

임의의 점 군 데이터로부터 쾌속조형을 위한 입력데이터의 자동생성

유동진[#]

Automatic Generation of the Input Data for Rapid Prototyping from Unorganized Point Cloud Data

Dong-Jin Yoo[#]

ABSTRACT

In order to generate the input data for rapid prototyping, a new approach which is based on the implicit surface interpolation method is presented. In the method a surface is reconstructed by creating smooth implicit surface from unorganized cloud of points through which the surface should pass. In the method an implicit surface is defined by the adaptive local shape functions including quadratic polynomial function, cubic polynomial function and RBF(Radial Basis Function). By the reconstruction of a surface, various types of error in raw STL file including degenerated triangles, undesirable holes with complex shapes and overlaps between triangles can be eliminated automatically. In order to get the slicing data for rapid prototyping an efficient intersection algorithm between implicit surface and plane is developed. For the direct usage for rapid prototyping, a robust transformation algorithm for the generation of complete STL data of solid type is also suggested.

Key Words : Unorganized point cloud data (임의의 점 군 데이터), Implicit surface (음 함수 곡면), Rapid prototyping (쾌속조형)

1. 서론

CAD(Computer Aided Design) 시스템 또는 3D 스캐너(scanner) 시스템으로부터 생성된 STL 파일의 폴리곤모델은 무수한 복잡한 형상의 홀(hole), 퇴화된 삼각형(degenerated triangle) 및 삼각형간의 겹침(overlap)과 같은 많은 기하학적 결함들을 포함하고 있다. 이런 결함들을 제거하기 위하여 많은 STL 편집기가 상용화되어 이용되고는 있으나

사용자의 많은 수작업이 요구되고 작업시간의 과다로 인한 시간 비용 문제 등 해결해야 할 문제점이 많이 남아 있는 것이 현실이다. STL 파일이 가지고 있는 이러한 문제점 때문에 최근 쾌속조형(rapid prototyping) 분야에서는 CAD 모델이나 STL 모델을 이용하지 않고 점 군 데이터로부터 제품을 직접 제작하는 직접쾌속조형법에 대한 연구가 진행되고 있다.^{1,4} 이러한 연구들의 주요 핵심은 점 군 데이터를 이용하여 슬라이싱(slicing) 데이터를

접수일: 2007년 4월 12일; 게재승인일: 2007년 9월 12일

[#] 교신저자: 대전대학교 컴퓨터응용 기계설계공학과

E-mail: djyoo@daejin.ac.kr Tel. (031) 539-2031

추출해 내는 것에 있는데 생성된 데이터의 신뢰도, 정확도 및 계산속도 등 고려해야 할 인자가 많은 문제로서 아직까지 국내에서는 이에 대한 연구가 없었다.

본 연구에서는 임의의 점 군 데이터를 지나는 부드러운 곡면을 생성한 후 생성된 곡면으로부터 직접 슬라이싱 데이터를 추출하는 새로운 방법을 제안하고자 한다. 본 연구에서는 임의의 점 군 데이터를 지나는 곡면을 생성하기 위하여 음함수 곡면기법⁵⁻⁸ 과 영역분할법⁹⁻¹² 을 이용하였는데 3 차원상에 존재하는 무수한 임의의 점들을 부드러운 하나의 곡면으로 표현할 수 있을 뿐 아니라 슬라이싱 데이터를 추출하기 위한 곡면상의 좌표값 및 법선벡터 등의 계산이 용이하다는 장점이 있다. 또한 저자의 선행연구결과¹¹ 를 통하여 보고된 적응적 영역분할법(ADDM)을 이용함으로써 수 십만, 수 백만 개의 불완전한 점 군 데이터로부터 수 분 내에 결함이 없는 슬라이싱 데이터를 얻는 것이 가능하도록 하였다. 아울러 상용시스템의 사용자를 위하여 생성된 곡면을 홀메움 및 오프셋과정을 통하여 다양한 형태의 솔리드로 자동변환 하는 방법도 제안하고자 한다.

2. 점 군 데이터로부터 음함수곡면의 생성

본 연구에서는 임의의 점 군 데이터를 통과하는 부드러운 곡면을 하나의 새로운 음 함수 곡면으로 정의하는 방법을 사용하였다. 또한 곡면의 보간 및 슬라이싱 데이터 생성에 소요되는 시간을 크게 줄이기 위하여 다양한 형상함수를 선택적으로 사용하는 적응적 영역분할법을 이용하였다. 적응적 영역분할법은 Fig. 1 에서 보는 바와 같이 점 군이 차지하고 있는 3 차원 공간을 복수개의 영역으로 나눈 후 각 영역마다 서로 다른 형상함수(shape function)를 사용하는 방법으로서 본 연구에서는 2 차 다항식과 3 차 다항식 및 레이디얼 기초함수를 사용하였다. 한 영역내의 모든 점들을 근사하게 통과하는 곡면을 Table 1 에서 보는 바와 같이 다양한 형상함수로 구성하였다. 형상을 표현하기 위하여 2 차 다항식곡면의 경우에는 6 개, 3 차 다항식곡면의 경우에는 10 개, 레이디얼 기초함수를 이용한 곡면의 경우에는 보간 점 수에 비례하는 개수의 계수를 갖게 된다. 따라서 임의의 점 군 데이터로부터 생성될 곡면에 편평한 부분이 많

은 경우에는 2 차 및 3 차 다항식곡면으로 충분히 표현이 가능하므로 제반 계산에 소요되는 시간을 크게 단축시킬 수 있다. 주어진 점 군 데이터로부터 곡면을 생성하는 과정 및 생성하고자 하는 곡면모델의 정확도의 조절 등에 관한 자세한 알고리즘은 저자의 선행연구¹⁰⁻¹² 에 상세히 기술되어 있으므로 본 논문에서는 생략하도록 한다.

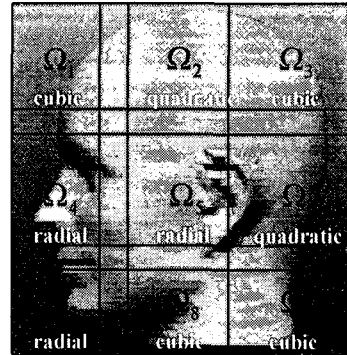


Fig. 1 Schematic diagram illustrating the concept of adaptive domain decomposition method

Table 1 Various types of shape functions

<p>Quadratic polynomial</p> $f(\mathbf{X}) = w - (a_0 + a_1u + a_2v + a_3u^2 + a_4uv + a_5v^2) \quad (1)$
<p>Cubic polynomial</p> $f(\mathbf{X}) = w - (a_0 + a_1u + a_2v + a_3u^2 + a_4uv + a_5v^2 + a_6u^3 + a_7u^2v + a_8uv^2 + a_9v^3) \quad (2)$
<p>Radial basis function</p> $f(\mathbf{x}) = \sum_{j=1}^N \lambda_j \phi(\mathbf{x} - \mathbf{c}_j) + P(\mathbf{x}) \quad (3)$ <p>where,</p> $\phi(\mathbf{x}) = \mathbf{x} ^2 \log(\mathbf{x})$ $P(\mathbf{x}) = P_0 + P_1x + P_2y + P_3z$

분할된 각 영역에서의 곡면방정식이 구해지면

혼합함수(blending function)에 의하여 전체 곡면방정식을 구할 수 있다.¹⁰

$$d_i(\mathbf{P}) = 1 - \prod_{r \in x,y,z} \frac{4(\mathbf{P}_r - \mathbf{S}_r)(\mathbf{T}_r - \mathbf{P}_r)}{(\mathbf{T}_r - \mathbf{S}_r)^2} \quad (4)$$

$$W_i(\mathbf{P}) = -6d_i^5 + 15d_i^4 - 10d_i^3 + 1 \quad (5)$$

$$w_i(\mathbf{P}) = \frac{W_i(\mathbf{P})}{\sum_{j=1}^{ND} W_j(\mathbf{P})} \quad (6)$$

$$F(\mathbf{P}) = \sum_{i=1}^{ND} f_i(\mathbf{P})w_i(\mathbf{P}) \quad (7)$$

여기서,

S: 각 영역의 최소 x,y,z 값

T: 각 영역의 최대 x,y,z 값

P: 구하고자 하는 점의 좌표값

w_i : i 번째 영역의 가중치

ND: 분할영역의 개수

F(P): 점 **P** 에서의 음 함수값이다.

3. 쾌속조형을 위한 입력데이터의 생성

3.1 슬라이싱 데이터의 추출

주어진 점 군 데이터들이 하나의 곡면으로 정의 되었으므로 평면과 이 음 함수 곡면과의 교차선을 구함으로써 슬라이싱 데이터를 추출할 수 있다. 음 함수 곡면과 평면과의 교차문제는 다음과 같은 비선형 연립방정식을 푸는 문제로 표현될 수 있다.¹³

$$F_1(\mathbf{X}) = 0 \quad (8)$$

$$F_2(\mathbf{X}) = \mathbf{U}_2 \cdot \mathbf{X} - p = 0 \quad (9)$$

$$F_3(\mathbf{X}) = (\mathbf{X} - \mathbf{X}_0) \cdot \mathbf{T}_0 - d = 0 \quad (10)$$

여기서,

\mathbf{U}_2 : 교차평면에 수직한 단위벡터

P: 원점과 평면과의 수직거리

\mathbf{X}_0 : 시작점

d: 전진거리이다.

식 (8)은 음 함수 곡면의 방정식을, 식 (9)는 교차시킬 평면의 방정식을, 식 (10)은 \mathbf{X}_0 에서 \mathbf{T}_0 방향으로 **d** 만큼 떨어져 있는 평면의 방정식을 의미한다. 식 (10)에서 \mathbf{T}_0 는 \mathbf{X}_0 위치에서의 음 함수 곡면의 법선벡터와 교차평면에 수직한 벡터에 수직한 단위벡터로서 식 (11)과 같이 정의된다.

$$\mathbf{T}_0 = \frac{\mathbf{U}_1 \times \mathbf{U}_2}{|\mathbf{U}_1 \times \mathbf{U}_2|} \Big|_{\mathbf{X}=\mathbf{X}_0} \quad (11)$$

여기서,

\mathbf{U}_1 : \mathbf{X} 위치에서 음 함수 곡면에 수직한 단위벡터이다.

식 (10)에서 **d** 는 현재의 교차점에서 나가야 할 전진거리로서 식 (12)와 같이 표현할 수 있다.¹³

$$d = L(1 - \frac{\delta}{\rho}) \quad (12)$$

$$L = \sqrt{4\delta(2\rho - \delta)} \quad (13)$$

여기서,

ρ : 현재 점에서의 곡률반경

δ : 허용오차이다.

현재 점에서의 곡률반경이 계산되고 허용오차가 주어지면 식 (8)과 식 (9) 및 식 (10)을 이용하여 식 (14)와 같은 뉴턴랩슨 반복계산식을 구성할 수 있다.

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ z_{i+1} \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} - \mathbf{J}^{-1} \begin{bmatrix} F_1(x_i, y_i, z_i) \\ F_2(x_i, y_i, z_i) \\ F_3(x_i, y_i, z_i) \end{bmatrix} \quad (14)$$

여기서,

$$\mathbf{J} = \begin{bmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial y} & \frac{\partial F_1}{\partial z} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial y} & \frac{\partial F_2}{\partial z} \\ \frac{\partial F_3}{\partial x} & \frac{\partial F_3}{\partial y} & \frac{\partial F_3}{\partial z} \end{bmatrix}$$

식 (8)로 표현된 음 함수 곡면방정식은 기존의 CAD 분야에서 널리 사용되고 있는 B-spline 이나 NURBS 곡면방정식에 비해 함수 값 및 법선벡터 등의 계산이 용이하고 계산시간도 빠르다는 장점을 갖고 있어 식 (14)의 반복계산에 소요되는 시간이 다른 곡면방정식을 이용하는 것 보다 상대적으로 작아 효율적인 교차선 계산을 수행할 수 있다. 초기 시작점 X_0 는 일정한 x 또는 일정한 y 지점에서 y 축과 평행한 직선 또는 x 축과 평행한 직선과 음 함수 곡면과의 교차점으로서 다음과 같이 비선형방정식의 해를 구함으로써 얻어질 수 있다. 일반적인 직선의 방정식은 식 (15)와 같은 벡터방정식으로 표현할 수 있다.

$$r = r_0 + \lambda m \tag{15}$$

여기서,

m : 직선방향의 단위벡터

r_0 : 직선의 시작점

λ : 직선의 길이를 나타내는 매개변수

이다.

식 (15)로 표현되는 직선과 식 (8)로 표현되는 음 함수 곡면과의 교차점은 다음과 같은 λ 에 관한 뉴턴랩슨 반복계산식으로부터 구할 수 있다.

$$\lambda_{i+1} = \lambda_i - \frac{F_i(\lambda)}{F_i'(\lambda)} \tag{16}$$

$$F_i' = n \cdot \nabla F_i \tag{17}$$

Fig. 2 에서 보는 바와 같이 초기점 X_0 를 시작으로 해서 T_0 방향으로 전진하면서 전체 교차선을 구하게 된다. 식 (12)와 식 (13)의 곡률반경 ρ 는 교차평면에 투영된 음 함수 곡면의 곡률반경으로서 본 연구에서는 X_0 와 X_0 에서 T_0 및 $-T_0$ 방향으로 약간 떨어져 있는 음 함수 곡면상의 점들을 구해 이 세 점을 지나는 원의 반지름으로 하였다. 형상이 복잡해지면 Fig. 3 에서 보는 바와 같이 한 교차평면에 대하여 교차선이 여러 부분에 생성될 수 있다. 이러한 경우에도 안정적인 교차선 계산 및 루프생성이 가능하도록 본 연구에서는 음 함수 곡면의 가시화 방법으로 널리 사용되고 있는 마칭 큐브 알고리즘(marching cube algorithm)

을 이용하였다. 마칭 큐브 알고리즘은 음 함수 곡면이 정의 되어 있는 공간을 작은 직육면체 또는 정육면체로 나눈 후 각 코너점에서의 함수값을 구하여 전부가 양 또는 음이면 지나가고 양 과 음의 값이 혼재 되어 있는 경우에는 적절한 방법에 의하여 삼각형을 구성하는 방법이다.¹⁰ 이러한 마칭 큐브 알고리즘의 2 차원 버전인 마칭 렉탱글 알고리즘(marching rectangle algorithm)에 의하여 개략적인 교차선의 수치 값 및 루프의 총 개수 등을 미리 파악한 다음 각 교차선의 시작점을 식 (16) 및 식 (17)을 이용하여 구하고 정확한 전체 교차선을 다시 생성하는 단계적인 방법을 제안함으로써 교차선 계산 및 루프 생성 알고리즘의 신뢰성 및 강건성 향상을 도모하였다.

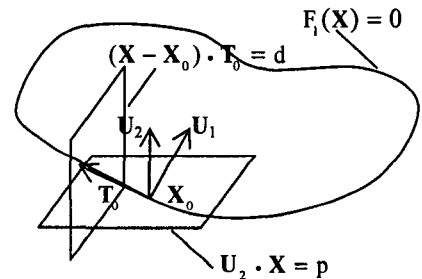


Fig. 2 Intersection between implicit surface and plane

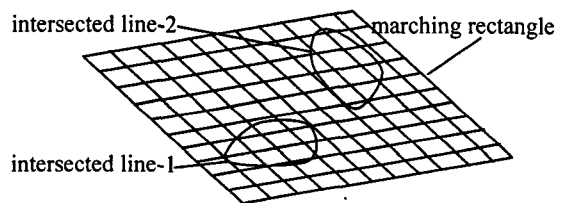


Fig. 3 Schematic diagram illustrating the concept of marching rectangle algorithm(MRA)

3.2 음 함수 곡면으로부터 솔리드로의 변환

생성된 곡면을 이용하여 RP 장비로 바로 입력될 수 있는 솔리드 형태의 STL 데이터를 만들기 위해서는 홀 메움 및 오프셋 (offset) 과정이 필요하다. 본 연구에서는 음 함수 곡면으로부터 마칭 큐브알고리즘 및 스무딩¹⁰ 을 이용하여 초기 삼각망을 구성한 후 열려진 곡면(open surface)의 경우에는 홀 메움에 의하여 닫혀진 형태의 솔리드로 변환하거나 오프셋과정을 통하여 셸(shell) 형태의

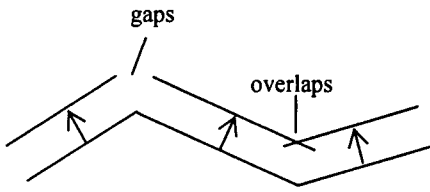


Fig. 4(a) Face normal offsetting

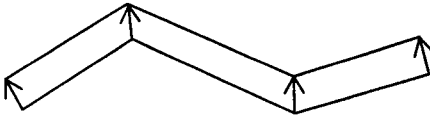


Fig. 4(b) Vertex normal offsetting

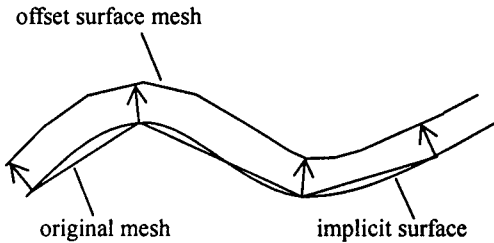


Fig. 4(c) Implicit surface normal offsetting

솔리드로 변환하였다. 닫혀진 곡면(closed surface)의 경우에는 바로 솔리드 STL 데이터로 만들 수도 있고 필요에 따라서 오프셋을 한 후 STL 데이터로 변환할 수 있다. 종래의 삼각망의 오프셋 방법에는 면을 법선 방향으로 오프셋 거리만큼 이동하는 면 오프셋 방법과 각 꼭지점을 평균 법선 방향으로 이동하는 꼭지점 오프셋 방법이 있다. 면 오프셋 방법은 오프셋의 정밀도는 우수하지만 꼭지점과 모서리에 연결해야 하는 공간과 삼각형간의 겹침이 발생해 데이터가 증가하고 후처리 프로그램 작성해야 하는 단점이 있고 꼭지점 오프셋 방법은 필요 없는 공간과 겹침이 발생하지 않으며 데이터 크기가 증가하지도 않지만 오프셋의 정밀도가 크게 떨어지는 단점이 있어서 이 두 방법의 장단점을 보완할 수 있는 효율적인 오프셋 방법이 요구된다. 본 연구에서는 Fig. 4(c)와 같이 보간된 음 함수 곡면으로부터 각 꼭지점에서의 법선벡터를 정확히 구해 오프셋 한 후 이 점 들을 다시 곡면으로 보간하므로써 정밀도도 향상시키고 공간과 겹침으로 인한 후처리 작업의 필요성도 해소시켰

다. 음 함수 곡면상의 한 점에서의 법선 벡터는 식(18) 과 같이 구하였다.

$$\mathbf{n} = \frac{\nabla F_1}{|\nabla F_1|} \quad (18)$$

여기서,

$$\nabla F_1 = \frac{\partial F_1}{\partial x} \mathbf{i} + \frac{\partial F_1}{\partial y} \mathbf{j} + \frac{\partial F_1}{\partial z} \mathbf{k}$$

이다.

4. 적용 및 결과

본 연구에서 제안한 임의의 점 군 데이터로 부터의 슬라이싱 데이터 추출 알고리즘의 효율성 및 타당성을 알아 보기 위하여 개발된 프로그램을 실제 쾌속조형제품 개발에 적용해 보았다.

Fig. 5 는 삼각망으로 구성된 초기 STL 파일로부터 점 군 데이터를 구성하고 점 군 데이터를 부드러운 하나의 음 함수 곡면으로 보간한 후 슬라이싱 데이터를 계산한 결과를 보여주고 있다. Fig. 5(a) 는 3D 스캐너를 이용하여 얻은 170 여 만개의 삼각망으로 구성된 초기 STL 파일의 형상을 보여 주고 있는데 스캐너 자체의 한계성으로 인한 측정데이터의 누락 및 서로 다른 시점에서 얻은 데이터를 병합하는 과정에서의 오류 등으로 인하여 그림에서 보는 바와 같이 복잡하고 커다란 불규칙한 형상의 홀들이 형상 전체에 걸쳐 분포되어 있으며 삼각형간의 겹침도 많아 STL 전용편집기를 이용해도 완전한 모델로 수정하기가 매우 힘든 모델이다. 점 군 데이터를 바로 슬라이싱 하는 방법²⁴ 들에 있어서는 교차선 계산에 소요되는 시간은 본 방법에 비하여 다소 적게 걸리나 Fig. 5 의 모델과 같이 형상에 큰 홀(hole)들이 존재하는 경우에는 교차선 계산 자체가 불가능하다. 또한 본 연구에서처럼 형상에 대한 닫힌 형태의 솔리드 모델 및 셸(shell) 형태의 솔리드와 같은 다른 응용을 위한 변형된 모델을 생성할 수도 없는 단점이 있다. Fig. 5(b) 는 초기 STL 파일로부터 얻은 점 군 데이터를 이용하여 음 함수 곡면을 생성한 후 z 방향으로 슬라이싱 한 후의 결과를 보여 주고 있다. 결과에서 볼 수 있는 바와 같이 초기 STL 파일에

홀 및 삼각형간의 겹침 등 많은 결함들이 있었으나 전체 형상이 하나의 완전한 곡면으로 재생성 되었으므로 슬라이싱 데이터를 얻는데 전혀 문제점이 없는 것을 확인할 수 있었다. IBM RS 43P 140 w/s 에서 보간 과정에 10 분, 슬라이싱 데이터추출에 30 초 정도 소요되었는데 원래 STL 데이터가 170 여 만개의 삼각형 망으로 구성되어 있었다는 것을 감안해 보면 제안된 방법이 효율성 면에 있어서도 장점을 갖고 있는 것을 확인할 수 있었다.

사실 RP 분야와 관련된 개발자 입장에서는 이와 같은 슬라이싱 데이터의 추출에 관심이 많겠지만 상용시스템의 사용자 입장에서는 3D 스캐너로부터 얻은 불완전한 점 군 데이터 또는 STL 파일을 가능하면 빠른 시간 안에 RP 장비의 완전한 입력데이터로 전환하는데 더 관심이 많을 것이다. Fig. 6 은 이런 관점에서 점 군 데이터를 음 함수 곡면으로 보간한 후 마칭큐브알고리즘 및 스무딩¹⁰에 의하여 생성된 삼각망을 다양한 형태의 솔리드 STL 파일로 변환한 결과를 보여 주고 있다. Fig. 6(a) 처럼 완전히 닫힌 형태의 솔리드 및 Fig. 6(b) 와 같은 셸(shell) 형태의 솔리드 등 다양한 형태의 솔리드로의 변환이 홀 매움 및 오프셋(offset) 과정을 통하여 용이하게 수행될 수 있도록 자동변환 프로그램들도 개발하였다.

또한 최근에는 제작하고자 하는 형상의 곡면기울기를 계산하여 임계 적층 두께를 계산하고 임계 적층 두께에 따른 슬라이싱 데이터를 추출하는 방법¹⁴도 연구 보고 되었는데 본 연구에서도 최적의 슬라이싱 데이터 추출이 가능하도록 해당 평면에서의 교차점에서의 모든 곡면기울기를 계산하여 이를 슬라이싱 간격설정에 반영하도록 하였다. Fig. 7 에서 보는 바와 같이 형상이 복잡하고 곡면의 기울기가 큰 부분에서는 조밀하게 슬라이싱 되고 비교적 형상이 단순하고 곡면의 기울기가 작은 부분은 슬라이싱 간격이 크게 조절되고 있음을 확인할 수 있다. Fig. 8 은 Fig. 6 의 두 형태의 솔리드를 RP 장비를 이용하여 제작한 결과를 보여 주고 있는데 역설계와 폐속조형기술의 통합화에 가장 큰 걸림돌이 되었던 스캔데이터가 가지고 있는 여러 기하학적인 문제의 해소에 본 연구 결과가 효율적으로 적용될 수 있는 가능성을 확인할 수 있었다.

Fig. 9 는 본 연구에서 제안한 음 함수 곡면 오프셋 알고리즘을 이용하여 Agripa 모델을 오프셋

한 결과를 보여주고 있는데 단순히 삼각망만의 데이터가 아닌 부드러운 곡면의 데이터를 이용하기 때문에 오프셋 된 후의 곡면도 매우 질이 좋음을 확인할 수 있었다.

Fig. 10 과 와 Fig. 11 은 많이 알려져 있는 데이터들¹⁵에 대하여 본 연구에서 개발한 프로그램을 이용하여 RP 장비의 입력데이터를 생성한 후 직접 형상을 제작한 결과들을 보여주고 있다. 372563 개의 점 데이터로 구성된 Igea 모델의 경우 보간 및 솔리드로의 변환에 12 분 25 초, 슬라이싱 데이터추출에 5 초 정도 소요되었고 610848 개의 점 데이터로 구성된 인체모델의 경우에는 보간 및 솔리드로의 변환에 19 분 10 초, 슬라이싱 데이터추출에 12 초 정도 소요되었다. 이상과 같은 다양한 점 군 데이터에 대한 적용 결과로부터 초기 스캔 점 데이터에서 RP 제품 제작까지의 전 과정이 하나의 시스템으로의 통합화가 가능하다는 것을 확인할 수 있었다. 별도의 수정프로그램에 의한 지루한 수작업이 거의 필요없게 되었을 뿐만 아니라 점 군 데이터로부터 직접 슬라이싱 데이터를 추출할 수 있고 필요하면 솔리드로의 자동변환도 가능하게 되어 폐속조형제품개발기간의 단축에도 크게 기여할 수 있을 것으로 기대 되어진다.

5. 결론

본 연구에서는 불완전한 초기 STL 파일이나 스캔된 점 군 데이터를 완전한 하나의 음함수 곡면으로 재생성한 후 폐속조형을 위한 슬라이싱 데이터를 추출하는 알고리즘을 제안하였다.

영역마다 서로 다른 형상함수를 선택적으로 사용하는 적응적영역분할법에 의하여 점 군을 음 함수 곡면으로 보간함으로써 대형의 점 군 데이터를 취급하는 것이 가능하게 되었고 보간 및 슬라이싱 데이터의 추출에 소요되는 시간을 크게 줄일 수 있었다.

마칭 텍셀 알고리즘에 의하여 개략적인 교차선을 구한 다음 이 데이터를 기초로 정확한 교차선을 다시 구하는 단계적인 방법을 제안함으로써 슬라이싱 알고리즘의 강건성을 향상시켰다.

제안된 방법은 RP 분야의 개발자 및 사용자 모두를 지원할 수 있도록 슬라이싱 데이터의 추출 알고리즘뿐만 아니라 홀매움 및 오프셋과정을 통하여 닫힌 솔리드나 셸 형상의 솔리드 등 다양한



Fig. 5(a) Initial scanned Agrippa model

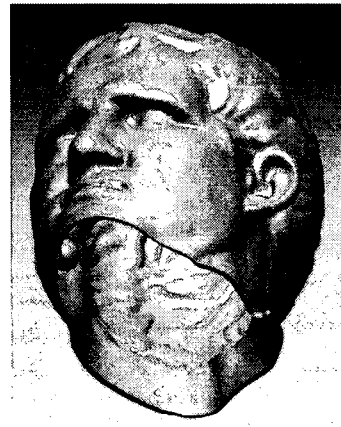


Fig. 6(b) Thickened solid model of Agrippa

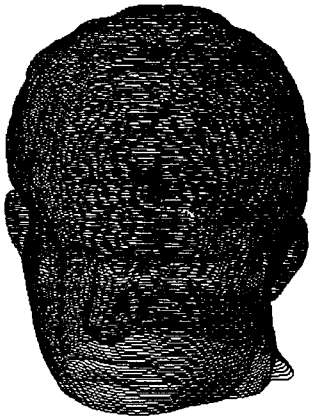


Fig. 5(b) Calculated slicing data of Agrippa model

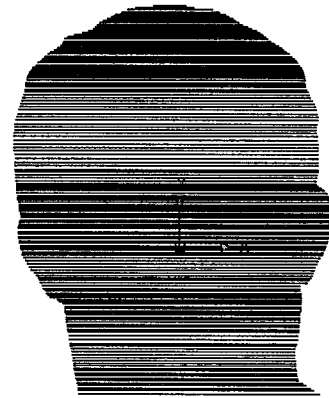
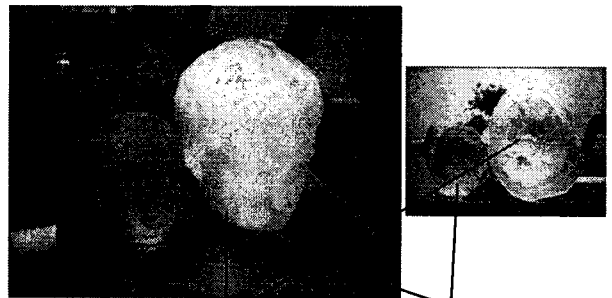


Fig. 7 Calculated slicing data of Agrippa model considering the inclination of surface



Fig. 6(a) Closed solid model of Agrippa



solid model of shell type

closed solid

Fig. 8 Fabricated Agrippa model from RP machine



Fig. 9(a) Offsets of Agrippa model(original model)

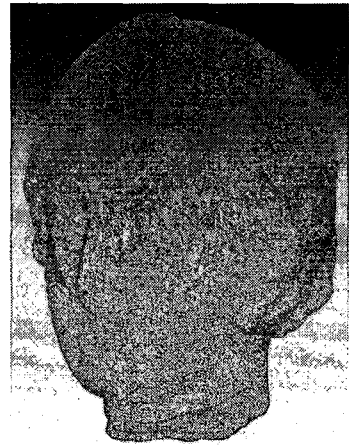


Fig. 10(a) Point cloud data of Igea model



Fig. 9(b) Offsets of Agrippa model(offset : -3mm)



Fig. 10(b) Slicing data of Igea model in x-direction

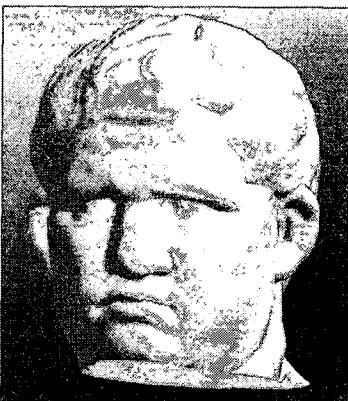


Fig. 9(c) Offsets of Agrippa model(offset : 3mm)



Fig. 10(c) Slicing data of Igea model in z-direction



Fig. 10(d) Solid STL data of Igea model



Fig. 11(b) Slicing data of human model

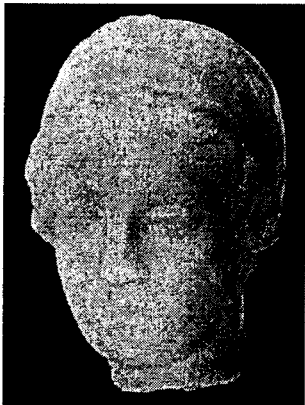


Fig. 10(e) Fabricated Igea model from RP machine



Fig. 11(c) Solid STL data of human model



Fig. 11(a) Point cloud data of human model

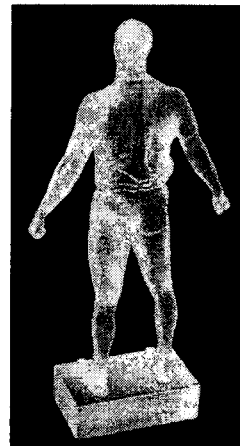


Fig. 11(d) Fabricated human model from RP machine

형태의 솔리드로의 자동 변환 가능성도 제시하였는데 큰 의의가 있다고 하겠다.

후 기

이 논문은 2007 학년도 대전대학교 학술 연구비 지원에 의한 것임.

참고문헌

1. Byun, H. S., Kang, E. C. and Lee, K. H., "Determination of the Orientation of the Scanned Point Data for Direct Rapid Prototyping," *Journal of the Korean Society for Precision Engineering*, Vol. 22, No. 10, pp. 31-38, 2005.
2. Liu, G. H., Wong, Y. S., Zhang, Y. F. and Loh, H. T., "Error-based segmentation of cloud data for direct rapid prototyping," *Computer Aided Design*, Vol. 35, No. 7, pp. 633-645, 2002.
3. Liu, G. H., Wong, Y. S., Zhang, Y. F. and Loh, H. T., "Modelling cloud data for prototype manufacturing," *Journal of Materials Processing Technology*, Vol. 138, No. 1-3, pp. 53-57, 2003.
4. Wu, Y. F., Wong, Y. S., Loh, H. T. and Zhang, Y. F., "Modelling cloud data using adaptive slicing approach," *Computer Aided Design*, Vol. 36, No. 3, pp. 231-240, 2004.
5. Turk, G. and O'Brien, J. F., "Modelling with implicit surfaces that interpolate," *ACM Transactions on Graphics*, Vol. 21, No. 4, pp. 855-873, 2002.
6. Carr, J. C., Beatson, R. K., Cherrie, J. B., Mitchell, T. J., Fright, W. R., McCallum, B. C. and Evans, T. R., "Reconstruction and representation of 3D objects with radial basis functions," In *Proceedings of SIGGRAPH 2001*, pp. 67-76, 2001.
7. Kojekine, N., Hagiwara, I. and Savchenko, V., "Software tools using CSRBFs for processing scattered data," *Computer & Graphics*, Vol. 27, No. 2, pp. 311-319, 2003.
8. Yoo, D. J., "A Study on Filling Holes of the Polygon Model using Implicit Surface Scheme," *Journal of the Korean Society for Precision Engineering*, Vol. 22, No. 3, pp. 107-114, 2005.
9. Ohtake, Y., Belyaev, A., Alexa, M., Turk, G. and Seidel, H. P., "Multi-level partition of unity implicits," *ACM Transactions on Graphics(TOG)*, Vol. 22, No. 3, pp. 463-470, 2003.
10. Yoo, D. J., "A Study on Filling Holes of Large Polygon Model using Implicit Surface Scheme and Domain Decomposition Method," *Journal of the Korean Society for Precision Engineering*, Vol. 23, No. 1, pp. 174-184, 2006.
11. Yoo, D. J., "Shape Reconstruction from Unorganized Cloud of Point using Adaptive Domain Decomposition Method," *Journal of the Korean Society for Precision Engineering*, Vol. 23, No. 8, pp. 89-99, 2006.
12. Yoo, D. J., "Shape Reconstruction from Large Amount of Point Data using Repetitive Domain Decomposition Method," *Journal of the Korean Society for Precision Engineering*, Vol. 23, No. 11, pp. 93-102, 2006.
13. Faux, I. D. and Pratt, M. I., "Computational geometry for design and manufacture," Ellis Horwood Ltd., pp. 257-277, 1979.
14. Yang, D. Y., Park, S. H. and Lim, T. W., "Fabrication of Three-dimensional Microstructure using Nano-Stereolithography process," *Journal of the Korean Society for Precision Engineering*, Vol. 2, No. 10, pp. 17-25, 2005.
15. <http://www.cyberware.com/samples>