$CAFL^{VM}$ 시스템을 위한 단면정보 추출기의 개발

공 용해*, 엄 태준**, 방 재철***, 김 기석*,
*순천향대학교 정보기술공학부,
**순천향대학교 기계공학과,
***순천향대학교 신소재화학공학부

A Slicer Development for $CAFL^{VM}$ System

Yong Hae Kong*, Tae Jun Eom**, Jae Cheol Bang***, Gee Seog Kim*,
*Division of Information Technology Engineering, Soonchunhyang University,
**Department of Mechanical Engineering, Soonchunhyang University,
***Division of Chemical Engineering, Soonchunhyang University

요 약

본 연구에서는 다종재료를 가공할 수 있는 RP(Rapid Prototyping)인 $CAFL^{VM}$의 구현을 위해 반드시 필요한 단면정보 추출기에 대하여 연구하였다. STL 3D 모형에서 단면정보를 추출하기 위해서, 패싯(facet)과 슬라이싱(slicing) 평면이 가질 수 있는 모든 경우를 고려한 알고리즘을 개발하였다. STL 파일의 구조적인 문제점 때문에 모호성과 처리량 증가 문제가 발생하지만, 본 연구에서는 참조 버퍼를 사용하여 이러한 문제점들을 효과적으로 해결하였으며, 이를 많은 STL 3D 샘플로 검증하였다.

1. 서론

레이저를 사용하여 재료를 절단하고 적층하여 시제품(prototype)을 제작하는 새로운 방식의 RP(Rapid Prototype)인 $CAFL^{VM}$(Computer Aided Fabrication of Lamination for Various Material) 시스템을 개발하고 원천기술 확보를 통하여 시스템 전체를 자체 제작하기 위해서는, STL(Stereolithography interface specification) 파일로 표현되는 3D 모형으로부터 단면정보 추출이 선행되어야 한다.

미국을 포함한 외국의 몇 개 기업에서 신속조형시스템(RP)으로 상품화하고 있지만, 아직 많은 기술적인 문제를 가지고 있을 뿐만 아니라, 사용 가능한 재료의 종류 및 형태가 제한되므로 금형 제작의 추가 공정이 필요하다[1][2]. 다품종 소량생산 환경에 적합하고 금속을 포함한 다양한 재료를 자유롭게 형상화할 수 있는 쾌속원형제작(AP: Agile Prototyping)을 위한 새로운 방식인 $CAFL^{VM}$ 시스템은 그림 1과 같이 레이저를 사용하여 다종 재료의 각 단면을 절단한 다음, 이러한 절단한 단면들을 적층하여 시제품을 완성하게 된다[3].

이러한 $CAFL^{VM}$ 시스템을 개발하기 위해서는 우선 STL 파일로부터 단면 정보를 추출해야 한다. 대부분의 RP들은 STL이라는 설계정보 교환 표준체계에 의거하여 운용되고 있는데, STL은 파일 자체의 구조적인 문제점 때문에 모호성과 처리량 증가 문제를 유발한다. 본 연구에서는 STL의 구조로 인한 문제를 해결하기 위해 참조 버퍼를 사용하고 있으며 단면 정보 추출을 효율적으로 수행할 수 있는 알고리즘을 연구하였다. 제안된 알고리즘과 참조 버퍼는 많은 STL 3D 샘플을 가지고 실험하였다. 실험 결과를 통하여 알고리즘과 참조 버퍼의 사용이 단면정보 추출에 효율적임이 검증되었다.

본 연구는 한국과학재단 목적기초연구(20000199) 지원으로 수행되었음.

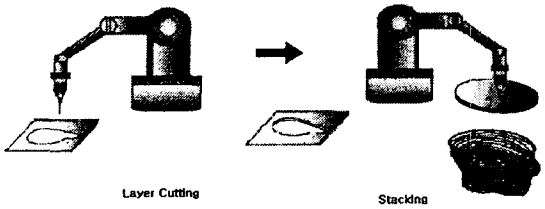

그림 1 레이저로 절단하여 적층하는 시스템

2. 단면 정보 추출 방법

2.1 STL에서의 교차 패싯 추출

먼저 입력된 STL 파일로부터 슬라이싱 평면과 교차되는 패싯을 찾는다. 패싯의 구조는 그림 2와 같다.

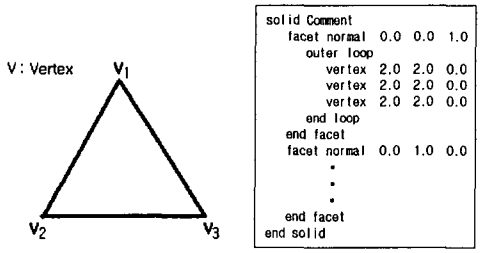

그림 2 패싯의 구조

교차점을 찾기 위해서는 삼각형(패싯)과 슬라이싱 평면이 교차할 때, 삼각형의 어떤 변(또는 꼭지점)과 교차하는지를 알아야 한다. 교차하는 변을 찾기 위해서, 패싯의 세점을 평면방정식에 대입하여 얻어낸 값을 이용하였다. 예를 들어, 평면방정식 $f(x, y, z)$와 점 $P(x, y, z)$가 있을 때,

(1) 점 P가 평면 상에 존재하면 $f(P) = 0$이고,
(2) 점 P가 평면보다 위쪽에 존재하면 $f(P) > 0$ 이고,
(3) 점 P가 평면보다 아래쪽에 존재하면 $f(P) < 0$ 인

관계를 가진다. 슬라이싱 평면과 패싯의 교차점을 찾는 알고리즘을 기술하면 그림 3과 같다.

교차점을 찾기 위해서는 먼저 교차 패싯을 찾아야한다. 교차 패싯을 찾은 후에는 교차 선분(삼각형의 변)을 찾고, 교차 선분에서 교차점을 찾는 순서대로 처리한다. 두 점(패싯의 꼭지점)의 좌표 값을 평면의 방정식에 대입하여 얻은 두 값을 곱하여 0보다 작은 값을 갖는 선분을 찾는 일이 교차점을 찾기 위해 수행해야 할 첫 번째 작업이다.

```
# 결과값(R) : 좌표를 평면방정식 P(x, y, z)에 대입하여
            얻은 값
# 중  점(M) : 3D 공간에서 두 점 좌표의 중간 좌표

① 교점의 정밀도(Accuracy)를 설정;
  중점 저장 버퍼(M) 설정;
  P(M)값 저장 버퍼(R) 설정;

② 매개 변수로 넘어온 두 점(A, B)을 가지고 중점(M)을 계산
  M = (A + B) ÷ 2

③ P(M) 값을 계산하여 결과값 저장 버퍼(R)에 저장
  R = P(M)

④ 방어 프로그래밍 루틴 : 무한 루프를 막기 위해
  IF ( R ≤ Accuracy)
     R = 0;

⑤ 종료 조건 검사 : R값이 0 이면 좌표 M을 반환
  IF ( R == 0 )     // Finish Condition
     return M;        // Return Cross_Point(M)
  ELSE
  {
     IF ( P(A) × P(M) < 0 )
          B = M;
     ELSE            // case of P(M) × P(B) < 0
          A = M;
     GOTO ②;
  }
```

그림 3 교차점 탐색 알고리즘

2.2 교차 선분에서 슬라이싱 점 탐색

패싯에는 교차점의 존재 유무 확인이 필요한 3개의 선분과 3개의 꼭지점이 있다. 교차점의 존재 유무를 확인하기 위해서는 패싯과 평면의 위치 관계를 알아야 한다. 3D 공간에서의 평면과 2D 패싯의 위치 관계는 그림 4와 같이 14가지 경우가 있다. STL 파일의 패싯 정보를 저장하기 위한 자료구조는 삼각형의 꼭지점 순서 정보를 유지해야 한다. 이 때문에, 평면과 패싯의 위치 관계를 분석할 경우에는, 평면이 한 꼭지점을 지나는 경우(즉, $f(V1) = 0$, $f(V2) = 0$, $f(V3) = 0$ 인 경우)를 동일한 경우로 간주해서는 안되고 각각 다른 경우로 나누어 분석하는 알고리즘으로 설계해야 한다.

사용자가 설정한 정밀도를 갖는 최종 물체를 얻기 위해서는 패싯 상의 슬라이싱 점 또한 미리 설정된 정밀도를 충족시켜야 한다. 교차점 탐색 알고리즘에서도 이러한 정밀도를 고려해줘야 한다. 만약 정밀도를 미리 정의해 주지 않으면, 탐색 알고리즘 때문에 시스템이 무한 루프(infinite loop)에 귀착되는 경우가 발생하게 된다. 정밀도가 낮으면, 최종적으로 얻어지는 성형 제품의 질이 낮아지고, 정밀도가 높으면 처리량(computation)이 많아져, 전체적으로 시스템의 성능이 낮아지는 문제가 발생한다. 구현된 단면정보 추출기에서는 경험적인 방법으로 정밀도를 구하여 사용하였다. 슬라이서(slicer)를 구현하기 위해서는 그림 4와 같은 평면과 패싯의 위치 관계를 고려한 구현 알고리즘이 필요하다.

* 가정 : $P()$ - 평면의 방정식
V1, V2, V3 - Facet의 세 꼭지점
L1, L2, L3 - Facet의 세 변

1. 평면이 V1을 지나는 경우
   $P(V1)=0$ & $P(V2)\times P(V3)>0$
2. 평면이 V2를 지나는 경우
   $P(V2)=0$ & $P(V3)\times P(V1)>0$
3. 평면이 V3을 지나는 경우
   $P(V3)=0$ & $P(V1)\times P(V2)>0$
4. 평면이 꼭지점 V1과 변 L2를 지나는 경우
   $P(V1)=0$ & $P(V2)\times P(V3)<0$
5. 평면이 꼭지점 V2와 변 L3을 지나는 경우
   $P(V2)=0$ & $P(V3)\times P(V1)<0$
6. 평면이 꼭지점 V3과 변 L1을 지나는 경우
   $P(V3)=0$ & $P(V1)\times P(V2)<0$
7. 평면이 꼭지점 V1과 꼭지점 V2를 지나는 경우
   $P(V1)=0$ & $P(V2)=0$
8. 평면이 꼭지점 V2와 꼭지점 V3을 지나는 경우
   $P(V2)=0$ & $P(V3)=0$
9. 평면이 꼭지점 V3와 꼭지점 V1을 지나는 경우
   $P(V3)=0$ & $P(V1)=0$
10. 평면이 변 L1과 변 L2를 지나는 경우
    $P(V1)\times P(V2)<0$ & $P(V2)\times P(V3)<0$
11. 평면이 변 L2와 변 L3을 지나는 경우
    $P(V2)\times P(V3)<0$ & $P(V3)\times P(V1)<0$
12. 평면이 변 L3과 변 L1을 지나는 경우
    $P(V3)\times P(V1)<0$ & $P(V1)\times P(V2)<0$
13. 평면이 세 꼭지점(V1, V2, V3)을 지나는 경우
    $P(V1)=0$ & $P(V2)=0$ & $P(V3)=0$
14. 평면과 삼각형(facet)이 만나지 않는 경우
    $(P(V1)>0$ & $P(V2)>0$ & $P(V3)>0)$ *or*
    $(P(V1)<0$ & $P(V2)<0$ & $P(V3)<0)$

그림 4 평면과 패싯의 위치 관계

평면이 패싯의 꼭지점을 지나는 경우에는 꼭지점의 좌표를 알고 있으므로 그림 3과 같은 교차점 탐색 알고리즘을 사용할 필요가 없어진다. 단면정보 추출기를 위한 알고리즘은 그림 4에 있는 모든 경우를 처리할 수 있어야 하고, 또 실시간으로 교차점과 관련된 화면 표시 작업을 처리할 수 있어야 한다. 수천 개에서 수만 개의 패싯으로 구성된 물체를 실시간으로 화면 표시하는 일은 처리량(computation) 증가를 유발한다. 이 때문에 패싯의 개수에 따라 차이는 있지만, 실시간 화면 표시에 문제가 발생할 수 있다.

패싯이 구성하고 있는 형상을 실시간으로 확대·축소하는 기능을 구현하는 것은 어렵지 않으나, 슬라이싱 모드에서는 확대·축소할 때마다 실시간으로 슬라이싱 평면과 패싯의 교차점을 계산하여 화면에 표시하는 작업을 수행해야 하기 때문에 많은 처리량이 발생하여 시스템이 느려진다.

구현된 단면정보 추출기에서는 이러한 문제점을 해결하기 위해, 참조 버퍼를 사용하였다. 참조 버퍼를 사용하는 경우에는 전체 패싯에 대하여 그림 4와 같은 위치 조건을 검사하지 않고 처리할 수 있다. 따라서 처리 속도가 느려지는 문제를 개선할 수 있다. 하지만, 슬라이싱 평면이 변경되었을 때의 처리는 전체 패싯에 대하여 새로운 평면과의 위치 관계를 조사해야 하므로 시스템이 느려지는 것을 피할 수는 없다.

3. 단면정보 추출기의 구현 및 실험결과

3.1 단면정보 추출기의 실험 환경과 구현

아래의 표 2와 같은 개발 도구로 시스템을 구현하여 실험에 사용하였고 그림 5와 같은 처리 단계를 사용하였다.

표 1 실험 환경 및 개발 도구

구분 \ 내용	세부 사항
실험 환경	1. 기 종 : Pentium II 400 MHz 2. RAM : 128 Mbytes 3. OS : Windows 98 SE
개발 도구	1. MS Visual Studio 6.0 C++ Compiler (Win32 API) 2. Graphic Library : OpenGL 1.1.3

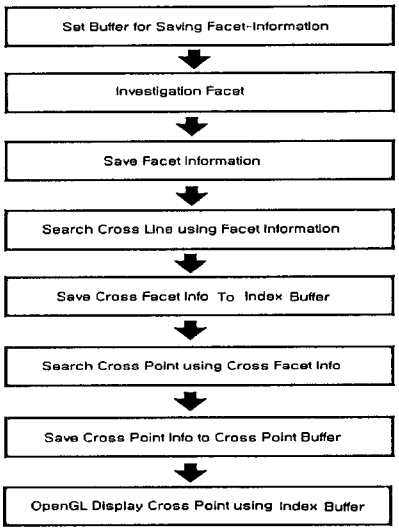

그림 5. 교차점 탐색을 위한 처리 단계

3.2 탐색된 패싯의 모호성 해결 방법

RP 기계장치의 표준 데이터라 할 수 있는 STL 파일의 자료구조 때문에, 슬라이싱 점을 찾는데 다음과 같은 몇 가지 문제점이 발생한다.

(1) 물체의 동일한 영역을 표현하는 패싯이 중복되어 있는 STL 파일이 많다.
(2) 패싯의 세 꼭지점에 대한 정보를 실수형으로 저장하기 때문에, 확대할 경우, 물체를 이루는 패싯 사이에 빈 공간(null space)이 발생할 위험성이 높다.

(3) 패싯을 이루는 삼각형 꼭지점의 순서 정보가 물체의 앞뒤를 나타내므로 임의로 순서로 바꿔 저장할 수 없다.
(4) STL 파일 내부의 패싯 데이터 위치(순서)가 패싯 사이의 인접성과는 무관하다.
(5) 정밀도를 표현하기 위해 많은 부동 소수점 연산을 필요로 한다.
(6) 3차원 물체를 나타내는 패싯의 꼭지점 좌표의 기준이 되는 원점 좌표가 없다.
(7) 패싯의 법선 벡터 값에 오류가 많다.

그림 6은 슬라이싱 평면과 패싯의 교차점을 탐색할 경우에 발생할 수 있는 모호성을 보여주고 있다(그림 6은 슬라이싱 평면이 삼각형의 한 변을 지나는 경우와 한 꼭지점을 지나는 경우에 발생하는 모호성을 설명해준다).

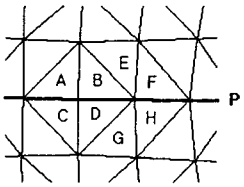

그림 6 교차 패싯 탐색의 모호성

그림 6에서 슬라이싱 평면 P와 관련된 패싯은 △A, △B, △C, △D, △E, △F, △G, △H 등이다. △E와 △G는 단지 한 꼭지점만이 평면에 존재하지만, 평면이 지나는 패싯으로 처리된다. 이런 경우, △B, △D, △E, △F, △G, △H가 만나는 점은 슬라이싱을 포인트 탐색을 어렵게 하는 모호성(ambiguity)을 가지고 있다. 또, 슬라이서의 동작 후에, 작동하는 경로 추적기에서 레이저의 경로를 찾을 때, 시간 비용을 초래하게 된다. 그림 4에 있는 조건 7, 8, 9가 이와 같은 경우를 발생시킨다. 구현된 단면정보 추출기에서는 모호성 문제를 해결하기 위해서, 그림 7과 같이 2가지 조건을 만족해야 교차 패싯으로 결정하도록 하였다.

```
조건 ① - 평면이 패싯의 한 꼭지점(V1)을 지나는 경우,
  : 꼭지점과 마주보는 선분(L2)을 평면이 지나는지 검사한다.
  IF ( P(V2)×P(V3) < 0 )
     교차 패싯으로 결정하고 참조 버퍼에 패싯에 대한
     정보(패싯 번호, 교차 꼭지점, 교선)를 저장
  ELSE
     교차 패싯이 아닌 것으로 결정

조건 ② - 평면이 Facet의 두 꼭지점(V1, V2)을 지나는 경우,
  : 나머지 한 꼭지점(V3)의 좌표를 평면방정식에 대입하여
    그 결과값을 가지고 결정한다.
  IF ( P(V1) > 0 )
     교차 패싯으로 결정하고 참조 버퍼에 패싯에 대한 정보
     (패싯 번호, 교차 꼭지점, 교선)를 저장
  ELSE
     교차 패싯이 아닌 것으로 결정
```

그림 7 교차 패싯으로 결정하기 위한 2가지 조건

3.3 단면정보 추출기의 구동 결과

그림 8은 교차 패싯의 모호성 문제 해결 방법으로 얻어낸 단면정보 추출기의 결과 화면이다.

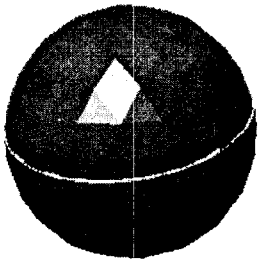

그림 8 슬라이싱을 수행한 솔리드(solid) 영상

슬라이싱 평면, 패싯이 만드는 3D 물체의 형상, 패싯의 개수 등에 따라 차이는 있지만 모호성을 제거한 후, 그 이전보다 수행 속도가 빨라졌다. 특히, 슬라이싱 모드에서 확대·축소 기능을 사용할 경우, 실행 속도가 향상되었다. 그림 8을 통해서 알 수 있듯이, 모호성을 일으키는 패싯들을 제거하여 패싯들이 제거되었다. 그림 9는 그림 8의 와이어 프레임 영상이다.

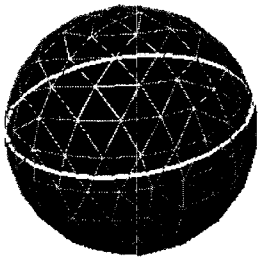

그림 9 그림 8의 와이어 프레임 영상

그림 9를 보면, 슬라이싱 점을 연결한 슬라이싱 선을 볼 수 있다. 다각형의 꼭지점 순서를 가지고 앞면

과 뒷면을 구분하여 화면 표시한 것이다. 그림 10은 그림 9의 슬라이싱 선을 슬라이싱 점으로 표시한 화면이다.

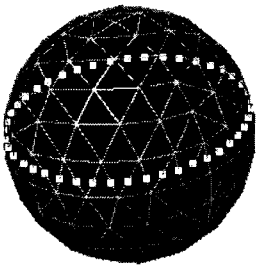

그림 10 슬라이싱 점 표시

그림 11은 슬라이싱 평면과 패싯이 가질 수 있는 여러 가지 위치 관계를 제안된 알고리즘으로 동시에 처리하여 화면 표시한 것이다.

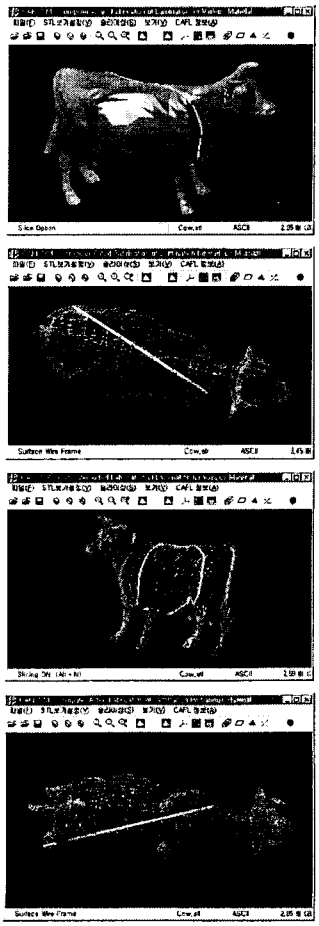

그림 11 복잡한 형상에서의 슬라이싱 예

4. 결론

다종 재료 가공용 쾌속 RP인 *CAFL*™을 개발하기 위해서는 반드시 STL 파일로부터 단면 정보를 추출 과정이 선결되어야 한다.

본 연구에서는 참조 버퍼와, 슬라이싱 평면과 패싯의 위치 관계를 고려한 알고리즘을 보였으며, STL 파일의 자료구조로 인해 발생하는 모호성을 해결한 알고리즘을 보였다. 구현된 단면정보 추출기는, 슬라이싱 평면이 패싯(삼각형)의 꼭지점이나 변을 지나는 특별한 경우나 두 변을 가로지르는 일반적인 경우에서나, 정확하게 동작하였다. 변을 지나는 경우, 슬라이싱 평면은 두 개의 패싯을 지나게 된다. 이러한 경우에 발생하게 되는 모호성과 처리량(computation)의 증가 문제를 제안된 알고리즘으로 해결할 수 있었다.

단면정보 추출기에서 얻어낸 단면 정보를 이용하여 *CAFL*™ 시스템에서 여러 종류의 재료를 레이저 절단할 수 있다. 추출기에서 얻어낸 단면정보를 이용하여 레이저 절단을 위한 경로 최적화와 탄젠트 절단(tangent cutting)을 위한 레이저의 궤적을 연구할 계획이다.

[참고문헌]

[1] J. D. Cawley, A. H. Heuer, W. S. Newman, and B. B. Mathewson, "Computer-Aided Manufacturing of Laminated Engineering Materials", Am. Ceram. Soc. Bull., 75, 1996

[2] E. A. Griffin, D. R. Mumm, and D. B. Marshall, "Rapid Prototyping of Functional Ceramic Composites", American Ceramic Society Bulletin, Vol. 75, No. 7, p. 65, 1996

[3] J. D. Cawley, A. H. Heuer, W. S. Newman, and B. B. Mathewson, "Computer-Aided Manufacturing of Laminated Engineering Materials", American Ceramic Society Bulletin, Vol. 75, No. 5, 1996