

온라인 3D 게임을 위한 대규모 실외 지형 구성

김혜선, 이동춘, 박찬용, 장병태
한국전자통신연구원
e-mail:hsukim@etri.re.kr

Construction of Large-scale outdoor terrain for online 3D game

Hye-sun Kim, Dong-chun Lee,
Chan-Yong Park, Byung-tae Jang
Electronics and Telecommunications Research Institute

요 약

3D 게임 산업이 발전하고 여러 사용자가 온라인 상에서 즐길 수 있도록 대형화되면서 그 규모 또한 대형화되고 있다. 본 논문은 대규모 실외용 지형을 생성하기에 적합하도록 계층화된 지형 구조를 소개하고 있으며, 기존의 지형 CLOD 알고리즘을 개선하여 넓고 구조화된 지형의 렌더링 속도 문제를 해결하였다. 이를 통하여 대규모 온라인 게임의 지형 구성 시 발생하는 지형 최적화 문제를 해결하고 실시간으로 게임을 진행할 수 있다.

1. 서론

최근 들어, 일반 개인의 시스템 처리 능력이 좋아지고 빠른 속도의 3D 그래픽 카드가 보급됨에 따라 3D 환경은 더 이상 고가의 그래픽 장비를 가지고 있는 특수 사용자만의 전유물이 아니다. 각각의 일반 사용자들 또한 자신의 데스크탑 PC를 이용해 언제든 3D 환경을 접할 수 있게 되었다. 이에 따라 종래의 2D 중심이었던 게임 산업이 3D 중심의 게임 개발로 급격하게 변하고 있다. 이들은 3D 그래픽 기술의 발전으로 고품격의 실감나는 렌더링 화면을 실시간으로 생성해낼 수 있는 능력을 가지고 있으며 3D 그래픽 기술의 발전에 더욱더 박차를 가하고 있다.

특히 인터넷의 보급으로 대규모 게임 맵에서 여러 사용자가 동시에 접속해서 게임을 즐기는 롤플레이팅(Roll Playing) 형태의 게임이 발전하면서 게임의 크기는 대형화되었다. 다수의 사용자들 한꺼번에 수용하기 위해 게임 지형은 더욱더 넓어졌고 이를 위해 특별히 게임 지형을 관리하는 방법이 사용되었다. 본 논문의 게임 지형 구성 방법 또한 이에 맞추어 대형 게임 지형을 효율적으로 생성하고 관리하는데 그 초점을 맞추고 있다.

지금까지 개발되어온 3D 게임들은 크게 사방이 벽으로 둘러싸인 실내 환경에서 게임이 진행되는 실내형(indoor) 게임과 넓은 실외에서 게임을 하는 실외용(outdoor) 게임으로 나눌 수 있다. 실내와 실외용 게임은 렌더링 방법이 확연히 틀리기 때문에 각각 따로 개발되어 왔다. 온라인 롤플레이팅 게임은 많은 사용자가 동시에 즐기기를 위해 좁은 실내가 아닌 넓은 실외용으로 주로 개발되고 있고, 본 논문의 지형 구성도 실외용에 최적화 되어 있다.

본 논문의 구성은, 먼저 2장에서 실내용 게임과 실외용 게임의 지형 구성에 있어서의 차이점에 대해 간략히 설명을 한 후, 3장에서 실외 게임의 지형 구성에 대해 설명을 한다. 3.1장에서 대규모의 지형을 관리하기 위한 지형의 계층적 구조가 소개되고 이를 실시간으로 렌더링하는 방법이 3.2장에서 설명된다. 마지막으로 결론에서는 본 논문의 지형 구조를 사용함으로써 이룰 수 있는 장점들을 기술하고 직접 게임에 적용한 결과 화면들을 보여준다.

2. 실내 게임과 실외 게임 비교

실내용 게임은 게임 지형을 방 단위로 분할하여 그리는 BSP 방법을 주로 쓰는데, Quake, Doom 등의 게임들이 이에 속한다. 실내에서 이루어지는 게임의 경우 폴리곤으로 이루어진 게임 지형을 방 단위로 나누어 렌더링하기 때문에 실외에 비해 한번에 그려야할 양이 비교적 적은 장점이 있다. 이러한 이유로 그래픽 가속 능력이 부족했던 시절에 주로 개발되었던 게임의 형태이다. 그림 1은 BSP로 실내 게임 지형을 방 단위로 분할해 놓은 모습을 보여주고 있다. 사용자가 각각의 벽으로 나누어 놓은 방에 들어가서 게임을 즐길 때, 사용자의 시야가 벽으로 가려지기 때문에 보이는 방만을 렌더링하면 된다. 또, 게임 전에 미리 각 방에서 보이는 다른 방들을 계산해놓아서 렌더링 부하 없이 실시간으로 게임을 즐길 수 있다.

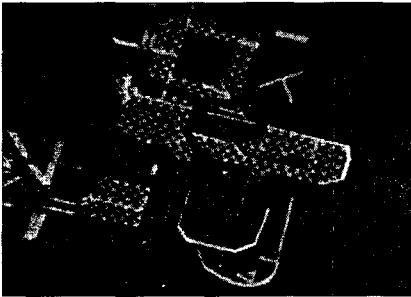


그림 1. BSP로 구성한 실내 게임 지형

그러나, BSP로 게임을 구성할 경우 지형이 벽으로 많이 나뉘고 사용자의 시야가 그 벽에 가려질수록 좋은 성능을 내는 특징 때문에, 벽이 전혀 존재하지 않는 실외용 게임 구성에는 부적합하다. 이로 인해 실외용 게임은 terrain CLOD(Continuous Level Of Detail) 방법을 쓰는데, 사용자의 시야가 넓은 이유로 실내용 게임보다 렌더링 시간이 많이 걸리는 것이 사실이다. 최근에는 그래픽 가속 능력이 발달하고, 넓은 지형을 요구하는 온라인 롤플레이 게임이 다수 개발됨에 따라 실외용 게임들이 많이 개발되고 있다. 실외용 게임 지형에 관한 자세한 설명은 3장에서 계속된다.

3. 실외 게임 지형 구성

실외 게임 지형은 넓은 3D terrain으로 구성된다. 이 또한 임의의 3D 메쉬로 표현하는 방법과 높이 정보값(height field)을 기반으로 하는 제한된 3D 메쉬로 표현하는 방법으로 나뉠 수 있다. 임의의 3D

메쉬로 표현할 경우 다양하고 자세한 지형을 표현할 수 있는 장점이 있는 대신 게임 진행 중에 지형의 높이 정보를 추출하기가 쉽지 않다는 단점이 있다. 반대로 높이 정보값(height field)을 기반으로 하는 3D 메쉬 표현법의 경우 게임 진행 중에 높이 정보는 추출하기 쉽지만 자세한 지형의 표현은 어렵다. 이처럼 두 가지 표현 방법에는 각기 장단점이 있지만, 빠른 속도로 진행되어야 하는 게임의 특성상 후자인 높이 정보값(height field)을 기반으로 하는 표현법을 사용한다.

3.1 계층적 지형 구조

대규모 지형 상에서 이루어지는 온라인 게임의 경우 한꺼번에 전체 지형을 처리하기가 힘들기 때문에 게임 지형을 셀(cell) 단위로 나누어서 관리한다. 그림 2의 (a)는 가상의 게임 지형을 4x4 셀로 나누어서 관리하는 예를 보여준다. 이 때 나누어진 셀들은 사용자가 위치한 셀을 중심으로 사방으로 9개의 셀들을 묶어서 AOI(Area Of Interest, 음영부분)를 이룬다. 온라인 게임의 경우 지형이 넓고 게임 사용자의 수가 많은 이유로 임의의 사용자의 게임 진행은 AOI 안에서만 처리한다.

그림 2의 (b)는 게임 지형의 계층적 구조를 보여준다. 하나의 게임 지형은 여러 개의 셀로 나뉘고, 그 셀은 텍스처(texture)가 입혀지는 타일(tile) 단위로 나뉘며, 가장 작은 높이 정보 값(height field)의 단위가 그리드(grid)가 된다. 특히, 3D 렌더링에서 속도에 중요한 영향을 미치는 요소 중에 하나가 텍스처 크기이므로, 타일의 크기를 적당하게 지정해야 최적의 게임 성능을 발휘할 수 있다. 타일의 크기가 너무 크다면 한번에 처리해야 하는 텍스처 크기가 너무 커져서 성능의 저하가 올 수 있으며, 반대로 타일의 크기가 너무 작아지면 이후 설명되는 CLOD가 제대로 적용되지 않아 문제가 생길 수 있다.

이 때 하나의 높이 정보 값(height field)만을 사용할 경우 단층의 지형만을 생성할 수 있으므로, 다리나 계단, 다층 건물 등을 표현하기 힘들다. 그래서 여러 개의 높이 정보 값(height field)을 겹쳐서 사용하는 다층 높이 정보 값(multi-layered height field)을 생성하는데, 그림 2의 (c)가 간단히 설명하고 있다. 그림 2의 (d)는 다층 높이 정보 값(multi-layered height field)을 사용해서 복층의 건물과 계단을 구현한 예를 보여주고 있다.

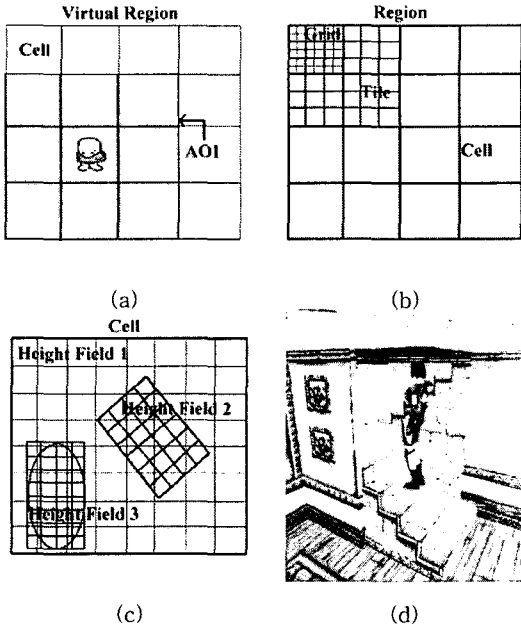


그림 2. 지형의 계층적 구조

3.2 지형 렌더링

일반적으로 높이 정보 값(height field)으로 구성된 게임 지형들은 렌더링 속도를 빠르게 하기 위해서 지형 CLOD 방법을 사용한다. 지형 CLOD 방법을 사용할 경우

- 렌더링 해야할 폴리곤의 수를 줄여준다.
- 사용자가 지정한 렌더링 레벨(level)을 조정할 수 있다.
- 실시간으로 지형 폴리곤의 레벨을 결정한다.
- 서로 다른 레벨의 폴리곤 사이를 부드럽게 연결한다.

위와 같은 장점이 있다. 그러나, 이러한 방법은 텍스처를 나눠서 렌더링하는 타일의 구조를 고쳐하지 않아서 전체 지형에 하나의 텍스처를 입힐 경우에만 사용할 수 있다. 또한 각각의 셀들을 따로 처리하기 때문에 간소화 과정에서 셀들 사이에 연결성이 없어질 수가 있다. 이에 따라 몇 가지 CLOD 조건들을 추가하면 아래와 같다.

- 텍스처를 입히는 타일 구조를 유지하기 위해 타일의 모서리 버텍스들은 간소화하지 않는다.

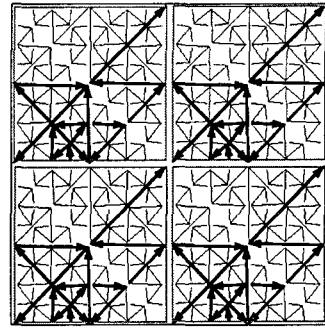
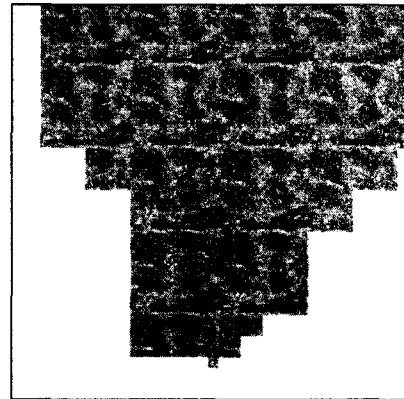
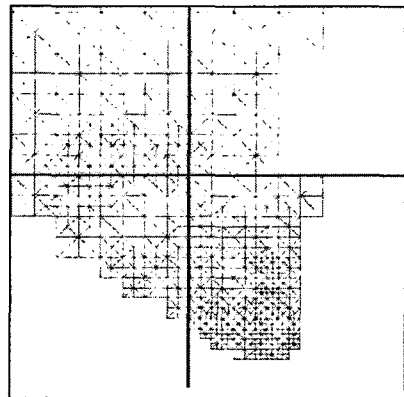


그림 3. 셀들간의 의존성 정보

· 셀들 간에 연결성을 보장하기 위해 연결된 셀들의 가장자리 버텍스들 사이에도 서로서로 의존성을 유지한다.(그림 3 참조)



(a)



(b)

그림 4. CLOD를 변형해서 적용한 결과

그림 4의 (a)는 간소화 과정에서 타일의 구조를 유지해 그대로 유지해서 여러 개의 텍스처를 하나의 게임 지형에 붙인 결과를 보여주고 있다. 그림 4의 (b)는 4개의 셀이 연결성을 유지하면서 렌더링 되는 모습을 보여준다.

4. 결론

본 논문에서는 온라인 게임을 위한 대규모의 게임 지형에 최적화된 실외용 지형의 구조와 렌더링 방법을 소개한다.

먼저, 빠른 게임 진행을 돕기 위해 지형을 여러 개의 물리적 및 논리적 단위로 계층화하여 사용하였다. 이렇듯 특수한 구조의 지형을 실시간으로 렌더링하기 위하여 지형 CLOD 방법을 개선하여 사용하는 방법을 제안하였다. 또한 이러한 구조를 사용하면 게임 지형을 계속해서 확장할 수 있다는 장점이 있다.

그림 5는 실제 본 논문에서 제안한 지형 구조를 이용하여 만든 게임 지형을 보여주고 있다.

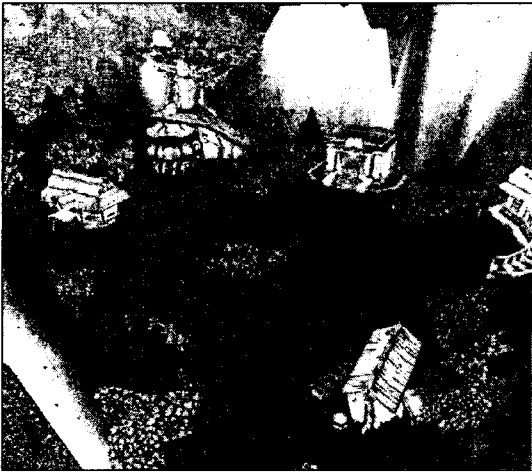


그림 5. 실제 게임에 적용한 예

참고문헌

- [1] Pankaj K. Agarwal, Edward F. Grove, T.M. Murali, and Jeffrey Scott Vitter "Binary Space Partitions For Fat Rectangles" 37th FOCS, 1996
- [2] E. Torres, "Optimization of the binary space partition algorithm (BSP) for the visualization of dynamic scenes", In Proceedings of EuroGraphics, pages. 507-518, 1990
- [3] Gary Simmons & Adam Hout, "Binary Space Partitioning Trees, Potential Visibility Sets, Constructive Solid Geometry, Portal Generation, Frustum Culling", Game Institute, 2001
- [4] Peter Lindstorm, David Kooler, William Ribarsky, Larry F. Hodges, Nick Faust, and Gregory A. Turner, "Real-Time, Continuous Level of Detail Rendering of Height Fields", Proceedings of ACM SIGGRAPH 96, pages. 109-118, 1996
- [5] Mark Duchaineau, Murray Wolinsky, David E. Sigiety, Mark C. Miller, Charles Aldrich, and Mark B. Mineev-Weinstein, "ROAMing Terrain: Real-time Optimally Adapting Meshes", Proceedings of ACM SIGGRAPH 97, 1997
- [6] Andreas Ogren, "Continuous Level of Detail In Real-Time Terrain Rendering", 2000