

## Stereolithography를 위한 STL파일로부터 단면정보 변환시스템의 개발

최 흥 태,\* 김 준 안,\* 이 석 희,\*\* 백 인 환\*\*

### Development of Cross-sectional Information Conversion System from STL file for Stereolithography

Hong-Tae Choi,\* Jun-An Kim,\* Seok-Hee Lee,\*\* In-Hwan Paik\*\*

#### ABSTRACT

This paper deals with conversion from the STL file to the Slice cross-sectional information for Stereolithography. The STL file is widely used for Stereolithography, but it is very difficult to convert STL file into Slice file directly. Because it consists of an ordered list of triangular net without any topological information other than the orientation of each facet. So, The system is accomplished by data flow through several intermediate stages such as Reference, .SL1, .SL2, .SL3, and .SLC file. The data processing is performed in 5 steps: 1) Create a Reference file including common information. 2) Modify STL file within the effective range of SL machine. 3) Calculate a point of intersection between plane equation and line equation. 4) Sort z values in ascending order using quick sort algorithm. 5) Search the adjacent points and formulate a closed loop using singly linked linear list. The system is developed by using Borland C++ 3.1 compiler in the environment of Pentium PC, and verified to be satisfactory by making some prototypes of electric household appliances.

**Key Words :** Stereolithography(광조형법), STL file(STL 파일), Slice cross-sectional information(슬라이스 단면정보), triangular net(삼각형망), vertex(정점), topological information(위상정보), SL machine(광조형 장치), singly linked linear list(단일 연결 리스트)

\* 부산대학교 생산기계공학과 대학원

\*\* 부산대학교 생산기계공학과 기계기술연구소

## 1. 서 론

공업 분야에서 제품이 생산되기 전에 다양한 모델이 시험 제작되어 외관 디자인, 부품 조립기구, 시험, 금형 제작 등에 매우 유용하게 이용된다. 특히 소비자의 기호가 다양화됨에 따라서 이들 제품들은 많은 곡면으로 이루어져 있고 점점 복잡해지고 있어 숙련자일지라도 설계도를 완전히 이해하기가 어려워지고 있다. 또한, 제품의 수명 주기가 점차 짧아지고 있기 때문에 갈수록 형상이 복잡해지고 있는 새로운 제품의 개발을 위한 시작품 제작 기간도 단축되어야 하는 어려움이 있다. 이러한 상황에서 현재 새로운 제품의 복잡한 형상 모델의 대부분은 고도의 숙련자에 의해 수작업으로 이루어지고 있으며 이러한 숙련자수는 해마다 감소하는 추세이다. 따라서 이러한 문제를 해결하고자 하는 일련의 노력으로 RP(Rapid Prototyping)분야가 활성화되어 현재 미국과 일본 등에서 상용 RP시스템들이 판매되고 있다. 국외에서는 白順義<sup>(1)</sup> 등이 시작품 등의 물리 모델로부터 단면 화상 처리 장치에 의한 단면 윤곽 데이터 생성에 관한 연구를 하였으며, 川嶋英嗣<sup>(2)</sup> 등은 CAD데이터를 STL(STereoLithography)데이터로 변환할 때 생성되어지는 STL데이터의 틈(gap) 및 중복이 생기는 문제점을 해결하기 위해 3차원 형상에 대해서 Delaunay 사면체 분할을 이용한 STL데이터 생성법을 제안하였다. Wozny<sup>(3)</sup>는 CAD모델로부터 직접적으로 물리층 고형화 공정(Physical Layer Solidification Process)이 이루어질 수 있는 SFF(Solid Freeform Fabrication)기법에 관한 내용을 다루었으며, 이 기법을 적용하기 전 데이터 변환 문제점들을 제시하고, 그 중 STL파일의 중복된 정점을 줄이기 위해 위상정보가 부족한 STL파일을 기초로 하여 정점(Vertex), 모서리(edges), 그리고 면(faces)의 색인 리스트(index list)를 가진 facet solid entity로 표현된 새로운 RPI포맷 파일을 제안하였다. 이러한 RPI포맷 파일의 색인 리스트 중의 색인 번호(index number)에 의해 각각의 facet들을 참조하고, STL파일의 중복된 정점(vertex)들을 줄여 나가도록 하였다. Dolenc<sup>(4)</sup> 등은 RP시스템을 위한 INSTANTCAM이란 소프트웨어를 개발하면서 Facetted Representation 방식의 STL파일을 바로 사용하지 않고 Surface Representation 방식 IGES 또는 VDAFS 교환파일을 STL파일로 변환하는 연구를 하였다. 그외 여러 논문들이 있지

만 대부분 RP시스템의 원리구현<sup>(5)</sup>과 SL(StereoLithography)을 이용한 사례연구<sup>(6)</sup> 등으로 이 또한 업계의 노하우 공개를 꺼려하는 탓인지 개략적인 내용만 언급하고 있다. 국내에서는 이우종<sup>(7)</sup> 등이 항공 산업이나 자동차 산업에서 널리 사용되어지는 CATIA 모델로부터 SLA(StereoLithography Apparatus)가 인지할 수 있는 파일 형태인 STL파일을 생성시키는 연구를 하였으며, 박형준<sup>(8)</sup> 등은 계층 구조로 이루어진 단면정보를 입력정보로서 삼각형망(Triangular net)을 형성하여 3차원 형상의 기하학적 모델을 구하는 연구를 하였다. 실제 SL(StereoLithography)을 이용하기 위해서는 STL파일로부터 단면정보 변환이 무엇보다 필수적이다. 그러나 아직 여기에 대한 연구는 미미한 상태이다. 따라서 본 연구는 Pro/ENGINEER와 I-DEAS에서 작성한 형상모델을 STL포맷으로 변환한 파일을 입력받아 0.1~0.2mm 두께의 얇은 층의 단면정보를 계산하여 SLICE파일을 생성하도록 하였다.

## 2. Stereolithography 시스템의 원리

SL은 광경화성 수지에 자외선 레이저를 주사하면 광에너지가 노출된 부분은 중합반응에 의해서 경화하는 성질을 이용하는 것을 기본 원리로 하고 있다. SL은 일반적으로 다음과 같은 공정을 거친다.

첫째, STL파일을 지원하는 CAD시스템에서 3차원 형상을 모델링한 후 STL파일로 변환 작업을 거친다. 둘째, STL파일의 삼각망의 정보들을 0.1~0.2mm 정도의 얇은 두께를 가진 xy평면과 평행한 수평층의 단면 데이터를 만드는 슬라이스(Slice) 작업을 행한다. 셋째, 생성된 슬라이스 파일을 이용하여 단면의 경계와 내부 레이저 주사 경로를 생성한다. 넷째, SL장치 구동에 맞는 NC 프로그램을 작성하여 SL장치로 전송한다. 다섯째, NC프로그램의 전송이 시작되면 Fig.1에서와 같이 원하는 형상의 맨 아랫부분부터 최상위 부분까지 순차적으로 한 슬라이스씩 적층되어 최종적으로 원하는 형상의 프로토타입(Prototype)을 얻을 수 있다.

## 3. STL파일의 포맷 및 문제점

STL파일은 삼각형망으로써 솔리드 모델의 표면을 나타내는데, 이 삼각형망의 정보는 Fig.2와 같이 삼각형의 세 정점 좌표(x, y, z)와 삼각형을 이루는 면의 법선

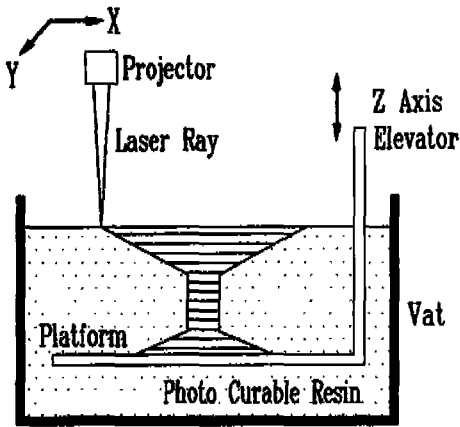


Fig. 1 Principle of Stereolithography

```

solid PART004          -> 파트이름 및 STL파일 시작
facet normal 0.000000e+00 0.000000e+00 -1.000000e+00  -> 법선벡터
outer loop
  vertex 1.000000e+01 -1.000000e+01 0.000000e+00  -> 점점 p1
  vertex 0.600000e+01 0.000000e+01 0.000000e+00  -> 점점 p2
  vertex 1.000000e+01 0.000000e+01 0.000000e+00  -> 점점 p3
endloop
endfacet
endsolid PART004      -> STL파일 끝
    
```

Fig. 2 STL file format

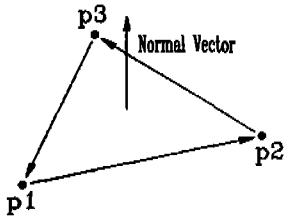


Fig. 3 Right Hand Rule

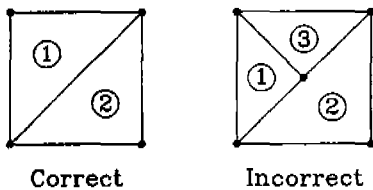


Fig. 4 Vertex-to-Vertex Rule

벡터로 구성된다. 이들 삼각형들은 첫째, 법선벡터와 관련되는 오른손 법칙(Right Hand Rule)을 만족해야 하며 Fig.3은 이와같은 오른손 법칙을 보여주고 있다.

둘째, 삼각형방은 이웃한 삼각형 정점끼리 연결되는

VV(Vertex-to-Vertex) Rule을 만족해야 한다. Fig. 4는 VV Rule의 한 예를 보여주고 있다.

본 연구에서는 Pro/ENGINEER와 I-DEAS CAD 시스템에서 생성된 STL파일을 입력하여 데이터 처리가 이루어지는데 각각의 CAD시스템에서 생성된 STL파일을 직접 다루는 데는 몇가지 문제점이 있다. 첫째, 3차원 모델을 STL파일로 변환시킬 때 STL파일내의 x, y, z좌표값의 자리수가 틀리고, 스트링을 대문자만으로 표시하거나 소문자만으로 표시하는 등 STL파일 포맷이 CAD시스템마다 조금씩 다르게 출력된다는 점이다. 따라서 본 연구에서는 STL파일에서 삼각형 세 정점의 좌표값을 추출하고 이를 SL장치 가공영역 범위내의 데이터로 수정하여 SL파일로 저장하였다.

둘째, STL파일은 Fig. 5와 같이 VV Rule을 만족하기 때문에 불필요한 모서리와 정점이 중복되게 존재한다. 본 연구에서는 단일 연결 리스트(Singly Linked Linear List) 기법을 사용하여 중복된 점을 제거하면서 단면정보를 계산한다. 셋째, CAD시스템에서 복합곡면의 접합부가 있는 형상을 STL파일로 변환할 때 가끔씩 이웃한 두 삼각형 사이에 미세한 틈과 중복이 생긴다. 이런 경우 슬라이스의 단면 데이터들이 페루프가 되지 않기 때문에 레이저 주사 경로점을 정확히 구할 수 없게 된다. 그러나 이런 문제점들은 선행 연구<sup>(2)</sup>에서 다루었기 때문에 본 연구에서는 이러한 문제점이 없는 STL파일을 입력한다.

네째, 법선벡터는 오른손 법칙에 의해 계산되어지는데 간혹 잘못된 법선벡터 즉, 두 이웃한 법선벡터의 방향이 반대가 되어 피비우스띠와 같은 조건이 존재하게 된다. 이와 같은 문제는 슬라이스의 단면정보를 계산하는 데는 영향을 미치지 않기 때문에 고려하지 않았다.

#### 4. STL파일로부터 단면정보 변환절차

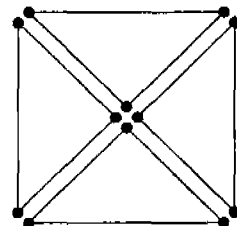


Fig. 5 Edge and vertex redundancy in STL format

본 연구를 통해 개발된 소프트웨어는 삼각형망으로 구성되어 있는 STL파일로부터 레이저 주사 경로점을 구하는데 필요한 단면정보를 가진 SLC파일로의 변환 기능을 한다. 슬라이싱(Slicing) 과정을 Fig.6에 도식적으로 보여주고 있다. 먼저 CAD상에서 솔리드 모델을 작성한 다음 삼각형망의 구조로 이루어진 STL파일로 변환시킨다. xy평면을 z축 방향으로 슬라이스 두께 만큼 이동시키면서 모든 삼각형과의 교점을 구한 후 페

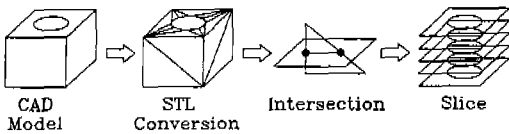


Fig. 6 Slicing Process

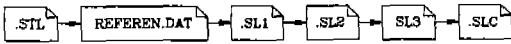


Fig. 7 Data flow from STL file to SLC file

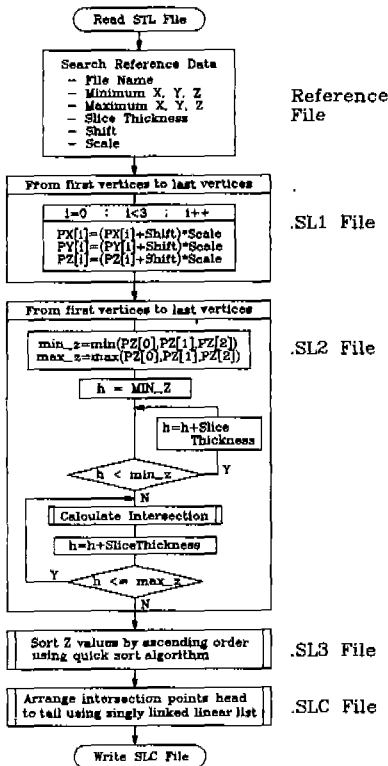


Fig. 8 Flow chart of Slicing Algorithm

루프를 구성하는 이웃한 점들을 순서대로 찾게 된다. 실제 데이터 흐름은 Fig. 7과 같이 파일 단위로 일괄 처리하게 되며, 전체 슬라이싱 알고리즘을 Fig. 8에 흐름도로 나타내었다.

#### 4.1 참조파일의 필요성

일반적으로 STL파일은 아주 단순한 형상을 제외하고는 대부분 수Mbyte에서 수십Mbyte 크기를 가진다. 따라서 파일 단위로 데이터 처리가 불가피하기 때문에 본 연구에서는 Fig. 7에서와 같은 데이터 흐름으로 처리하였다. 그러나 이와 같은 데이터 흐름에서 매번 STL파일을 읽어서 처리할 경우 처리 속도와 공통 정보의 공유 문제가 발생하기 때문에 STL파일에서 슬라이스 단면 데이터를 생성할 때 필요한 정보들을 참조파일에 기록하여 이용하도록 하였다. 참조파일에 기록되는 내용은 제작하고자 하는 조형물의 STL파일명(확장명 제외), 입력된 STL파일중에서 전체 삼각형 정점 좌표 중 최소값, 최대값, 슬라이스 두께, SL장치의 최대 레이저 주사 범위내의 가공을 위한 STL파일을 수정할 값들로 이루어져 있으며 Fig. 9는 참조파일의 포맷을 보여주고 있다. 이상과 같은 참조파일은 STL파일로부터 SLC파일 생성까지의 데이터 흐름에서 생성되는 SL1, SL2, SL3파일에 참조가 되는 정보를 가지고 있다.

먼저, STL파일의 오버헤드 부분을 제거한 순수한 삼각형망의 x, y, z좌표값만을 읽어서 참조파일의 SHIFT와 SCALE 값으로 SL장치의 레이저 주사 범위내의 값으로 수정하여 SL1파일을 만든다.

#### 4.2 삼각형망의 선분과 z=h평면의 교차점 계산

삼각형의 세 정점을 SL1파일로부터 읽어서 이들 점으로 이루어진 선분과 z=h평면과의 교차점을 구하게 된다. 이와 같은 교차점을 구할 때 z=h평면과 만나는 점

FILE NAME	: FLU
MIN X Y Z	: 68.005 68.005 97.856
MAX X Y Z	: 240.005 240.005 111.754
SLICE THICK	: 0.200
SHIFT	: 0.000
SCALE	: 1.000

Fig. 9 Reference file format

을 찾기 위해 SL1파일의 전체 삼각형망의 선분과 교차하는지의 여부를 판별하는 것은 검색과 처리 시간에서 비효율적이기 때문에 본 연구에서는 SL1파일에서 차례로 세 점의 좌표값씩 읽어서 하나의 삼각형에 대해 슬라이스 두께만큼 슬라이싱을 하도록 하였다.

먼저, 주어진 선분과 평면과의 관계는 다음과 같이 4가지 경우로 나눌 수 있다.

- (1) 만나지 않는 경우
- (2) 한 점에서 만나는 경우
- (3) 두 점에서 만나는 경우
- (4) 세 점에서 만나는 경우

위의 4가지 경우 중 두 점에서 만나는 경우의  $x, y, z$  좌표값만을 SL2파일에 저장하게 된다.

그 이유는 STL파일은 반드시 VV Rule을 지켜야 하기 때문에  $z=h$ 평면과의 교차점을 계산하게 되면 중복되는 점이 최소한 2개 이상 존재하게 된다. 따라서 필요한 점은 두 점에서 만나는 경우의 교차점만으로 충분하다.

다음은 교차점을 구하는 방법에 대해 설명하고 있다. STL파일의 각각의 삼각형 세정점으로 세 선분을 구할 수 있다. 두 점  $p1(x1, y1, z1)$ 과  $p2(x2, y2, z2)$ 로 이루어진 선분은 다음과 같은 매개변수( $t$ )를 이용한 직선의 방정식으로 표현할 수 있다.

$$x = x1 + t(x2 - x1) \tag{1}$$

$$y = y1 + t(y2 - y1) \tag{2}$$

$$z = z1 + t(z2 - z1) \tag{3}$$

여기서, 매개변수  $t$ 의 범위를 0과 1사이에서 두면 두 점 사이의 선분이 된다.

일반적인 평면 방정식  $ax+by+cz+h=0$ 은 법선벡터  $n=(a, b, c)$ 를 갖는 평면이다. 따라서  $xy$ 평면은  $a=b=h=0, c=1$ 이므로  $z=0$ 의 평면 방정식이 된다. 그러므로  $z$ 축으로  $h$ 만큼 이동한  $xy$ 평면 방정식은 다음과 같다.

$$z = h \tag{4}$$

위 식 (3)과 (4)에 의해서 매개변수  $t$ 가 결정이 되고 주어진 선분과  $z=h$ 평면과의 교점은 다음식으로 구할 수 있다.

$$x = x1 + (x2 - x1) \left( \frac{h - z1}{z2 - z1} \right)$$

$$y = y1 + (y2 - y1) \left( \frac{h - z1}{z2 - z1} \right)$$

$$z = h$$

이상과 같이 구한 교점  $x, y, z$ 값들을 SL2파일에 기록한다. 그러나 SL2파일은 SL1파일의 삼각형 순서대로 교점을 구하여 저장되어 있기 때문에  $z$ 값이 일정하지 않다. 그래서 슬라이스 단면 데이터 즉,  $z$ 값이 같은 데이터끼리 정렬(sort)할 필요가 있다. 본 연구에서는 다른 정렬 알고리즘에 비하여 평균적 수행 효율면에서 수행 시간이 아주 짧은 퀵 정렬(Quick Sort) 알고리즘을 사용하여  $z$ 값 오름차순으로 정렬한 다음 Fig. 10과 같은 포맷의 SL3파일을 생성한다.

#### 4.3 각 슬라이스의 페루프 연결점 추출

한 층의 슬라이스 단면 데이터는 반드시 페루프를 형성하는 연결점 순서대로 배열되어 있어야 한다. 그러나 SL3파일은 이러한 연결점들이 순서없이 배열되어 있다. 그러나 SL3파일에서는 이러한 연결점들을 한 쌍의 중복된 점으로 가지기 때문에 이러한 중복된 점을 찾아내어 페루프를 형성할 수가 있다. 이 같은 중복된 점은 STL파일의 VV Rule을 따른 것이다.

각 슬라이스의 페루프를 연결점을 추출하기 위해 루프의 시작점( $x, y, z$ )에서 출발하여 인접한 점들을 계속 찾아가면서 다시 시작점을 만나게 되면 하나의 페루프를 형성하게 된다. 특히 일상생활에서 접할 수 있는 대부분의 형상이 중실형상보다는 일정한 두께를 가진

<b>SLICE No. #</b>		
px1	py1	pz1
:		
pxm	pym	pzm
:		
<b>SLICE No. #</b>		
px1'	py1'	pz2
:		
pxn	pyn	pzn
:		

Fig. 10 SL3 file format

중공형상이 훨씬 많다. 따라서 이런 중공형상의 단면은 대개 페루프가 2개 이상 존재를 하게 된다. 이런 경우 루프끼리의 구분이 이루어져야 한다. 본 연구에서는 페루프 연결점을 찾는 방법으로 단일 연결 리스트 자료구조 기법을 사용하였다. 이 기법은 Fig. 11과 같은 구조체를 정의하여 한 슬라이스 단면 데이터의 갯수에 관계없이 동적으로 메모리에 할당(x, y)한다.

단일 연결 리스트의 예를 Fig. 12에 도식적으로 나타내었다. Fig. 12의 (a)는 8점으로 이루어진 사각형의 단면 데이터들이 순서없이 쌍으로 존재하는 예를 나타낸 것으로 (b)의 step 1과 같이 배열되어 있다고 가정하면, (c)와 같은 단계를 거쳐 step 4에서 사각형의 이웃한 단면 데이터들이 각 노드의 포인터에 의해 중복된 점을 제외하면서 순서대로 가리키게 된다.

Fig. 12의 (c)에 대한 세부 설명은 다음과 같다. Fig. 11에서 정의된 구조체 멤버인 x, y, next를 하나로 묶어서 노드(node)라고 부른다. 첫 노드의 주소를 head에 할당하고, 마지막 노드의 next에 NULL값을 할당한다. 인접한 좌표값을 메모리 상에서 이동함이 없이 노드 주소를 가리키는 포인터(pointer)에 의해 각각

의 연결점을 찾게 되는데 먼저, 같은 점을 찾으면 그 노드의 주소를 cur에 할당하고, 다음 주소를 가리키는 next에 NULL을 대입한다. 그리고 이전과 이후의 노드주소는 pre, nxt 포인터 변수에 할당한다. 기존의 NULL을 가진 노드는 이전 연결점의 노드 주소를 가리킨다. 이와 같은 방법을 반복하여 찾은 점이 최초의 시작점과 일치하면 하나의 페루프를 형성하게 된다. 하나의 페루프 데이터 갯수가 슬라이스 단면 데이터 갯수보다 작으면 이는 두개 이상의 페루프가 존재하는 경우로 나머지 슬라이스 단면 데이터 노드의 시작 위치를 가리키는 head와 NULL을 가진 next에 의해 연결점 추출이 가능하다. 위와 같이 페루프 연결점을 찾게 되면 이러한 단면정보를 SLC파일에 기록하게 된다. SLC파일은 Fig. 13과 같은 포맷으로 SLICE No.# 다음의 SOL과 EOL 사이에 하나의 페루프를 이루는 점들의 집합으로 이루어져 있으며 이는 임의의 점으로부터 페루프가 시작하여 이웃한 점들이 시계 또는 반시계방향으로 일정하게 나열된 x, y, z좌표값을 가지며 마지막에 시작점을 다시 한 번 더 기록한다. 한 슬라이스에 복수개의 페루프가 존재할 때는 SLICE No.# 다음에 각각의 페루프 사이의 관계(한 페루프가 다른 페루프에 포함되거나 독립되어 있는 관계 등)에 상관없이 페루프 갯수만큼 SOL과 EOL로 나타낸다.

```

struct joint_point (
    float x;
    float y;
    struct joint_point *next;
)
struct joint_point *head(MAX), *pre, *cur, *nxt;
    
```

Fig. 11 Definition of structure joint\_point

5. 적용예

개발된 프로그램은 펜티엄 마이크로프로세서가 장착

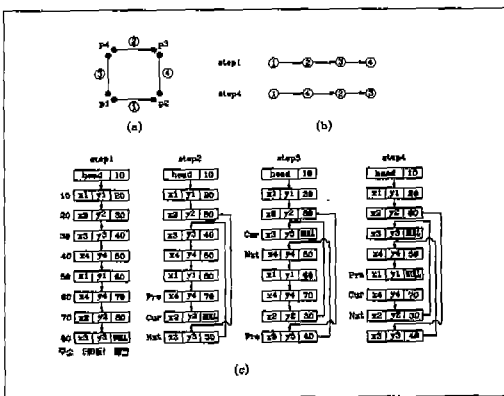


Fig. 12 Closed Loop using Singly Linked Linear List

```

SLICE No. #
SOL
px1    py1    pz
:
pxm    pym    pz
:
px1    py1    pz
EOL
SOL
px1'   py1'   pz
:
pxn'   py'n'  pz
:
px1'   py1'   pz
EOL
:
    
```

Fig. 13 SLC file format

된 IBM 호환기종에서 Borland C++ 3.1 컴파일러를 사용하여 작성하였다. Fig. 14는 컵 형상의 STL파일을 파일 크기 및 총 삼각형 수와 함께 와이어 프레임 방식으로 화면에 나타낸 것이다. Fig. 15는 Fig. 14의 STL파일을 Slice 단면정보로 변환한 후, 이를 총

SLICE 수와 현재 SLICE의 번호를 화면에 나타내고, 이전 슬라이스, 다음 슬라이스, 임의 슬라이스를 볼 수 있는 버튼을 보여주고 있다. Fig. 16은 세탁기 모터에 사용되는 클러치 부분 형상의 STL파일을 보여주고 있으며, Fig. 17은 Fig. 16의 STL파일의 슬라이스 단

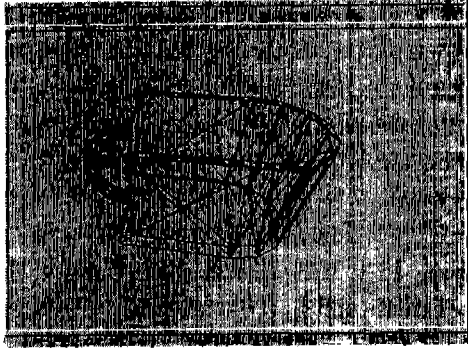


Fig. 14 Example(I) of STL file Visualization

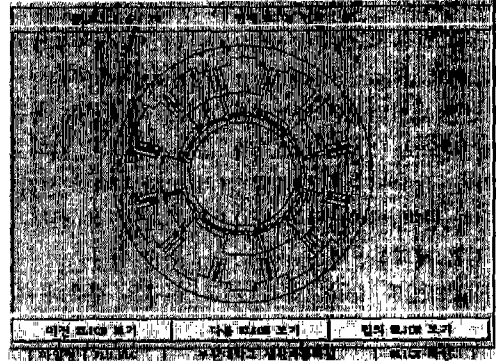


Fig. 17 Example(II) of Slice Information Verification

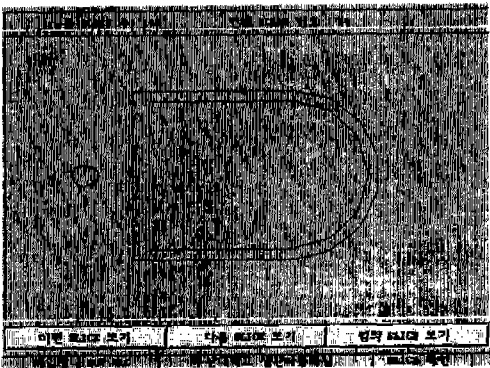


Fig. 15 Example(I) of Slice Information Verification

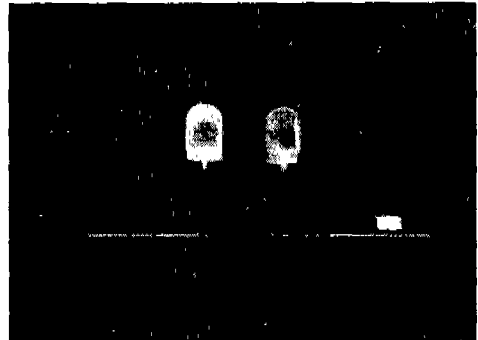


Fig. 18 Finished part(I)

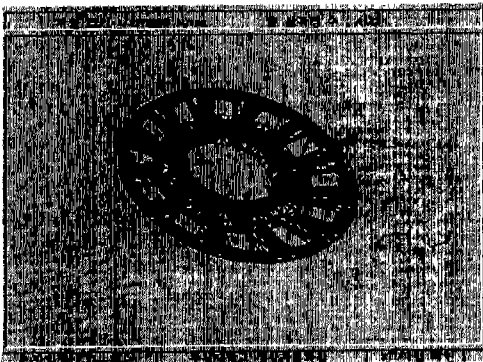


Fig. 16 Example(II) of STL file Visualization

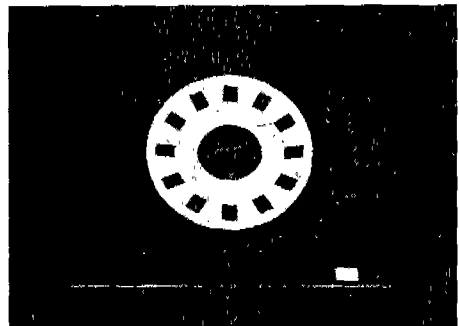


Fig. 19 Finished part(II)

면정보를 보여준다. Fig. 18과 Fig. 19는 우레탄 아크릴레이트계열의 광경화성 수지를 가지고 자외선 영역의 He-Cd 레이저와 500×400×400 크기의 작업영역을 가진 SL장치를 사용하여 제작한 프로토타입이다.

## 6. 결 론

SL운영을 위한 소프트웨어 개발의 일부분으로써 삼각형망으로 이루어진 STL파일을 슬라이스 두께만큼 z축으로 일정하게 슬라이싱하여 단면정보로 변환시키는 프로그램을 개발하였다.

본 논문에서는 Pro/ENGINEER와 I-DEAS에서 변환된 STL파일을 대상으로 하였으나 STL파일을 지원하는 모든 CAD시스템에서 변환된 STL파일도 적용할 수 있다. 슬라이싱 처리 시간을 줄이기 위해서 공통 정보를 가진 참조파일을 생성하고, 이를 이용하여 파일 단위로 일괄 처리 하였으며, 슬라이스별 정렬은 정렬 효율이 좋은 릭 정렬 알고리즘을 적용하였다. 그리고 메모리상에서 이웃한 점들을 검색할 때 배열의 값을 직접 이동시키지 않고 포인터로써 지시하는 단일 연결 리스트 기법을 사용함으로써 단면정보로 변환시키는 처리 시간을 상당히 줄일 수 있었다. 실제 여러 가지 전자제품의 부품에 적용하여 원하는 형상의 프로토타입을 얻을 수 있었으며, 향후 정밀도에 관한 연구가 이루어지면 여러 분야에서 시작품 제작에 많이 기여할 것으로 사료된다.

## 후 기

본 연구는 부산대학교 정밀정형 및 금형가공연구센터 연구 과제로 LG전자 창원 생산기술센터의 연구비 지원으로 이루어졌으며 이에 관계자 제위께 감사 드립니다.

## 참고문헌

1. 白順義紀, 橋原弘之, 五十嵐悟, 齋藤勝政, “複合意匠形狀のリバースエンジニアリングに關する研究”, 日本精密工學會春季大會學術講演會講演論文集

A51, pp.337~ 338. 1995.  
 2. 川嶋英嗣, 田中文基, 岸浪建史, “Delaunay四面體分割によるSTLデータの生成法”, 日本精密工學會春季大會學術講演會講演論文集 C63, pp.823~824, 1995.  
 3. M. J. Wozny, “DATA DRIVEN SOLID FREEFORM FABRICATION”. IFIP Transactions B-3: Human Aspects in Computer Integrated Manufacturing, pp.71~82, 1992.  
 4. A. Dolenc, I. Mäkelä, R. Hovtun, “Better Software for Rapid Prototyping with INSTANTCAM”, IFIP Transactions B-3: Human Aspects in Computer Integrated Manufacturing, pp.449~456, 1992.  
 5. E. Sachs, M. Cima, J. Cornie, “Three-Dimensional Printing: Rapid Tooling and Prototypes Directly from a CAD Model”. Annals of the CIRP Vol. 39, pp.201~204, 1990.  
 6. Takeo Nakagawa, Akitake Makinouchi, Jie Wei, “3-D Plotting of Finite Element Sheet Metal Forming Simulation Results by Laser Stereolithography”, Annals of the CIRP Vol. 41, pp.331~333, 1992.  
 7. 이우중, 이용한, 홍유석, “Rapid Prototyping System을 위한 형상정보 변환절차”, 대한산업공학회지, 제18권 제1호, pp.63~80, 1992.  
 8. 박형준, 김광수, “단면정보를 이용한 형상의 재구성”, 한국정밀공학회지, 제10권 제4호, pp.81~93, 1993.  
 9. Pro/ENGINEER Interface Guide for Release 11.0, Parametric Technology Corporation.  
 10. SLA-500 User Reference Manual, 3D Systems, Inc.