

이미지 피라미드 기반 다층 높이 맵을 사용한 변위 매핑 기법

전영재⁰ 오경수
 숭실대학교 미디어학부
 {dkreformer, oks}@ssu.ac.kr

Displacement mapping using an image pyramid based multi-layer height map

Chun Youngjae⁰ Oh Kyongsu
 Department of Media, Soongsil University

요 약

많은 수의 정점 정보를 사용하지 않는 대신 기하의 높이 정보를 저장한 높이 맵을 사용하여 복잡한 표면을 표현하는 기법은 많이 연구되어 왔다. 하지만 단층으로 구성된 높이 맵은 표면으로부터의 각 위치 당 한 개의 높이 정보를 저장하고 있으므로 복잡하고 오목한 물체를 표현할 수 없다. 이 논문에서는 다층으로 구성된 높이 맵을 사용하여 단일 높이 정보로는 재구성할 수 없는 복잡한 물체를 정확하게 그리는 방법에 대하여 소개한다. 우리는 그리고자 하는 장면의 높이 값을 텍스처의 각 채널에 높이에 따라 순차적으로 저장한 다층 높이 맵의 높이 정보가 2채널 마다 쌍을 이루며 기하 블록을 구성하는 점에 착안하였다. 안전하고 정확한 광선 탐색은 다층 높이 맵의 1, 3번 째 채널은 높이의 최대값을 2, 4번 째 채널은 높이의 최소값을 사용하여 이미지 피라미드를 구성함으로써 이루어진다. 이런 구조에서의 광선 탐색은 선형 탐색에 기반 한 기존의 방식들이 스크림에서 화질이 현저히 낮아지는 문제를 개선하였으며 실시간에서 표현이 가능하다.

Abstract

Many methods which represent complex surfaces using height map without a number of vertex have been researched. However, a single layer height map cannot present more complex objects because it has only one height value on each position. In this paper, we introduce the new approach to render more complex objects, which are not generated by single layer height map, using multi layer height map. We store height values of the scene to each texture channel by the ascending order. A pair of ordered height values composes a geometry block and we use this property. For accurate ray search, we store the highest value in odd channels and the lowest value in even channels to generate quad tree height map. Our ray search algorithm shows accurate intersections between viewing ray and height values using quad tree height map. We solve aliasing problems on grazing angles occurred in previous methods and render the result scene on real-time.

키워드 : 변위매핑, 실시간 렌더링, GPU 기반 광선탐색

Keywords : Displacement mapping, Real-time Rendering, GPU based Ray Searching

1. 서론

한정된 자원을 사용하여 시각적인 오류를 일으키지 않도록 복잡한 장면을 그려내는 것은 컴퓨터 그래픽스 연구 분야에 있어서 중요한 논점이다. 한정된 자원이란 저장 공간, 장면

을 그리는데 필요한 계산 시간 등을 말한다. 게임 등에서 많이 활용되고 있는 범프 매핑(Bump Mapping)은 높이 정보를 조명 계산에 사용하여 단순한 면에 복잡한 음영을 주는 방식이다. 이와 달리 “릴리프 매핑”(Relief Mapping)[1]이나 “시차 폐색 매핑”(Parallax Occlusion Mapping)[2]은 각 픽셀

에서 광선을 쏘아 기하에 도달한 경우, 도달 지점의 텍스처 공간으로 광선을 변환하여 높이 맵의 높이 값과의 교차점을 구하는 방식으로 기하의 굴곡을 표현하였다. 하지만 교차점을 구하기 위해 선형 탐색과 이진 탐색을 사용한 앞선 방법은 시야각이 작아짐에 따라 광선의 이동거리가 급증하여 많은 시각적 오류를 낳는다. 이를 해결하기 위해 이미지 피라미드 기반의 높이 맵을 만들고 광선의 진행거리를 안전하게 책정한 “이미지 피라미드를 이용한 변위 맵의 실시간 렌더링”(Pyramidal Displacement Mapping)[3]은 많은 시각적 오류를 개선하였다. 이미지 피라미드 기법에 기반 한 높이 맵은 상위 레벨의 저해상도 높이 맵을 만들기 위해 참조되는 텍셀 값들 중 가장 큰 높이 값을 선택한다. 따라서 광선 탐색 시 루트 레벨부터 탐색을 시작한 광선은 현재 레벨에서 가장 높은 높이 까지만 진행하므로 안전하고 정확하게 광선과 높이 값의 교차점을 찾을 수 있다.

기존의 변위 매핑 기법들은 단층으로 구성된 높이 맵을 사용하는 것이 일반적이었다. 따라서 한 지점에 대해 한 개의 높이 정보가 제공 되고 이는 표면으로부터 불룩하게 솟아 오른 물체에 대해서만 표현이 가능하다는 단점을 안고 있다. Policarpo et al.[4]은 이런 문제를 해결하기 위해 다층으로 구성된 높이 맵을 제안하여 높이 맵으로 표현할 수 있는 외형의 한계를 확장하였다. 하지만 선형 탐색과 이진 탐색을 병행하여 광선과 높이 값의 교차점을 검출하였기 때문에 앞서 언급한 대로 시각적인 오류가 발생한다. 이 논문에서 우리는 이를 개선하기 위해 정확하게 교차점을 찾아줄 수 있는 PDM(Pyramidal Displacement Mapping)[3]의 방식을 접목하였으며 그 과정에서 생기는 문제점들을 해결하였고 기존의 방식보다 개선된 화질로 복잡한 물체를 표현할 수 있었다.

이 논문의 나머지 부분들은 다음과 같다. 2절에서는 관련 연구에 대해서 소개하고 3절에서는 이 논문에서 사용하는 다층 높이 맵의 자료구조에 대해 설명한다. 4절에서는 3절에서 소개된 다층 높이 맵 자료구조를 사용한 광선 탐색 과정에 대해 설명하며 5절에서 그 실험 결과를 서술한다. 마지막으로 6절에서는 결론에 대하여 토의한다.

2. 관련연구

GPU를 사용한 실시간 변위 매핑은 많은 방식으로 연구되어 왔다[1-3]. 초기의 주요 관점은 눈으로 대표되는 카메라에서 시작하는 광선과 높이 맵에 저장된 높이 값과의 충돌 지점을 계산하는 것이었다. 릴리프 매핑[1]과 시차 페색 매핑[2]은 광선 탐색 방식이 간단하여 빠른 렌더링 속도를 보장하지만 카메라의 시점이 스킨각(Grazing angle)에 가까워짐

에 따라 심각한 시각적 오류가 발생한다. 이는 시점 광선과 높이 정보의 올바른 교차점을 찾지 못하는 선형탐색, 이진 탐색에 의한 문제점이다. 이런 문제를 해결하기 위해 PDM[3]은 입력 높이 맵을 쿼드트리(Quad Tree)로 재구성하고 부모 노드는 자식 노드들 중 가장 큰 높이 값을 저장하게 하였다. 쿼드 트리의 루트에서 탐색을 시작한 시점 광선은 가장 높은 높이 까지만 전진을 할 수 있으므로 광선이 교차점을 지나치거나 찾지 못하는 문제점을 해결하였다. 하지만 지금까지 언급된 연구에선 높이 정보를 한 층으로 구성하였기 때문에 표면에서부터 불룩한 모양은 표현할 수 있으나 중간 부분이 오목하거나 복잡한 물체를 표현할 수 없었다.

높이 맵을 사용하지 않고 기하의 모양 자체를 3차원 텍스처에 저장한 후 그리는 방법에서도 많은 연구가 이루어 졌다. 이 방법은 단층 높이 맵으로 구성되는 표면보다 더 복잡한 형태를 표현할 수 있으나 저장공간, 처리속도의 문제가 있다. Porumbescu et al. [5]은 기본 기하를 확장하여 기본 기하를 감싸는 쉘 공간(shell space)을 생성하고 쉘 공간을 사면체로 분할하였다. 분할된 사면체들의 쉘 공간은 무게중심 좌표계를 사용하여 텍스처 공간과 일대일대응이 되며 최종적으로 광선 추적법(Ray Tracing)을 사용하여 복잡한 표면을 임의의 기하에 매핑하였다. 이 방식의 단점으로는 기본 기하의 굴곡이 심한 부분에서 왜곡이 발생할 수 있고 광선 추적법을 사용하기 때문에 실시간으로 처리가 불가능하다는 점이다. Chen et al.[6]은 투명도를 가지는 메소구조를 표현하기 위해 물체를 불균질 쉘과 균질 코어로 모델링한다. 쉘은 물체의 표면에서 일어나는 빛의 전달을 표현하기 위한 모델이며 표면에서 일어나는 광원의 산란 정보는 볼륨 포톤 매핑을 사용하여 지점과 방향에 대한 5차원 함수로 접근할 수 있도록 저장한다. 이렇게 미리 계산된 볼륨 데이터는 입력 메쉬의 정해진 두께에 따라 배치되며 광원의 산란을 계산하여 최종 렌더링 결과를 낸다. 결과 화면은 면하산란을 포함한 포톤 매핑(약 3190초 소요)과 유사한 정도의 화질을 보이나 렌더링 속도는 약 40배 가량 빠르나(약 77초 소요) 실시간 어플리케이션에 적용하기 힘든 문제점이 있다.

Wang et al.[7]은 일반적인 메소구조(Meso-structure)를 표현하기 위해 광선의 시작점과 방향에 따른 변위와의 거리를 볼륨 텍스처에서 미리 계산하여 5차원 함수로 저장하는 방식을 제안하였다. [7]에서 제안된 GDM(Generalized Displacement Mapping)의 모델링은 볼륨 텍스처의 해상도와 시점 샘플링 횟수에 따라 화질이 결정되므로 가까운 시점에서의 고 화질을 확보하기 위한 5차원 자료구조는 많은 저장 공간을 필요로 한다. 이는 전처리 단계에서 많은 부담으로 작용하며 그래픽스 하드웨어 상의 저장 공간에서 한 번에 다룰 수

없는 크기였기에 SVD(Singular Value Decomposition)를 사용하여 데이터 압축을 하였다.

Policarpo et al.[4]은 다층의 높이 맵을 사용할 경우 표현할 수 없는 복잡하고 오목한 물체를 표현하기 위해 그림 1과 같이 텍스처의 한 픽셀 당 최대 4개의 높이 정보를 저장하였다. 순서대로 저장된 4개의 높이 정보는 두 쌍으로 이루어져서 광선 탐색의 중간 단계에서 광선이 물체의 내부에 속하는지 외부에서 정지하였는지를 확인하기 위해 참조된다. 하지만 시점 광선과 높이 값과의 교차점을 구하기 위해 먼저 선형 탐색을 수행하고 중간 결과를 재사용하여 이진 탐색하였기 때문에 그림 2와 같은 예리한 각도에서 시각적 오류를 발생한다.

이 논문에서는 Policarpo et al.[4]이 제시한 방법의 문제점을 해결하기 위해 보다 정확한 광선 탐색 방식인 PDM[3]을 사용하였으며 이를 적용하기 위해 높이 맵 데이터를 새로운 방법으로 재구성 하였다(3절에서 설명). 결과적으로 스피큘에서의 오류가 눈에 띄게 줄어든 다층 변위 매핑 결과들 상호대화 가능한 속도로 표현할 수 있었다.

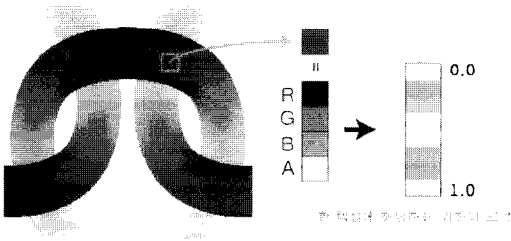


그림 1. 다층 높이 맵의 구조.

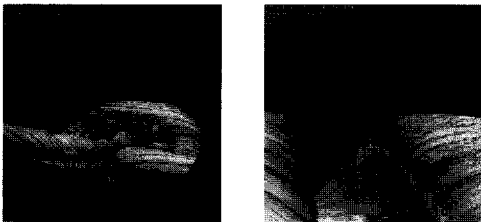


그림 2. 다층 높이 맵 릴리프 매핑의 이진 탐색에 따른 시각적 오류.

3. Max-Min 다층 높이 맵

실시간 변위 매핑 기법은 정점으로 이루어진 실제 기하체를 그리는 대신에 높이 맵을 사용하여 기하 본래의 모습과 가깝게 그릴 수 있는 유용한 기법이다. 하지만 광선 탐색

시 참조하는 높이 맵의 한 픽셀 당 한 개의 높이 값만을 저장하였기 때문에 이는 더욱 복잡한 형태의 기하를 표현할 수 없다는 단점으로 작용한다. 앞서 언급한 바와 같이 Policarpo et al.[4]은 4채널로 구성된 높이 맵을 사용하여 가능한 한도 내에서 복잡한 물체를 그릴 수 있었지만 광선 탐색 방식이 기본적으로 릴리프 매핑[1]의 방식 사용하므로 시각적인 오류를 유발할 수 있다.

우리는 이런 시각적 오류를 개선하기 위한 방법인 PDM[3]에서 다중 구조의 높이 맵을 사용할 수 있도록 먼저 Max-Min 다중 높이 맵을 작성하였다. 다중 높이 맵은 한 지점의 높이 값이 높은 순서대로 각 채널에 하나씩 저장한다. 우리는 다중 텍스처에 저장된 높이 값들이 정렬되어있고 두 채널 마다 장면을 블록화 한다는 점에 착안하였다. 쿼드 트리를 구성하는 과정에서 부모노드의 값을 결정할 때 홀수 채널은 최대값을 짝수 채널은 최소값을 저장하는 방식으로 쿼드 트리를 구성하는 방식이 우리가 제안하는 높이 맵의 구성이다. 그림 3에서 리프 레벨에 해당하는 원래의 높이 맵에서 1레벨에 해당하는 상위 높이 맵을 만들 때 이 논문에서 제안한 방법으로 작성을 하면 오른쪽 상단의 높이 맵을 얻을 수 있다. 결과적으로 루트 레벨에서는 복잡한 물체의 바운딩 박스 같은 가장 투박한 모양을 나타내고 리프 레벨로 가면 갈수록 세밀한 원래의 모습을 되찾게 된다. 이런 방식의 높이 값 재구성은 PDM[3]과 마찬가지로 리프 레벨 보다 높은 레벨에서 광선이 진행 할 수 있다면 최대한 진행할 수 있게 해준다. Max-Min 다중 높이 맵의 작성은 전처리 과정에 해당하며 입력 높이 맵에 대해서 단 1회의 작업만이 실행된다.

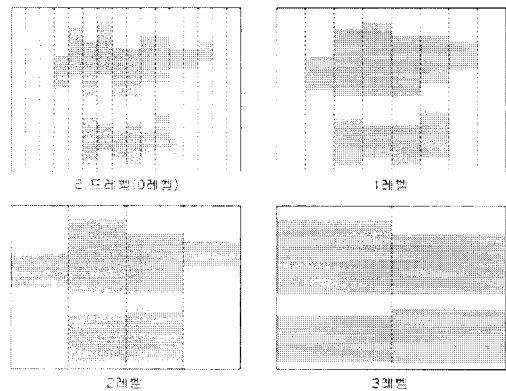


그림 3. 가장 높은 높이 값과 가장 낮은 높이 값을 번갈아가며 쿼드트리를 작성한 Max-Min 다중 높이 맵.

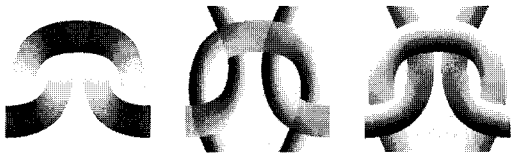


그림 4. 4채널로 구성된 다층 높이 맵과 각 지점의 법선 정보를 저장한 두 개의 법선 맵.

4. Max-Min 다층 높이 맵에서의 광선탐색

논문에서 제시하는 방법은 입력으로 1개의 높이 맵과 2개의 법선 맵을 사용하며 각 텍스처는 4채널로 이루어진 RGBA형식의 .tga파일이다(그림 4.). 다층 높이 맵은 전처리 과정을 이미지 피라미드 기반의 Max-Min 다층 높이 맵으로 재작성 되고 이렇게 작성된 높이 맵에서 올바른 높이 맵과의 교차점을 찾기 위해 광선 탐색을 수행한다. 2개의 법선 맵은 각각 정규화 된 법선 벡터를 구성하는 x성분과 y성분의 값을 저장한다. 채널 별로 해당 높이 정보에 해당하는 법선 정보를 저장하며 z축 성분은 실행 시간에 픽셀 세이더에서 피타고라스의 정리를 사용하여 계산한다.

4.1 이미지 피라미드 방식에 기반 한 광선탐색

이 논문에서 수행한 광선 탐색은 이미지 피라미드 방식에 기반 한 다층 높이 맵에서 이루어지므로 기존의 방식보다 정확하게 광선과 높이 정보와의 교차점을 찾을 수 있다. 먼저 한 픽셀의 색상을 결정하기 위해 일반적인 GPU기반 변위맵핑 방법과 같이 그리고자 하는 물체의 텍스처 공간으로 시점 벡터를 변환하여 광선의 탐색을 시작한다. 기존 연구인 PDM은 현재 광선의 높이와 높이 맵의 높이를 비교하여 광선의 높이가 더 높은 경우 광선을 안전한 깊이까지 진행시켰다. 하지만 우리의 방법에서는 광선의 현재 위치에서 4개의 깊이 값을 깊이 맵으로부터 읽어서 광선과의 차를 구하여 그 부호를 확인한다. 부호가 양수라면 광선이 기하의 외부에 존재하는 경우이므로 광선을 안전한 거리까지 진행시키고 음수라면 기하의 내부에 존재하는 경우이므로 정지한 채 맵의 레벨을 낮추도록 하였다. 이는 광선과 다층 높이 맵과의 교차점을 정확하게 찾을 수 있는 방법으로 기존의 방법에 비해 개선된 사항이다. 광선을 진행시킬 수 있다고 판단되면 우리는 그 다음 단계로 광선이 안전하게 전진할 수 있는 거리를 결정한다. 진행 거리는 현재 광선의 위치에서 높이 맵의 높이 값들을 읽어오고 그 중 현재 광선의 높이보다 더 낮으면서 가장 가까운 높이를 선택한다. 이 방식은 하향식 광선 탐색을 가정하고 있으며 높이 차의 절대 값이 가장 작은 이동거리를 선택함으로써 안전한 광선의

진행을 보장한다. 여기서 노드 넘어가기(Node Crossing) 방식은 기존 PDM의 방식을 따른다. 그림 5.에서는 광선 탐색의 과정을 순서도로 간략히 보여준다. 광선 탐색이 종료 되면 두 개의 법선 맵으로부터 법선 벡터의 x, y성분을 읽는다. 여기서 우리는 광선 탐색을 마친 지점의 높이 정보를 참조하여 가장 근접한 채널을 선택하고 그 채널에 해당하는 법선 정보를 읽어온다. 그리고 각 접선 벡터 성분은 정규화 되어있으므로 피타고라스의 정리를 사용하여 z성분을 계산한다. 이후 광원의 방향에 따른 밝기 계산은 계산된 법선 벡터를 사용하여 수행된다.

4.2 Max-Min 다층 높이 맵 생성 시의 문제점

지금까지 제안된 다층 높이 맵의 재구성과 광선 탐색을 다루었다. 불행하게도 Max-Min 다층 높이 맵 생성 시, 리프 레벨에서는 올바른 높이 순서를 갖는 G, B 채널의 높이 정보가 이미지 피라미드 방식으로 재구성을 하는 동안 역전될 수 있다. 이 경우 4.1절에서 제안된 광선 탐색을 그대로 사용할 경우 예상치 못한 광선의 진행으로 인해 그림 8.과 같이 올바른 교차점을 찾지 못하고 뚫리는 결과를 낳는다. 그림 7.에서는 상위 레벨의 텍셀을 정하기 위한 쿼드트리 작성 중 생기는 문제에 대하여 보여주고 있다.

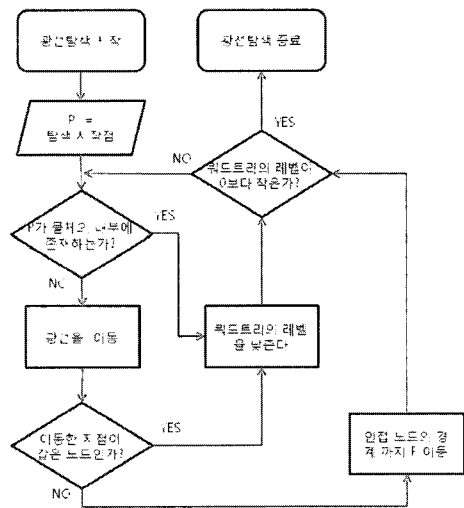


그림 5. 광선 탐색의 순서도

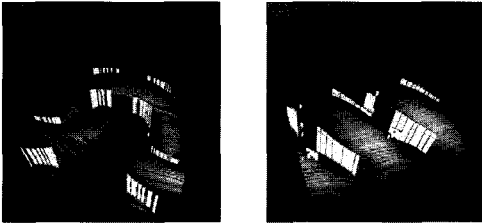


그림 6. Max-Min 다중 높이 맵 생성 도중 높이 값의 역전되는 부분.(밝게 표시된 부분)

3절에서 설명된 Max-Min 맵 생성 방식에 따르면 R, B 채널은 상위 레벨의 텍셀을 결정할 때 참조되는 4텍셀의 높이 값 중 가장 큰 값을 취하고 G, A 채널은 가장 작은 값을 취한다. 우리는 이런 과정을 통해 장면의 전반적인 모양부터 세밀한 모양까지 밀맵의 레벨에 따라 표현됨을 예상하였다. 하지만 4채널 다중 높이 맵은 4채널을 전부 사용하는 지점과 2채널만 사용하는 지점 그리고 어떤 높이 값도 저장되어 있지 않은 지점으로 분류 되고 각 지점이 연결되는 부분에서는 Max-Min 맵 작성 시 문제가 발생할 수 있다. 4채널을 사용하는 지점과 2채널을 사용하는 지점이 인접하여(그림 6.에 표시된 밝은 부분) 상위 레벨의 텍셀을 정하는 과정에서 참조 된다면 그림 7.과 같은 변위 역전이 일어날 가능성이 있다. 이는 광선이 물체의 외부에 존재할 경우 전진한다는 공식을 항상 만족시킬 수 없는 문제를 일으킨다. 우

리는 이 문제를 해결하기 위해 광선의 진행 여부 결정 단계에서 현재 위치에서의 G, B 채널의 높이 값이 역전되어 있는지 확인한다. 만약 두 높이 중 B채널의 값이 더 높다면 광선을 진행하지 않고 밀맵 레벨을 낮추고 그렇지 않다면 정상적으로 진행을 시킨다. 4.1절에서 제안된 방법으로 광선 탐색을 수행한 경우 그림 8.에서와 같이 구멍(Hole)이 생길 수 있다(오른쪽 그림은 광선 탐색 결과의 높이 정보를 보여준다). 반면에 그림 9.는 수정된 광선 탐색의 결과를 보여주고 있으며 이 논문에서 제시하는 방법은 임의의 다중 높이 맵에 대해 올바른 매핑 결과를 보장한다. 개선된 광선 탐색 방식에서는 광선이 전진하지 못하고 한 자리에서 잠시 머무는 경우가 더 많아지나 이는 광선 탐색 알고리즘에 거의 부하를 주지 않아서 렌더링 속도 역시 많은 차이 없는

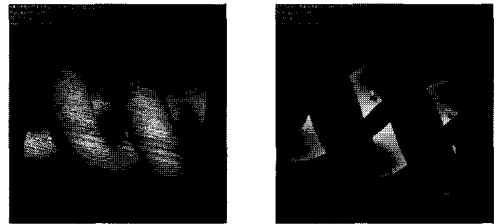


그림 8. 정렬된 높이 순서를 갖지 못하는 다중 높이 맵에서의 오류.

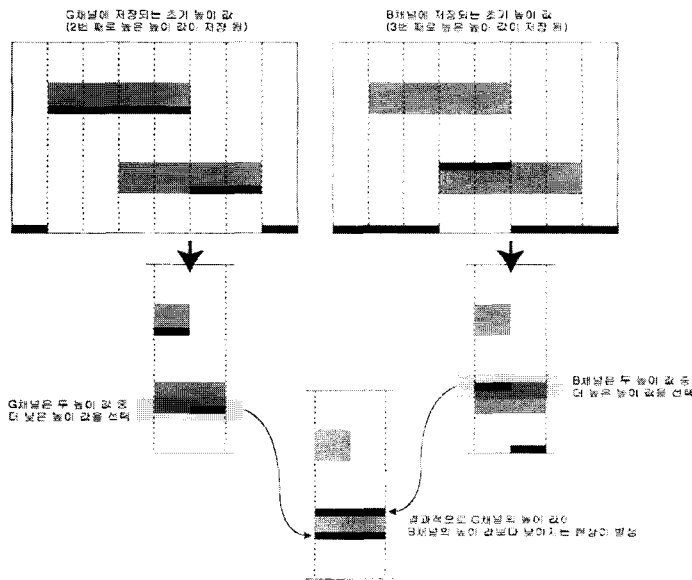


그림 7. 상위 레벨의 텍셀 값을 결정하기 위한 Max Min맵 생성 중 생기는 변위 역전

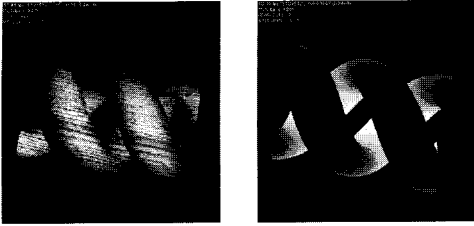


그림 9. 수정된 광선 탐색을 사용한 다층 높이 맵 렌더링 결과. 퍼포먼스를 보였다.

5. 결과

우리는 512*512 해상도의 화면을 렌더링 하기 위해 ATI사 Radeon X1900 512M 그래픽 카드를 사용하였으며 DirectX 3D 9.0, Shader Model 3.0 환경에서 이 논문의 구현 내용을 실험하였다. CPU는 전처리 과정에서 Max-Min 다층 높이 맵을 만드는 과정에서 주로 사용되며 Intel Pentium 4 3.0GHz를 사용하였다. 모든 결과 이미지는 256*256 해상도를 가지는 RGBA 32비트 높이 맵, 법선 맵 그리고 컬러 맵을 사용하여 렌더링 하였다. 그림 10.의 오른쪽 그림들은 이 논문에서 제시된 기법으로 렌더링 한 결과이며 평균적으로 155fps로 성능을 보였다. 같은 높이 맵을 기존의 방식[4]으로 렌더링 한 결과(그림 10.의 왼쪽, 약 400fps)와 비교하여 렌더링 속도는 상당량 감소하지만 충분히 실시간에 렌더링이 가능한 정도이며 대신에 화질은 눈에 띄게 향상되었다. 그림 11.에서는 기존의 방식[4]에서 언급된 대로 R채널만 사용하여 단층 높이 맵을 사용한 결과와 기존의 단층 높이 맵 렌더링 방식과 동일한 결과를 얻을 수 있음을 보이며 이 때, 계산 부하가 절반가량 줄어들었다(약 400fps).

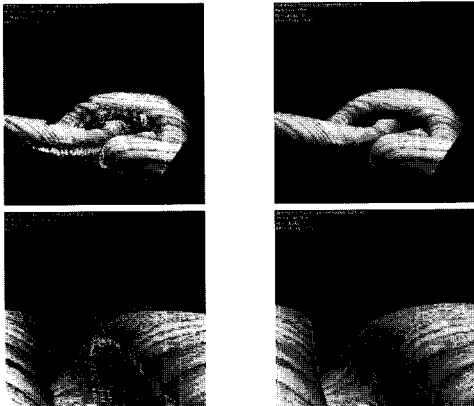


그림 10. 기존의 기법과 우리의 기법으로 다층 높이 맵을 렌더링 한 결과의 비교화면.

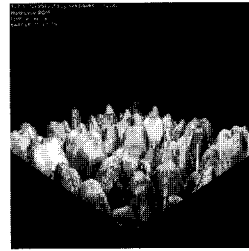
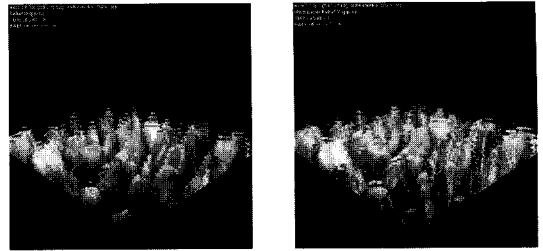


그림 11. 단층 높이 맵과 다층 높이 맵의 한 채널만 사용하여 렌더링 한 결과의 비교화면. 왼쪽 상단은 릴리프 맵핑, 오른쪽 상단은 다층 높이 맵의 릴리프 맵핑, 하단은 우리들의 결과 화면으로 뚜렷한 화질 차이를 보여준다.

6. 결론 및 향후과제

우리는 이 논문에서 여러 층으로 구성된 높이 정보를 이미지 피라미드 기반의 자료구조로 정리하고 정확한 광선 탐색 방식을 사용하여 단층 높이 맵으로 표현할 수 없는 오목하고 복잡한 형태의 물체를 그리는 방법에 대해 소개하였다. 기존의 변위 맵핑들은 게임 같은 실행 프로그램에 적용 시 쌍방향 소통이 가능할 정도로 계산이 빠르다. 따라서 건물이나 지면의 표현 등에 자주 사용되지만 단층 높이 맵의 사용 때문에 표현에 한계가 있다. 이 논문에서 제안하는 방법은 이런 표현의 한계를 완화하는 대안으로 제시될 수 있을 만큼의 화질과 빠른 계산 속도를 보장한다. 현재, 다층 높이 맵으로 구성되는 장면의 많은 물체들에게 개별적인 ID를 부여하고 각각 다른 색상의 텍스처를 입혀서 단일 물체가 아닌 하나의 장면 전체를 재구성하거나 실시간으로 높이 정보를 수정하여 장면을 변화 시키는 작업들은 추가적인 연구로 남아있다.

감사의 글

- 본 연구는 숭실대학교 교내 연구비 지원으로 이루어졌음.
- 본 연구는 정보통신부 및 정보통신연구진흥원의 IT신성장 동력핵심기술개발사업의 일환으로 수행하였음. [2006-S-045-01, 기능 확장형 초고속 렌더러 개발]

참고 문헌

- [1] Policarpo, F., Oliveira, M. M., and Comba, J., Real-time relief mapping on arbitrary polygonal surfaces, *In Proceeding of ACM Symposium on Interactive 3D Graphics and Games 2005*, pp. 155-162, 2005.
- [2] Brawley, Z., and Tatarchuk, N., Parallax Occlusion Mapping: Self-Shadowing, Perspective-Correct Bump Mapping Using Reverse Height Map Tracing, *In ShaderX3: Advanced Rendering with DirectX and OpenGL*, pp. 135-154, 2004.
- [3] 오경수, 기현우, 이미지 피라미드를 이용한 변위 맵의 실시간 렌더링, *하계 한국게임학회 학술발표대회*, pp. 209-215, 2006.
- [4] Policarpo, F., and Oliveira, M. M., Relief mapping of Non-height-field surface details, *13D '06: Proceedings of the 2006 symposium on Interactive 3D graphics and games*, pp. 55-62, 2006.
- [5] Porumbescu, S. D., Burge, B., Feng, L., Joy, K. I., Shell maps, *In Proceeding of ACM SIGGRAPH 2005*, pp. 626-633, 2005.
- [6] Chen, Y., Tong, X., Wang, J., Lin, S., Guo, B., and Shum, H.-Y., Shell texture functions, *ACM Transaction of Graphics - Proceedings of SIGGRAPH 2004*, pp. 343-352, 2004.
- [7] Wang, X., Tong, X., Lin, S., Hu, S., Guo, B., and Shum, H.-Y., Generalized displacement maps, *In Eurographics Symposium on Rendering 2004, EUROGRAPHICS, Keller and Jensen, Eds., EUROGRAPHICS*, pp. 227-233, 2004.