PROGRESSIVE TRANSMISSION OF 3D OBJECT BASED ON BALLS AND CONES UNION FROM MEDIAL AXIS TRANSFORMATION

Florent Dupont, Benjamin Gilles and Atilla Baskurt

LIRIS – Université Claude Bernard Lyon1
Bâtiment NAUTIBUS, 8 boulevard Niels Bohr
69622 Villeurbanne Cedex, France
fdupont@ligim.univ-lyon1.fr

ABSTRACT

3D discretized objects become more and more present in many applications for computer graphics. The need to store and transmit these data grows permanently. This paper introduces a new scheme of 3D object progressive transmission. It is based on an union of cone sections and balls selected from the medial axis and optimized for a better scalability in terms of bit rates and quality. This representation of a 3D object is a lossless approach and is very efficient for compression and progressive transmission, as shown in the results presented in this paper.

1. INTRODUCTION

Usually, polygonal meshes are used for visualizing 3D data. Several methods have been proposed on 3D surface mesh compression, mesh simplification and detail approximation in order to reduce the transmission time of 3D objects [7][8].

This paper deals with compression and transmission of 3D discrete data. For voxelized objects, skeletons and medial axis (MA) are often used in order to characterize shapes, and some works concern the generation of graphs to describe objects [9][11]. Borgefors and Nyström [4], Nilsson and Danielsson [10] present methods to have an efficient shape representation with medial axis. Giblin and Kimia [6] propose an hypergraph skeletal representation based on a local analysis of the MA. For the reconstruction part, Amenta [1] computes the interior of the union of balls with continuous methods.

In this study, we propose an efficient and scalable representation of a 3D object from the medial axis. Section 2 presents the MA computation based on the discrete distance map and the local maxima determination. We show in Section 3 how to optimize the number of points of the MA in order to reduce the transmission time on a network. We also present the coding scheme in order to have rapidly a good quality for the reconstructed object. In Section 4 we propose an other method based on "cone sections". We define these "cone sections" and how to use them in order to reduce the number of transmitted elements. Section 5 presents the results for the evaluation of both the compression performance (the quantified results of the compression rates obtained compared to other methods) and the quality of the progressivity.

2. MEDIAL AXIS

2.1. Discrete Distance Map

To compute the MA, one way is to obtain first a discrete distance map where each voxel of the 3D object has the value of the minimal distance from the edge [3]. This method is reversible but does not guarantee the connectivity preservation (this point is not important for a compression purpose). Discrete distances are considered [12] and they are based on 3D voxels neighborhoods. We have implemented four different configurations that lead to four different distances: D6, D18, D26 and chamfer distance with 3-4-5 parameters.

2.2. Local Maxima

The MA is the set of centers of the maximal balls included in the object and is obtained from the distance map by computing the local maxima in the neighborhood considered with the chosen distance.

3. “BALLS” METHOD

3.1. Medial Axis optimization

The idea is to transmit medial axis balls, decomposing the object, for compression and progressive transmission purposes. The medial axis is redundant because associated balls overlap. The aim of our optimization method is to suppress the biggest number of balls while ensuring an exact reconstruction. Our method consists in computing for each medial axis point the number of voxels that belong to the associated ball and not to others. We call this characteristic the "intrinsic volume". The list of medial axis points is sorted by increasing radius. Then the intrinsic volume is computed for the first point of the list. If this value is null the point called « Zero volume » is suppressed from the list. This procedure is repeated while there are still unprocessed points. Characteristics of all points have to be updated at the end of the algorithm because of their mutual influence. The sorting stage allows a huge reduction of the number of points from the medial axis (small balls are preferentially suppressed from the medial axis).

The optimized medial axis is a list of points (X,Y,Z coordinates) with associated radius. The list is sorted by decreasing radius in order to achieve rapidly a good quality of the reconstructed object.

3.2. Coding Scheme

Coordinates and radii are coded into a binary file with the minimum number of bits dedicated to each coordinate and radius. Then, the decoding is performed in real time. Every transmitted ball is also reconstructed in real time. Hence, the number of LOD (Levels Of Detail) is very high. A more efficient entropic coder could be implemented to reduce the size of the transmitted file.

The two optimization phases are very important for the progressive transmission aim. They are successively applied on data. The medial axis computation and the optimization stages can be computed for all the norms (Section 2) and the better one can be retained. This leads to the smallest binary file even if computation time is higher (compression can be processed offline).

4. CONE SECTIONS

Representing a volume with medial axis points is equivalent to decompose it with a union of balls. The thicker and smoother the volume is, the more efficient in terms of compression of information, the method is. For stringy objects, the method is less efficient. To remedy this, the idea consists to use other primitives than balls. Here, we propose a reversible method that uses cones sections to decrease the size of data to be transmitted whilst ensuring exact reconstruction. For thin and smooth objects, the number of medial axis points can be very large. However, the number of cone sections to decompose them exactly is generally small. This point has been well depicted by Reinders, and al [11]. Decomposing an object into a minimal set of cone sections is almost impossible by looking to all possible solutions. However, by combining medial axis balls, which are included in cone sections, it is possible to create cone sections describing well the object. For example, a cylinder is represented with only one cone section which links two extreme balls. We define cone sections as the set of the balls between two balls where radius are linearly interpolated (Figure 1).

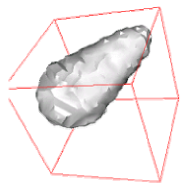

Figure 1. An example of cone section with chamfer metric

The proposed algorithm, by taking as the input the medial axis balls, intents to select the best links between these balls in order to describe the object in the most compact way. The computation of the optimal solution would be too heavy as the number of spatial and temporal combinations to look at is very large. To make the problem computationally feasible, we assumed that transmitting the biggest volume, for each level of detail, would give a good solution in order to minimize the slope of the distortion/time curve.

First, we compute an exhaustive list of cone sections which are totally included in the object: every pair of medial axis balls is tested to see if the cone section linking them belongs to the object. Then, volumes of primitives (balls and cones sections) are computed, in number of voxels, and stored into a list. This is the initialization step (See Figure 2).

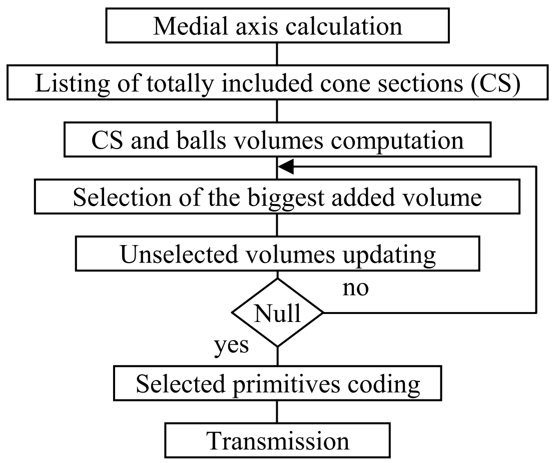

Figure 2. Coding scheme of the “cone sections” method

Then, the loop aims to select, for each iteration, the primitive that brings the biggest volume to the object, until the object is totally reconstructed. The selection is made by taking the biggest volume from the volumes list.

As soon as a primitive is selected, its volume has to be subtracted from the others. So the list is updated and then corresponds to the list of added volume. Furthermore, when it contains only null volumes, it means that any voxel cannot be added to the object; the union of selected primitives is at that time exactly the object.

The coding of the list of primitives is quite straightforward because we just need to transmit balls and links. For a ball, we code its coordinates, its radius, and any number of links to other balls. They are represented by relative indexes interleaved by 1-bit separators. We have chosen an arithmetic coding for indexes in order to profit from the non-uniform distribution of relative indexes.

5. RESULTS

Results given in this section were obtained with different volumes (See Table 1). We expect two kinds of results from this study: the compression efficiency with the comparison to standard encoders and the quality of the progressive method.

			
Lobster 77 x 83 x 16	Vertebra 66 x 66 x 66	Hand_1 37 x 63 x 66	Hand_2 19 x 32 x 34

Table 1. Test images

5.1. Data compression

After coding balls and/or cone sections, the size of the binary file obtained can be compared to the one given by classical encoders like GZIP and JBIG. We have implemented the progressive 2D compression scheme JBIG in 3D by considering a volume as a succession of 2D slices. A drawback is the necessity to have at least one dimension as a power of 2.

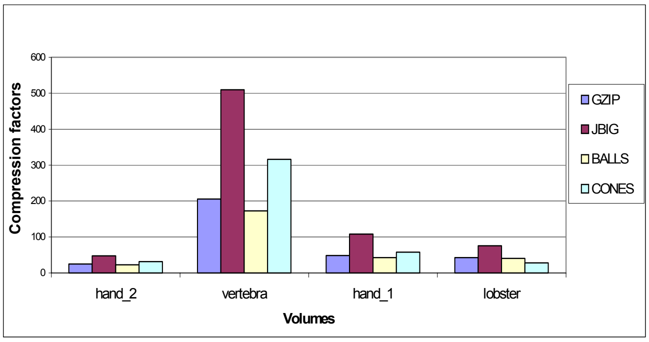

Figure 3. Compression factors

Figure 3 shows the compression factors (the size of raw file over the size of compressed one) for our methods (the best norm is used for each object), also for GZIP and JBIG.

For the compression of natural objects, our methods are a bit weaker than the two others, especially for thin objects. However, they present the progressive transmission functionality of 3D objects, which makes them interesting, i.e. for web 3D applications.

5.2. Progressive transmission

Figure 4 shows the distortion's evolution in function of the percentage of transmitted data, which is proportional to the time at a fixed network's rate. The distortion is the error on the reconstructed volume from received data in comparison to the original one. Distortion is calculated using the metric adopted by MPEG–4:

D = (nb of pixels in error) / (nb of pixels into object)

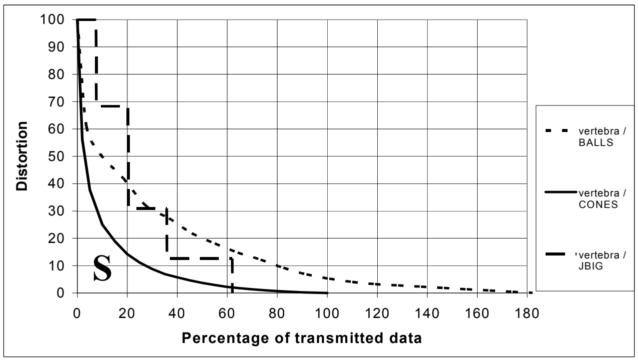

Figure 4. Distortion/time curves

In Figure 4, the corresponding curves for the JBIG method, for our "balls" and "cones" methods are shown. With our "cones" method, even if the transmission is a bit longer than JBIG in some cases, we have a very good approximation of the volume earlier (at about 20% of transmitted data) as seen in Figure 5. This figure illustrates these results with reconstructed images for the different methods at regular interval time. For Figures 4 and 5, the time or percentage of transmitted data is normalized with the "cones" method. The point called 100% corresponds to the total transmission of the file obtained with the "cones" method but not for others (the transmission for the "balls" method is complete at 180%).

To compare these two methods (and the JBIG one as well), a good criterion is to consider the area under the distortion/time curve (S in Figure 4). As we want to reduce the transmission time and the distortion, we want to minimize this area. The quality of the method is calculated with this criterion and results are presented in Figure 6.

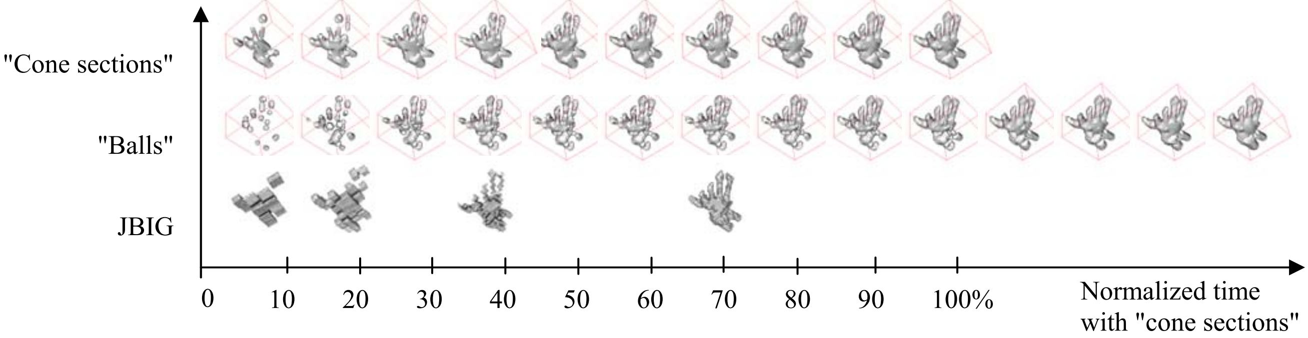

Figure 5. Reconstructed images

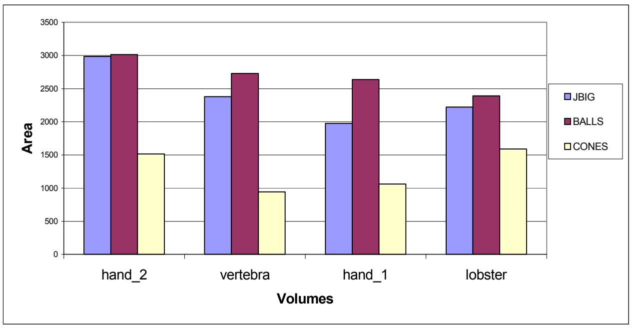

Figure 6. Area comparison

In all cases, our "cones" method gives better results than JBIG and our "balls" method.

6. CONCLUSION AND FUTURE WORK

This paper presents a new method for transmitting progressively 3D binary volumes. This method is able to give quickly a good approximation of a 3D object to the end-user. For complex volumes, the compression rates are better than the ones obtained with classic encoders. The use of data which contains 3D topological information will permit to compute 3D transformations and geometric properties easily. It opens perspectives for volume animation using transformations applied on MA and for indexing and retrieval volumes in large databases. For an on-line demonstration, the web site [5] can be consulted.

7. ACKNOWLEDGMENTS

Our thanks to Bertrand Marmond for the demonstration web site and to Florence Denis for the 3D hand image.

8. REFERENCES

[1] N. Amenta, R.K. Kolluri, "The medial axis of a union of balls", Computational Geometry, Vol. 20 No. 1-2, 2001, 25-37.

[2] H. Blum, "A transformation for extracting new descriptors of shape", In Symposium on Models for the Perception of Speech and Visual Form, M.I.T. Press, 1964, 139-146.

[3] G.Borgefors, "Distance Transformation in Arbitrary Dimension"s, Computer Vision, Graphics and Image Processing, Vol. 27, 1984, 321-345.

[4] G. Borgefors, I. Nyström, "Efficient shape representation by minimizing the set of centres of maximal discs / spheres", Pattern Recognition Letters, Vol. 18, 1997, 465-472.

[5] Demonstration web site: http://ligim.univ-lyon1.fr/ /textes/equipes/IMAGE/projets/fdupont2.htm.

[6] P. Giblin, B.B. Kimia, "A Formal Classification of 3D Medial Axis Points and their Local Geometry", CVPR00, 566-573.

[7] A. Guéziec, G. Taubin, F. Lazarus and W. Horn, "Simplicial Maps for Progressive Transmission of Polygonal Surfaces", VRML'98, ACM, February 98, 25-31.

[8] H. Hoppe, "Progressive Meshes", SIGGRAPH 1996, 99-108.

[9] R. Kresch and D. Malah, "Skeleton-Based Morphological Coding of Binary Images", IEEE Trans. Image Processing, Vol. 7, No.10, Oct. 1998, 1387-1399.

[10] F. Nilsson and P-E. Danielsson, "Finding the Minimal Set of Maximum Disks for Binary Objects", Graphical Models and Image Processing, 59(1), Jan. 1997, 55-60.

[11] F. Reinders, M.E.D. Jacobson, F.H. Post, "Skeleton Graph Generation for Feature Shape Description", Proc. Data Visualization 2000, 73-82.

[12] Remy, E., and Thiel E., "Medial Axis for chamfer distances: computing look-up tables and neighborhoods in 2D or 3D", Pattern Recognition Letters, Volume 23, Issue 6, April 2002, 649-661.