

Method Study of Point Model Fast High Quality Rendering Based on KD-Tree

Lv Hong-wei

Engineering University of PAP, ShannXi, Xi'an 710086

Keywords: point model; K-D tree

Abstract: In order to achieve fast and high quality rendering for point model, anisotropic quantitative method is proposed. After having used this method, the coordinate figures of sampling points in point model were quantified again; and the quantitative results have been represented as encoding of space partition methods and results in the process of constructing KD-tree. The paper proposed a data compression method in multiresolution structure and realized multiresolution rendering of point model by traversing the KD-tree breadth-first. It established the details control of multiresolution rendering, improved the drawing efficiency of point model. The simulation study was carried out in the paper. Results show that the method could not only reduce the geometry data volume of point models directly, but also reduce the sampling point number without debasing the numerical precision of the model geometry data, so as to further reduce data volume of point model. Some experimental results demonstrate that the algorithm is fast, and can achieve high quality rendering of point model.

1 Introduction

Take point as a kind of basic geometric model presentational element, the biggest advantage is that we can represent any complex model without recording topology data. But, after representing as a point collection for one model, the resolution is fixed. So in order to ensure the display quality, interpolations are needed, and for the complex models, geometric information and attribute information of the millions or even hundreds of millions points is still needed to record (Danny, 2001; M.Zwicker, 2002; Grossman, 2008; Szymon, 2000). Therefore, designing a reasonable data structure to represent point model, realizing the model data compression and multi-resolution rendering, so as to promote the storage and transmission efficiency of point model, this is based on the point presentation technology to solve the problem in the first place. Control level of detail (LOD) strategy has been widely applied in the current 3D data processing. When applying LOD technology to point model processing, you should first establish a multiresolution data structure corresponding to the whole model through adjusting continuously the partial resolution of input model (Leila, 1997; Carsten, 2003; Liviu, 2002). Then extract related record information from this data structure at the request of different resolutions in order to control distribution of different levels of detail on model surface (Carsten, 2003; Liviu, 2002; Carsten, 2002; Jonathan, 2001; Levoy, 1985). However, the calculation process of existing multi-resolution model generating algorithm is relatively complicated, and usually requires that each input sampling point has an accurate normal vector and that the size of each point is known, thus is now unavailable for the overall control of large scale sampling point set data.

2 K-D tree theory

The sampling point set $P = \{p_0, p_1, p_2, \dots, p_{n-1}\} \subset \mathbb{R}^3$, containing n sampling points on a given surface S , with quantitative accuracy h , satisfies Nyquist condition and is adequate for reflecting the geometry and topology information of surface. In order to construct the midpoint segmentation KD tree of the sampling point set, the first thing is to determine the local coordinate system and the initial bounding box size of the sampling point set, and then determine the splice plane.

Q , center of the bounding box, is

$$Q = aI_1 + bI_2 + cI_3 \quad (1)$$

In the formula, a, b, c are averages of the distribution range of sampling point set along the direction of the characteristic vector I_1, I_2, I_3 respectively.

$$\begin{cases} a = \frac{1}{2} (\max_{0 \leq i \leq n-1} (I_1 \cdot p_i) + \min_{0 \leq i \leq n-1} (I_1 \cdot p_i)) \\ b = \frac{1}{2} (\max_{0 \leq i \leq n-1} (I_2 \cdot p_i) + \min_{0 \leq i \leq n-1} (I_2 \cdot p_i)) \\ c = \frac{1}{2} (\max_{0 \leq i \leq n-1} (I_3 \cdot p_i) + \min_{0 \leq i \leq n-1} (I_3 \cdot p_i)) \end{cases} \quad (2)$$

The local coordinate system of sampling point set P is a coordinate system with Q as its coordinate system origin. and the direction of characteristic vector I_1, I_2, I_3 corresponding to that of x, y, z axis. Determine the splice plane is then segment the current model space along any of the three coordinate axis x, y, or z. In this article, the direction of segmentation is determined by the relation between characteristic values.

3 Point model fast high quality modeling

Anisotropic quantitative is out of sync segmentation that along the axis direction to the space when compress geometry information. As shown in figure 1, where there contains 4 points in the gray area, coordinate values can be orderly quantified for 4 points: (0.265, -0.746), (0.467, -0.712), (0.552, -0.716), (0.914, -0.736) in the quadtree, the quantitative accuracy of every coordinate component is 0.125, respectively need to use 3 bits to express. However, in KD-tree, the four sampling points can be quantified for one point: (0.467, -0.712); the quantitative accuracy of the x coordinate component is 0.6, and it needs 1 bit, the quantitative accuracy of the y component is 0.135, and it needs 3 bits. When drawing, the effect of the quadtree can achieve for using one proper size point to draw, and the size of the drawing size was just implied in the quantitative accuracy, it can be calculated by the formula (3):

$$\begin{cases} L_x = \frac{L_x^0}{2^{N_x}} \\ L_y = \frac{L_y^0}{2^{N_y}} \end{cases} \quad (3)$$

In the formula, “ L_x ” and “ L_y ” are respectively the length along the x and y direction; and “ L_x^0 ” and “ L_y^0 ” are respectively the length along the x and y direction in the initial segmentation space; “ N_x ” and “ N_y ” are respectively the numerical precision which after the sample points quantitative along the x and y coordinates, that is to say: the bit number.

The multi-resolution rendering of point model can be realized by the KD tree breadth-first traversal. Set μ for the corresponding threshold of a resolution, and a represents the 6 bounding surfaces that each node corresponding to in the KD-tree, that is

$$A = \pi(l_1^2 + l_2^2 + l_3^2) \quad (4)$$

Calculate ϵ , if $\epsilon \leq \mu$, then draw the surface, and stop processing the child node. Otherwise, if it is not a leaf node, continue to process the child nodes; if it is a leaf node, draw the node also.

4 Results

In order to validate the proposed algorithm performance, we can employ the five models sampling points set data to do simulation experiment of constructing midpoint segmentation KD-tree, and the experiment is carried out in VC++6.0 environment by using C++ and OpenGL in a microcomputer which equipped with Intel P4 2.8 GHz, NVIDIA GeForce FX 5200 graphics card and 512 MB internal memory. The experimental models are shown in figure 1. Table 1 is the coding efficiency

of KD-tree. Figure 2 is the differences results between the original normal vectors of sampling points for different models and the normal vectors that according to the method to code.

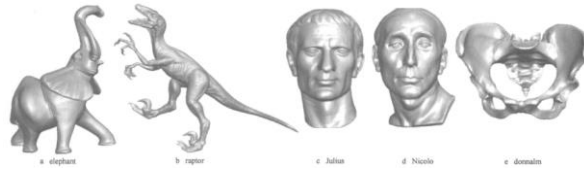


Figure 1: the experimental model

Table 1.: the coding efficiency of KD-tree

Model	N1	N2	N3	L1(bit)	L2(bit)	N3/N2	L2/N3
elephant	52901	52901	53202	1656123	1517824	0.9590	30.01
raptor	41882	41882	40100	1346791	1350879	0.9800	32.55
julius	149120	149120	81912	2411011	1840912	0.5557	22.83
nicolo	76772	76772	71658	2251023	2205000	0.9456	30.77
donnalm	55810	55810	51887	1964121	1952352	0.9559	36.99

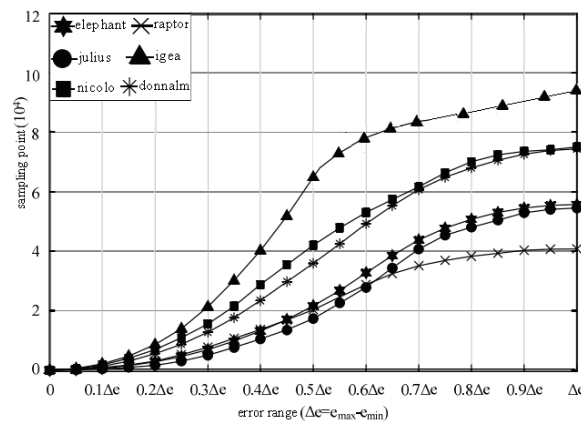


Figure 2: the error analysis of normal vector coding

5 Summary

In this paper, according to the space equal features of midpoint segmentation KD-tree, anisotropic quantitative method was proposed. In order to achieve fast and high quality rendering for point model, anisotropic quantitative method is proposed. After having used this method, the coordinate figures of sampling points in point model were quantified again; and the quantitative results have been represented as encoding of space partition methods and results in the process of constructing KD-tree. The paper proposed a data compression method in multiresolution structure and realized multiresolution rendering of point model by traversing the KD-tree breadth-first. It established the details control of multiresolution rendering, improved the drawing efficiency of point model. The simulation study was carried out in the paper. Results show that the method could not only reduce the geometry data volume of point models directly, but also reduce the sampling point number without debasing the numerical precision of the model geometry data, so as to further reduce data volume of point model. Some experimental results demonstrate that the algorithm is fast, and can achieve high quality rendering of point model.

References

- [1] Danny To, Rynson WH Lau and Mark Green, "An Adaptive Multi-Resolution Method for Progressive Model Transmission," Teleoperators and Virtual Enviroments, Vol. 10, No. 1, 2001, pp. 62-74.

- [2] M.Zwicker, M. Pauly, O. Knoll, M. Gross. Pointshop 3d: an interactive system for point-based surface editing. Proceedings of the 29th annual conference on Computer graphics and interactive techniques. ACM Press. 2002.322–329.
- [3] J.P. Grossman, William J. Dally. Point sample rendering. In Proceedings Eurographics Rendering Workshop 98. Eurographics. 1998. 181–192.
- [4] Szymon Rusinkiewicz, Marc Levoy. Qsplat: A multiresolution point rendering system for large meshes. In Proceedings SIGGRAPH 2000. ACM SIGGRAPH. 2000.343–352.
- [5] Leila De Floriani, Paola Magillo and Enrico Puppo, Building and Traversing aSurface at Variable Resolution, Proc. IEEE Visualization 1997, pp. 103-110.
- [6] Carsten Dachsbacher, Christian Vogelgsang, Marc Stamminger. Sequential point trees. In Proceedings ACM SIGGRAPH 03. ACM Press. 2003. 657–662.
- [7] Liviu Coconu, Hans-Christian Hege. Hardware-oriented point-based rendering of complex scenes. In Proceedings Eurographics Workshop on Rendering. 2002. 43–52.
- [8] Carsten Dachsbacher, Christian Vogelgsang, Marc Stamminger. Sequential point trees. In Proceedings ACM SIGGRAPH 03. ACM Press. 2003. 657–662.
- [9] Jonathan D. Cohen, Daniel G. Aliaga and Weiqiang Zhang, “Hybrid Simplification:Combining Multi-Resolution Polygon and Point Rendering,” Proc. IEEE Visualization 2001, pp. 37-44.
- [10]Levoy M., Whitted T. The use of points as a display primitive. Technical Report TR 85-022, University of North Carolina at Chapel Hill, 1985.