

RP를 위한 슬라이스 데이터로부터 STL 포맷의 3차원 형상 재구성 Reconstruction of 3D shape in STL file using slice data for RP

허성민*(부산대 대학원), 최홍태(경남전문대), 이석희(부산대 기계공학부)

Sung-Min Hur* (Graduate School, Pusan National University)

Hong-Tae Choi (Kyung-Nam Junior College), Seok-Hee Lee (Pusan National University)

ABSTRACT

This study presents a way to construct 3D shape in STL format from 2D slice data. Nowadays shape reconstruction has been done in many areas, the application of this method is important especially in Reverse Engineering which reconstructs original shape from cross-section data. Current RP(Rapid Prototyping) is used not only for the verification of a part designed but also for the production and tooling in more effective way. In RP technology, data should be prepared in STL format. In this paper, the way to make 3D shape data in STL format from 2D slice data is described which can be used to reconstruct an original shape in RP equipment

Key Words : Rapid prototype(급속조형), STL format, CT

1. 서론

오늘날 CT(Computed Tomography)는 의료 분야 뿐만 아니라 공업 분야에서도 광범위하게 이용되고 있다. 이미 종래의 의료학적 의미에서의 X-ray를 뛰어넘어 산업계에서는 CT scanner와 같은 기구를 이용하여 물체의 외부 형상뿐만 아니라 내부 형상에 관한 이미지를 제공하고, 물체 내부의 결함(flaw)을 관찰하는데도 사용하고 있다. 또, 종래의 기술로는 나타내기 힘든 복잡한 기하학적 3차원 형상을 측정 시에도 사용된다. 공업적으로는 이런 CT 데이터를 solid model로 변환해주는 시스템(CT-Modeller System)이 이미 나와 있고, 이런 CT를 이용한 공학적인 분야가 기하학적 데이터가 없는 물체의 원래의 형상을 그대로 복원하는 Reverse Engineering이며, 이것을 실현하는 가장 핵심적인 기구가 RP 장치이다. RP 장치 중 가장 대표적이라고 할 수 있는 StereoLithography Apparatus(SLA) 시스템의 경우, 조형 대상의 3차원 CAD 모델로부터 STL 변환기를

통해 삼각형망으로 구성되어 있는 STL 포맷으로의 변환이 요구된다. 본 연구에서는 2차원 단면 데이터를 STL 포맷 형식의 3차원 형상으로 복원시켜서 원래 형상의 3차원 구성 및 RP로의 응용을 목표로 한다.

2. 관련연구

국외에서는 Robert N.Yancey⁽¹⁾가 CT를 통해서 얻어진 물체의 데이터에서 원래의 형상을 재구성하는 reverse engineering에 대해서 연구하였다. James W. Kristoff는 MRI데이터로부터 CAD 데이터로 변환시키는 연구를 하였으며⁽²⁾, 실제로 사람의 MRI 데이터를 이용한 향상된 의료장비의 개발을 위해서 CAD 모델을 써서 제품 생산에 이용하는 방법에 대해서 설명하였다.⁽³⁾

국내에서는 박형준⁽⁴⁾ 등이 단면 정보를 이용한 형상을 삼각형 분할을 통해서 재구성하는 알고리즘을 제시하였지만, STL 포맷에 대한 삼각형 정점의 관

계, 법선벡터에 대한 언급이 없어 RP에서 직접 이용할 수 없다. 김남국⁽⁶⁾ 등은 웹을 이용한 의료진단용 볼륨 데이터 시각화 시스템을 구성하였는데, MRI (Magnetic Resonance Image)로부터 얻어진 물체의 데이터를 이용하여 기존의 Marching cube 알고리즘의 단점을 보완하여 모델의 크기를 줄이는 Decimation 알고리즘을 이용하여 이를 가상공간에서 시각화시켰다.

3. 본 론

슬라이스 데이터로부터 STL 포맷의 데이터로 변환하기 위해서 각각의 슬라이스 데이터로부터 입력받은 정보에 따라 서로 삼각형 분할(triangulation)을 할 부분을 선택한다. 선택된 부분을 삼각형 분할한 후, 이를 STL 포맷으로 다시 형식화한다. 이와 같은 과정을 Fig. 1에서 단계별로 나타내고 있다.

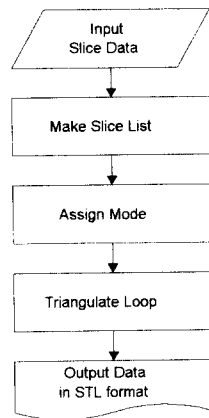


Fig. 1 Flow chart of main process

3.1 입력 슬라이스 파일 데이터

오늘날 CT scanner나 MRI의 발달에 의해서 물체의 슬라이스 데이터를 쉽게 얻을 수 있다. 이러한 장치는 이미 의학적인 분야의 이용을 뛰어넘어 공학에 응용되고 있으며, 슬라이스 데이터로부터 경계선 추출기법을 이용해서 각각의 슬라이스가 보유하고 있는 정보를 구한다. 본 연구에서 사용된 입력 파일의 형식은 Fig. 2와 같으며 포함된 정보는 다음과 같다.

- 각 슬라이스의 번호를 나타내는 Slice NO.
- 각 루프의 시작을 나타내는 SOL
- 각 루프를 구성하는 정점의 x, y, z좌표
- 각 루프의 끝을 나타내는 EOL

각 루프를 이루는 정점은 반시계방향의 순서대로 배열되어 있다.

SLICE No.1

SOL

43.000 45.000 0.001

13.001 45.000 0.001

.....

43.000 45.000 0.001

43.000 45.000 0.001

EOL

Fig. 2 Input Slice File

3.2 STL 포맷의 특징

STL 파일은 삼각형의 세 정점과 삼각형을 이루는 면의 법선 벡터로 구성되어 있다. 오류가 없는 STL 포맷을 형성하기 위해서는 물체를 구성하는 삼각형들이 vertex-to-vertex rule과 오른손 법칙을 만족시켜야 한다. vertex-to-vertex rule은 모든 삼각형의 정점은 반드시 이웃한 삼각형의 정점과 만나서 연결되어야 하며, 오른손 법칙은 Fig. 3에서와 같이 삼각형 세 정점의 배열 순서에 따라 오른손을 감았을 때 엄지손가락이 가리키는 방향이 법선벡터의 방향이 되며 항상 면의 바깥쪽을 향한다.

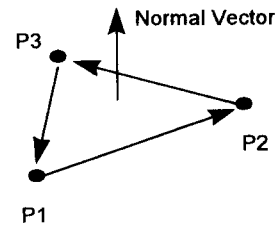


Fig. 3 Right hand rule

한 직선 위에 있지 않는 3점 $P_0(x_0, y_0, z_0)$, $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$ 를 지나는 평면 방정식은 다음 (1)식과 같다.

$$\begin{vmatrix} x-x_0 & y-y_0 & z-z_0 \\ x_1-x_0 & y_1-y_0 & z_1-z_0 \\ x_2-x_0 & y_2-y_0 & z_2-z_0 \end{vmatrix} = 0 \quad (1)$$

(1)식을 전개하면 $ax + by + cz + d = 0$ 으로 된다. 여기서 a, b, c, d 는 다음 (2)식~(5)식과 같다.

$$a = \{(y_1 - y_0)(z_2 - z_0) - (z_1 - z_0)(y_2 - y_0)\} \quad (2)$$

$$b = \{(z_1 - z_0)(x_2 - x_0) - (x_1 - x_0)(z_2 - z_0)\} \quad (3)$$

$$c = \{(x_1 - x_0)(y_2 - y_0) - (y_1 - y_0)(x_2 - x_0)\} \quad (4)$$

$$d = -(ax_0 + by_0 + cz_0) \quad (5)$$

이 관계로부터 단위 법선벡터를 구하면 그 성분들은 다음 (6)식~(8)식과 같다.

$$n_x = \frac{a}{\sqrt{(a^2+b^2+c^2)}} \quad (6)$$

$$n_y = \frac{b}{\sqrt{(a^2+b^2+c^2)}} \quad (7)$$

$$n_z = \frac{c}{\sqrt{(a^2+b^2+c^2)}} \quad (8)$$

본 연구에서는 슬라이스 데이터에 있는 정점들을 연결하므로 vertex-to-vertex rule을 만족시키며, 선택된 삼각형의 정점의 순서를 오른손 법칙을 만족하도록 출력시킨다.

3.3 슬라이스 연결 리스트 구성

입력된 슬라이스 데이터를 통해서 슬라이스 연결 리스트를 구성한다.

전체 슬라이스는 z축 방향의 각 슬라이스로 구성되며, 각 슬라이스는 여러 개의 루프로 구성되며, 각 루프는 여러 개의 정점으로 구성된다. 이와 같은 연결 관계를 이용하여 슬라이스 데이터를 읽으면서 전체 슬라이스를 구성하는 연결 리스트를 완성한다. 이상의 관계를 도식화한 연결 리스트가 Fig. 4이다.

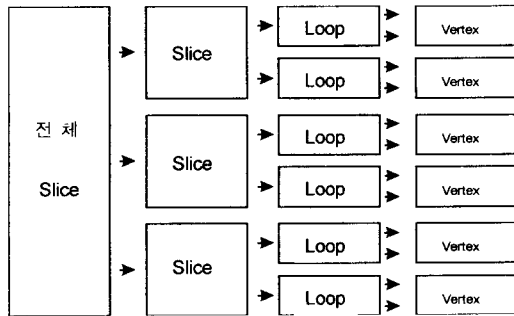


Fig. 4 Total slice list

3.4 각 슬라이스별 루프의 분류

각 슬라이스에 대한 루프와 그 다음 인접한 슬라이스에 대한 루프를 XY 평면에 투영하였을 때의 상관 관계를 Fig. 5와 같이 3가지 형태로 분류하였다.

첫째, Fig. 5 (a)와 같이 한 슬라이스에서 두 루프 사이에 포함관계를 가지는 상태, 이것을 본 연구에서는 nestmode라고 한다. 특히 각 루프의 최대값, 최소값을 고려하면 안쪽 루프와 바깥쪽 루프를 판별할 수 있다.

둘째, Fig. 5 (b)와 같이 한 슬라이스의 루프에서 다음 인접한 슬라이스의 루프간에 교점이 있는 상

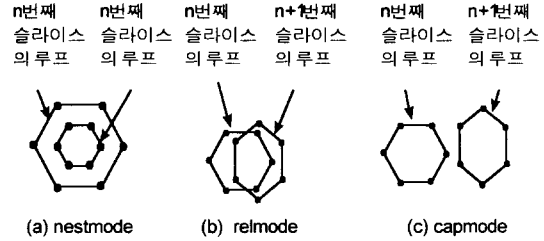


Fig. 5 Classification of loop state

태, 이것을 relmode라고 한다.

셋째, Fig. 5 (c)와 같이 한 슬라이스에서의 루프가 다음 인접한 슬라이스에서의 루프와 아무런 교점이 없이 분리된 상태, 이것을 capmode라고 한다.

전체 슬라이스 연결 리스트를 통해서 입력된 데이터를 다시 읽어서 한 슬라이스내의 루프 사이의 nestmode의 관계, 두 슬라이스내의 루프 사이의 relmode, capmode를 표시한다.

nestmode의 루프의 경우는 최대, 최소값을 통해서 안쪽과 바깥쪽을 판별할 수 있는데, 이 정보를 각각의 루프에 표시한다. relmode의 경우 n번째 슬라이스의 루프는 자기와 교점을 가지고 있는 n+1번째 슬라이스의 다른 루프에 대한 정보를 가져야 하므로 그 정보를 저장할 수 있는 관계리스트(relation list)를 가지고 relmode라고 판별될 때마다 관계리스트를 하나씩 생성시킨다. 이를 통하여 각 루프는 자기와 교점을 가진 루프에 대한 정보를 알 수 있으며 어떤 루프라도 교점을 가지지 못한 경우 capmode를 설정한다.

3.5 일반적 다각형의 삼각형 분할과 슬라이스간의 삼각형 분할의 비교

각 루프의 삼각형 분할은 크게 두 종류로 나눌 수 있다. 첫째, capmode에 있는 루프면 자체는 다각형(polygon)을 형성하여 다른 슬라이스내의 어느 루프와도 삼각형 분할에 영향을 주지 않으므로 Fig. 6과

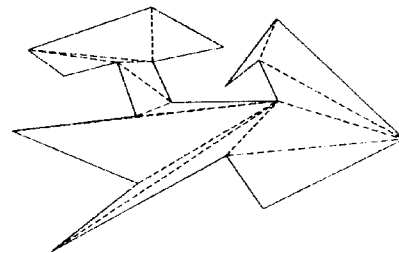


Fig. 6 Triangulation of polygon

같은 일반적인 다각형을 삼각형으로 분할을 할 수 있다.⁽⁶⁾

위와 같은 방법으로 삼각형 분할을 할 수 있는 부분이 물체의 가장 아랫면에 해당하는 슬라이스와 가장 윗면에 해당하는 슬라이스내의 루프이다. 이 면들은 자체로 하나의 폐곡선을 이루고 있다. 그리고 슬라이스 중에 capmode로 설정된 루프 역시 자체로 하나의 다각형을 형성하므로 동일한 방법으로 삼각형 분할을 행할 수 있다

둘째, 슬라이스간의 루프 사이에서 삼각형 분할을 할 경우이다.

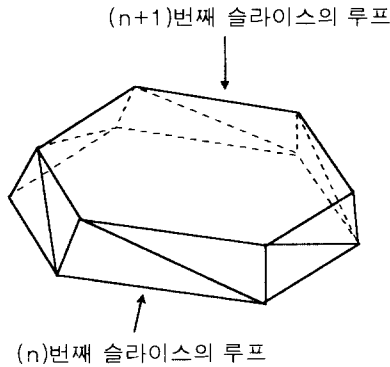


Fig. 7 Triangulation between loop in different slice

Fig. 7과 같이 분할된 삼각형의 정점은 $n \rightarrow n+1 \rightarrow n$ 또는 $n+1 \rightarrow n \rightarrow n+1$ 의 슬라이스의 루프를 거치는 관계로 되어 있다. 그리고 한 루프의 정점을 무조건 다음 루프의 정점과 연결시키면 루프의 정점 수가 다른 경우에 뒷부분에서의 정점 수가 남는 부분에서의 삼각형 분할을 고려해야하고 (Fig. 8에서 A 이후 부분), 삼각형의 한 변이 길어지는 부분이 발생하여 그 결과로 왜곡된 STL 포맷의 삼각형이 생성될 수 있다. (Fig. 8에서 B 부분).

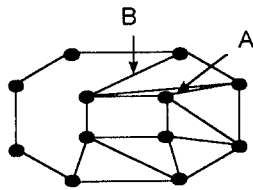


Fig. 8 Problem on triangulation

이 점들을 고려하여 각 정점이 $n \rightarrow n+1 \rightarrow n$ 또는 $n+1 \rightarrow n \rightarrow n+1$ 의 슬라이스의 루프를 거치는 관계를 순차적으로 나타내기 위해서 한 루프의 한 정점이 다른 루프의 한 정점과 선택되는 관계를 나타내는

선택 리스트(selection list)가 필요하다.

삼각형 분할의 단계로서, 첫째, 한 슬라이스의 각 정점은 다른 슬라이스의 가장 가까운 점과 연결하기 위해서 동일점을 판별해야 한다. 동일점은 한 슬라이스와 그 다음 슬라이스의 루프상의 정점의 x, y 좌표가 일치하거나 아주 가까운 거리내에 있는 경우이고 삼각형 분할 이전에 미리 판별한다. n번째 슬라이스의 루프의 정점 좌표와 n+1번째 슬라이스의 루프의 정점 좌표를 각각 $V_{n,x}, V_{n,y}$ 와 $V_{n+1,x}, V_{n+1,y}$ 라고 하면 n번째 점과 n+1 번째 점과의 거리는 다음 (9)식과 같다.

$$d = \sqrt{(V_{n,x} - V_{n+1,x})^2 + (V_{n,y} - V_{n+1,y})^2} \quad (9)$$

d가 허용한도 내에 드는 경우 같은 점으로 판별하여 선택리스트를 하나 생성시킨다.

둘째, 삼각형의 형상을 등변 삼각형에 가깝게 만들기 위해서 루프를 둘러싼 정점 수가 많은 루프의 각각의 정점에서 루프를 둘러싼 정점 수가 적은 루프의 가장 가까운 거리에 있는 각각의 정점을 서로간에 선택한다. 이상의 과정을 루프를 둘러싼 정점 수가 많은 루프의 모든 정점에 대해서 실행한 후, 각각의 선택된 정점에 대한 선택리스트를 생성시킨다.

세째, 위의 과정을 수행하였을 때 선택되지 않은 정점을 찾기 위해서, 기존의 선택 리스트에 포함된 n번째 리스트에서 한 루프의 n번째 정점과 다른 루프의 n번째 정점의 다음 정점과의 선택 관계, 또는 다른 루프의 n번째 정점과 한 루프의 n번째 정점의 다음 정점과의 선택 관계를 조사하여 그 정점들의 리스트가 선택 리스트에 존재하지 않는 경우 두 정점의 대응 관계에서 거리가 작은 쪽의 정점들의 쌍을 선택 리스트에 연결 리스트로 생성시킨다.

네째, 위의 과정을 통해서 정점의 수가 적은 루프의 선택되지 않은 정점을 찾기 위해서 선택리스트를 조사한다. 이 루프의 정점 중 선택되지 않은 정점을 조사하여 구한 후 위의 세 가지 단계와 같은 과정을 거치면서 이미 선택리스트에서 선택된 정점끼리를 제외한 나머지 정점끼리 선택이 되면 선택리스트를 생성시킨다.

다섯째, 이상의 선택 리스트의 정점의 대응을 나타내는 집합을 순차적으로 나타내기 위해서 선택리스트의 정점을 판단하여 그것의 두 정점 중 어느 하나의 다음 정점을 가진 선택리스트를 찾아야 하며 다시 이와 같은 과정을 반복하여 순차적으로 연결하며, 이렇게 연결된 상태에서 $n \rightarrow n+1 \rightarrow n$ 또는 $n+1 \rightarrow n \rightarrow n+1$ 의 슬라이스의 루프를 거치는 관계로 표시한다. 이때 삼각형의 법선 벡터를 위해서 right hand rule을 만족하도록 삼각형 세 정점의 배열 순서를

조정한다. 이상의 과정을 나타낸 선택 리스트를 그림으로 표시하면 Fig. 9와 같다.

1	2	3	4

- 1 : 동일점점의 선택
- 2 : 정점의 수가 많은 루프의 정점에서 작은 루프의 정점의 가장 가까운 정점을 선택
- 3 : 1, 2 의 과정을 통해서 선택된 정점의 다음 정점끼리의 선택
- 4 : 1, 2, 3 과정에서 선택되지 않는 나머지 정점의 1, 2, 3 과정의 반복

Fig. 9 Contents of selection list

3.6 각 루프사이의 삼각형 분할

각 루프에는 삼각형 분할 이전에 그 슬라이스 내에서의 다른 루프와의 포함 관계 또는 다음 슬라이스에서의 다른 슬라이스와의 접점을 가지는 관계를 표시하는 nestmode, relmode, capmode가 표시되었다.

nestmode를 가진 루프는 안 루프 또는 바깥 루프임을 표시하는 정보를 가지고 있으므로 다음 슬라이스의 루프와 안 루프는 안 루프끼리, 바깥 루프는 바깥 루프끼리 삼각형 분할을 행한다.

capmode를 가진 루프는 그것만을 폐곡선에 대한 삼각형 분할을 독립적으로 행한다.

relmode를 가진 루프는 그것이 가진 관계리스트를 조사하여 그 루프가 접하고 있는 모든 루프를 조사하여 삼각형 분할한다. 이때, 두 슬라이스간의 교차하는 루프끼리의 수가 1 : 1인 경우는 전술한 바와 같이 슬라이스사이의 삼각형 분할을 행한다.

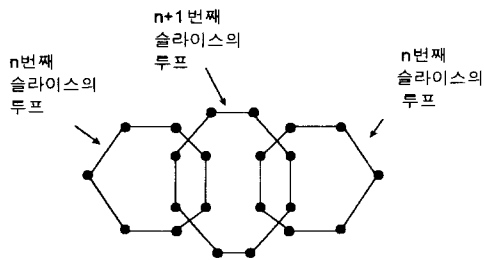


Fig. 10 1 : m triangulation

인접한 두 슬라이스를 평면에 투영하여 만나는 교점의 수가 1 : m인 Fig. 10과 같은 경우는 다음과 같이 여러 개의 조각으로 Fig. 11과 같이 분리할 수 있는데 각각의 분리된 루프는 이전 루프에서의 정보

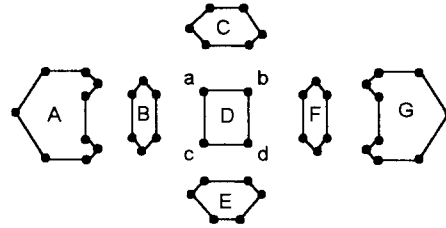


Fig. 11 Disassembly in 1:m case

를 가지고 있으므로 삼각형 분할을 할 수 있다. Fig. 11의 A, G는 슬라이스간의 삼각형 분할을 통해서 해결하며, 남은 부분인 CDE가 이루는 부분은 삼각형 분할의 편의를 위해서 B와 F의 C 영역내의 가장 아래 점인 a, b를 연결하고 B와 F의 E영역내의 가장 윗점인 c, d를 연결하여 내부의 폐곡선으로 연결된 D를 만들어 각 영역을 C, D, E로 분리한다. C, E는 슬라이스간의 삼각형 분할로써 해결할 수 있으며 D의 경우는 원래 n+1번째 슬라이스만의 영역이지만 n번째 슬라이스의 접합되는 부분이므로 삼각형 분할의 편의상 B, F와 같이 n번째 슬라이스와 공유하는 부분으로 처리한다. 이상의 분해 과정을 통해서 삼각형 분할을 하면 Fig. 12와 같은 결과를 얻을 수 있다.

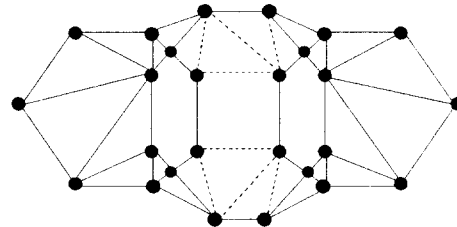


Fig. 12 Triangulation in 1:m case

두 슬라이스간의 루프끼리의 접하는 수가 m : m인 경우는 슬라이스와 슬라이스간에 많은 자유도가 예상되며, 규칙적인 형상의 연속인 경우에는 삼각형 분할이 가능할 것이다. 단, 이 경우는 슬라이스 사이의 간격을 줄임으로써 m : m의 경우의 발생을 최소화함으로써 형상 오차를 억제할 수 있다.

그리고 루프가 2개 이상의 mode를 보유하고 있을 경우는 각각의 mode에 대해서 위의 과정을 실행한다.

3.7 STL 포맷으로 출력

이상의 삼각형분할을 통해서 생성된 삼각형을 STL 포맷의 오른손 법칙을 만족하도록 정점의 순서를 배치한다. 바깥 루프의 삼각형 분할의 경우에는 오른손 법칙을 만족하도록 Fig. 13과 같이 a→b→c의 차례로 행하며, 안쪽 루프의 삼각형 분할의 경우에는

바깥면이 반대쪽을 향하는 방향으로 오른손 법칙을 만족하도록 삼각형의 정점을 배치한다.

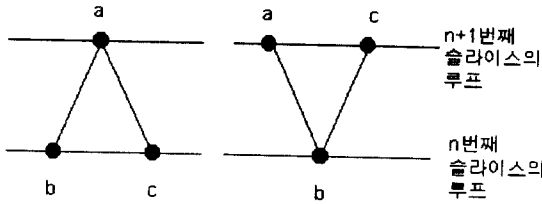


Fig. 13 Sequence of Vertex in consideration of right hand rule

4. 적용예

이상의 알고리즘을 Visual C++4.1와 TGS사의 Open Inventor를 사용하여 프로그래밍하여 Fig. 14와 같은 결과를 얻을 수 있으며, Fig. 15는 컵 형상의 2차원 슬라이스 파일을 읽어서 프로그래밍한 결과를 3차원 형상으로 구성한 예이다. 이것은 VRML로서 가상공간에 3차원 형상으로 구현한 것이다.

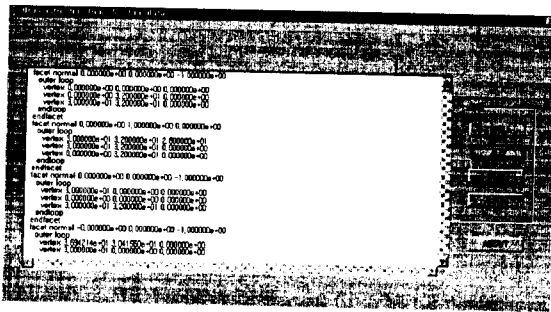


Fig. 14 Output data

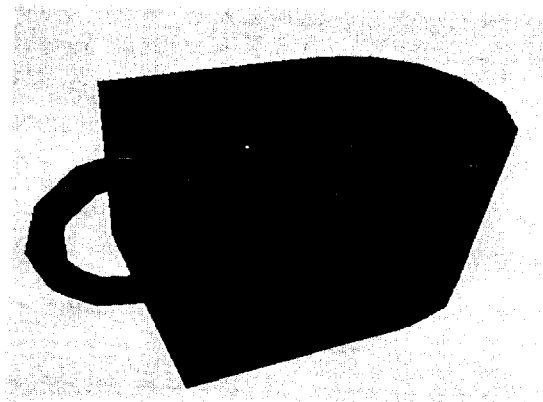


Fig. 15 Reconstruction of cup figure

5. 결론 및 고찰

본 연구에서는 2차원 슬라이스 데이터를 이용하여 3차원 형상을 STL 파일 형식으로 구성하였다. 주어진 입력 데이터를 인접한 슬라이스 사이의 연결점의 관계를 규명하여 삼각형 분할을 하고 정점과 법선 벡터를 고려하여 STL 포맷으로 출력하였다. 또, 각각의 슬라이스에서 루프가 접합되는 방식에 따라 capmode, nestmode, relmode로 나누어서 각 경우에 맞는 삼각형 분할의 방법을 제시하여 각 mode가 복합되어 있는 경우에도 해결할 수 있게 하였다. 그러나 교차하는 루프사이의 수가 많아지면 다루기 어려운 부분도 있기 때문에, 이 부분의 해결책으로 슬라이스 데이터의 z축 간격을 줄이는 방법을 제시하였지만, 이럴 경우 특정 부분만을 다시 더 작은 슬라이스 간격으로 잘라서 처리하는 프로세스는 추후 연구과제로 한다.

참고문헌

- [1] Robert N.Yancey, Dennis S.Eliassen, S. Trent Neel, James H.Stanley, "Reverse Engineering using Computed Tomography", Proceedings of the Fifth International Conference on Rapid Prototyping, Dayton, Ohio, pp. 141~149, 1994.
- [2] Kristoff J, "Converting MRI Scans to CAD Models for Device Engineering", in MD&D West Conference Proceedings, Santa Monica, CA, Canon Communication, pp. 103~109, 1996.
- [3] Kristoff J, "Mapping Human Anatomy from MRI Data for Improved Product Development", MD&DI 1997, <http://www.devicelink.com/mddi/archive/97/03/017.html>
- [4] 박형준, 김광수, "단면정보를 이용한 형상의 재구성", 한국정밀공학회지, 제10권, 제4호, pp. 81~93, 1993.
- [5] 김남국, 김철영, 김영호, 강석호, "웹을 이용한 의료진단용 볼륨 데이터 시각화 시스템", '97 한국 CAD/CAM 학회 학술 발표회 논문집, pp. 349~354, 1997.
- [6] Joseph O'Rourke, "Computational Geometry", Cambridge University press, pp. 35~51, 1993.