

# NURBS를 이용한 Skin Animation의 속도 향상에 관한 연구

강종호\*, 이우석, 김경식  
호서대학교 게임공학전공

## A Study on the Skin Animation for Improving its Rendering Speed Using NURBS

Jong-Ho Kang\*, Woo-Seock Yi, Kyung-Sik Kim  
Dept. of Game Engineering, Graduate school of Hoseo University

### 요 약

과거 로(low)-폴리곤 3D 메시는 모델러에 의해 각 정점들의 제어가 가능했지만, 앞으로의 대세가 전망되는 하이-폴리곤의 경우 모든 정점들을 일일이 제어하는 것은 불가능하다. 따라서 하이-폴리곤 3D 메시는 넵스로 구성될 것이다. 본 연구는 이러한 하이-폴리곤 3D 메시가 스킨 애니메이션될 때 하이-폴리곤 3D 메시의 넵스 구조와 넵스의 계산 속도를 개선하여 애니메이션의 속도를 향상시키는 것을 연구하였다. 본 연구의 결과는 3D 애플리케이션에서 애니메이션을 사용할 때, 하이-폴리곤일수록 애니메이션 연산이 증가하는 것을 억제하고 하이-폴리곤을 사용하는 애니메이션의 부담없는 사용과 넵스의 LOD 적용을 실시간에 가능하게 한다.

### 1. 서론

#### 1.1 연구 배경과 목적

초기의 3D 게임들의 3D 메시들은 낮은 하드웨어 성능에 따른, 1000개 이하의 폴리곤으로 구성된 로-폴리곤(low polygon) 3D메시 객체들을 사용하였다. 현재는 약 1800~2200정도의 폴리곤들을 사용한다. 보통 2000개 정도의 폴리곤이면 대부분의 원하는 객체들을 표현할 수 있지만, 머리카락이나 옷 또는 피부의 주름 등의 세밀한 표현들이 불가능하기 때문에 아직도 좀 더 많은 폴리곤을 가지는 3D 메시가 요구되고 있다. 지금의 하드웨어의 발전 속도를 고려할 때, 앞으로의 3D 메시는 폴리곤의 수가 약 10000개를 넘는 하이-폴리곤(high polygon) 3D 메시가 전망된다. 3D 그래픽 모델러에 의해 만들어지는 로-폴리곤의 경우, 사람의 손으로 3D 메시들의 각 정점들을 제어하는 것이 가능하지만 10000개가 넘는 하이 폴리곤의 경우, 사람의 손으로 일일이 제어하는 것이 불가능하기 때문에, 모든 3D 그래픽 모델러들은 넵스(NURBS : Non-Uniform Rational B-Spline)를 이용하여 3D 메시들을 개발한다[5,6]. 즉, 모든 하이-폴리곤 메시들은 넵스로 구성되어 있다.

모델러에 의해 만들어진 3D 메시들은 게임 애플리케이션에서 사용되어진다. 게임 애플리케이션은 항상 낮은 하드웨어 사양에서 최대의 성능을 발휘하도록 계획되어져 왔고, 언제나 게임 프로그램들의 궁극적인 대상은 하드웨어들의 최대 성능을 이끌어 내는 것 즉, 계산 처리 속도였다.

본 논문은 3D 그래픽의 Animation과 Light&Material부분을 넵스를 이용하여 속도를 향상시키는 목적으로 연구한 결과를 기술하였다.

#### 1.2 본 연구의 특징

다른 논문에서 NURBS가 모델의 외형을 이루는 기하적 측면에 적용한 것에 비하여 본 논문은 NURBS를 계산의 절감을 위한 도구로 사용한 것에 차별성의 의의를 둔다. 본 연구의 동기는 대세가 예측되는 하이-폴리곤 3D 메시가 넵스를 기반으로 만들어지는 것에 착안하여, 넵스의 계산량을 줄이는 것이 넵스를 기반으로 하는 메시의 계산량을 줄일 수 있다는 것이었다. 이 연구의 결과는 하이-폴리곤에서 같은 폴리곤을 갖는 3D 메시보다 더 높은 효율을 나타냈고, 이러한 결과는 하이-폴리곤 캐릭터 애니메이션을 사용하는 높

은 퀄리티의 3D 애플리케이션에서 속도를 절감시키는 기술로 활용을 가능케 할 것이다.

## 2. 관련연구

### 2.1 캐릭터 애니메이션

캐릭터 애니메이션은 인물이나 동물, 또는 의인화된 무생물 개체를 표현하기 위한 것이다[1]. 3차원 캐릭터 애니메이션의 제작에 있어서 캐릭터의 생명감을 표현하기 위해 물리적 법칙을 비롯한 일련의 규칙에 의해 자동 혹은 반자동적으로 동작을 생성하는 기법을 적용하는데 이를 통칭하여 동작 제어 기술이라 한다. 동작 제어 기법은 크게 키프레임(Key-frames) 기법, 운동학적(Kinematics) 기법, 역학(Dynamics) 그리고 동작 편집(Motion Editing) 방법등으로 분류할 수 있다[2]. 키프레임 기법은 시간 축에 따라 캐릭터의 각 관절의 굴절 각도와 캐릭터 자체의 공간적 위치 등을 애니메이터가 수작업을 통해 동작을 제어하는 방법으로 키프레임 간의 연결 동작을 위해 보간 동작의 생성이 필요하다[3]. 운동학적 기법(Kinematics)은 키프레임을 이용하고 키프레임 간의 자연스러운 연결 동작을 위해 적절한 보간 기법을 적용하며, 각 관절의 집합 위치(joint position)와 각도로부터 관절 끝(end effector)의 위치를 운동학적 방법으로 산출하여 동작을 생성하는 방법이다[2]. 운동학적 기법으로는 계층구조의 상위에 있는 관절을 조작하여 동작을 제어하는 전운동학(Forward Kinematics) 방법과 계층구조의 하위에 있는 관절을 조작하여 동작을 제어하는 역운동학(Inverse Kinematics) 방법 등이 있다[3]. 게임에서는 운동학적 기법을 뼈대 애니메이션(Skeletal Animation)이라 한다. 뼈대 애니메이션은 척추동물과 같이 뼈대를 가지는 생물들의 신체 구성원리를 흉내낸 것으로서 스티칭(stitching)과 스킨닝(skinning) 방식이 있다[4]. 스티칭은 하나의 다각형의 정점들이 서로 다른 뼈대들에 부착되어 두 뼈대의 움직임으로 인해 생기는 메시의 틈을 하나의 다각형으로 덮는 효과를 만들어 내는 것이다. 반면, 스킨닝은 하나의 정점이 여러 개의 뼈대에 영향을 받는 방법이다. 스킨닝의 장점은 완전한 연속성을 보장해 준다는 것이다.

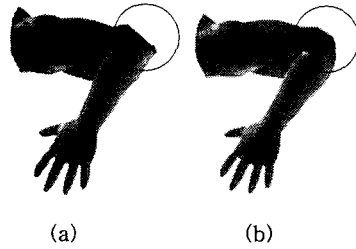


그림 1 (a) 스티칭 기법 (b) 스킨닝 기법

### 2.2 NURBS[5]

NURBS(Non-Uniform Rational B-Spline)는 충분한 연속성 보장, 국부적인 변형 가능성, 곡면식의 일반성, 곡면제어의 편이성 등 여러 가지 장점을 가지고 설계를 보다 용이하게 하므로 곡면설계에 자주 이용된다. NURBS 곡면의 생성에서 제어점들이 결정되면 NURBS 곡면이 결정된다[6]. 본 논문에서는 NURBS의 일종인 베지어 패치만을 사용한다. 베지어 패치는 베지어 곡선으로 구성되며, Tensor-product Bezier Surface라고도 불리운다[7]. 그림 2는 베지어 패치의 형태이다.

$$Q(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{ij} B_{i,j}(u, v)$$

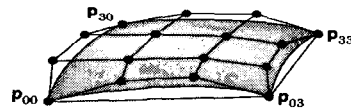


그림 2 베지어 패치와 제어점

### 2.3 MetaNURBS의 구성

MetaNURBS는 3D 도구인 LightWave3D에서 사용되는 베지어 패치로 구성된 객체를 말한다. 그림 3은 베지어 패치로 구성된 MetaNURBS이다.

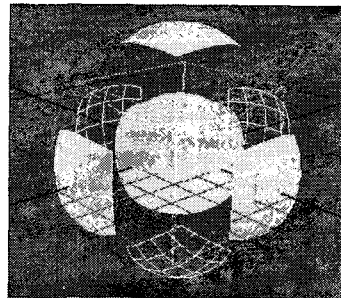


그림 3 베지어 표면으로 구성된 MetaNURBS

### 3. 연구 방법과 내용

스킨 애니메이션에서, 뼈대의 가중치를 받아 변형되는 정점들의 계산에는 적어도 두 번의 3x3 회전행렬을 연산하게 된다. 위치에 한번 그리고 노멀 벡터에 한번이다. 즉, 한 정점 당 최소한 9번의 곱셈과 6번의 덧셈이 필요하다. 만일 बे지어 패치에 의해서 생성되는 정점 연산이 스킨 애니메이션에서 필요한 연산보다 더 적다면, बे지어 패치를 사용한 스킨 애니메이션이 더 빨라질 수 있는 것이다. 그림 4는 각 레벨에 따른 बे지어 패치의 표면 생성 점의 개수이다.

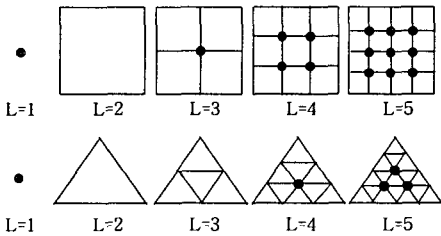


그림 4 बे지어 패치의 레벨에 따른 표면 분할

본 연구에서는 Casteljau 알고리즘을 이용하여 정점의 계산 수를 줄였다. 그림 5는 레벨 3에서의 MetaNURBS 객체이다. 이 객체는 각 बे지어 패치에서 하나의 표면점만을 가진다.

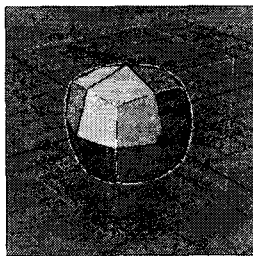


그림 5 레벨 3 분할의 MetaNURBS에서의 하나의 बे지어 표면 점

레벨 3에서의 बे지어 패치에서 하나의 표면점은 Casteljau 방법을 이용하여 다음과 같은 방법으로 얻을 수 있다. 우선 Casteljau 알고리즘을 이용하여 बे지어 곡선의 중간점을 구한다.

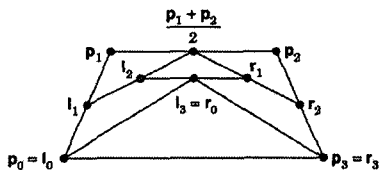


그림 6 Casteljau 알고리즘을 이용한 중간 점 구하기

$$\begin{aligned}
 i_0 &= p_0 & r_3 &= p_3 \\
 i_1 &= \frac{1}{2}(p_0 + p_1) & r_2 &= \frac{1}{2}(p_2 + p_3) \\
 i_2 &= \frac{1}{2}(i_1 + \frac{1}{2}(p_1 + p_2)) & r_1 &= \frac{1}{2}(r_2 + \frac{1}{2}(p_1 + p_2)) \\
 i_3 &= r_0 = \frac{1}{2}(i_2 + r_1)
 \end{aligned}$$

레벨 3일 때의 बे지어 곡선의 중간 정점은  $i_3 = \frac{1}{8}(P_0 + 3P_1 + 3P_2 + P_3)$  이다. 다음은 레벨 3의 बे지어 패치에서의 표면 정점을 구한다.

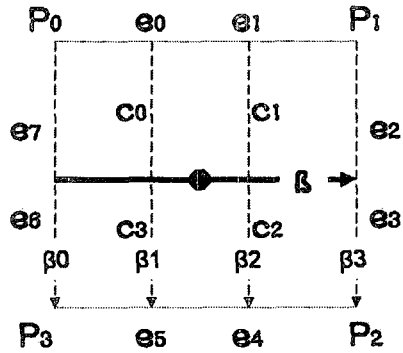


그림 7 레벨 3의 Casteljau 알고리즘을 이용한 बे지어 곡선들로 레벨 3의 बे지어 표면 점을 생성

$$\begin{aligned}
 \beta_0 &= P_0 \rightarrow e_7 \rightarrow e_8 \rightarrow P_3 = \frac{1}{8}(P_0 + 3e_7 + 3e_8 + P_3) = \lambda_0 \\
 \beta_1 &= e_0 \rightarrow c_0 \rightarrow c_3 \rightarrow e_5 = \frac{1}{8}(e_0 + 3c_0 + 3c_3 + e_5) = \lambda_1 \\
 \beta_2 &= e_1 \rightarrow c_1 \rightarrow c_2 \rightarrow e_4 = \frac{1}{8}(e_1 + 3c_1 + 3c_2 + e_4) = \lambda_2 \\
 \beta_3 &= P_1 \rightarrow e_2 \rightarrow e_3 \rightarrow P_2 = \frac{1}{8}(P_1 + 3e_2 + 3e_3 + P_2) = \lambda_3
 \end{aligned}$$

$$\begin{aligned}
 \beta &= \lambda_1 \rightarrow \lambda_2 \rightarrow \lambda_3 \rightarrow \lambda_2 = \frac{1}{8}(\lambda_1 + 3\lambda_2 + 3\lambda_2 + \lambda_3) \\
 &= \frac{1}{64}(\sum P_i + 3\sum E_j + 9\sum C_k)
 \end{aligned}$$

레벨 3에서는 하나의 표면 정점만이 생성되고 이 표면정점은

$$\frac{1}{64}(\sum P_i + 3\sum E_j + 9\sum C_k)$$

이와 같이 각 레벨에 따른 बे지어 패치의 정점들을 계산할 수 있다.

#### 4. 결론 및 향후 연구과제

본 연구의 목적은 앞으로의 대세가 예측되는 NURBS로 구성된 하이폴리곤 3D 메시를 저장하여 렌더링 속도를 향상시키는 것이다. 이 기술은 같은 폴리곤을 가진 메시에서 더 빠른 효율을 보인다. 또한 넘스로 구현되어 있기 때문에 기본적으로 넘스의 분할 레벨을 이용하여 LOD가 가능하다. 그러나 3D 메시가 반드시 넘스로 구현되어야하고 LOD의 단계가 단계 레벨의 제곱(2<sup>2</sup>, 3<sup>2</sup>, 4<sup>2</sup>...)이기 때문에 사용의 폭이 좁다. 더구나 프로그램 구현 수준이 중급 이상이 요구될 정도로 구현이 어렵다. 그럼에도 불구하고 이 기술은 스킨 애니메이션의 속도를 개선하는 한 방법에 큰 의미를 가진다. 스킨 애니메이션은 한 정점당 두 번의 3x3 행렬 연산이 필요한 선형증가이기 때문이다. 3D 메시가 하이폴리곤일 수록 증가하는 연산을 피할 수 없다. 본 연구 결과의 특징은 3D 메시가 하이폴리곤일 수록 연산의 증가율은 감소한다. 베지어 표면을 구성하는 베지어 곡선의 선형미분을 위한 처음의 기본 연산 이후, 정점의 생성이 미분으로 얻어진 증가량들의 3번의 덧셈으로 이루어지기 때문이다. 앞으로 이 기술은 베지어 표면만이 아닌 다양한 NURBS 표면들을 포괄적으로 수용하고 프로그램 부분의 좀더 많은 최적화를 통하여 렌더링 속도를 개선해야한다. 그림 8과 그림 9는 본 연구에 대한 결과이다.

#### [참고문헌]

- [1] 송경민, 최소 모션 정보를 이용한 3차원 캐릭터 애니메이션, 전남대학교 석사학위논문, pp.19-21, 2002.02.
- [2] 송미영, 조형제, "다양한 지형에서의 적응적인 걷기 동작 생성", 정보과학회논문지:소프트웨어 및 응용, 제 30권, 제 11호, 동국대학교, pp.1093-1094, 2003. 12.
- [3] "CG 애니메이션," 정보과학회지, 제 17권, 제 2호, 통권 제117호, 1999.02.
- [4] Mark Deloura의 공저/류광 역, "Game Programming Gems", 정보문화사, 2001.
- [5] Piegl, L. and Tiller, W., The NURBS Book, Springer, 1995.
- [6] 정형배, 나승수, 박종환, "NURBS Surface Global Interpolation에 대한 한 방법", 한국 CAD/CAM 학회 논문집 제 2권 제 4호, pp.237-238, 1997.12.
- [7] Tomas Akenine-moller, Eric Haines, Real-Time Rendering, 2th Ed., A K Peters, 2002.

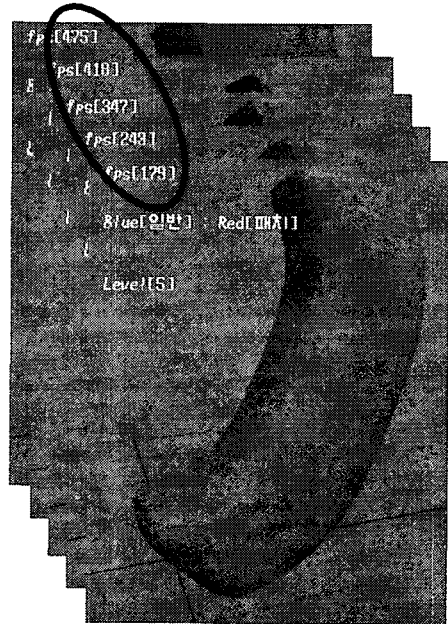


그림 8 일반 3D 메시를 이용한 스킨 애니메이션 결과, 베지어 패치의 표면 분할 레벨에 따른 동등한 폴리곤 수를 가진다.

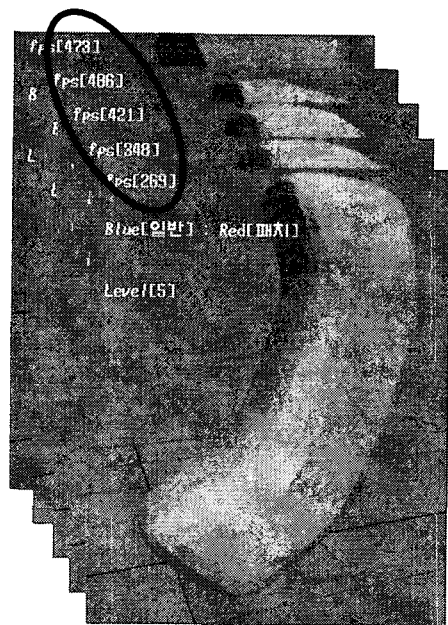


그림 9 베지어 패치의 각 표면 분할 레벨에 따른 스킨 애니메이션의 결과, 일반 3D 메시를 이용할 때보다 프레임이 높은 것을 알 수 있다.