progressive forest split compression." *Proc. of SIGGRAPH'98* (1998): 123-132.
7. Rossignac J. "Edgebreaker: connectivity compression for triangle meshes." *IEEE Trans. on Visualization and Computer Graphics* 5.1 (1999): 47-61.
8. Samus N., Osharovska O.V. "Image mesh coding by Edgebreaker algorithm." *Digital Technologies* 15 (2014): 119-124.

ЛІТЕРАТУРА:

1. Smolic A., Mueller K., Stefanoski N., Ostermann J., Gotchev A., Gozde B. Akar, Koz. A. and Triantafyllidis G. "Coding Algorithms of 3DTV – A Survey." *IEEE Trans. On Curcuits and Systems for Video Technology* 17.11 (2007): 1606-1621.
2. http://signal.ee.bilkent.edu.tr/Theses/kkoseThesis.pdf.
3. Peng J., Kim C.-S., Kuo C.-C. Jay "Technologies for 3D triangular mesh compression: a survey." *Journal of Visual Communication and Image Representation* 16.6 (2005): 688-733.
4. http://www.inf.tu-dresden.de/content/institutes/smt/cg/publications/paper/diss_gumhold.pdf.
5. Taubin G., Rossignac J. "Geometry compression through topological surgery." *ACM Transactions on Graphics* 17.2 (1998): 84-115.
6. Taubin G., Gueziec A., Horn W. and Lazarus F. "Progressive forest split compression." *Proc. of SIGGRAPH'98* (1998): 123-132.
7. Rossignac J. "Edgebreaker: connectivity compression for triangle meshes." *IEEE Trans. on Visualization and Computer Graphics* 5.1 (1999): 47-61.
8. Самусь Н. Сіткове кодування зображення за алгоритмом Edgebreaker / Самусь Н., Ошаровська О.В. // Цифрові технології: збірник. – Одеса: Одес. нац. академія зв'язку ім. О.С.Попова. – 2014. – №15. – С. 119-124.