A Freehand Sketch Interpreter System for Constructing 3D Solid Models

Weizhong Liu†, Kunio Kondo†, Jun Mitani‡,

† Department of Information and Computer Sciences, Saitama University,
{wzliu,kondo}@ke.ics.saitama-u.ac.jp, ‡ RIKEN, jmitani@riken.jp

1. Introduction

Freehand sketch interpreter system is a new type of CAD system which increases flexibility of data input for 3D models and will become the future trend of CAD system[1-3].

This paper discusses a freehand sketch interpreter system for constructing complex 3D solid models. The essential original idea of this paper is to create complex 3D object with a template topology library step by step from freehand strokes. The library is called T-LIB. The input strokes are analyzed as a 2D edge graph. Once the edge graph can be matched to a template of T-LIB, the 3D shape can be reconstructed (referring to Fig.1(a,b)) or modified (referring to Fig.1(c,d)). Designers can use or feel comfortable to use our system for realizing their idea sketches.

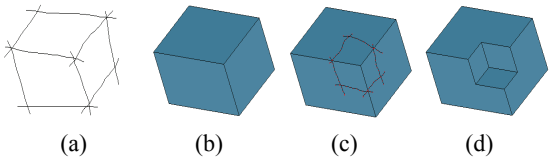


Figure 1: Sketch interpreter system. (a)Input freehand strokes. (b)Create a basic 3D shape. (c)Input strokes on the 3D shape. (d)Generate a new modified 3D shape.

2. Interpretation of Sketch Input

2.1 Sketch Plane

The strokes are drawn on a plane which is called sketch plane. Referring to Fig.2, it can be regarded as a plane between 3D scene and screen of user interface.

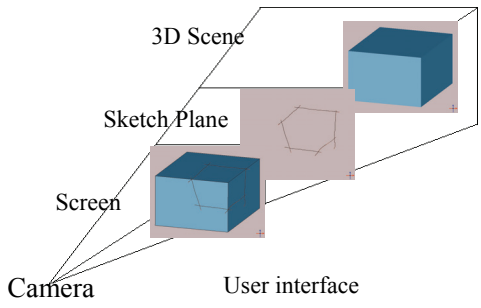


Figure 2: Sketch Plane.

2.2 Edge Graph

Edge Graph is a 2D topology graph. It is calculated from the strokes and the projected edges of 3D mesh which intersect with the strokes.

Here, we use a proper threshold to judge whether the strokes meet at one vertex or not. We implement a process called T-Extracting to get the proper intersecting edges from 3D mesh. And then project them to sketch plane. From the input strokes and projected edges, we can build a 2D edge graph.

2.3 T-LIB

T-LIB is a template topology library. It consists of 2D Edge Graph and the corresponding algorithm to create and modify 3D objects. Fig.3 illustrates the templates of cube, triangular prism and triangular pyramid. New template definition is feasible.

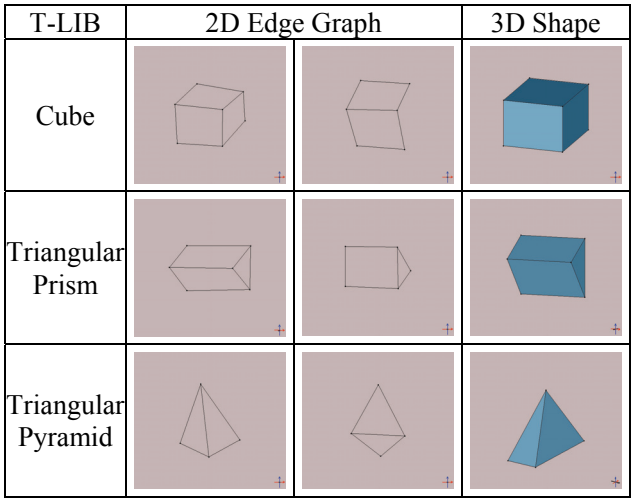


Figure 3: T-LIB.

3. Application Interface System

3.1 System Procedure

The system is shown as the following Fig.4.

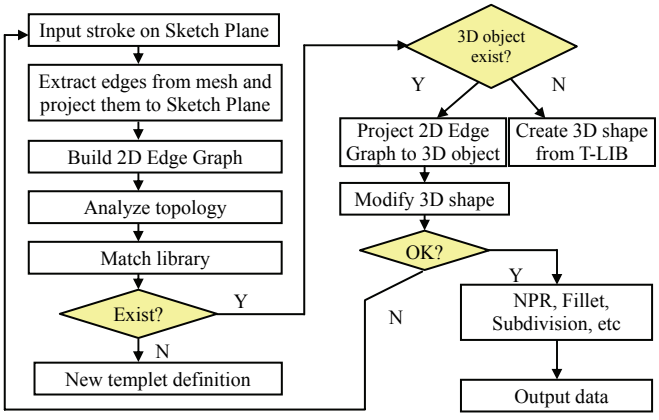
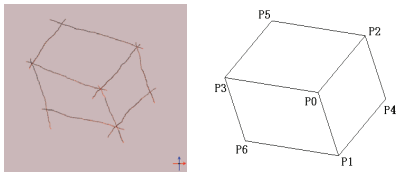


Figure 4: Application interface system

3.2 3D Shape Reconstruction from T-LIB

The algorithm of cube shape reconstruction is discussed in this section. The strokes are shown in Fig.5(a), while the extracted 2D edge graph is shown in Fig.5(b). The 3D cube is calculated by evaluating a camera normal and using parallel projection.



(a) (b)

Figure 5: 3D cube shape reconstruction from T-LIB

Let vector $n=\{n_x, n_y, n_z\}$ be the camera normal, $V_i=\{v_i^x, v_i^y, v_i^z\}$ be the vertex of the 3D cube, while $P_i=\{p_i^x, p_i^y, p_i^z\}$ be the vertex of cube edge graph.

Let $v_0=(0,0,0)$, $a=n_x/n_z$, $b=n_y/n_z$, with the principle of parallel projection, we get the following equations:

$$v_1^x v_2^x + v_1^y v_2^y + v_1^z v_2^z = 0 \quad (1)$$

$$v_2^x v_3^x + v_2^y v_3^y + v_2^z v_3^z = 0 \quad (2)$$

$$v_3^x v_1^x + v_3^y v_1^y + v_3^z v_1^z = 0 \quad (3)$$

$$v_1^x = a v_1^z + p_1^x, \quad v_1^y = b v_1^z + p_1^y \quad (4)$$

$$v_2^x = a v_2^z + p_2^x, \quad v_2^y = b v_2^z + p_2^y \quad (5)$$

$$v_3^x = a v_3^z + p_3^x, \quad v_3^y = b v_3^z + p_3^y \quad (6)$$

Let $c=(a p_1^x+b p_1^y)/(a^2+b^2+1)$, $d=(a p_2^x+b p_2^y)/(a^2+b^2+1)$, $e=(a p_3^x+b p_3^y)/(a^2+b^2+1)$, $f=(p_1^x p_2^x + p_1^y p_2^y)/(a^2+b^2+1)$, $g=(p_2^x p_3^x + p_2^y p_3^y)/(a^2+b^2+1)$, $h=(p_3^x p_1^x + p_3^y p_1^y)/(a^2+b^2+1)$, from equation (1-6), we get the following equations:

$$v_1^z v_2^z + d\, v_1^z + c\, v_2^z + f = 0 \quad (7)$$

$$v_2^z v_3^z + e\, v_2^z + d\, v_3^z + g = 0 \quad (8)$$

$$v_3^z v_1^z + c\, v_3^z + e\, v_1^z + h = 0 \quad (9)$$

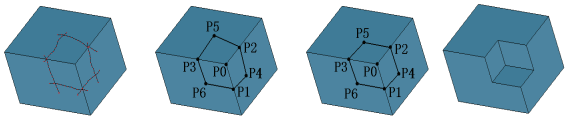
From equation (7-9), we can deduce the equation (10):

$$(g-de)(v_1^z)^2 + 2(cg-cde)\, v_1^z + (fh+gc^2-cdh-cef) = 0$$

Then v_1, v_2, v_3 can be calculated, other vertices can be calculated with the parallel projection. In such a way, the basic 3D shape can be reconstructed.

3.3 3D Shape Modification

The 3D shape modification process is illustrated as the following figures in Fig.6:



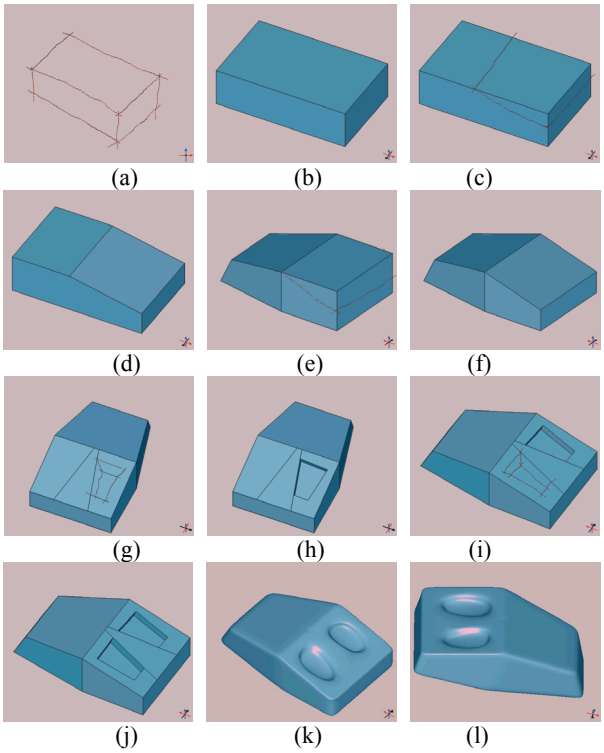
(a) (b) (c) (d)

Figure 6: Modification process. (a)Input stroke. (b)Extract edge graph, (c)Adjust edge graph, (e)Modify 3D shape.

Here the angle between the stroke and edge of 3D shape is used as a threshold to adjust the edge graph for getting the parallel information.

4. Examples

Fig.7 is an example we made by using our sketch system for constructing a mouse model.



(a) (b) (c)
(d) (e) (f)
(g) (h) (i)
(j) (k) (l)

Figure 7: An example for constructing a mouse model. (a)Input strokes. (b)Create basic 3D shape. (c-j)Input strokes again and modify the 3D shape. (k)Implement fillet operation and adaptive subdivision. (l)Another view of Fig.(k).

5. Conclusions

In this paper, we propose a freehand sketch interpreter system for constructing complex 3D solid models. The idea comes from a template topology library called T-LIB, which is used in both 3D shape reconstruction and modification procedures. The basic 3D shape is created by evaluating a camera normal and using parallel projection. The complex 3D shape can be gotten by modify the basic 3D shape with T-LIB step by step from freehand strokes. With our system, designers can feel very comfortable for realizing their idea sketches.

References

[1] S. Sugishita, K. Kondo, H. Sato, S. Shimada, *Sketch Interpreter for geometric modeling*, International Conf. of Computer Aided Geometric Design 1994, 1994.7.

[2] K. Matsuda, S. Sugishita, Z. Xu, K. Kondo, H. Sata, S. Shimada, *Freehand Sketch System for 3D Geometric Modeling*, Shape Modeling International, pp. 55-62, 1997.3

[3] P.A.C. Varley, H. Suzuki, J. Mitani, R.R. Martin, *Interpretation of Single Sketch Input for Mesh and Solid Models*, International Journal of Shape Modeling, Vol. 6, No. 2(2002) 207-241.