

LOD를 적용한 지형 자동 생성 시스템 구현

○
신정은, 조창석
한신대학교 정보통신학과

An Implementation of terrain manufacturing system using LOD algorithm.

○
Jeong-Eun Shin, Chang-Suk Cho
Dept. of Information Science and Telecommunications, Hanshin University

요 약

본 연구에서는 전투 시뮬레이션 게임 구현을 목적으로 하여 LOD를 적용한 실시간 지형 렌더링 시스템을 제작하였다. 이를 위하여 지형의 렌더링을 위한 폴리곤 수를 절감시키는 방식으로 메모리를 관리 할 수 있도록 하였고, 현재 위치와 시점의 거리에 따라 가까운 거리는 상세하게 표현하고 먼 거리는 지형의 외형을 최대한 유지하면서 폴리곤 수를 줄여 렌더링 하도록 하였다.

1. 서론

전투 시뮬레이션 게임에서는 3차원상의 지형 위를 마치 지상을 돌아다니듯 전부를 보여주고 한번에 많은 공간을 한 화면에 보여주어야 하므로 엄청난 양의 컴퓨터 메모리를 요구하게 된다. 이러한 이유로 3차원상의 움직임을 보다 부드럽고 무리를 주지 않게 하기 위해서 연구되어 온 것이 LOD(Level Of Detail) 방법이다[1-4].

3차원상의 지형을 실시간에 렌더링하기 위하여 많은 연구들이 수행되어 왔고, LOD 지형 알고리즘 중 Hoppe[1]는 임의의 메쉬에 삼각형들을 추가하는 적응적 메쉬에 기반한 알고리즘을 제시하였고, Lindstrom[2]은 지형의 조각(Patch)을 표현하기 위해 사용되는 쿼드트리라고 하는 구조체를 이용해 지형을 재귀적으로 분할하는 방식을 제안하였다.

Duchaineau[3]는 이진삼각형 트리에 기반한 알고리즘(Real-time Optimally Adapting Meshes: ROAM)을 제시하였고, 이는 각 패치를 이동면 삼각형으로 취급하고 이의 정점에서 빗변의 중점을 분할하여 두 개의 새로운 이동면(우측) 삼각형을 만들어 재귀적인 분할을 하도록 하는데 그 핵심이 있다.

본 논문에서 구현된 방식은 Duchaineau의 방식에서 삼각형 분할 방식을 채택하였으며, 지형의 고도와 좌

표 생성은 비트맵을 이용하여 지정하였다. 이를 위하여 비트맵의 각 색상 값을 높이필드의 각 높이로 사용하였고 그 높이 값에 따라 격자의 고저를 표현함으로써 지형을 생성하고, 사용자가 원하는 대로 지형 데이터를 뽑을 수 있도록 하였다.

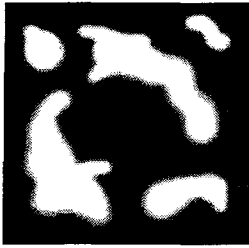
지형을 렌더링 할 때에는 좌표값과 그 좌표값에 해당하는 높이값을 갖는 각 점들을 삼각형으로 연결하여 표현하게 되는데 이렇게 표현되는 지형에 LOD를 적용하여 최적화된 지형을 렌더링하게 되는 것이다.

LOD 방식에 있어서 지형의 상세도 레벨을 선택하는 것은 사용자의 시점을 기준으로 한다는 공통된 개념을 가지고 있다. 그러므로 관찰자에게 가까운 곳 즉 잘 보이는 곳은 상세히 표현해 주고 먼 곳이나 보이지 않는 곳은 덜 상세히 표현함으로써 시스템의 성능에 무리를 주지 않도록 하는 기능인 것이다.

2. 3차원 지형 자동생성

2.1 Height Field 지형생성

본 연구에서는 비트맵의 각 색상 값을 높이필드의 각 높이로 사용하였고 그 높이 값에 따라 격자망의 고저를 표현함으로써 지형을 간단히 생성할 수 있는 기법을 사용하였다.



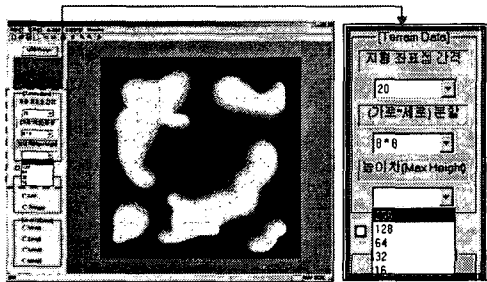
[그림 1] 그레이스케일 비트맵

비트맵 파일로부터 0에서 255사이의 명암값들을 추출하여 높이값으로 사용하였으며, 밝을수록 높이가 높은 것이고 어두울수록 낮은 것이다. [그림1]은 그레이 레벨로 지형이 표시된 비트맵 파일을 나타낸다.

2.2 지형 자동생성 시스템

Height Field 기법으로 지형을 생성하도록 하였고, 이에 의해 만들어진 비트맵 파일에 따라 지형을 자동으로 생성하게 하였다. 비트맵 파일은 33의 배수인 198, 264, 330bit등의 그레이레벨 파일을 사용하며 이는 삼각형 분할이 잘 이루어지는 최소 단위가 33이기 때문이다.

본 시스템에서는 사용자가 지형의 데이터를 어떻게 설정하느냐에 따라서 지형이 생성된다. 지형의 데이터 설정에는 지형의 좌표 점 간격, 지형(가로*세로)분할 파일 생성, 지형의 최대 높이 값 설정이 있다. [그림 2]는 지형 자동 생성 시스템의 설정 화면을 보여주고 있다.



[그림 2] 지형 자동생성 시스템의 설정 화면

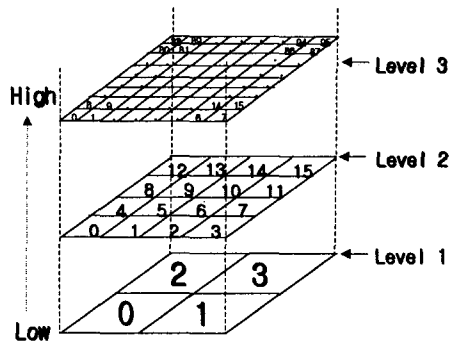
3. LOD 방법에 의한 렌더링

수천개의 삼각형들로 이루어진 복잡한 세부모델의 렌더링 결과는 모델이 시점 부근에 있는 경우에는 시각적으로 미려한 결과를 나타낸다. 다각형으로 미세하게 분할될수록 렌더링 하는데 드는 시간이 커지며 상대적으로 훌륭한 시각적 품질을 얻을 수 있다. 그러나

동일한 모델이 시점에서 멀리 있는 경우 렌더링 된 모델의 화면 공간 적용범위는 몇 개의 픽셀뿐이므로 수천개의 삼각형에 의해 제공된 세부 묘사가 그다지 두드러지지 않는다.

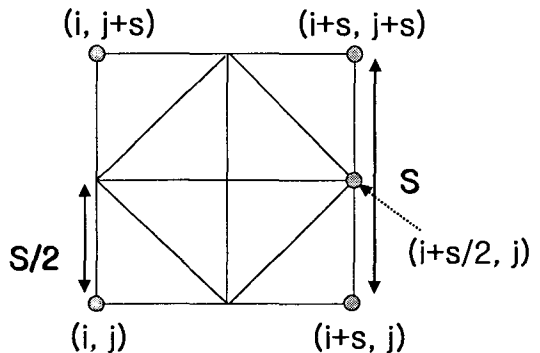
이러한 경우, 시간과 시각적 품질의 절충은 의미가 없다. 최종 렌더링이 몇 개의 픽셀만을 다룬다면 처리되는 삼각형의 수도 이에 비례해야 하며 이러한 원리에 근거한 것이 LOD 방법이다.

3.1 지형 토폴로지



[그림 3] 레벨에 따른 지형 토폴로지

[그림 3]에서와 같이 지형이 $N \times N$ 으로 생성되면 노드가 순서대로 구성되고, 코너 점에 대한 값을 저장하여 레벨의 설정에 따라서 재귀적인 함수를 호출하면, 삼각형의 패치가 생성되어 지형을 렌더링 한다.



[그림 4] 블록의 코너점 계산

[그림 3]에 있는 레벨2의 5블록을 현재 시점이 머물고 있는 곳으로 한다면 이 블록 5번 지형은 [그림 4]처럼 상세히 분할되어 렌더링 된다.

[그림 4]에 있는 지형의 S는 [그림 3]에 있는 5번

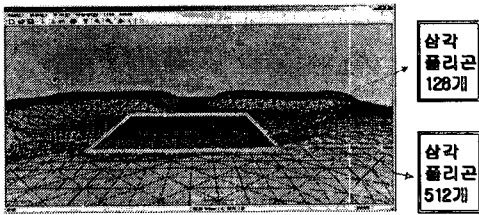
정방형 블록의 가로, 세로 길이를 나타내고, 이 블록의 정점(vertex) 수가 $N \times N$ 이라면 S는 N값을 가지게 되며 이로부터 각 노드의 코너(corner) 점을 구할 수 있다. 4개의 코너 좌표점 (i, j), (i+s, j), (i, j+s), (i+s, j+s)는 5번 블록의 나타내는 영역의 최소, 최대좌표값이 된다.

해당 블록이 시점을 기준으로 한 일정 범위 내에 들게 되면 LOD가 적용(활성화)되며, LOD가 활성화 되면 블록 길이의 S/2 지점에 정점이 생기면서 삼각형이 재귀적으로 분할되어 상세화 되는 것이다.

3.2 레벨 & 노드

하나의 지형 파일은 하나의 노드로 구성하는데 $N \times N$ 개의 정점을 가진 지형에서는 $N \times N$ 개의 노드가 생성된다.

현재 시점이 위치한 곳의 노드를 상세화 하며, 노드의 기본 레벨에서 다음 레벨로 상세화 하는데 이것은 재귀적인 함수를 호출함으로써 삼각형을 상세화 한다.



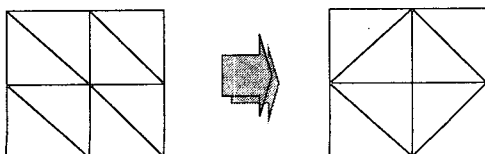
[그림 5] 기본 레벨보다 상세화 된 지형

위의 [그림 5]는 현재 시점이 기본 레벨보다 한 단계로 상세화 된 지형으로 기본 레벨의 삼각형 폴리곤의 개수보다 3배의 삼각형 폴리곤의 개수로 지형을 렌더링 하는 것을 볼 수 있다.

그러므로 시점으로부터 멀리 있는 곳은 현재 시점이 위치하고 있는 곳보다 삼각 폴리곤 수가 1/3 정도로 감소되어 렌더링 속도가 빨라진다.

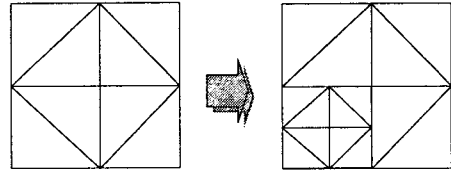
3.3 격자 구성

격자 구성은 기존의 규칙 삼각망 구성과 달리 LOD 제어를 위한 격자 구성을 한다.



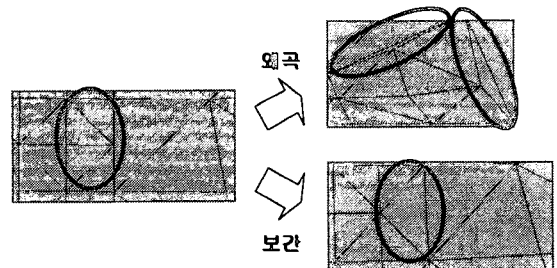
[그림 6] 격자 구성

블록에서 상세화 될 때 루트 블록에서 자식블럭으로 분화되어 가며, 루트블럭에서 분기된 자식블럭의 꼭지점은 [그림 8]에서처럼 다양한 높이로 나타난다.



[그림 7] 블록 간소화 후 최소 삼각형

그러나 상세화를 하였을 때 각 꼭지점에 대해 다양한 꼭지점의 높이 때문에 지형의 왜곡 현상이 일어난다. 이는 [그림 8]에서 관찰할 수 있다. 이러한 지형의 왜곡을 보간하기 위해서는 상세화 된 부분과 그렇지 않은 부분 사이의 경계부분을 보간 처리 한다. 보간 처리는 왜곡된 부분의 폴리곤 형성 방식을 바꿔줌으로써 왜곡된 부분을 자연스럽게 처리하는 것이다.



[그림 8] 왜곡된 삼각형 보간

LOD를 적용함에 있어 유의할 점은, 실시간 렌더링을 목표로 하되 삼각형 개수를 무조건 줄일 것이 아니라, 상세화 과정에서 발생하는 왜곡과 같은 오차를 최소화 할 수 있어야 한다. 상기의 보간 처리 방식을 이용하여 왜곡 현상을 최소화 할 수 있었고, 이에 따라 하위 레벨에서 상세화 된 상위 레벨까지 자연스럽게 처리되는 것을 볼 수 있었다.

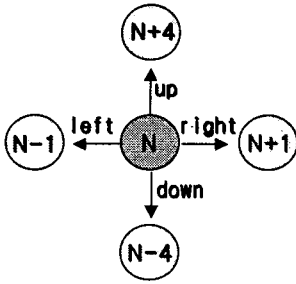
3.4 지형 렌더링

레벨에 따라서 격자를 구성하고 상세화 한 후에 왜곡된 삼각형을 처리하고 렌더링을 한다. 각 활성 블록의 삼각형들의 렌더링은 현재의 시점 위치의 이동과 연계되며 시점이 이동한다면 노드 역시 변환 이동을 해야 한다.

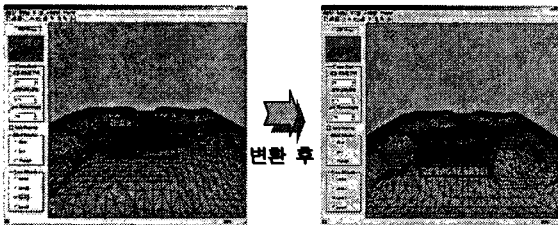
다음 [그림 9]는 시점 이동에 따른 노드 변환을 보

여주는 것으로 노드를 계산하여 렌더링하게 한다.

현실감 있게 보이기 위해 맵핑을 적용하였다.



[그림 9] 시점 이동에 따른 노드 변환



[그림 10] 시점 이동에 따른 렌더링

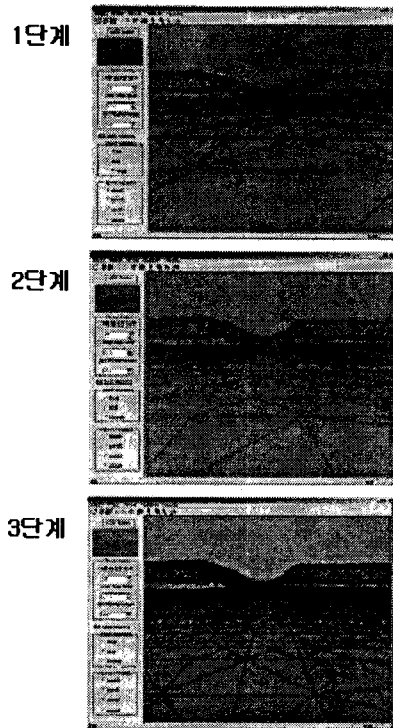
3.5 레벨에 따른 격자 상세화

본 시스템에서는 상세도 레벨에 따라 지형이 렌더링 되도록 구현되었고, 레벨은 1에서 4정도로 설정할 수 있도록 되어있다. 레벨의 설정은 시스템에서의 자동 설정 및 사용자가 설정이 가능하도록 되어 있으며 상위 레벨일수록 상세화 된 격자가 구성된다.

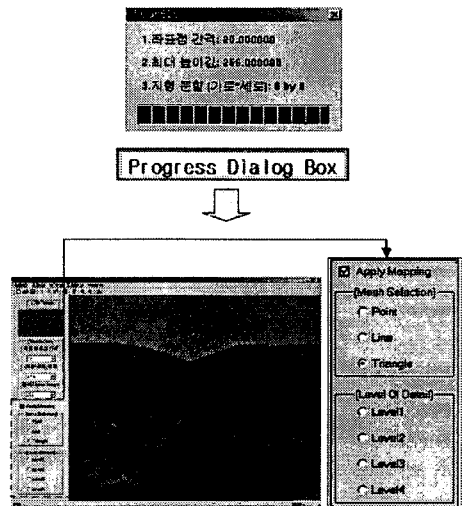
4. LOD를 적용한 지형 자동생성 프로그램

Height Field 지형생성 방식으로 지형도를 제작한 후 제작된 지형에 LOD를 적용하는 방식에 의해 전투 시뮬레이션 게임용 지형 자동생성 시스템을 제작하였다. 지형이 생성이 되고 레벨에 따라 상세도 적용이 되면 사용자는 여러가지 방식으로 렌더링을 제어할 수 있게 된다. 그러므로 렌더링 할 때 레벨에 따라 렌더링 할 수 있으며, 그래픽적으로 좀 더 현실감 있게 맵핑을 적용 할 것인지에 대한 적용 여부를 체크하는 기능을 추가 하였다.

위의 [그림 10]은 Height Field기법으로 좌표점 간격 20, 최대높이 256, 지형 분할(가로*세로) 8*8로 설정하여 생성된 지형이다. 그리고 렌더링 할 때에는 지형 폴리곤(polygon) 형태를 삼각형으로, 생성된 지형에 가장 기본 레벨인 레벨1을 적용하고 그래픽적으로



[그림 11] 레벨에 따른 격자 상세화



[그림 12] LOD를 적용한 지형 자동생성 시스템

5. 결론

Height Field 기법으로 지형을 생성하였고, 생성된 지형에 LOD를 적용하고, 지형이 사용자의 설정에 맞도록 자동적으로 생성될 수 있는 시스템을 구현하였다. 본 시스템은 탱크전 용 전투시물레이션 게임의 지형 생성용으로 제작된 것이며, 사실적인 렌더링을 위하여 맵핑을 적용시켰고, 3차원 지형을 시물레이션 할 수 있도록 하기 위해 3D 시점 이동을 적용하였다.

[참고문헌]

- [1]Peter Lindstrom, David Koller, William Ribarsky, Larry F. Hodges, Nick Faust and Gregory A. Turner. "Real-time, continuous level of detail rendering of height field". Proceedings of SIGGRAPH'96, pp.109-118, 1996.
- [2]Hugues Hoppe. "Progressive meshes". Proceedings of SIGGRAPH'96, pp.99-108, 1996.
- [3]M. A. Duchaineau, M. Wolinsky, D. E. Sigeti, M.C.Miller, C.Aldrich, M.Mineev, "ROAMing Terrain: Real-time Optimally Adapting Meshes," Visualization 97.
- [4]David H.Eberly, "3D Game Engine Design", (주)민프레스, 2001, pp383-424.