ERROR ANALYSIS OF 3D POLYGONAL MODEL: A SURVEY

Mr. Devendra Singh Rajput[1] and Dr. R Rama Kishore[2]

[1]Associate Manager, CSC India Pvt. Ltd. Noida, India
dsrajputindian@yahoo.co.in
[2]Assistant Professor, USIT, GGS Indraprastha University, Delhi
ram_kish@yahoo.com

ABSTRACT

Various applications of computer graphics, (like animation, scientific visualization, and virtual reality) involve the manipulation of geometric models. They are generally represented by triangular meshes due to its wide acceptance to process on rendering systems. The need of realism and high visual fidelity and the latest advances on scanning devices has increased complexity and size of triangular meshes. The original 3D model gets modified because of activities like approximation, transmission, processing and storage etc. Mostly the modification occurs due to simplification approaches which primarily use geometric distance metric as their simplification criteria. But it is hard to measure a small distance error accurately whereas other geometric or appearance error (like high curvature, thin region, color, texture, normals and volumetric) has greater importance. Hence it is essential to understand the applicability of various parameters to evaluate the quality of 3D model. This paper briefly surveys the various errors analysis techniques, error metrics and tools to assess the quality of 3D mesh models.

KEYWORDS

Error Metrics, Geometry and Attribute error, Analysis, 3D Mesh Quality, Error Analysis Tools.

1. INTRODUCTION

Large meshes are common in computer graphics for example when using devices such as CT MRI range cameras or satellite data. The large meshes put a strain on storage capacity communication and rendering hardware. In real-time or interactive applications, 3D polygonal models with millions of polygons are burdensome even on fast graphics hardware. Three-dimensional mesh models are generally composed of a large set of connected vertices and faces required to be rendered and/or streamed in real time. Using a high number of vertices/faces enables a more detailed representation of a model and possibly increases the visual quality while causing a performance loss because of the increased computations. Therefore, a trade-off often emerges between the visual quality of the graphical models and processing time, which results in a need to judge the quality of 3-D graphical content. As some methods are quick in execution however provides poor approximations; some methods generate good quality approximations but poor in execution times. Several operations in 3-D models need quality evaluation. For example, transmission of 3-D models in network-based applications requires 3-D model compression and streaming, in which a trade-off must be made between the visual quality and the transmission speed. Most of the operations cause certain modifications to the 3-D shape. To bring 3-D graphics to the masses with a high fidelity, different aspects of the quality of the user experience must be understood.

DOI : 10.5121/ijcses.2012.3205

In spite of extensive use of 3D polygonal models in geometric modelling and computer graphics, there is no agreement on the most appropriate way to estimate simple geometric attributes such as curvatures on polygonal surfaces. Many surface-oriented applications need an approximation of the first and the second order differential properties. Unfortunately, since 3D polygonal models are piecewise linear surfaces, the concept of continuous curvatures is not common. The discrete curvature can be computed by many schemes, and is useful to enhance the shape description of polygonal surfaces. In data reduction scheme, it is proposed how to determine the principal curvatures and their associated directions by least-squares parabolic fitting of the adjacent vertices, though the difficult task of selecting an appropriate tangent plane was left to the user. Although discrete curvature is useful for describing characteristics of polygonal model, it is rarely formalized into an error metric. The proposed survey should be able to establish a way to know that, which all attributes are appropriate to perform error analysis. So that an adequate quality of 3D mesh model can be obtained.

1.1. Quality of 3D Polygon models

Polygonal representations of 3D objects have become a necessity in various application domains. One of the main reasons of their popularity is that today's graphics hardware is highly specialized in displaying polygons, especially triangles, at interactive rates. Apart from visualization, the flexibility and simplicity of polygons greatly facilitate in designing, processing, transmitting, animating and interacting with 3D objects. Polygonal models can be created in various ways. Few models are designed very interactively using 3D modelling software. Some are reproduced from raw data collected by imaging devices, such as scattered points (in 3D scanning) and gray scale volumes (in bio-medical imaging). Also, other 3D representations, such as NURBS, subdivision surfaces, Constructive Solid Geometry, and implicit surfaces, often need to be transformed into polygonal forms for visualization and computation. For a practical use, a polygonal model needs to satisfy some correctness criteria expected by the target application. The polygon meshes should represent the external surface of any 3D model. Whereas, the polygonal surface should be closed, manifold and free from self-intersections. Geometric correctness [9] is especially useful in engineering and manufacturing, where numerical computations are needed for solid objects, like finite element analysis, and for real production, like quick prototyping. The manifolds are essential to need in geometry processing for calculating differential quantities on surfaces, like normals and curvatures. The production of high quality 3D polygon meshes is not a simple activity. To ensure that a given mesh is adequately smooth needs a regress investigation with directional or point light origins to detect any visible unpleasant scenes on the surface. Curvature plots, applying false color to texture the polygon mesh as per the different curvatures. This can quickly indicate any issue or potential problems as they would reveal the difference of curvatures in absolute way. Whereas, the identical appearance is the ultimate objective for approximations used in rendering systems, it is usually simple to consider geometric measures of error instead. A geometric similarity can be used as an idea for visual similarity. Achievement of geometrically trustworthy results can also produce approximations that will be useful in applications. A geometrically correct polygonal mesh should also have the similar topology as the solid it represents. Especially, topological features, like connected components, must be preserved. Topological correctness [9], at the other hand, further ensures that the polygonal model doesn't produce additional complexity to the solid it represents, like redundant handles and disconnected pieces. This may unnecessarily complicate geometry processing activity such as simplification, parameterization, and segmentation. Here the word "topology" particularly refers to the topology of a 3D solid, rather than how polygons are connected (e.g., being manifold). If the inputs mesh is non-manifold it needs to cut into manifold pieces. In case the face properties are compromised, they need to be enforced by cutting the faces, however in practice no polygonal model encountered, where the face properties had been violated. All features are preserved during approximation. In further sections error analysis techniques (error metrics) and tools are discussed.

2. ERROR ANALYSIS TECHNIQUES AND METRICS

The error metric is a measure of difference between two polygonal models. Small error between two models means to be very similar to each other. An error metric is usually defined as the geometric distance between an original and a simplified model. Some error metrics combine other attributes – color, normal, and texture coordinator – but these methods are too complex to be represented altogether. Local error metric evaluates the error using local geometric information. Most decimation methods use this metric. Global error metric provide error bound that applies globally. Simplification envelope makes two offset surfaces from the given model, and samples triangles that exist between them. Clustering and super face also use global error metric. Many of simplification methods use geometric details for their error metric. In order to evaluate the quality of any 3D polygonal model various elements of its geometry, topology and appearance are to be considered. In this paper the error metrics are categorized on following:

- Geometry based metrics
- Attribute based metrics

The geometry based metrics consists of distance, tangent slope, curvature, volume and area etc as parameters for error measurement and analysis. However, the appearance and attribute based metrics consists of color, texture, normal, roughness, distortion and strain etc. In this survey, all above parameters are discussed along with some key examples.

2.1. Geometry Based Metrics

In this section geometry based error metrics (distance, tangent and curvature etc) are discussed.

An efficient approach to measuring error represented in quadratic form [6] as the distance between the simplified vertices and the planes of the original surface. Though, the simplest estimation of how similar two meshes are is defined by the root mean square (RMS) difference. One of the most popular and earliest metrics for comparing a pair of models with different connectivity is the Hausdorff distance. This metric [1] calculates the similarity of two point sets by computing one-sided distances. The two-sided Hausdorff distance is computed by taking the maximum of D (A, B) and D (B, A); **H (A, B) = max (D (A, B), D (B, A))**

The Hausdorff distance has been used to find the geometric error between a pair of 3-D mesh models in the Metro tool by Cignoni et al. [4]. In this approach, the mean distance between a pair of meshes is found by dividing the surface integral of the distance between the two meshes by the area of one of the surfaces. The computation of this integral on discrete 3-D models requires a sampling method for fast computation. It is also observed a sampling implementation of the Hausdorff distance in the MESH tool. The Hausdorff distance computes the final distance between two surfaces as the maximum of all point wise distances. Rather than taking the maximum, extensions have been proposed to provide a better indication of the error across the entire surface.

Garland and Heckbert [6] developed an algorithm for surface simplification which quickly produces good quality approximations of polygonal models. The algorithm is used to simplify polygon models and preserves surface error approximations using quadric matrices. QEM (Quadratic Error Metric) uses Euclidean distance between a point and a face as an error metric. Error metrics of geometric approximations, mesh simplification with error control, and data reduction scheme also include such type of error metrics.

Garland and Heckbert [15] used methods from approximation theory and differential geometry to show that in the limit if triangle area becomes zero on a differentiable surface, then quadric error is directly related to curvature of surface. Within this limit, the triangulation minimizes quadric error metric and achieves an optimal triangle aspect ratio where it minimizes the geometric error. QEM combines geometric coordinator and attribute coordinator, and estimates error in the high dimension space. New QEM also includes attribute, however evaluates in R3 space.

Tangential error metric is defined [10] by the magnitude of a difference vector between two normal vectors of tangent planes. Given a vector at a point on a curve, that vector can be decomposed uniquely as a sum of two vectors, one tangent to the curve, called the tangential component of the vector, and another one perpendicular to the curve, called the normal component of the vector. Similarly a vector at a point on a surface can be broken down the same way. More generally, given a sub-manifold N of a manifold M, and a vector in the tangent space to M at a point of N, it can be decomposed into the component tangent to N and the component normal to N.

Jeong and Kim [11] mentioned as from a theoretical point of view polygonal surfaces do not have any curvature at all, since all polygonal faces are flat and the curvature is not properly defined along edges and at vertices because the surface is not differentiable there. But thinking a polygonal surface as a piecewise linear approximation of an unknown smooth surface, one can try to estimate the curvatures of that unknown surface using only the information that is given by the polygonal surface itself. A discrete curvature error metric measures a certain difference of discrete curvatures between an original and a simplified polygonal model. The discrete curvature error metric is the variance value of discrete curvatures of neighbour vertices of a collapsed edge.

The optimized curvature had been used as simplification criteria. Although curvature is useful for describing characteristics of polygonal model, it is not usually used in simplification because of its difficulty of computation and estimation. Discrete curvature doesn't require any smooth approximation, but is computed by geometric reasoning. It characterizes surface on the basis of geometric attributes assigned to edges and triangles, therefore it can describe the surface shape. In static polyhedron simplification, the Gaussian curvature is used and error zone is defined with a sphere as error bound at each vertex. LOD [11] generation with discrete curvature error metric is useful to preserve the shape of the original surface.

In paper [16] for defining discrete curvatures of polygon meshes, there are three cost functions obtained using a simple model. The results derived by the various cost functions are also compared. Functioning on data sampled from some simple 3D models. They compared the approximation error of the final optimal triangle meshes to the sampled model in different norms. All three cost functions leads to similar results, and none of these are superior to the others.

In memory-less simplification an error metric is based on geometric properties of the mesh such as volumes and areas. Image-driven simplification [12] algorithm defines an image metric, which is a metric based on pixel-wise differences between two images, and simplifies a mesh using image metrics between images from several views.

Several applications require accurate level-of-detail (LOD) [11] simplification of 3-D meshes for fast processing and rendering optimization. Watermarking of 3-D models requires evaluation of quality due to artifacts produced. Indexing and retrieval of 3-D models require metrics for judging the quality of 3-D models that are indexed.

2.2. Attribute Based Metrics

In this section appearance and attribute based error metrics are discussed. Many 3-D mesh models contain per-vertex attributes in addition to the vertex position, such as color, normal, and texture coordinates. Also, in sharp creases of the models, there may be multiple normals per-vertex, or there may be several color values on the boundaries, causing discontinuities in the attributes. Discontinuities of a model, such as creases, open boundaries, and borders between differently colored regions, are often among its most visually significant features. Therefore, their preservation is critical for producing quality approximations.

Several other metrics, using different perceptual principles, exists to better estimate the perceived quality of 3-D meshes. These solutions can be categorized as roughness-based structure-based and strain-energy-based metrics. Since each of these categories focuses on different aspects of perception, it is unlikely for one of them to estimate the perceived visual quality for all scenarios. In this case, blending metrics of several categories may be a possible solution.

Many models have surface properties beyond simple geometry. In computer graphics, the most common are surface normals, colors, and textures. To produce approximations which faithfully represent the original, it is needed to maintain the properties like Surface normals and Euclidean Attributes as well as the surface geometry. Primarily the color and Texture properties are considered as described with examples.

In a colored surface example [7] Garland and Heckbert could measure error as $Q(\mathbf{v}\text{pos}) + R(\mathbf{v}\text{rgb})$ using separate quadrics Q and R for position ($\mathbf{v}\text{pos} = [x\ y\ z]\text{T}$) and color ($\mathbf{v}\text{rgb} = [r\ g\ b]\text{T}$), respectively. Perceptual color spaces are not Euclidean in RGB. Discontinuities of a model, such as creases, open boundaries, and borders between differently colored regions, are often among its most visually significant features. Therefore, their preservation is critical for producing quality approximations.

Cohen developed an algorithm capable of re-parameterizing texture maps as a surface is simplified [7]. Garland and Heckbert has introduced an extended algorithm to their previous work, whereas it can just as easily be used to simplify surfaces with texture maps. In the example they looked down at a square height field of the eastern half of North America. The surface is textured with a satellite photograph with height.

Some solutions evaluate the quality of processed 3-D models based on their differences from the original model in their surface roughness (or smoothness). These solutions use the observation that operations on 3-D mesh either introduce a kind of noise related to roughness or cause smoothing of the surface details. Roughness is an important perceptual property, as we cannot determine the effect of a small distortion if it is on a rough region of the model, and we can detect defects on smooth surfaces more easily. This perceptual attribute, called the masking effect, states that one visual pattern can hide the visibility of another.

Structural distortion-based metrics [2] consider the assumption that the human visual system is good at extracting the structural information of a scene in addition to local properties. A mesh structural distortion measure (MSDM) had been developed based on 2-D images.

A solution based on the strain energy [2] on the mesh as a result of elastic deformation. Mesh models are assumed to be elastic objects; as shells composed of triangular faces of negligible thickness. The assumption is that triangle faces do not bend, and each triangle is deformed along its plane by ignoring any rigid body motion.

3. ERROR ANALYSIS TOOLS

Many applications produce or manage extremely complex surface meshes (e.g. volume rendering, solid modelling, and 3D range scanning). Excessive surface complexity results non interactive rendering, secondary-to-main memory bottlenecks while managing interactive visual simulations or network saturation in 3D distributed multimedia systems. In spite of the constant improvement in processing speed, the performances needed by interactive graphics applications are in many cases much higher than those provided by current technology. Significant results had been reported in the recent years, aimed at reducing surface complexity while assuring approximation in good appearance.

A general comparison of the simplification approaches is not easy, because the criteria to drive the simplification process are highly differentiated and there is no common way of measuring error. In fact, many simplification approaches do not return measures of the approximation error introduced while simplifying the mesh. For example, given the complexity reduction factor set by the user, some methods try to optimize" the shape of the simplified mesh, but they give no measure on the error introduced. Other approaches let the user define the maximal error that can be introduced in a single simplification step, but return no global error estimate or bound. Some other recent methods adopt a global error estimate or simply ensure the introduced error to be under a given bound 4. But the field of surface simplification still lacks a formal and universally acknowledged definition of error, which should involve shape approximation and also preservation of feature elements and mesh attributes (e.g. color). For these reasons, any error analysis tool that would measure the actual geometric/difference between the original and the simplified meshes would be strategic both for researchers to design new simplification algorithms, and for users, to allow them to compare the results of different approaches on the same mesh and to choose the simplification method that best fits the target mesh.

3.1. Metro

METRO [4] designed to compensate for a deficiency in many simplification methods. Metro allows one to compare the difference between a pair of surfaces (e.g. a triangulated mesh and its simplified representation) by adopting a surface sampling approach. It has been designed as a highly general tool, and it does no assumption on the particular approach used to build the simplified representation. It returns both numerical results (meshes areas and volumes, maximum and mean error, etc.) and visual results, by coloring the input surface according to the approximation error. Metro has been defined as a tool which is general and simple to implement. It compares numerically two triangle meshes, which describe the same surface at different levels of detail (LOD) [11]. Metro requires no knowledge on the simplification approach adopted to build the reduced mesh. Metro evaluates the difference between two meshes, on the basis of the approximate distance. The tool simplifies meshes either by merging/collapsing elements or by re-sampling vertices, using different error criteria to measure the correctness of the approximated surfaces. Any level of reduction may be obtained; on the condition that a sufficiently approximation threshold is set.

3.2. Nebula

NEBULA [8] tool measures the differences between a triangle mesh and its simplified approximation based on rendered images. The basic principle behind this tool is that current public domain tool measuring errors are geometry-based, and we think that the quality of simplified mesh should be evaluated more closely to the human vision model, by using what we perceived on the screen.

3.3. Polymeco

POLYMECO [14] tool allows polygonal mesh analysis and comparison. As per the applications of polygonal meshes, sometimes they need to be processed, like for simplification. Such processing introduces error, whose evaluation is required for a specific application. This tool enhances the way users perform mesh analysis and comparison. This happens through providing an environment where many visualization options are available and may be used in a coordinated manner. The Polygonal Mesh Comparison Tool, POLYMECO, has the following innovative features:

1. It provides an integrated environment, where several models and comparison results are managed simultaneously.
2. Comparison of results using: a) statistical data representations as box plots.
   b) Colored models with the possibility of using a common color scale;
3. Color scale adjustment in order to encompass a specific range of values.
4. Probe tool to obtain the error value associated with mesh vertices or faces.

3.4. Maya API

The MAYA API can be used as an error measuring approach [13] which is capable of measuring error of the polygonal models. This can be is implemented by API programming which is eligible to use as a plug-in in MAYA. This can measure the approximation errors of the model which has been modified.

4. Conclusions and Future Work

Different error analysis approaches have different strengths and weaknesses in terms of the quality of geometric and attribute correctness. It has absolute relevance with the quality of 3D mesh approximations, running times and memory overhead. This survey of error analysis metrics and tools brings following conclusions:

1. Useful results had been observed from the examples discussed; aim to reduce the mesh complexity, assuring geometric quality and also good appearance based on attribute metrics.

2. In order to evaluate the errors, this is important to identify the attributes from geometric and attribute dependent factors. It is more relevant if applications of 3D models are considered.

3. The error analysis tools mentioned in the paper are able to measure and analyze with most of the geometric and attribute based parameters discussed in error metrics section earlier.

4. A general comparison of the error analysis approaches is not easy, because the criteria to drive the process are highly differentiated as no common way of measuring error.

The findings in this paper indicates that appropriate combinations of geometric and appearance based attributes needs to be incorporate into the error analysis approaches or metrics. The combinations may include attributes like distance, tangent vector, curvature, color and texture etc. This mostly depends upon the application domain for which error analysis is to be performed. In future work, each error metric has its scope in their respective applications. However more precisely, choosing appropriate attributes for a specific application should produce more relevant evaluation of 3D models.

REFERENCES

[1]. Nicolas Aspert, Diego Santa-Cruz, Touradj Ebrahimi, "*MESH : Measuring Errors between Surfaces using the Hausdorff Distance*" In Proc. of the IEEE International Conference in Multimedia and Expo (ICME) 2002, vol. 1, pp. 705-708, Lausanne, Switzerland, August 26-29.

[2]. Abdullah Bulbul, Tolga Capin, Guillaume Lavoué , Marius Preda, "*Assessing Visual Quality of 3D Polygonal Models*," IEEE Signal Processing Magazine, vol. 28, No. 6, pp. 80-90, 2011.

[3]. P. Cignoni, C. Montani, and R. Scopigno, "*A Comparison of Mesh Simplification Algorithms*," Computers & Graphics, vol. 22, no. 1, 1998, pp. 37-54.

[4]. P. Cignoni, C. Rocchini, R. Scopigno: Metro: Measuring error on simplified surfaces. Comput. Graph. Forum 17, 2 (1998), 167–174.

[5]. Eric Shaffer, Michael Garland, "*A Multiresolution Representation for Massive Meshes*" IEEE Transactions on Visuliztion and computer graphics, VOL. 11, NO. 2, MARCH/APRIL 2005

[6]. M. Garland, P. Heckbert, "*Surface Simplification using Quadric Error Metrics*," Proceedings of SIGGRAPH 97, Page 209-216, Aug 1997.

[7]. M. Garland, P. Heckbert, "*Simplifying Surfaces with Color and Texture using Quadric Error Metrics*," Proceedings of IEEE Visualization'98, 1998,

[8]. Fu-chung Huang, Bing-yu Chen, Yung-yu Chuang, "*Nebula: A Mesh-Error Measuring Tool Based on Rendered Images*" The Eurographics Association and Blackwell Publishing 2006. Volume 0 (1981), Number 0 pp. 1–6, 2006

[9]. T. Ju, "*Fixing Geometric Errors on Polygonal Models: A Survey*," Journal of Computer Science and Technology, 24(1):19-29, January 2009.

[10]. Sun-Jeong Kim, Soo-Kyun Kim and Chang-hum Kim, "*Discrete Differential Error Metric for Surface Simplification*", computer Graphics and Applications Proceedings, 276-283, 2002.

[11]. Kim SJ, Jeong WK, Kim CH, "*LOD generation with discrete curvature error metric*," In proceedings of 2nd Korea Israel Bi-National Conference on Geometrical Modeling and Computer Graphics in the WWW Era 1999: 97-104.

[12]. P. Lindstrom and G. Turk, "*Image-driven simplification*," ACM Trans. Graph., vol. 19, pp. 204–241, July 2000.

[13]. R. Rama kishore, Yogesh Singh, B.V.R.Reddy, " *Measurement of Error in 3D Polygonal Model Using MAYA API*" et. al. / International Journal of Engineering Science and Technology Vol. 2(4), 2010, 574-586

[14]. Silva, S.; Madeira, J.; Santos, B.S.; IEETA, Aveiro, Portugal, "*PolyMeCo - a polygonal mesh comparison tool,*" Information Visualization Proceedings. On page(s): 842 – 847, Issue Date: 6-8 July 2005

[15]. Paul S. Heckbert and Michael Garland, "*Optimal Triangulation and Quadric-Based Surface Simplification*," Journal of Computational Geometry: Theory and Applications, 1999, volume 14, pages 49-65.

[16]. N. Dyn, K. Hormann, S.-J. Kim, and D. Levin. "*Optimizing 3D Triangulations Using Discrete Curvature Analysis,"* Mathematical Methods for Curves and Surfaces: Oslo 2000, Tom Lyche and Larry L. Shumaker (eds.), pp. 135-146, ISBN 0-8265-1378-6, 2001.