Proceedings of the 7th WSEAS International Conference on Automation & Information, Cavtat, Croatia, June 13-15, 2006 (pp179-184)

Describing Solids with Complex Surface Textures in a Simple Format

DONGXING WANG, HUIWEN LENG AND SU CHEN
School of Mechatronics and Automobile Engineering
Yantai University
No. 32, Qingquan Street, Laishan District, Yantai 264005
CHINA

Abstract: - Describing a surface colored model in a proper format is very meaningful to its formation by Color Rapid Prototyping. The proposed format describes the geometry of a solid by a polyhedron consisting of triangular facets, which approaches the geometry of the solid. It also uses a simple method to describe complex surface textures. It uses affine mapping from a rectangular texture image to each facet to describe simple texture mapping on the surface, and it describes complex texture mapping on the surface by simply storing the mapping result image of the complex texture mapping. The format is both simple enough that CAD models in this format can be easily sliced, and capable of describing solids with complex surface textures. Therefore, it is good to be an intermediate format for describing surface colored models in Rapid Prototyping field.

Key-Words: - Modeling; Format; Color; Texture; Surface; Rapid Prototyping

1 Introduction

Compared with a mono color model, a colored one can give more information, more clearly. It is more convenient and more persuasive when used in communication between designers, manufacturers, distributors and users [1].

Before 2000, there had not been any rapid prototyping facility capable of directly fabricating full color models. Complex and time-consuming handwork is required to change a mono color model to a full colored one. This increases the cost and time for making a colored model. Forming colored models directly by Rapid Prototyping techniques has been the hope of the industry.

Recently, Color Rapid Prototyping attracts more and more attention. Some research projects have already been done [2-4]. Z Corp., a manufacturer of rapid prototyping machines, has even developed commercial 3D color printers [4].

Most colored models are used only for visualization purpose. They need to be colored only on their surfaces, such as models used for assembly inspection, building models, and jewelry models. They are called surface colored models.

A surface colored model may be simply colored, for instance, a polyhedron, each of whose facets is painted in only one color, and different facets may be painted in different colors. It may also has complex surface detail, for instance, there are always textures on the surface of an actual wooden solid or a block of marble. In addition, there may be texts, trademarks, and/or decorating decals on the surface of a solid.

A CAD model must be created first in order to form a surface colored model.

1.1 Methods for describing a surface colored model

Description of a surface colored model includes both its geometry and its surface detail, i.e., surface color and/or texture(s).

There are several methods for describing the geometry of a solid, such as Constructive Solids Geometry and Boundary Representation. In RP field, for simplicity of slicing, the most widely used method is using a polyhedron consisting of triangular facets to approach the geometry, which is a kind of Cell Representation.

Surface details of surface colored models are diverse. For a simple one, e.g. a polyhedron, each of whose facets is painted in only one color, its surface detail can be described by assigning each facet a color value. While for a complex one, texture-mapping methods [5] should be used to describe its surface detail.

By the methods for generating texture patterns, texture-mapping methods can be classified into two groups; procedural texture mapping methods and image based texture-mapping methods.

A procedural texture mapping method generates a texture pattern directly by a procedural way [6-8].

An image-based texture mapping method establishes a mapping relationship between points on a 3D surface and those in a 2D image. The color of each point on the surface can then be obtained by mapping the point to the image.

It is easy to establish the mapping relationship between an image and a parameterized surface, such as a plane, a cylinder, or a sphere. For example, the mapping relationship between a plane and an image can be established using projective transformation, bilinear transformation, or affine transformation [9]. Among these transformation methods, the projective transformation is the most general. However, more conditions are required to determine the transformation parameters, and more calculation is required, as well. For the bilinear transformation, its reverse is usually not single valued, so it can only be used in cases requiring only the forward transformation. The affine transformation [10], with six degrees of freedom, is the simplest. It can be determined by the coordinates of three non-collinear points in object space, and the coordinates of their images in texture space. Therefore, for a triangular facet, the coordinates of whose three vertices are known, if the coordinates of the images in texture space, the texture coordinates, of its three vertices are given, the affine mapping between the facet and the texture image can be determined uniquely.

For surfaces which can not be parameterized, such as a polyhedral surface consisting of many small planar or curved patches, the texture on such a surface can be described using two-part texture mapping [11].

In addition, there are still many other methods for describing the surface detail of a solid [12, 13].

By their complexity, the above texture mapping methods can be classified into two groups.

(1) Simple texture mapping methods: refer to those using affine transformation, which includes translation, rotation, scale, reflection, and their multiplications [10]. These methods map a line to a line, a triangle to a triangle, and a rectangle to a parallelogram.

(2) Complex texture mapping methods: refer to those which are not simple texture mapping methods, including methods using bilinear or projective transformation, two-part texture mapping methods, procedural texture mapping methods, and so on.

1.2 Existing formats capable of describing surface colored models

The CAD model of a surface colored model must be output in a proper format. In RP field, the STL format [14], for its simplicity and the advantage that files in this format can be easily sliced, has been widely used for describing mono color solids. However, it does not offer color information, thus cannot be used to describe surface colored models. At present, formats capable of describing a surface colored model include the polygonal format (PLY) [15], the virtual reality modeling language (VRML) [16], and the Color STL format [17, 18]. The PLY format is simple, but there is not a ready scheme for describing surface textures. Although it is simple to describe the geometry of a solid using polygons with more than three edges, slicing of CAD models in such a format would be more difficult. VRML can describe surface color and textures, but it contains much information that has nothing to do with RP. Its complexity undoubtedly brings inconvenience to the slicing of CAD models in this format. Because of its complexity also, user usually only uses a subset of it [19], description of complex surface textures is actually unavailable. Just like the STL format, the Color STL format is also simple. Files in this format can also be easily sliced. Though it can be used to describe simple surface color, it cannot be used to describe surface textures.

Thus, it is meaningful to establish a format, which is both simple, and capable of describing solids with complex surface textures.

2 The Proposed Format

A new format was proposed, which is simple enough that CAD models in this format can be easily sliced, and capable of describing arbitrarily complex surface textures in a simple way. It describes the geometry of a solid in a similar way as STL format does, so it is called the extended STL format, abbreviated as ETL.

Considering that the new format is required to describe solids with complex surface detail, two files are used to describe a surface colored model. One is for describing the geometry of the model, called the geometry description file (GDF). The other is for describing its surface detail, called the texture mapping description file (TMDF).

2.1 Describing the geometry

It is a simple way to describe the geometry of a solid of arbitrary complexity using a polyhedron consisting of triangular facets, which approaches the geometry of the solid. The CAD model in a format describing the geometry of a solid in this way can be easily sliced, as in the case of the STL format.

2.2 Describing surface color and/or textures

For the convenience of slicing, the ETL format uses a simple method to describe surface color and/or texture(s), as described below.

(1) In ETL format, a solid, even one with curved surface, is described as a polyhedron made up of many triangular facets. Its surface detail can then be described by the color or texture defined on each of its triangular facets.

(2) If the color on a facet is identical, then use this color to describe the facet.

(3) Else, the color on the facet is not identical. Then use texture mapping to describe the facet. If the texture on the surface region where the facet resides is previously defined by simple texture mapping methods, then still use simple texture mapping methods to describe the texture on the facet. Else, the previous description of the texture on the facet must be relatively complex, then rasterize the mapping result on the facet, save it as a triangular image, and use this image to describe the texture on the facet. This is actually also a simple texture mapping method.

2.2.1 Describing texture images

In ETL format, there are two kinds of texture images, ordinary rectangular texture images, and triangular texture images (TTIs). The latter are just images of the texture mapping results on triangular facets.

2.2.1.1 Describing rectangular texture image

Rectangular texture images are just the ordinary rectangular 2D images. Therefore, the existing image file formats, for instance, BMP, PCX, TIFF, and JPG, etc., are all applicable to their description.

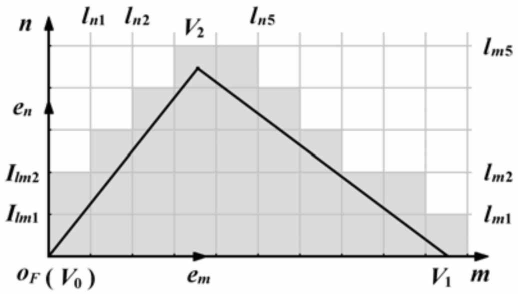


Fig.1 Rasterizing the texture mapping result on a triangular facet

2.2.1.2 Describing the texture mapping result on a facet using a TTI

Let V_0, V_1, and V_2 to denote the vertices of a facet, which are arranged in counter-clockwise direction, if seen from the outside of the described solid. Find the directional longest edge of the facet, for instance, $\boldsymbol{V}_0\boldsymbol{V}_1$. Set V_0 as the origin, the direction of $\boldsymbol{V}_0\boldsymbol{V}_1$ as the positive direction of an axis, to establish a rectangular facet coordinate system, o_Fmn, as shown in Fig. 1.

Provided that Δm by Δn is the size of a pixel used in rasterizing the texture mapping result on the facet. Paralleling to o_Fn, at a spacing of Δm, draw a series of lines $\{\boldsymbol{l}_{n1}, \boldsymbol{l}_{n2},\}$. Paralleling to o_Fm, at a spacing of Δn, draw another series of lines $\{\boldsymbol{l}_{m1}, \boldsymbol{l}_{m2},\}$. The two series of lines constitute a grid, as shown in Fig. 1.

A mesh in the grid corresponds to a pixel in the rasterizing result. The pixels (meshes) at least partially inside the triangle constitute the TTI.

Establishing the facet coordinate system this way can reduce the number of lines in the resulting TTI. It can also insure the vertex not on axis o_Fm appearing only in the first quadrant, and the angles between the longest edge and the other two edges must be acute. Thus, it can simplify the rasterizing process.

For each pixel in the TTI, map it to texture space using the previous texture mapping definition(s) on the facet, and assign it a color equaling to the average color in its image region in texture space. By this method, the color of all the pixels in the TTI can be worked out.

A TTI is described in rows. In order to reduce the required storage space, store only the effective pixels in each row, i.e. the starting column number, the color of each active pixel in the row, the length of the row or the ending column number.

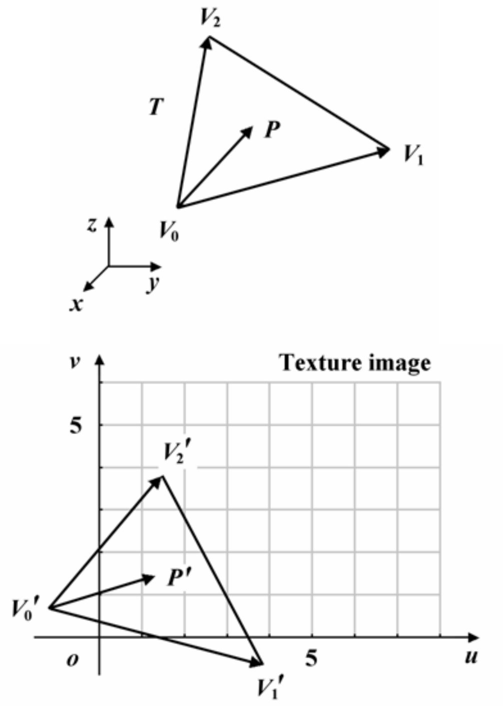


Fig. 2 Simple texture mapping from a rectangular image to a triangular facet

2.2.2 Describing textures on a facet

There are accordingly two methods for describing the texture on a facet, simple texture mapping from a rectangular texture image to a facet, and using a TTI to describe the texture on a facet. The latter is actually also a simple texture mapping method. The former is for describing simple texture mapping from a rectangular texture image to a planar surface region, the latter is for describing complex texture mapping on a surface region.

2.2.2.1 Describing simple texture mapping from a rectangular texture image to a facet

Establish a texture coordinate system as shown in Fig. 2, the rectangular texture image is to be mapped by a simple texture mapping method to triangular facet T in R^3, i.e., the coordinate transformation between the texture coordinate system and R^3 is affine. Based on the knowledge of Affine Geometry, the simple mapping relationship between the facet and the texture image can be uniquely determined by the coordinates of the three vertices of T, and the coordinates of their images in texture space, i.e., their texture coordinates.

By their content, rectangular texture images can be classified into two groups. One group is those for describing the materials of solids such as wood, stone, or brick. Such a texture image usually covers the whole surface of a solid, and shows some periodicity. Another group is those appearing only in some local surface regions, such as trademark images and decorating decals. These images are usually not tiled repeatedly. Therefore, there should be flags to indicate whether a texture image is tiled or mirrored repeatedly along each coordinate axis.

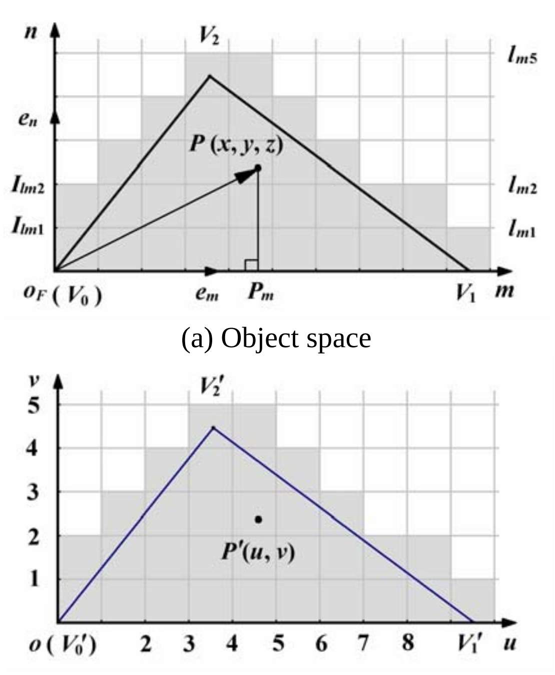
So, the complete description of a simple texture mapping from a rectangular texture image to a facet includes tiling flags in directions of u and v axes, mirroring flags in directions of u and v axes, the identifying number of the rectangular texture image, and texture coordinates of the vertices of the facet.

2.2.2.2 Describing the texture on a facet using a TTI

It is a simple and definite way to describe the texture on a triangular facet directly using a TTI created by the method described in 2.2.1.3. In addition, storing a triangular image requires less space than storing a rectangular one.

Texture mapping from such a TTI to a facet is simple and straightforward, as shown in Fig. 3.

In the creation of a TTI for a facet, the origin of the facet coordinate system is set as the start point of the directional longest edge of the facet. To avoid confusion, the sequence number of the vertex set as the origin of the facet coordinate system should be recorded. The size of the pixel used in rasterization is saved in the file header of the TMDF, as described in 2.2.3. Therefore, describing the texture on a facet using a TTI requires the sequence number of the vertex set as the origin of the facet coordinate system and the description data of the TTI.



(a) Object space

(b) Texture space

Fig.3 Texture mapping from a TTI to a triangular facet

2.2.2.3 Describing multiple texture mappings

The surface texture of an actual solid is diverse. Some reflects the material of the solid, such as that on the surface of a wooden solid or a piece of marble. Some may simply be a color, such as that on the surface of a painted solid. There may also be some decals, such as a label or a decorating pattern, on the surface of a solid.

If there are multiple complex texture mapping definitions, or a complex texture mapping definition plus some simple texture mapping definitions, on the surface region where a facet resides, then rasterize the mapping result on the facet, save the rasterizing result as a triangular image, and use the method described in 2.2.2.2 to describe the texture on the facet.

If there are only a few simple texture-mapping definitions on the surface region where a facet resides, then the description of the texture on the facet is just the descriptions of these simple texture mappings in sequence, respectively. Afterwards, when calculating the average color of a polygonal region, say A, on the facet, just use each simple mapping definition to determine whether there are vertices of *A* mapped to the effective region of the texture image. If it does, define the average color of *A* equaling to the average color of the pixels in the effective region of *A*'s image. The later calculation result overwrites the previous one.

A complete texture mapping description item (TMDI) starts with an IDByte of unsigned char type.

If its MSB is zero, then the TMDI uses a TTI to describe the texture on the facet, and its value is just the sequence number of the vertex set as the origin of the facet coordinate system when creating the TTI. The following is the description data of the TTI.

Else, its MSB is one, indicating that the TMDI consists of descriptions of a few simple texture mapping definitions. Its value minus 128 is the number of simple texture-mapping definitions. Then follows descriptions of these simple texture mappings.

For briefness, the case where there is only simple color definition on a facet is also regarded as simple texture mapping. Here, only the color of the facet is required.

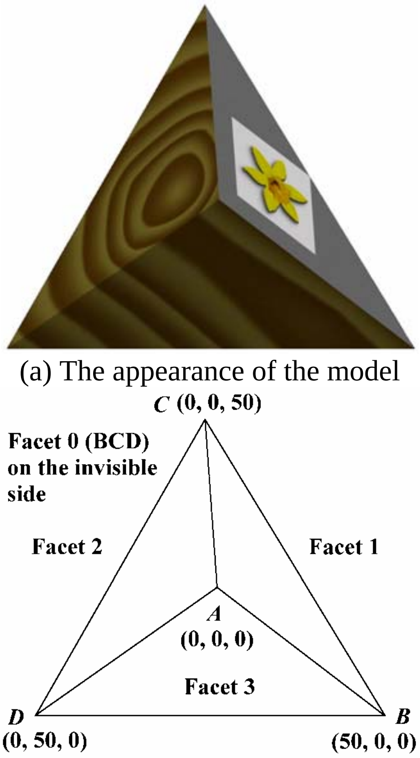
2.2.3 The format of a complete TMDF

A complete TMDF starts with a file header, which includes some global information, including file format and version (unsigned char Format_ID[8]), a flag byte indicating whether the described solid is a mono color one, and whether there is only simple color definition for each facet (unsigned char ColorFlag), the number of bits of color (unsigned char ColorBits), the global color (unsigned char GlobalColor[3]), the length of the color palette (unsigned short ColorPaletteLength), the size of the pixel used in rasterizing the mapping results on facets (float Deltam, Deltan), the number of triangular facets (unsigned long NumberofFacets), etc. If ColorBits is zero or one, then there is also a 24 bits color palette at the end of the file header.

If the described solid is a mono color one, then its TMDF has only a file header. Its 24-bit color is stored in GlobalColor[3].

If there is only simple color information, then its TMDF consists of a file header as described above, and a list of the color or color indices to the color palette, of each facet.

If neither the described solid is a mono color one, nor there is only simple color information, then TMDIs, as described in 2.2.2.3, are required to describe the texture on each facet. Because different TMDIs may be of different length, for fast accessing the TMDI of a facet by the sequence number of the facet, a list of the offsets of the TMDI of each facet is required, which immediately follows the file header. Next follows all the TMDIs.



(a) The appearance of the model

(b) Its geometry

Fig.4 A colored model with texture on its surface

3 An example

A tetrahedron with color, textures, and a decal on its surface, as shown in Fig. 4 (a), is taken as an example to illustrate the ETL format. Its geometry is shown in Fig. 4 (b). The color of Facet 0, which is on the invisible side, is RGB(150, 150, 150). The base color of its front face, Facet 1, is RGB(128, 128, 128). There is also a rectangular decal on Facet 1, whose edges are all 20 mm. On its left side, Facet 2, and its bottom side, Facet 3, there are wood textures.

The description of the tetrahedron in ETL format includes a GDF and a TMDF, which describe the geometry and the surface detail of the model.

The GDF consists of a file header, a list of the coordinates of all vertices, and a list of the description data of each triangular facet, which is actually a list of the indices of the three vertices of the facet. The geometry data is as shown in Fig.4 (b).

The TMDF consists of a file header, a list of the offsets of the TMDI of each facet, and all the TMDIs.

The TMDI of facet 0 describes just a simple texture mapping, which includes only a simple color definition.

The TMDI of facet 1 includes descriptions of two simple texture-mapping definitions. The first simple texture mapping description describes only a simple color definition. The second simple texture mapping description describes a simple texture mapping from a rectangular texture image, without tiling or mirroring.

The TMDIs of facet 2 and 3 describe the surface detail of the two facets directly using the triangular mapping result images on the two facets.

4 Conclusions

The proposed format describes the geometry of a solid in a simple way. It also uses a simple method to describe arbitrarily complex surface textures. It uses affine mapping from a rectangular texture image to each facet to describe simple texture mapping on the surface, and it describes complex texture mapping on the surface by simply storing the mapping result image of the complex texture mapping. The format is simple enough that CAD models in this format can be easily sliced. Therefore, it is good to be an intermediate format for describing surface colored models in Rapid Prototyping field.

The method for slicing a CAD model in ETL format is very similar to that for slicing a digital model in Color STL format [20]. To get the data for driving a color rapid prototyping facility, two further procedures, thickening the colored regions in a color volume dataset [21] describing the slicing result, and halftoning the thickened color volume dataset [22], are also required.

References:

[1] Y.G. Wang, *Rapid Prototyping Technology*, Wuhan: Huazhong University of Science and Technology Press, 1999, pp. 43-50 and 285-286. (In Chinese)

[2] W.M. Ling and I. Gibson, Possibility of colouring SLS prototype using the ink-jet method, *Rapid Prototyping Journal*, Vol. 5, No. 4, 1999, pp. 152-153.

[3] J. Jang, W.C. Huang and B.Z. Jang, 3-D color model making apparatus and process, US Patent No. 6,165,406, Dec 26, 2000.

[4] http://www.zcorp.com

[5] D. Hearn and M.P. Baker, *Computer Graphics*, Beijing: Tsinghua University Press, 1998, pp. 553-558. (In Chinese)

[6] Y.X. Tang, *Computer Graphics*, Beijing: Science Press, 1990, pp. 384-385. (In Chinese)

[7] D.R. Peachy, Solid texturing of complex surface, *Computer Graphics*, Vol. 3, 1985, pp. 279-286.

[8] J.S. Fu and D.C. Zhang, Representation and rendering of wood texture, *Journal of Computer-Aided Design and Computer Graphics*, Vol. 7, No. 2, 1995, pp. 109-14. (In Chinese)

[9] P.S. Heckbert, *Fundamentals of Texture Mapping and Image Warping (Master's Thesis)*, University of California, Berkeley, 1989, pp. 11-21.

[10] D.Z. Fang and Y.P. Chen, *Projective Geometry*, Beijing: Higher Education Press, 1983, pp. 52-96. (In Chinese)

[11] E. Bier and K. Sloan, Two-part texture mapping, *IEEE Computer Graphics & Application*, Vol. 6, No. 9, 1986, pp. 40-53.

[12] J.L. Jiang, Research on the regional texture mapping on the polygonal primitivity, *Journal of Anqing Teachers College (Natural Science)*, Vol. 9, No. 2, 2003, pp. 61-4. (In Chinese)

[13] H.Q. Sun and H.J. Bao, Interactive texture mapping for polygonal models, *Computational Geometry*, Vol. 15, 2000, pp. 41-49.

[14] M. Szilvasi-Nagy and G. Matyasi, Analysis of STL Files, *Mathematical and Computer Modeling*, Vol. 38, 2003, pp. 945-960.

[15] http://www.cc.gatech.edu/projects/largemodels/ply.html

[16] W.M. Ling and I. Gibson, Specification of VRML in Color Rapid Prototyping, *International Journal of CAD/CAM*, Vol. 1, No. 1, 2002, pp. 1-9.

[17] D.X. Wang, D.M. Guo and J.Y. Jia, RPM Oriented Representation of the Colored Models, *Machine Design and Research*, No. 2, 2002, pp. 62-63. (In Chinese)

[18] http://rapid.lpt.fi/rp-ml-2001/2598.html.

[19] W.D. Rase, Physical Models of GIS Object by Rapid Prototyping, ISPRS Commission IV, Symposium, Ottawa, Canada, July 9-12, 2002.

[20] D.X. Wang, D.M. Guo, J.Y. Jia and H.W. Leng, Slicing of CAD Models in Color STL Format, *Computers in Industry*, Vol. 57, No. 1, 2006, pp. 3-10.

[21] D.X. Wang, D.M. Guo, J.Y. Jia and L.H. Jiang, Thickening the Colored Regions in a Color Volume Dataset, *Computers & Graphics*, Vol. 29, No. 5, 2005, pp. 588-593.

[22] D.X. Wang, D.M. Guo, J.Y. Jia and L.H. Jiang, Halftoning colour volume datasets based on subdivision, *Computer-Aided Design*, Vol. 36, No. 13, 2004, pp. 1371-1377.