(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2008년01월22일
(11) 등록번호 10-0796735
(24) 등록일자 2008년01월15일

(51) Int. Cl.
G06T 17/10 (2006.01) *G06T 15/00* (2006.01)
(21) 출원번호 10-2007-0091334
(22) 출원일자 2007년09월10일
심사청구일자 2007년09월10일
(56) 선행기술조사문헌
KR1020050112148 A
KR1020020087964 A

(73) 특허권자
인하대학교 산학협력단
인천 남구 용현동 253 인하대학교
(72) 발명자
신병석
경기 안양시 동안구 갈산동 샘마을한양아파트 114동 402
임석현
서울 서초구 방배1동 888-34 롯대캐슬파크 C-1001
(74) 대리인
김건우

전체 청구항 수 : 총 3 항 심사관 : 장기정

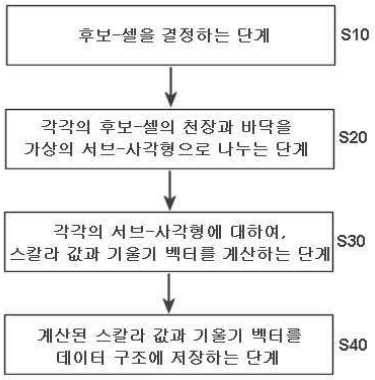
(54) 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법

(57) 요 약

본 발명은 볼륨 광선 투사법(volume ray casting)을 위한 데이터 구조(data structure)를 생성하는 방법에 관한 것으로서, 본 발명에 따른 데이터 구조 생성 방법은, (1) 각각의 셀(cell)에 대하여 투명-셀(transparent cell)인지 여부를 확인하여 후보-셀(candidate cell)을 결정하는 단계 - 여기서, 3차원 픽셀인 복셀(voxel) 8개로 구성되는 큐브(cube) 구조를 셀이라 하며, 셀을 이루는 8개의 복셀이 모두 투명한 경우를 투명 셀이라고 하고, 최종 영상에 영향을 주는 셀을 후보-셀이라고 함 - ; (2) 결정된 상기 각각의 후보-셀에 대하여, 천장과 바닥을 가상의 서브-사각형(sub-square)으로 나누는 단계; (3) 나누어진 상기 각각의 서브-사각형에 대하여, 쌍선형 보간법(bilinear interpolation)을 이용하여 스칼라 값과 기울기 벡터를 계산하는 단계; 및 (4) 계산된 상기 스칼라 값과 기울기 벡터를 데이터 구조로서 저장하는 단계를 포함하는 것을 그 구성상의 특징으로 한다.

본 발명의 볼륨 광선 투사법을 위한 데이터 구조 생성 방법에 따르면, 후보-셀의 천장과 바닥을 가상의 서브-사각형으로 나누고, 나누어진 각각의 서브-사각형에 대하여, 쌍선형 보간법을 이용하여 스칼라 값과 기울기 벡터를 계산하여 데이터 구조로서 저장함으로써, 기존의 방법과 비교하여 렌더링 시간을 크게 줄일 수 있으면서, 동시에 전처리에 소요되는 시간 및 메모리를, 후보-셀을 서브-셀로 나누는 기존의 방법에 비하여 대폭 감소시킬 수 있다.

대표도 - 도2



특허청구의 범위

청구항 1

(1) 각각의 셀(cell)에 대하여 투명-셀(transparent cell)인지 여부를 확인하여 후보-셀(candidate cell)을 결정하는 단계 - 여기서, 3차원 픽셀인 복셀(voxel) 8개로 구성되는 큐브(cube) 구조를 셀이라 하며, 셀을 이루는 8개의 복셀이 모두 투명한 경우를 투명 셀이라고 하고, 최종 영상에 영향을 주는 셀을 후보-셀이라고 함 - ;

(2) 결정된 상기 각각의 후보-셀에 대하여, 천장과 바닥을 가상의 서브-사각형(sub-square)으로 나누는 단계;

(3) 나누어진 상기 각각의 서브-사각형에 대하여, 쌍선형 보간법(bilinear interpolation)을 이용하여 스칼라 값과 기울기 벡터를 계산하는 단계; 및

(4) 계산된 상기 스칼라 값과 기울기 벡터를 데이터 구조로서 저장하는 단계

를 포함하는 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법.

청구항 2

제1항에 있어서,

상기 단계 (4)에서, 상기 데이터 구조를 참조하기 위한 후보-셀들의 색인을 인덱스 구조로서 저장하는 것을 특징으로 하는 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법.

청구항 3

제1항 또는 제2항에 있어서,

상기 단계 (2)에서, 후보-셀의 기울기 벡터의 변화 정도를 판단한 후, 상기 기울기 벡터의 변화가 소정의 기준 값보다 크다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 증가시키고, 상기 기울기 벡터의 변화가 소정의 기준 값보다 작다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 감소시키는 것을 특징으로 하는 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법.

청구항 4

삭제

명 세 서

발명의 상세한 설명

기 술 분 야

<1> 본 발명은 볼륨 광선 투사법(volume ray casting)을 위한 데이터 구조(data structure)를 생성하는 방법에 관한 것으로서, 보다 구체적으로는 각각의 후보-셀(candidate cell)의 천장과 바닥을 가상의 서브-사각형(sub-square)으로 나누고, 나누어진 각각의 서브-사각형에 대하여, 쌍선형 보간법(bilinear interpolation)을 이용하여 스칼라 값과 기울기 벡터를 계산하여 데이터 구조로서 저장함으로써, 기존의 방법과 비교하여 렌더링 시간을 크게 줄일 수 있으면서, 동시에 전처리에 소요되는 시간 및 메모리를, 후보-셀을 서브-셀로 나누는 기존의 방법에 비하여 대폭 감소시킬 수 있는 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법에 관한 것이다.

배 경 기 술

<2> 볼륨 렌더링(volume rendering)은 볼륨 데이터(volume data)로부터 의미 있는 정보를 추출하여 가시화하는 기법이다. 볼륨 렌더링은 크게 표면 렌더링(surface rendering) 기법과 직접 볼륨 렌더링(direct volume rendering) 기법으로 나뉜다. 볼륨 광선 투사법은 직접 볼륨 렌더링 기법 중 하나로서, 고화질의 영상을 생성할 수 있으므로 많이 사용되고 있다. 볼륨 광선 투사법은 크게 빈공간을 도약하는 단계와 색상값을 계산하는 단계로 나뉜다. 볼륨 광선 투사법의 가장 큰 단점은 렌더링 속도가 느리다는 것인데, 이를 해결하기 위한 많은 연구들이 최종 영상에 기여하지 않는 빈공간을 빠른 속도로 도약하는 방법에 집중되어 왔다. 하지만, 빈공간을 도약한 후 색상값을 계산하기 위해서는 결국 주 메모리(main memory)를 여러 번 임의로 참조해야 하기 때문에

렌더링 속도가 저하될 수밖에 없다. 이와 같은 렌더링 속도의 저하는, 불투명한 복셀들이 많을 경우 더욱 심각하게 된다.

<3> CPU 기반의 볼륨 렌더링 기법을 가속하기 위하여 많은 방법이 제안되어 왔다. Yagel 등은 광선의 진행 경로에 따른 샘플의 위치와 재샘플링 가중치를 미리 계산하여 놓은 원형(template)을 이용하여 샘플링 연산을 단축시키는 방법을 제안하였는데, 이 방법은 직교 투영(orthogonal projection)에만 가능하며 투시 투영(perspective projection)이 필요한 분야에는 이용하기 어렵다는 단점이 있다. 쉬어-왑(shear warp) 볼륨 렌더링은 메모리 참조 효율성을 높여서 색상값 계산 단계를 가속하는 방법으로서, 대화적 분류(interactive classification) 환경에서 캐시 미스(cache miss)를 유발하며, 메모리 참조 효율성을 높이기 위해 재샘플링 필터로 쌍선형 보간법(bilinear interpolation)을 이용하기 때문에 영상의 질이 떨어진다는 문제점이 있다. Parker 등은 워크스테이션 상에서 메모리 접근법을 향상시켜 렌더링 속도를 가속하는 방법을 제안하였는데, 이 방법은 등위면 렌더링(isosurface rendering)에만 국한되었으며, 실시간 렌더링을 위해서는 128개의 CPU을 필요로 하였다. 인텔사의 MMX™(MultiMedia eXtension) 기술을 이용하여 캐시 연산을 가속하는 볼륨 광선 투사법 시스템도 발표되었는데, 이 시스템은 실시간에 가까운 렌더링 속도를 보이지만 영상의 크기가 256× 256으로 제한되어 있다는 문제점이 있다. 만약 그 이상의 해상도를 원할 경우에는, 256× 256의 영상을 확대할 수밖에 없는데, 이 경우 영상이 블러링(blurring)하게 보이는 문제점이 발생한다. Mora 등은 볼륨 광선 투사법에 스플래팅 개념을 도입하여 셀들을 동일한 큐브 모양으로 투영하는 객체-기반 볼륨 광선 투사법(object order volume ray casting)을 제안하였는데, 이 방법은 렌더링 속도가 빠르지만 투시 투영에는 이용할 수 없다는 문제점이 있다. 메모리의 참조를 임의로 접근하고 CPU의 캐시(cache) 공간에 저장되어 있는 기울기 벡터(gradient vector)를 효율적으로 활용하여 볼륨 광선 투사법의 단점을 해결하는 방법도 소개되었는데, 이 방법을 사용하면 범용 PC에서 초당 10프레임 이상의 속도를 보여주지만, 역시 직교 투영에만 가능하다는 단점이 있다. 또한 쓰레드 레벨에서의 병렬성(thread-level parallelism, TLP)과 볼륨 데이터의 접근 순서를 변경하여 렌더링 속도를 가속시키는 방법도 제안되었다.

<4> 최근에 볼륨 광선 투사법에서 색상값 획득 속도를 가속하기 위한 데이터 구조가 제안되었는데, 이 방법은 전처리 과정에서 후보-셀들을 가상의 N_{ss}^3개의 서브-셀로 나눈 후, 각각의 서브-셀마다 삼선형 보간법(trilinear interpolation)으로 계산된 스칼라 값과 기울기 벡터를 저장한 후, 이들을 렌더링 단계에서 사용하였다. 도 1은 상기 방법에 따라서 한 개의 후보-셀을 N_{ss}^3개의 서브-셀들로 나눈 후, 삼선형 보간법으로 계산된 스칼라 값과 기울기 벡터를 저장하는 것을 나타내는 도면이다. 도 1에 도시된 예에서는, 한 개의 후보-셀이 64(=4^3)개의 서브-셀들로 나뉜다. 상기 방법은 렌더링 시간을 줄일 수 있다는 장점은 있지만, 후보-셀들의 개수가 증가할 경우 전처리에 많은 시간이 요구되며, 또한 메모리의 소비도 크게 증가된다는 문제점이 있다.

발명의 내용

해결 하고자하는 과제

<5> 본 발명은 기존에 제안된 방법들의 상기와 같은 문제점들을 해결하기 위해 제안된 것으로서, 기존의 방법과 비교하여 렌더링 시간을 크게 줄일 수 있으면서, 동시에 전처리에 소요되는 시간 및 메모리를, 후보-셀을 서브-셀로 나누는 기존의 방법에 비하여 대폭 감소시킬 수 있는 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법을 제공하는 것을 그 목적으로 한다.

과제 해결수단

<6> 상기한 목적을 달성하기 위한 본 발명의 특징에 따른, 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법은,

<7> (1) 각각의 셀(cell)에 대하여 투명-셀(transparent cell)인지 여부를 확인하여 후보-셀(candidate cell)을 결정하는 단계 - 여기서, 3차원 픽셀인 복셀(voxel) 8개로 구성되는 큐브(cube) 구조를 셀이라 하며, 셀을 이루는 8개의 복셀이 모두 투명한 경우를 투명 셀이라고 하고, 최종 영상에 영향을 주는 셀을 후보-셀이라고 함 - ;

<8> (2) 결정된 상기 각각의 후보-셀에 대하여, 천장과 바닥을 가상의 서브-사각형(sub-square)으로 나누는 단계;

<9> (3) 나누어진 상기 각각의 서브-사각형에 대하여, 쌍선형 보간법(bilinear interpolation)을 이용하여 스칼라

값과 기울기 벡터를 계산하는 단계; 및

<10> (4) 계산된 상기 스칼라 값과 기울기 벡터를 데이터 구조로서 저장하는 단계를 포함한다.

<11> 바람직하게는, 상기 단계 (4)에서, 상기 데이터 구조를 참조하기 위한 후보-셀들의 색인을 인덱스 구조로서 저장할 수 있다.

<12> 바람직하게는, 상기 단계 (2)에서, 후보-셀의 기울기 벡터의 변화 정도를 판단한 후, 상기 기울기 벡터의 변화가 소정의 기준 값보다 크다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 증가시키고, 상기 기울기 벡터의 변화가 소정의 기준 값보다 작다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 감소시킬 수 있다.

<13> 더욱 바람직하게는, 후보-셀의 영상의 변화 정도를, 기울기 벡터의 변화 정도를 이용하여 판단할 수 있다.

효 과

<14> 본 발명의 볼륨 광선 투사법을 위한 데이터 구조 생성 방법에 따르면, 후보-셀의 천장과 바닥을 가상의 서브-사각형으로 나누고, 나누어진 각각의 서브-사각형에 대하여, 쌍선형 보간법을 이용하여 스칼라 값과 기울기 벡터를 계산하여 데이터 구조에 저장함으로써, 기존의 방법과 비교하여 렌더링 시간을 크게 줄일 수 있으면서, 동시에 전처리에 소요되는 시간 및 메모리를, 후보-셀을 서브-셀로 나누는 기존의 방법에 비하여 대폭 감소시킬 수 있다.

발명의 실시를 위한 구체적인 내용

<15> 이하에서는 첨부된 도면들을 참조하여, 본 발명에 따른 실시예에 대하여 상세하게 설명하기로 한다.

<16> 먼저 볼륨 광선 투사법에서의 색상값 계산 과정을 간단히 살펴본 후, 렌더링 단계에서의 상기 색상값 계산 시간을 단축시킬 수 있는 본 발명에 따른 데이터 구조 생성 방법을 구체적으로 살펴보기로 한다.

<17> 볼륨 광선 투사법에서 색상값 계산 과정은 크게 3개의 절차로 구성된다. (1) 첫 번째로, 광선이 임의의 샘플점에 놓일 경우 정확한 투명도 값을 판단하기 위하여 재샘플링 필터(resampling filter)를 이용하여 샘플점을 둘러싸고 있는 8개의 복셀들로부터 정확한 스칼라 값을 계산한다. 볼륨 광선 투사법에서는 일반적으로 삼선형 보간법(trilinear interpolation)을 사용하기 때문에 처리 시간이 길고 주 메모리를 임의로 참조한다는 문제점을 가지고 있다. (2) 두 번째로, 해당 샘플점에서의 기울기 벡터를 계산하기 위해서, 샘플점을 둘러싸고 있는 8개의 복셀들에서 기울기 벡터를 계산한다. 가장 단순한 중앙 차분법(central difference)을 사용하더라도 주 메모리에 대한 약 48번의 임의-접근이 필요하다. (3) 세 번째로, 두 번째 절차에서 계산된 8개의 기울기 벡터들을 삼선형 보간법을 이용하여 해당샘플점에서의 기울기 벡터를 구한다. 이렇게 구해진 기울기 벡터로부터 조명 모델을 통하여 원하는 색상을 획득한다.

<18> 상기와 같은 과정을 해당 광선에서의 모든 샘플점에 대하여 수행하며, 이들 연산들을 영상의 모든 화소에 대하여 반복한다. 이들 단계를 가속하는 가장 확실한 방법은 주 메모리를 임의로 접근하는 부분들을 미리 전처리 단계에서 계산하고, 그 결과를 렌더링 단계에서 참조하는 것이다. 본 발명에서는, 전처리 단계에서 계산한 결과들을 저장할 수 있고 렌더링 단계에서 이를 참조할 수 있는 데이터 구조를 생성하는 방법을 제안한다.

<19> 도 2는 본 발명의 일 실시예에 따른 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법을 나타내는 도면이다. 도 2에 도시된 바와 같이, 본 발명에 따른 데이터 구조 생성 방법은, (1) 후보-셀을 결정하는 단계(S10), (2) 결정된 각각의 후보-셀의 천장과 바닥을 가상의 서브-사각형으로 나누는 단계(S20), (3) 나누어진 각각의 서브-사각형에 대하여, 스칼라 값과 기울기 벡터를 계산하는 단계(S30), 및 (4) 계산된 스칼라 값과 기울기 벡터를 데이터 구조에 저장하는 단계(S40)를 포함한다. 단계 S10에서, 임의의 셀이 후보-셀인지 여부는 해당 셀이 투명-셀인지 여부를 확인함으로써 결정할 수 있다. 후보-셀은 최종 영상에 영향을 주는 셀을 의미하는 것으로, 셀을 구성하는 8개의 복셀이 모두 투명한 투명-셀이 아닌 경우에 후보-셀로 결정할 수 있다. 단계 S20에서, 후보-셀의 영상의 변화 정도에 따라서 후보-셀을 나누는 서브-사각형의 개수를 조정할 수 있다. 즉, 영상의 변화가 크다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 증가시키고, 영상의 변화가 작다고 판단된 후보-셀에 대해서는 나누는 서브-사각형의 개수를 감소시킬 수 있다. 후보-셀의 영상의 변화 정도는 후보-셀의 기울기 벡터의 변화 정도를 통해 확인할 수 있다. 이와 관련하여, 추후 도 6을 참조하여 더욱 상세하게 설명하기로 한다. 단계 S30에서, 스칼라 값과 기울기 벡터는 쌍선형 보간법을 이용하여 계산될 수 있다. 단계 S40에서, 후보-셀들의 색인을 인덱스 구조에 저장하여, 데이터 구조를 참조할 수 있다.

<20> 도 3은 본 발명의 일 실시예에 따라서 후보-셀을 서브-사각형으로 나누는 것을 나타내는 도면이다. 8개의 복셀들로 구성된 육면체를 셀(cell)이라 하고, 하나의 셀 내의 8개의 복셀이 모두 투명할 경우 투명-셀(transparent cell)이라 하며, 최종 영상에 영향을 주는 셀을 후보-셀(candidate cell)이라 하는데, 도 3에 도시된 바와 같이, 본 발명에서는 각각의 후보-셀들의 천정과 바닥을 N_{ss}^2개의 가상의 사각형(서브-사각형; sub-square)으로 나눈다. 다시 말해, 본 발명의 방법에 따르면, 1개의 후보-셀은 $2N_{ss}^2$개의 서브-사각형으로 나뉘게 된다. 도 3에 도시된 실시예에서는, 1개의 후보-셀을 32(=2×4^2)개의 서브-사각형으로 나누고 있다.

<21> 1개의 후보-셀을 서브-사각형으로 나누고 나면, 다음 단계로 볼륨 광선 투사법의 색상값 계산 단계를 가속하기 위해 필요한 데이터들을 계산한다. 임의의 샘플점에 대하여, 만약 삼선형 보간된 스칼라 값과 기울기 벡터가 미리 저장되어 있다면 주메모리를 임의로 참조하는 연산을 줄여서 렌더링 속도를 대폭 단축할 수 있다. 앞서 언급한 바와 같이, 이와 같은 방법이 기존에 제안되었지만, 그 방법은 후보-셀의 개수가 많을 경우 필요한 데이터의 양이 지나치게 증가하는 문제점이 있다. 이러한 문제점을 해결하기 위하여, 본 발명에서는 후보-셀을 여러 개의 서브-셀로 나누는 대신에, 후보-셀의 천정과 바닥을 서브-사각형으로 나누는 방법을 사용한다. 이들 서브-사각형에는 쌍선형 보간법(bilinear interpolation)으로 계산된 스칼라 값과 기울기 벡터가 저장된다. 본 발명에서 제안한 방법을 사용할 경우, 후보-셀을 서브-셀로 나누는 기존의 방법과 비교하여, 다음 수학식 1 및 수학식 2에서 확인할 수 있는 바와 같이, $2/N_{ss}$만큼 메모리를 절약할 수 있고, $1/N_{ss}$만큼 전처리 시간을 줄일 수 있다.

수학식 1

<22> $$\frac{2N_{ss}^2}{N_{ss}^3}=\frac{2}{N_{ss}}$$

수학식 2

<23> $$\frac{3t_{lint}\times 2N_{ss}^2}{7t_{lint}\times N_{ss}^3}\approx\frac{1}{N_{ss}}$$

<24> 상기 수학식들에서, t_{lint}는 한 번의 선형 보간을 수행하는 데 소요되는 시간을 나타낸다.

<25> 각각의 서브-사각형들에 대하여, 쌍선형 보간법으로 계산된 스칼라 값과 기울기 벡터는 데이터 구조(Data-Storage)에 저장되는데, 저장된 데이터 구조를 참조하기 위해서는 후보-셀들의 색인을 저장하고 있는 자료 구조가 추가로 필요하다. 이를 인덱스 구조(Index-Storage)라 지칭한다. 인덱스 구조에는, 복셀의 값들이 투명할 경우에는 null 값이 저장되며, 그렇지 않을 경우에는 누적한 색인이 저장된다. 도 4는 본 발명에서 제안한 방법에 따라서, 데이터 구조와 인덱스 구조를 생성하는 과정을 나타내는 도면이다. 도 4에 도시된 바와 같이, 각각의 후보-셀을 참조할 수 있는 인덱스 구조가 생성되며, 이와 동시에 각각의 후보-셀을 서브-사각형들로 나눈 후, 나누어진 각각의 서브-사각형에 대하여 선형 보간법에 의하여 스칼라 값과 기울기 벡터가 계산되어 데이터 구조에 저장되게 된다. 도 4에 도시된 본 발명의 일 실시예에 따른 데이터 구조를 참조하면, 데이터 구조는 한 쌍의 천장(ceiling)과 바닥(floor)에 대하여 함께 생성될 수 있으며, 천장과 바닥에 대한 각각의 스칼라 값 및 천장과 바닥에 대한 각 방향으로의 기울기 벡터가 함께 저장될 수 있다. 도 4에서는, l=0으로 인덱스된 천장과 l=16으로 인덱스된 바닥이 함께 저장되고, 또한 l=1로 인덱스된 천장과 l=17로 인덱스된 바닥이 함께 저장되고 있다.

<26> 이와 같은 방법으로 데이터 구조를 생성할 경우의 메모리 소요량(ϕ)은 다음 수학식 3과 같이 계산될 수 있다.

수학식 3

<27> $$\phi=N_{cc}(4\times 2N_{ss}^2)+4N^3$$

<28> 여기서, N_{cc}는 후보-셀의 총 개수를, N^3은 볼륨데이터의 크기를 각각 나타낸다. 인덱스 구조의 크기는 후보-셀의 개수에 의존적이지만, 일반적으로 4 바이트의 자료구조로 충분하다.

<29> 본 발명에 따라서 전처리 단계에서 생성된 데이터 구조가, 렌더링 단계에서 어떻게 이용될 수 있는지를 도 5를 참조하여 설명하기로 한다. 도 5는 본 발명에 따라서 생성된 데이터 구조를 렌더링 단계에서 이용하는 모습을 나타내는 도면이다. 렌더링 단계에서는, 시점이 고정되면 먼저 영상의 픽셀에서 가상의 광선이 발사된다. 투명한 공간을 도약한 후, 임의의 후보-셀의 특정 샘플점에 도달하게 되면, 도 5에 도시된 바와 같이, 해당 샘플점에서 가장 가까운 천정과 바닥의 서브-사각형을 선택한다. 해당 서브-사각형들의 쌍선형 보간법으로 미리 계산된 스칼라 값들에 대하여, 한 번의 선형 보간을 수행함으로써 정확한 삼선형 보간법으로 계산된 스칼라 값을 구한다. 만약 이 값이 투명하다고 판단될 경우 다음 샘플점으로 이동하며, 불투명할 경우 역시 미리 계산된 쌍선형 보간된 기울기 벡터의 x, y, z 성분들에 대하여 각각 한 번씩의 선형 보간을 수행하여 삼선형 보간된 기울기 벡터를 구한다. 이 기울기 벡터를 이용하여 색상값을 획득한다. 이와 같은 연산은 불투명도가 1.0 이하가 될 때까지 반복하며, 영상의 모든 픽셀에 대하여 수행된다. 이상에서 확인할 수 있는 것처럼, 본 발명에 따라 생성된 데이터 구조를 이용할 경우, 주 메모리를 임의로 참조하는 대부분의 연산들이 미리 계산되어 있으므로, 렌더링 시간이 대폭 단축될 수 있다.

<30> 본 발명에서 제안한 방법이 후보-셀을 서브-셀로 나누는 기존의 방법에 비하여 전처리에 소요되는 시간 및 메모리의 양을 줄일 수 있지만, 후보-셀의 개수가 증가할 경우 역시 많은 메모리가 소요된다는 문제점에 직면할 수 있다. 이와 같은 문제점을 해결하기 위하여, 본 발명에서는 가변 기울기 벡터 조절법을 함께 사용한다. 가변 기울기 벡터 조절법은, 한 개의 후보-셀을 다수 개의 서브-사각형으로 나눔에 있어서 영상의 변화가 심한(즉, 기울기 벡터의 변화가 큰) 후보-셀에 대해서는 서브-사각형의 개수를 증가시키고, 그렇지 않을 경우에는 서브-사각형의 개수를 상대적으로 줄이는 방법이다. 본 발명의 실시예에서는 사용자가 정의한 값에 따라 3단계로 구별하여 차별적으로 서브-사각형의 개수를 할당한다.

<31> 이와 같이 가변 기울기 벡터 조절법을 사용할 경우, 서브-사각형의 개수가 변하게 되므로, 이들을 저장하고 있는 인덱스 구조의 저장 방식도 달라져야 한다. 기존의 방법을 변경 없이 적용할 경우 3개의 인덱스 구조가 필요하게 되는데, 이는 메모리의 소요량을 증가시킨다. 이와 같은 문제점을 해결하기 위하여, 본 발명에서는 기울기 벡터에 따라 서브-사각형의 개수를 3단계로 나눈 후, 이를 기존의 정렬 알고리즘(sorting algorithm)을 이용하여 정렬하고 각각의 단계에서의 총 후보-셀의 개수를 저장하는 방법을 제안한다. 렌더링 단계에서는 현재 인덱스 구조에 저장되어 있는 색인과 각각의 단계에서의 후보-셀의 개수를 더하여 데이터 구조를 참조한다. 이렇게 할 경우, 한 개의 인덱스 구조를 가지고도 효과적으로 가변 기울기 벡터 조절법을 적용할 수 있다. 물론 전처리 과정에서 한 번의 정렬 과정이 필요하지만 전체 볼륨을 한 번 탐색하는 것에 불과하기 때문에 메모리 소요량 증가에 비해서는 무시할만한 수준이다. 도 6은 이상 설명한 내용을 나타내는 도면으로서, 기울기 벡터의 변화 정도에 따라서 서브-사각형의 개수를 차등적으로 나누는 것을 나타내는 도면이다.

<32> 실험 결과

<33> 본 발명에서 제안된 방법은 AMD사의 애슬론 64× 2 4200+ CPU와 2GB의 주메모리가 탑재된 시스템에서 실험되었다. 실험 데이터는 256^3의 크기를 갖는 엔진블록 데이터와 분재 데이터, 및 256× 256× 161 크기의 치아 데이터가 이용되었다. 도 7은 실험에서 사용된 각각의 데이터에 대한 불투명도 전이 함수의 모양을 나타내는 도면으로서, 도 7a는 엔진블록 데이터, 도 7b는 분재 데이터, 도 7c는 치아 데이터에 대한 불투명도 전이 함수의 모양을 나타내고 있다.

<34> 다음 표 1은 본 발명의 방법에 따라서 데이터 구조를 생성하는 데 소요되는 전처리 시간 및 메모리 사용량을, 후보-셀을 서브-셀로 나누는 기존의 방법(서브-셀 방법)의 전처리 시간 및 메모리 사용량과 비교하여 나타내는 표이다. 기존의 서브-셀 방법은 1개의 후보-셀을 64(=4^3)개의 서브-셀로 나누었으며, 본 발명에서 제안한 방법은 가변 기울기 벡터 조절법을 위하여, 기울기 벡터의 한 성분이라도 150 이상일 경우 한 개의 후보-셀을 32(=2× 4^2)개의 서브-사각형으로 나누었으며, 50과 149 사이일 경우 18(=2× 3^2)개, 그 이하일 경우 8(=2× 2^2)개로 나누었다.

표 1

<35>

데이터	엔진블록	분재	치아
총 후보-셀 개수 (복셀)	1,031,929	829,726	1,000,096
후보-셀 결정 시간 (초)	1.31	0.78	0.71

서브-셀 기반 데이터 구조 및 인덱스 구조 생성 시간(초)	30.93	26.84	29.45
총 전처리 시간 (초)	**32.24**	**27.62**	**30.16**
서브-셀 기반 데이터 구조의 메모리 소요량 (MB)	1,057	850	1,024
인덱스 구조의 메모리 소요량 (MB)	34	34	34
총 메모리 소요량 (MB)	**1,091**	**884**	**1,058**

<36>

데이터	엔진블록	분재	치아
총 후보-셀 개수 (복셀)	1,031,929	829,726	1,000,096
후보-셀 결정 시간 (초)	1.31	0.78	0.71
본 발명에서 제안한 데이터 구조 및 인덱스 구조 생성 시간(초)	7.76	6.81	7.87
총 전처리 시간 (초)	**9.07**	**7.59**	**8.58**
본 발명에서 제안한 데이터 구조의 메모리 소요량 (MB)	93	80	100
인덱스 구조의 메모리 소요량 (MB)	34	34	34
총 메모리 소요량 (MB)	**127**	**114**	**134**

<37> 상기 표 1로부터, 본 발명에서 제안한 방법에 따라서 데이터 구조를 생성하는 경우에, 기존의 서브-셀 방법을 따르는 경우와 비교하여, 전처리 시간(약 1/4로 감소) 및 메모리 사용량(약 1/9로 감소)이 대폭 감소한 것을 확인할 수 있다.

<38> 다음 표 2는 본 발명에서 생성된 데이터 구조를 이용할 경우의 렌더링 시간을, 기존의 서브-셀 방법의 렌더링 시간과 비교하여 나타내는 표이다. 본 발명에서 제안한 방법이나 서브-셀 방법은, 모두 색상값 계산의 속도를 가속하는 방법이기 때문에, 빈공간을 도약하기 위한 방법이 별도로 필요하다. 본 실험에서는, 빈공간 도약 기법으로서, 기존의 최대-최소 옥트리 자료구조와 거리-맵 자료구조를 이용하였다.

표 2

<39>

데이터	엔진블록	분재	치아
최대-최소 옥트리 (A)	1.64	1.40	1.80
최대-최소 옥트리 + 서브-셀 방법 (B)	0.70	0.67	0.80
효율 (1-B/A, %)	**57.3**	**52.1**	**55.6**
거리-맵 (C)	1.43	1.20	1.58
거리-맵 + 서브-셀 방법 (D)	0.69	0.59	0.75
효율 (1-D/C, %)	**51.7**	**50.8**	**52.5**

<40>

데이터	엔진블록	분재	치아
최대-최소 옥트리 (E)	1.64	1.40	1.80
최대-최소 옥트리 + 제안한 방법 (F)	0.79	0.76	0.93
효율 (1-F/E, %)	**51.8**	**45.7**	**48.3**
거리-맵 (G)	1.43	1.20	1.58
거리-맵 + 제안한 방법 (H)	0.79	0.69	0.83
효율 (1-H/G, %)	**44.8**	**42.5**	**47.5**

<41> 표 2에서 사용된 영상의 크기는 256×256이며, 빈공간 도약 시간이 포함되어 있다. 상기 표에서 단위는 초이며, 표 2로부터 47%의 속도 향상이 가능함을 확인할 수 있다. 이는 기존의 서브-셀 방법과 비교하여 효율이 6% 정도만 저하된 것으로, 앞서 표 1로부터 확인한 전처리 시간 및 메모리 소요량의 이득에 비하면 미미한 정도라 할 수 있다.

<42> 마지막으로, 도 8을 참조하여 실제 렌더링한 결과 영상을 살펴보기로 한다. 도 8은 본 발명에서 제안한 데이터 구조를 이용한 경우의 렌더링 결과 영상을 나타내는 도면으로서, 좌로부터 각각 엔진블록, 분재, 치아의 렌더링 결과 영상을 나타내고 있다. 결과 영상의 품질을 수치적으로 비교하기 위하여 RMSE(Root Means Square Error) 값을 계산해 보았다. RMSE 값은 각각 3.02, 4.02, 2.86으로 나타났다. 이를 평균하면 3.36의 결과를 얻을 수 있는데, 이는 렌더링 속도 향상에 비하여 매우 작은 값이라고 할 수 있다. 또한 효과적인 비교를 위한 다른 방법으로서 인지 비교법(Perceptual identical comparison method)도 사용하였는데, 각각 0.27%, 0.68%, 0.11%의 값을 나타내었다. 이 값들을 평균하면 0.35%의 비교 결과를 얻을 수 있는데, 만약 이 결과가 0%라면 인간은 구분을 못하는 수준이라는 것을 감안할 때 0.35%의 결과는 렌더링 속도 향상에 비해 만족할만한 수준이라고 할 수 있다.

<43> 이상 설명한 본 발명은 본 발명이 속한 기술분야에서 통상의 지식을 가진 자에 의하여 다양한 변형이나 응용이 가능하며, 본 발명에 따른 기술적 사상의 범위는 아래의 특허청구범위에 의하여 정해져야 할 것이다.

도면의 간단한 설명

<44> 도 1은, 종래 제안된 방법에 따라서, 한 개의 후보-셀을 N_{ss}^3개의 서브-셀들로 나눈 후, 삼선형 보간법으로 계산된 스칼라 값과 기울기 벡터를 저장하는 것을 나타내는 도면.

<45> 도 2는 본 발명의 일 실시예에 따른 볼륨 광선 투사법을 위한 데이터 구조를 생성하는 방법을 나타내는 도면.

<46> 도 3은 본 발명의 일 실시예에 따라서 후보-셀을 서브-사각형으로 나누는 것을 나타내는 도면.

<47> 도 4는 본 발명에서 제안한 방법에 따라서, 데이터 구조와 인덱스 구조를 생성하는 과정을 나타내는 도면.

<48> 도 5는 본 발명에 따라서 생성된 데이터 구조를 렌더링 단계에서 이용하는 모습을 나타내는 도면.

<49> 도 6은 기울기 벡터의 변화 정도에 따라서 서브-사각형의 개수를 차등적으로 나누는 것을 나타내는 도면.

<50> 도 7은 실험에서 사용된 각각의 데이터에 대한 불투명도 전이 함수의 모양을 나타내는 도면.

<51> 도 8은 본 발명에서 제안한 데이터 구조를 이용한 경우의 렌더링 결과 영상을 나타내는 도면.

<52> <도면 중 주요 부분에 대한 부호의 설명>

<53> S10 : 후보-셀을 결정하는 단계

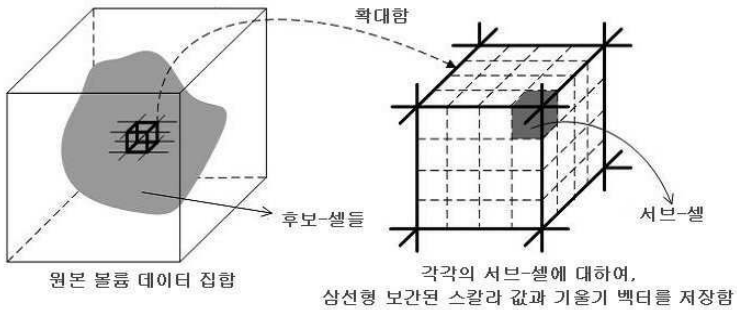
<54> S20 : 각각의 후보-셀의 천장과 바닥을 가상의 서브-사각형(sub-square)으로 나누는 단계

<55> S30 : 각각의 서브-사각형에 대하여, 스칼라 값과 기울기 벡터를 계산하는 단계

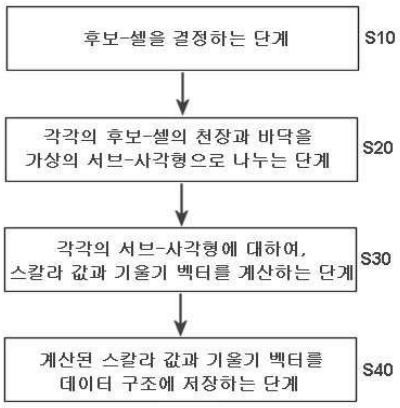
<56> S40 : 계산된 스칼라 값과 기울기 벡터를 데이터 구조에 저장하는 단계

도면

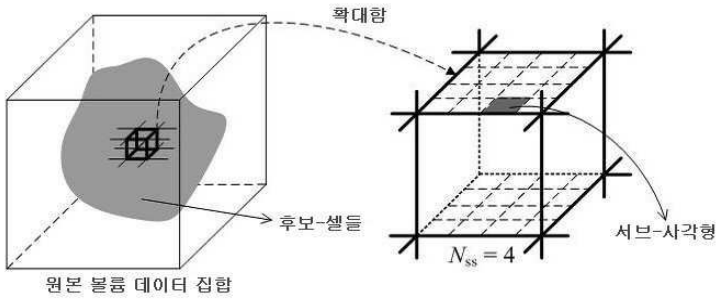
도면1



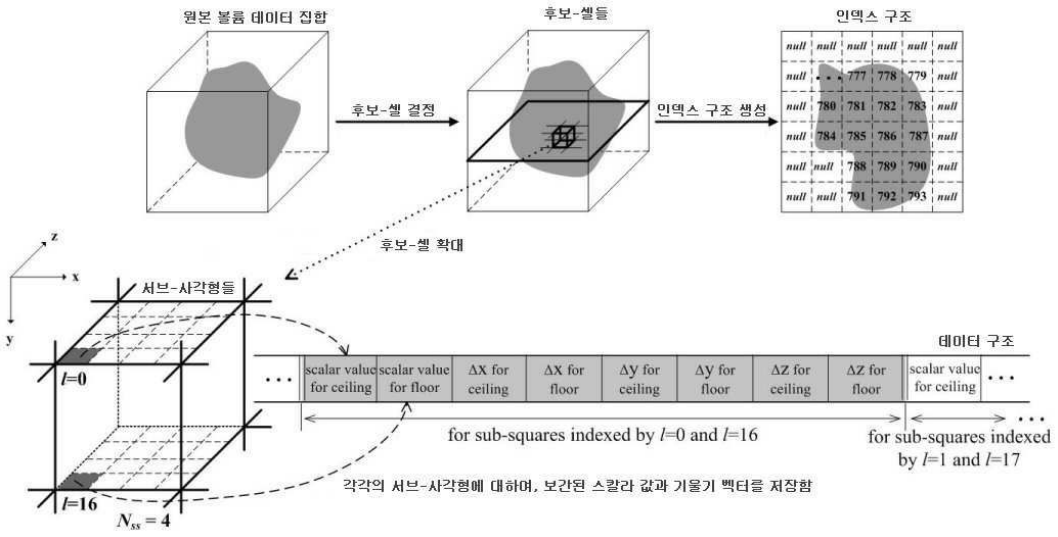
도면2


후보-셀을 결정하는 단계
S10
각각의 후보-셀의 천장과 바닥을 가상의 서브-사각형으로 나누는 단계
S20
각각의 서브-사각형에 대하여, 스칼라 값과 기울기 벡터를 계산하는 단계
S30
계산된 스칼라 값과 기울기 벡터를 데이터 구조에 저장하는 단계
S40

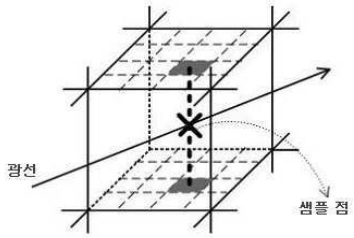
도면3


확대함
후보-셀들
서브-사각형
$N_{ss} = 4$
원본 볼륨 데이터 집합

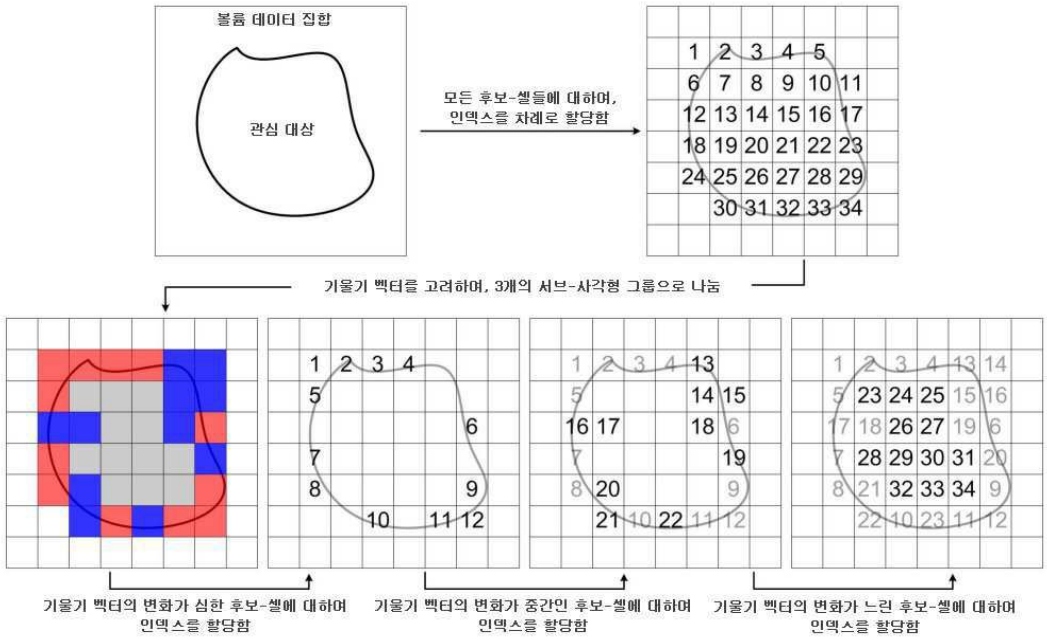
도면4


원본 볼륨 데이터 집합
후보-셀 결정
후보-셀들
인덱스 구조 생성
인덱스 구조
후보-셀 확대
서브-사각형들
l=0
l=16
$N_{ss} = 4$
데이터 구조
scalar value for ceiling
scalar value for floor
ΔX for ceiling
ΔX for floor
Δy for ceiling
Δy for floor
ΔZ for ceiling
ΔZ for floor
scalar value for ceiling
for sub-squares indexed by l=0 and l=16
for sub-squares indexed by l=1 and l=17
각각의 서브-사각형에 대하여, 보간된 스칼라 값과 기울기 벡터를 저장함

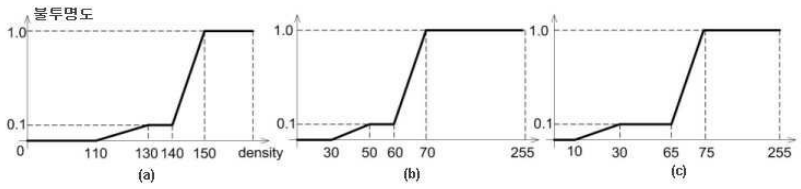
도면5


광선
샘플 점

도면6


볼륨 데이터 집합
관심 대상
모든 후보-셀들에 대하여,
인덱스를 차례로 할당함
기울기 벡터를 고려하여, 3개의 서브-사각형 그룹으로 나눔
기울기 벡터의 변화가 심한 후보-셀에 대하여
인덱스를 할당함
기울기 벡터의 변화가 중간인 후보-셀에 대하여
인덱스를 할당함
기울기 벡터의 변화가 느린 후보-셀에 대하여
인덱스를 할당함

도면7


불투명도
1.0
0.1
0
110
130 140
150
density
(a)
1.0
0.1
30
50
60
70
255
(b)
1.0
0.1
10
30
65
75
255
(c)

도면8

