



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년10월15일

(11) 등록번호 10-1560283

(24) 등록일자 2015년10월07일

(51) 국제특허분류(Int. Cl.)

G06T 15/00 (2006.01)

(21) 출원번호 10-2014-0049254

(22) 출원일자 2014년04월24일

심사청구일자 2014년04월24일

(56) 선행기술조사문헌

KR1020120106930 A*

KR1020110100795 A

KR1020120019720 A

KR1020100128493 A

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

세종대학교산학협력단

서울특별시 광진구 능동로 209 (군자동, 세종대학교)

주식회사 실리콘아츠

서울특별시 송파구 오금로11길 55, 213호, 214호, 215호, 216호, 217호, 218호(방이동, 현대주상복합빌딩)

(72) 발명자

박우찬

서울 종로구 통일로 246-20, 108동 1604호 (무악동, 무악현대아파트)

권혁주

서울특별시 송파구 석촌호수로 169, 131동 1101호 (잠실동, 레이크 팰리스)

(뒷면에 계속)

(74) 대리인

정부연

전체 청구항 수 : 총 19 항

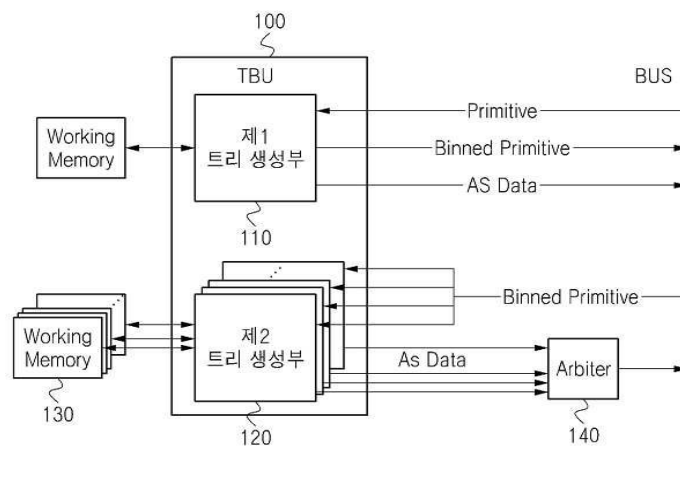
심사관 : 조우연

(54) 발명의 명칭 3차원 영상 생성 방법, 이를 수행하는 3차원 영상 생성 장치 및 이를 저장하는 기록매체

(57) 요약

(a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하는 단계 및 (b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 단계를 포함하고, 상기 (a) 단계는 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하는 단계를 포함하는 3차원 영상 생성 방법이다.

대표도 - 도1



(72) 발명자

허진석

서울특별시 강동구 풍성로54길 40-13 (성내동)
102호

김우현

서울특별시 광진구 동일로42길 10 (군자동)201호

이 발명을 지원한 국가연구개발사업

과제고유번호 10043633

부처명 지식경제부

연구관리전문기관 한국산업기술평가관리원

연구사업명 부품소재산업경쟁력향상사업

연구과제명 100M ray/s급 스마트 TV/게임기용 실시간 레이트레이싱 GPU칩 개발

기 여 율 1/1

주관기관 (주)실리콘아츠

연구기간 2012.12.01 ~ 2015.11.30

명세서

청구범위

청구항 1

(a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하는 단계; 및

(b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 단계를 포함하고,

상기 (a) 단계는 해당 서브 공간이 상기 KD-tree(K-Dimension tree)의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하고 상기 서브 공간(KD-tree 노드)의 상기 원시 데이터 개수를 기초로 상기 서브 공간의 분할 방식을 결정하는 단계를 포함하는 KD-tree 생성 방법.

청구항 2

제1항에 있어서, 상기 임계치는

상기 해당 서브 공간을 정렬 방식으로 분할할 때 요구되는 내부 메모리의 크기를 기초로 결정되는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 3

제1항에 있어서, 상기 내부 노드는

상기 주어진 공간의 분할 면과 분할 면에 의해 분할된 복수의 서브공간으로 구성되는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 4

제1항에 있어서, 상기 (a) 단계는

상기 원시 데이터의 개수가 상기 임계치를 초과한다면 상기 해당 서브 공간을 비닝(binning) 방식으로 분할하여 상기 복수의 서브 공간들을 생성하는 단계를 더 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 5

제1항에 있어서, 상기 (a) 단계는

상기 원시 데이터의 개수가 상기 임계치를 초과하지 않는다면 상기 해당 서브 공간을 정렬(sorting) 방식으로 분할하여 복수의 서브 공간들을 생성하는 단계를 더 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 6

제5항에 있어서, 상기 (a) 단계는

상기 해당 서브 공간에 대한 정렬 방식의 분할을 병렬적으로 수행하여 상기 복수의 서브 공간을 생성하는 단계를 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 7

제1항에 있어서, 상기 (a) 단계는

상기 복수의 서브 공간들의 분할 순서를 상기 KD-tree의 방문 순서로 변환하여 상기 복수의 서브 공간들을 메모리에 저장하는 단계를 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 8

제7항에 있어서, 상기 복수의 서브 공간들을 메모리에 저장하는 단계는

상기 주어진 공간의 분할을 통해 생성된 노드의 주소 및 노드의 데이터를 입력 받는 단계; 및

상기 노드의 데이터가 저장될 버스트 노드 엔트리(burst node entry)의 위치를 탐색하는 단계를 포함하고,

상기 버스트 노드 엔트리의 위치를 탐색하는 단계는 노드 주소와 버스트 주소를 비교하는 단계를 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 9

제7항에 있어서, 상기 복수의 서브 공간들을 메모리에 저장하는 단계는

노드 데이터가 저장될 공간이 없는 버스트 노드 엔트리를 탐색하는 단계를 포함하는 것을 특징으로 하는 KD-tree 생성 방법.

청구항 10

(a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하고,

(b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 트리 생성 유닛을 포함하고,

상기 트리 생성 유닛은 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하고 상기 서브 공간(KD-tree 노드)의 상기 원시 데이터 개수를 기초로 상기 서브 공간의 분할 방식을 결정하는 단계를 포함하는 KD-tree 생성 장치.

청구항 11

제10항에 있어서, 상기 임계치는

상기 해당 서브 공간을 정렬 방식으로 분할할 때 요구되는 내부 메모리의 크기를 기초로 결정되는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 12

제10항에 있어서, 상기 내부 노드는

상기 주어진 공간의 분할 면과 분할 면에 의해 분할된 복수의 서브공간으로 구성되는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 13

제10항에 있어서, 상기 트리 생성 유닛은

상기 원시 데이터의 개수가 상기 임계치를 초과한다면 상기 해당 서브 공간을 비닝(binning) 방식으로 분할하여 상기 복수의 서브 공간들을 생성하는 제1 트리 생성부를 포함하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 14

제10항에 있어서, 상기 트리 생성 유닛은

상기 원시 데이터의 개수가 상기 임계치를 초과하지 않는다면 상기 해당 서브 공간을 정렬(sorting) 방식으로 분할하여 복수의 서브 공간들을 생성하는 적어도 하나의 제2 트리 생성부를 포함하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 15

제14항에 있어서, 제2 트리 생성부는

상기 해당 서브 공간에 대한 정렬 방식의 분할을 병렬적으로 수행하여 상기 복수의 서브 공간을 생성하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 16

제10항에 있어서, 상기 트리 생성 유닛은

상기 복수의 서브 공간들의 분할 순서를 상기 KD-tree의 방문 순서로 변환하여 상기 복수의 서브 공간들을 메모리에 저장하는 노드 스케줄러를 포함하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 17

제16항에 있어서, 상기 노드 스케줄러는

상기 주어진 공간의 분할을 통해 생성된 노드의 주소 및 노드의 데이터를 입력받고 상기 노드의 데이터가 저장될 버스트 노드 엔트리(burst node entry)의 위치를 탐색하는 푸시 엔트리 선택부를 포함하고,

상기 푸시 엔트리 선택부는 노드 주소와 버스트 주소를 비교하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 18

제16항에 있어서, 상기 노드 스케줄러는

노드 데이터가 저장될 공간이 없는 버스트 노드 엔트리를 탐색하는 팝 엔트리 선택부를 포함하는 것을 특징으로 하는 KD-tree 생성 장치.

청구항 19

(a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하는 기능; 및

(b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 기능을 포함하고,

상기 (a) 단계는 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하고 상기 서브 공간(KD-tree 노드)의 상기 원시 데이터 개수를 기초로

상기 서브 공간의 분할 방식을 결정하는 기능을 포함하는 KD-Tree 생성 방법에 관한 컴퓨터 프로그램을 기록한 기록매체.

발명의 설명

기술 분야

[0001] 본 발명은 3차원 영상 생성 기술에 관한 것으로, 보다 상세하게는, 3차원 영상에서 필요한 공간 분할을 효율적으로 수행할 수 있는 3차원 영상 생성 방법, 3차원 영상 생성 장치 및 이를 저장하는 기록매체에 관한 것이다.

배경 기술

[0002] 일반적으로, KD-tree(k-dimension tree)는 3차원 영상 생성시 사용되고 k-차원의 공간에 포인트들(즉, 데이터)을 정리하기 위한 공간-분할 데이터 구조를 가지는 k-차원 트리인 다차원 검색 트리로 정의될 수 있다. KD-tree는 다차원의 탐색 키(예를 들어, 범위 탐색 및 근접 탐색)를 이용하여 탐색을 하는 경우에 유용하게 사용될 수 있고 KD-tree는 BSP(binary space partitioning) 트리의 한 종류에 해당한다.

[0003] 한국공개특허 제10-2011-0124834호는 하드웨어를 이용하여 KD-tree를 생성하는 방법에 관한 것으로, SAH(Surface area heuristic) 방식으로 노드의 코스트가 최소가 되는 위치를 찾아 이를 기준으로 KD-tree를 생성하는 방법을 개시한다. 이러한 KD-tree 생성 방법은 리스트 데이터의 인덱스를 참조하여 메모리로부터 읽어 들인 트라이앵글 데이터에 기초하여 SAH 방식으로 현재 노드의 코스트가 최소가 되는 위치를 찾아 리스트 데이터의 좌우를 분류하고, 현재 노드의 코스트가 최소가 되는 위치를 찾지 못한 경우에는 리프노드를 생성할 수 있다.

[0004] 한국공개특허 제10-2012-0090543호는 복셀의 가시성에 기반하여 KD-tree를 구축하기 위한 방법을 개시한다. 이러한 KD-tree를 구축하는 방법은 복셀의 가시성에 기반한 비용 함수가 KD-tree를 구성하기 위해 사용되고 비용 함수를 최소화 하는 평면을 사용하여 복셀을 2개의 부-복셀들로 분할함으로써 KD-tree가 구성될 수 있다.

선행기술문헌

특허문헌

[0005] (특허문헌 0001) 한국공개특허 제10-2011-0124834호
(특허문헌 0002) 한국공개특허 제10-2012-0090543호

발명의 내용

해결하려는 과제

[0006] 본 발명의 일 실시예는 상위 노드는 비닝(binning) 방식으로 생성하고 하위 노드는 정렬(sorting) 방식으로 병렬적으로 생성하여 트리의 질을 유지하면서 고속으로 트리를 생성할 수 있는 3차원 영상 생성 방법을 제공하고자 한다.

[0007] 본 발명의 일 실시예는 원시데이터의 개수를 기준으로 상위 노드 및 하위 노드를 구분하고, 원시데이터의 개수가 정렬방식이 수행될 수 있는 개수인 경우에는 정렬 방식을 이용하고 그렇지 않은 경우에는 비닝 방식을 이용하여 트리를 생성하는 것을 통해 트리의 질을 유지하면서 고속으로 트리를 생성할 수 있는 3차원 영상 생성 방법을 제공하고자 한다.

[0008] 내부 메모리에 트리를 생성하는데 필요한 모든 데이터를 저장하여 트리 생성 시 외부 메모리로 접근하는 경우를 줄일 수 있는 3차원 영상 생성 방법을 제공하고자 한다.

[0009] 본 발명의 일 실시예는 노드의 탐색 순서대로 노드를 정렬하여 메모리에 저장하는 것을 통해 버스트 접근이 가능하게 하고 이를 통해 메모리 접근 효과를 최대로 얻을 수 있는 3차원 영상 생성 방법을 제공하고자 한다.

과제의 해결 수단

- [0010] 실시예들 중에서, 3차원 영상 생성 방법은 (a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하는 단계 및 (b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 단계를 포함하고, 상기 (a) 단계는 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하는 단계를 포함한다.
- [0011] 일 실시예에서, 상기 임계치는 상기 해당 서브 공간을 정렬 방식으로 분할할 때 요구되는 내부 메모리의 크기를 기초로 결정될 수 있다.
- [0012] 일 실시예에서, 상기 내부 노드는 상기 주어진 공간의 분할 면과 분할 면에 의해 분할된 복수의 서브공간으로 구성될 수 있다.
- [0013] 일 실시예에서, 상기 (a) 단계는 상기 원시 데이터의 개수가 상기 임계치를 초과한다면 상기 해당 서브 공간을 비닝(binning) 방식으로 분할하여 상기 복수의 서브 공간들을 생성하는 단계를 더 포함할 수 있다.
- [0014] 일 실시예에서, 상기 (a) 단계는 상기 원시 데이터의 개수가 상기 임계치를 초과하지 않는다면 상기 해당 서브 공간을 정렬(sorting) 방식으로 분할하여 복수의 서브 공간들을 생성하는 단계를 더 포함할 수 있다.
- [0015] 일 실시예에서, 상기 (a) 단계는 상기 해당 서브 공간에 대한 정렬 방식의 분할을 병렬적으로 수행하여 상기 복수의 서브 공간을 생성하는 단계를 포함할 수 있다.
- [0016] 일 실시예에서, 상기 (a) 단계는 상기 복수의 서브 공간들의 분할 순서를 상기 KD-tree의 방문 순서로 변환하여 상기 복수의 서브 공간들을 메모리에 저장하는 단계를 포함할 수 있다.
- [0017] 일 실시예에서, 상기 복수의 서브 공간들을 메모리에 저장하는 단계는 상기 주어진 공간의 분할을 통해 생성된 노드의 주소 및 노드의 데이터를 입력받는 단계 및 상기 노드의 데이터가 저장될 버스트 노드 엔트리(burst node entry)의 위치를 탐색하는 단계를 포함하고, 상기 버스트 노드 엔트리의 위치를 탐색하는 단계는 노드 주소와 버스트 주소를 비교하는 단계를 포함할 수 있다.
- [0018] 일 실시예에서, 상기 복수의 서브 공간들을 메모리에 저장하는 단계는 노드 데이터가 저장될 공간이 없는 버스트 노드 엔트리를 탐색하는 단계를 포함할 수 있다.
- [0019] 실시예들 중에서, 3차원 영상 생성 장치는 (a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하고, (b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 트리 생성 유닛을 포함하고, 상기 트리 생성 유닛은 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크한다.
- [0020] 일 실시예에서, 상기 임계치는 상기 해당 서브 공간을 정렬 방식으로 분할할 때 요구되는 내부 메모리의 크기를 기초로 결정될 수 있다.
- [0021] 일 실시예에서, 상기 내부 노드는 상기 주어진 공간의 분할 면과 분할 면에 의해 분할된 복수의 서브공간으로 구성될 수 있다.
- [0022] 일 실시예에서, 상기 트리 생성 유닛은 상기 원시 데이터의 개수가 상기 임계치를 초과한다면 상기 해당 서브 공간을 비닝(binning) 방식으로 분할하여 상기 복수의 서브 공간들을 생성하는 제1 트리 생성부를 포함할 수 있다.
- [0023] 일 실시예에서, 상기 트리 생성 유닛은 상기 원시 데이터의 개수가 상기 임계치를 초과하지 않는다면 상기 해당 서브 공간을 정렬(sorting) 방식으로 분할하여 복수의 서브 공간들을 생성하는 적어도 하나의 제2 트리 생성부를 포함할 수 있다.
- [0024] 일 실시예에서, 제2 트리 생성부는 상기 해당 서브 공간에 대한 정렬 방식의 분할을 병렬적으로 수행하여 상기 복수의 서브 공간을 생성할 수 있다.
- [0025] 일 실시예에서, 상기 트리 생성 유닛은 상기 복수의 서브 공간들의 분할 순서를 상기 KD-tree의 방문 순서로 변

환하여 상기 복수의 서브 공간들을 메모리에 저장하는 노드 스케줄러를 포함할 수 있다.

[0026] 일 실시예에서, 상기 노드 스케줄러는 상기 주어진 공간의 분할을 통해 생성된 노드의 주소 및 노드의 데이터를 입력받고 상기 노드의 데이터가 저장될 버스트 노드 엔트리(burst node entry)의 위치를 탐색하는 푸시 엔트리 선택부를 포함하고, 상기 푸시 엔트리 선택부는 노드 주소와 버스트 주소를 비교할 수 있다.

[0027] 일 실시예에서, 상기 노드 스케줄러는 노드 데이터가 저장될 공간이 없는 버스트 노드 엔트리를 탐색하는 팝 엔트리 선택부를 포함할 수 있다.

[0028] 실시예들 중에서, 3차원 영상 생성 방법에 관한 컴퓨터 프로그램을 기록한 기록매체는 (a) 주어진 공간의 분할 방식을 결정하여 상기 주어진 공간을 복수의 서브 공간들(각각은 KD-tree의 노드를 형성함)로 분할하는 기능 및 (b) 상기 복수의 서브 공간들 각각을 상기 주어진 공간으로 설정하여 상기 단계 (a)를 반복하는 기능을 포함하고, 상기 (a) 단계는 해당 서브 공간이 상기 KD-tree의 내부 노드에 해당하면 상기 내부 노드에 있는 원시 데이터의 개수가 임계치를 초과하는지 여부를 체크하는 기능을 포함하는 3차원 영상 생성 방법에 관한 컴퓨터 프로그램을 기록한다.

발명의 효과

[0029] 본 발명의 일 실시예에 따른 3차원 영상 생성 방법은 상위 노드는 비닝(binning) 방식으로 생성하고 하위 노드는 정렬(sorting) 방식으로 병렬로 생성하여 트리의 질을 유지하면서 고속으로 트리를 생성할 수 있다.

[0030] 본 발명의 일 실시예에 따른 3차원 영상 생성 방법은 원시데이터의 개수를 기준으로 상위 노드 및 하위 노드를 구분하고, 원시데이터의 개수가 정렬방식이 수행될 수 있는 개수인 경우에는 정렬 방식을 이용하고 그렇지 않은 경우에는 비닝 방식을 이용하여 트리를 생성하는 것을 통해 트리의 질을 유지하면서 고속으로 트리를 생성할 수 있다.

[0031] 본 발명의 일 실시예에 따른 3차원 영상 생성 방법은 내부 메모리에 트리를 생성하는데 필요한 모든 데이터를 저장하여 트리 생성시 외부 메모리로 접근하는 경우를 줄일 수 있다.

[0032] 본 발명의 일 실시예에 따른 3차원 영상 생성 방법은 노드의 탐색 순서대로 노드를 정렬하여 메모리에 저장하는 것을 통해 버스트 접근이 가능하게 하고 이를 통해 메모리 접근 효과를 최대로 얻을 수 있다.

도면의 간단한 설명

[0033] 도 1은 본 발명의 일 실시예에 따른 트리 생성 유닛을 설명하는 블록도이다.

도 2(a)는 한 축에 대해서 비닝 방식으로 생성된 분할 면의 일 실시예이다.

도 2(b)는 한 축에 대해서 정렬 방식으로 생성된 분할 면의 일 실시예이다.

도 3은 도 1의 트리 생성 유닛을 포함하는 레이 트레이싱 장치를 설명하는 블록도이다.

도 4은 본 발명의 일 실시예에 따른 작업 메모리를 나타내는 블록도이다.

도 5(a)는 본 발명의 일 실시예에 따른 노드 스케줄러 및 노드 메모리를 나타내는 블록도이다.

도 5(b)는 노드 스케줄러의 실시예를 나타내는 블록도이다.

도 6는 레이 트레이싱의 과정을 설명하는 블록도이다.

도 7은 개시된 기술에서 사용되는 가속 구조와 기하학적 데이터를 설명하는 블록이다.

도 8은 본 발명의 일 실시예에 따른 트리 생성 유닛을 설명하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0034] 본 발명에 관한 설명은 구조적 내지 기능적 설명을 위한 실시예에 불과하므로, 본 발명의 권리범위는 본문에 설명된 실시예에 의하여 제한되는 것으로 해석되어서는 아니 된다. 즉, 실시예는 다양한 변경이 가능하고 여러 가지 형태를 가질 수 있으므로 본 발명의 권리범위는 기술적 사상을 실현할 수 있는 균등물들을 포함하는 것으로

이해되어야 한다. 또한, 본 발명에서 제시된 목적 또는 효과는 특정 실시예가 이를 전부 포함하여야 한다거나 그러한 효과만을 포함하여야 한다는 의미는 아니므로, 본 발명의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.

[0035] 한편, 본 출원에서 서술되는 용어의 의미는 다음과 같이 이해되어야 할 것이다.

[0036] "제1", "제2" 등의 용어는 하나의 구성요소를 다른 구성요소로부터 구별하기 위한 것으로, 이들 용어들에 의해 권리범위가 한정되어서는 아니 된다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다.

[0037] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결될 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다고 언급된 때에는 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 한편, 구성요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0038] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한 복수의 표현을 포함하는 것으로 이해되어야 하고, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이며, 하나 또는 그 이상의 다른 특징이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0039] 각 단계들에 있어 식별부호(예를 들어, a, b, c 등)는 설명의 편의를 위하여 사용되는 것으로 식별부호는 각 단계들의 순서를 설명하는 것이 아니며, 각 단계들은 문맥상 명백하게 특정 순서를 기재하지 않는 이상 명기된 순서와 다르게 일어날 수 있다. 즉, 각 단계들은 명기된 순서와 동일하게 일어날 수도 있고 실질적으로 동시에 수행될 수도 있으며 반대의 순서대로 수행될 수도 있다.

[0040] 본 발명은 컴퓨터가 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현될 수 있고, 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광 데이터 저장 장치 등이 있으며, 또한, 캐리어 웨이브(예를 들어 인터넷을 통한 전송)의 형태로 구현되는 것도 포함한다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

[0041] 여기서 사용되는 모든 용어들은 다르게 정의되지 않는 한, 본 발명이 속하는 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한 이상적이거나 과도하게 형식적인 의미를 지니는 것으로 해석될 수 없다.

[0042] 도 1은 본 발명의 일 실시예에 따른 트리 생성 유닛을 설명하는 블록도이다.

[0043] 도 1을 참조하면, 트리 생성 유닛(Tree Build Unit)(100)은 제1 트리 생성부(110) 및 적어도 하나의 제2 트리 생성부(120)를 포함할 수 있다.

[0044] 제1 트리 생성부(110)에서는 비닝(binning) 방식을 이용하여 트리를 생성한다. 여기에서, 비닝 방식은 3개의 축(x, y, z)에 대해서 미리 정해진 수만큼 바운딩 박스(bounding box) 공간을 분할하여 분할 면에 대해서만 SAH(Surface area heuristic) 값의 계산을 수행한다. 여기에서, SAH는 광선 추적(ray-tracing)을 위한 가속 구조를 생성하는 방법으로서, 임의의 광선과 부딪치는 원시데이터(primitive)를 찾기 위한 값들(예를 들어, 노드 방문 횟수, 원시데이터와 교차되는지 여부를 계산하는 횟수 등)을 계산하여 계산 결과 중 가장 좋은 값을 가지는 분할 면을 기준으로 공간을 분할하는 방법이다.

[0045] 적어도 하나의 제2 트리 생성부(120)에서는 정렬(sorting) 방식을 이용하여 트리를 생성한다. 여기에서, 정렬 방식은 3개의 축(x, y, z)에 대하여 전체 원시데이터의 시작 값 및 끝 값을 추출하고 3개 축을 시작 값 및 끝 값으로 분할하여 6개의 리스트 데이터를 생성한다. 여기에서, 리스트 데이터는 원시데이터의 인덱스로 구성된 배열이다. 리스트 데이터를 생성한 후에 각 리스트 데이터에 대하여 정렬을 하고 정렬된 원시데이터의 시작 값 및 끝 값에서 분할 면을 생성한다. 다음으로, 생성된 분할 면에서 SAH 값을 계산하여 가장 좋은 값을 가지는 분

할 면을 기준으로 하여 공간을 분할하는 방법이다.

- [0046] 보다 구체적으로, 적어도 하나의 제2 트리 생성부(120)는 제1 트리 생성부(110)를 통하여 원시 데이터 정보를 얻고 복수의 제2 트리 생성부(120)가 병렬로 동작할 수 있으며, 출력은 중재기(arbiter)(140)와 연결되어 적어도 하나의 제2 트리 생성부(120)로부터 입력받은 노드 및 리스트 데이터들(AS 데이터)을 외부 메모리(External Memory)(340)에 저장한다.
- [0047] 제1 트리 생성부(110) 및 적어도 하나의 제2 트리 생성부(120)에서 트리가 생성될 때 내부에 있는 작업 메모리(Working memory)(130)가 사용된다. 작업 메모리(130)에 대해서는 도 4을 참조하여 설명한다.
- [0048] 도 2(a)는 한 축에 대해서 비닝 방식으로 생성된 분할 면의 일 실시예이다.
- [0049] 도 2(a)를 참조하면, 바운딩 박스(210a)는 분할 면(220a) 및 원시데이터(230a)를 포함할 수 있다. 일 실시예에서, 분할 면(220a)은 5개의 분할 면들로 미리 정해져 있고, 6개의 원시데이터(230a)에 대하여 각 분할 면(220a)에서의 SAH 값을 계산할 수 있다. 계산 결과, 가장 좋은 값을 가지는 분할 면을 선택하여 바운딩 박스(210a)를 분할할 수 있다.
- [0050] 도 2(b)는 한 축에 대해서 정렬 방식으로 생성된 분할 면의 일 실시예이다.
- [0051] 도 2(b)를 참조하면, 바운딩 박스(210b)는 분할 면(220b) 및 원시데이터(230b)를 포함할 수 있다. 일 실시예에서, 분할 면(220b)은 원시데이터(230b)의 정렬된 시작과 끝 값에서 생성되고, 원시데이터(230b)에 대하여 각 분할 면(220b)에서의 SAH 값을 계산할 수 있다. 계산 결과, 가장 좋은 값을 가지는 분할 면을 선택하여 바운딩 박스(210b)를 분할할 수 있다.
- [0052] 도 3은 도 1의 트리 생성 유닛을 포함하는 레이 트레이싱 장치를 설명하는 블록도이다.
- [0053] 도 3을 참조하면, 레이 트레이싱 장치(300)는 CPU(Central Processing Unit)(310), 시스템 메모리(System Memory)(320), DRTX(Dynamic Ray Tracing Accelerator)(330) 및 외부 메모리(External Memory)(340)를 포함할 수 있다.
- [0054] CPU(310)는 3차원 어플리케이션을 처리할 수 있으며, 3차원 게임 엔진 등의 어플리케이션(311), API(Application Programming Interface)(312) 및 씬 매니저(Scene Manager)(313) 중 적어도 하나를 포함할 수 있다.
- [0055] 시스템 메모리(320)는 3차원 어플리케이션에 필요한 그래픽 데이터 정보를 저장할 수 있으며, 원시 정적 씬(Primitive Static Scene)을 저장하는 PSS 영역(321), 원시 동적 씬(Primitive Dynamic Scene)을 저장하는 PDS 영역(322) 및 텍스처 매핑을 위한 밍맵(MIP-MAP)을 저장하는 텍스처 맵(Texture Map) 영역(323)을 포함할 수 있다. 여기에서, 원시 고정 씬은 전체 프레임에 대하여 변하지 않는 원시 데이터 정보이기 때문에 트리가 생성될 때 한번만 필요하므로, 3차원 어플리케이션 구동 시 트리 생성 결과인 고정 가속 구조는 DRTX(330)의 외부 메모리(340)로 보내진다. 또한, 동적 씬은 프레임마다 원시 데이터 정보가 변하기 때문에 매번 트리 생성 유닛(100)을 통해 트리를 생성하고 생성 결과인 동적 가속 구조는 DRTX(330)의 외부 메모리(340)로 보내진다.
- [0056] DRTX(330)는 도 1의 트리 생성 유닛(Tree Build Unit)(100)을 포함하며, 버스 인터페이스 유닛(Bus Interface Unit)(331), 프리미티브(Primitive)(332), 가속 구조 데이터(AS Data)(333), 작업 메모리(Working Memory)(130), 가속 구조 캐시(AS Cache)(335), 텍스처 캐시(Texture Cache)(336), 컬러 결과 버퍼(Color Result Buffer)(337), 스택 메모리(Stack Memory)(338) 및 광선 추적 유닛(Ray Tracing Unit)(339)를 더 포함할 수 있다. 일 실시예에서, 그래픽 데이터 정보를 기초로 공간 분할 구조체를 구축하고, 생성된 공간 분할 구조체를 기초로 레이 트레이싱을 수행하며, 수행된 레이 트레이싱의 결과를 CPU(310)로 전송하고, 레이 트레이싱 속도를 모니터링하여 그래픽 데이터 정보에 대한 공간 분할 구조체를 재구축할 수 있다.
- [0057] 외부 메모리(340)는 DRTX(330)에서 처리되는 정보를 임시로 저장할 수 있으며, 기하학적 정보 저장 영역(341), 정적 씬 가속 구조 저장 영역(342), 동적 씬 가속 구조 저장 영역(343), 텍스처 맵 저장 영역(344) 및 컬러 정보 저장 영역(345)을 포함할 수 있다.

- [0058] 도 4는 본 발명의 일 실시예에 따른 작업 메모리를 나타내는 블록도이다.
- [0059] 도 4를 참조하면 작업 메모리(Working memory)(130)는 원시 데이터(Primitive data)(410), 원시 데이터 인덱스(Primitive index)(420), 임시 공간(Temporary space)(430) 및 가속 구조 데이터(Acceleration Structure data)(440)가 저장되는 메모리들을 포함할 수 있다.
- [0060] 원시 데이터(410) 메모리에는 적어도 하나의 제2 트리 생성부(120)로부터 받은 원시 데이터의 x, y, z 좌표 정보가 저장되고, 원시 데이터 인덱스(420)메모리에는 원시 데이터를 정렬하는 과정에서 정렬된 순서대로 원시 데이터들의 인덱스 번호가 저장된다. 또한, 임시 공간(430) 메모리에는 분할 면을 기준으로 분류를 수행하면서 좌우로 분할된 원시 데이터의 인덱스 번호가 임시로 저장되고, 가속 구조 데이터(440) 메모리에는 적어도 하나의 제2 트리 생성부를 수행하면서 생성된 노드 및 리스트 데이터가 저장된다. 또한, 작업 메모리(130)는 내부 메모리에 해당한다.
- [0061]
- [0062] 도 5(a)는 본 발명의 일 실시예에 따른 노드 스케줄러 및 노드 메모리를 나타내는 블록도이다.
- [0063] 도 5(a)를 참조하면 노드 스케줄러(Node scheduler)(510)는 노드 데이터 매니저(Node Data Manager)(511), 푸시 엔트리 선택부(Push Entry Selector)(512) 및 팝 엔트리 선택부(Pop Entry Selector)(513)를 포함할 수 있고 노드 메모리(520)는 복수의 데이터 배열 필드(data array field)(521)를 포함할 수 있다. 버스트 노드 엔트리(530)는 노드 데이터 매니저(511)의 존재필드, 풀 필드 및 버스트 필드 및 데이터 배열 필드(521)를 포함할 수 있다.
- [0064] 노드 데이터 매니저(511)는 노드 메모리에 존재하는 데이터 배열 필드(521)들의 정보를 관리하는 작업을 수행하며 다수의 존재 필드(exist field), 풀 필드(full field) 및 버스트 필드(burst field)로 구성된다. 여기에서, 존재 필드는 데이터 배열 필드(521) 안에 노드 데이터가 존재하는지 여부에 대한 정보를 가지고, 풀 필드는 데이터 배열 필드(521)에 데이터를 더 저장할 수 없는지에 대한 정보를 가지고, 버스트 필드는 노드 데이터가 버스트 접근을 할 외부 메모리주소를 가진다.
- [0065] 푸시 엔트리 선택부(512)는 제1 및 제2 트리 생성부(110 및 120)에서 생성된 노드 데이터가 저장될 버스트 노드 엔트리(burst node entry)(530)를 선택하고, 노드 스케줄러(510)의 외부로부터 노드 데이터(node data) 및 노드 주소(node address)에 대한 정보를 받는다. 여기에서, 노드 주소는 노드 데이터가 노드의 탐색 순서대로 저장될 외부 메모리 주소를 의미한다. 또한, 푸시 엔트리 선택부(512)는 노드 스케줄러(510) 내부에 있는 노드 데이터 매니저(511)로부터 각 버스트 노드 엔트리(530)의 정보를 입력받는다. 그리고 나서, 푸시 엔트리 선택부(512)는 입력된 데이터를 기반으로 노드 데이터가 저장될 버스트 노드 엔트리(530)의 위치를 결정하고, 위치가 선택되면 푸시 엔트리 선택부(512)는 노드 주소의 데이터 오프셋(data offset)을 확인한 후 해당 버스트 노드 엔트리(530)의 데이터 배열 필드(521)에 노드 데이터를 저장한다. 여기에서, 노드 주소는 버스트 주소와 데이터 주소를 포함하고, 데이터 오프셋은 데이터 배열 필드(521)에서 노드 데이터가 저장될 위치정보를 갖는다.
- [0066] 팝 엔트리 선택부(513)는 선택된 버스트 노드 엔트리(530)의 데이터 배열 필드(521)를 외부로 출력시킨다. 보다 구체적으로, 팝 엔트리 선택부(513)는 노드 데이터 매니저(511)로부터 모든 버스트 노드 엔트리(530)의 풀 필드에 있는 정보를 받고, 유효한 정보가 존재하는지 확인한다. 유효한 정보가 존재할 경우, 해당 버스트 노드 엔트리(530)의 버스트 필드와 데이터 배열 필드(521)를 외부로 출력시킨다.
- [0067] 데이터 배열 필드(521)은 복수의 노드 데이터를 저장할 수 있는 메모리 공간으로서 외부 메모리(340)의 버스트 접근 사이즈(burst access size)와 동일한 크기를 가진다. 또한, 노드 데이터 매니저(511)의 존재 필드, 풀 필드 및 버스트 필드는 노드 메모리(520)의 데이터 배열 필드(521)와 함께 하나의 버스트 노드 엔트리(530)를 구성한다.
- [0068] 보다 구체적으로, 노드 스케줄러(510)는 노드 주소와 노드 데이터를 입력으로 받고, 푸시 엔트리 선택부(512)에서 입력된 노드 데이터가 저장될 버스트 노드 엔트리(530)의 위치를 찾는다. 여기에서, 위치를 찾는 방법은 다음과 같다. 모든 존재 필드를 확인한 후 노드 데이터가 존재하는 버스트 노드 엔트리(530)들을 찾는다. 그 다음 버스트 노드 엔트리(530)들의 버스트 필드와 입력된 노드 주소의 버스트 주소를 비교하여 동일한 주소를 갖는 버스트 필드를 찾는다. 버스트 필드가 찾아진 경우, 푸시 데이터에서 해당 버스트 노드 엔트리(530)에 노드 데이터를 저장한다. 동일한 버스트 필드를 찾지 못한 경우, 노드 데이터가 존재하지 않는 버스트 노드 엔트리

(530)를 선택하고 버스트 필드를 변경한다. 선택 방법은 존재 필드가 유효하지 않은 가장 상위의 버스트 노드 엔트리(530)를 우선으로 한다. 변경 방법은 선택된 버스트 노드 엔트리(530)의 버스트 필드에 입력된 노드 주소의 버스트 주소를 저장하는 것이다. 그 다음 푸시 데이터에서 초기화된 버스트 노드 엔트리(530)의 데이터 배열 필드(521)에 노드 데이터를 저장한다.

[0069] 노드 데이터의 저장이 완료되면 팝 엔트리 선택부(513)에서 모든 풀 필드 정보를 확인하여 더 이상 노드 데이터를 저장할 공간이 없는 버스트 노드 엔트리(530)를 찾는다. 만일 해당되는 버스트 노드 엔트리(530)가 존재하지 않는다면 노드 스케줄러(510)를 종료한다. 그렇지 않다면 해당 버스트 노드 엔트리(530)의 버스트 필드와 데이터 배열 필드(521)에 존재하는 데이터를 출력하고 노드 스케줄러(510)를 종료한다.

[0070] 도 5(b)는 노드 스케줄러의 실시예를 나타내는 블록도이다.

[0071] 노드의 생성 순서인 0, 1, 2, 3, ... 순서로 노드에 대한 정보가 노드 스케줄러(510)에 입력되면, 도 4(a)에서 설명한 과정을 거쳐 노드 메모리(520)에 노드의 탐색 순서에 따라 7, 3, 1, 0, ... 순서로 저장될 수 있다. 또한, 버스트 접근 사이즈는 4이고, 이에 따라 하나의 데이터 배열 필드(521)에는 4개의 노드 데이터가 저장될 수 있다. 즉, 노드의 생성 순서는 0, 1, 2, 3, ... 이지만 노드 메모리에 저장된 순서는 7, 3, 1, 0, ... 으로서 탐색 순서로 저장됨을 볼 수 있다.

[0072] 도 6는 레이 트레이싱의 과정을 설명하는 블록도이다.

[0073] 도 6를 참조하면, 각 픽셀(Pixel) 당 카메라(610) 위치로부터 최초 레이(Primary Ray)(P)를 생성하여 레이(P)와 만나는 물체(620)를 찾기 위한 계산을 수행한다. 해당 레이(P)와 만나게 된 물체가 굴절의 성질을 갖는 물체(620)나 반사의 성질을 갖는 물체(631)(632)인 경우에는, 해당 레이(P)와 물체가 만난 위치에서 굴절효과를 위한 굴절 레이(Refraction Ray)(F) 및 반사 효과를 위한 반사 레이(Reflection Ray)(R)를 생성하고, 라이트(650) 방향으로 그림자 레이(Shadow Ray)(S)를 생성할 수 있다. 일 실시예에서, 그림자 레이(S)가 다른 물체(640)와 만나면 해당 그림자 레이(S)가 생성된 지점에 그림자가 생성될 수 있다.

[0074] 도 7은 개시된 기술에서 사용되는 가속 구조와 기하학적 데이터를 설명하는 블록이다.

[0075] 도 7에서, 가속 구조(Acceleration Structure, AS)는 KD-tree를 사용한다고 가정하였다. KD-tree는 공간 분할 구조체(Spatial Partitioning Structure)의 일종으로, 레이-삼각형 교차 테스트(Ray-Triangle Intersection Test)를 위하여 사용될 수 있다. 예를 들어, KD-tree는 박스 노드(Box Node)(710), 내부 노드(Inner Node)(720) 및 리프 노드(Leaf Node)(730)를 포함할 수 있다. 일 실시예에서, 리프 노드(730)는 기하학적 데이터에 포함된 적어도 하나의 삼각형 정보를 포인팅 하기 위한 삼각형 리스트를 포함할 수 있다. 예를 들어, 삼각형 정보는 삼각형의 세 점에 대한 정점 좌표, 법선 벡터 및 텍스처 좌표를 포함할 수 있다. 일 실시예에서, 기하학적 데이터에 포함된 삼각형 정보가 배열로 구현된 경우에는 리프 노드에 포함된 삼각형 리스트는 배열 인덱스에 상응할 수 있다. 일 실시예에서, 내부 노드(720)는 바운딩 박스 기반의 공간 영역을 가지고 있고, 공간 영역은 다시 2개의 영역으로 나뉘어서 두개의 하위 노드에 할당 될 수 있다. 즉, 내부 노드(720)는 분할 면과 이를 통해 분할된 두개 영역의 서브트리로 구성되고, 리프 노드(730)는 일련의 원시 데이터만을 포함하게 된다. 공간을 분할하는 분할 면은 임의의 광선과 부딪치게 되는 원시데이터를 찾기 위한 값(예를 들어, 노드 방문 횟수, 원시데이터와 교차 되는지 여부를 계산하는 횟수 등)이 최소가 되는 지점을 찾아야 하고, SAH(Surface area heuristic)은 해당 지점을 찾기 위해 사용하는 방법에 해당할 수 있다.

[0076] 도 8은 본 발명의 일 실시예에 따른 트리 생성 유닛을 설명하는 흐름도이다.

[0077] 먼저, 원시 데이터의 개수가 임계치보다 많은지 여부를 확인한다(단계 S801). 여기에서, 임계치는 하나의 제2 트리 생성부(120)가 처리할 수 있는 최대 원시데이터의 개수이다. 확인 결과, 임계치보다 많은 경우 비닝 방식으로 공간을 분할하고(단계 S802), 분할한 공간에 대하여 단계 S801로 돌아가 원시데이터의 개수가 임계치보다 많은지 여부를 다시 확인한다.

[0078] 원시 데이터의 개수가 임계치보다 적은 경우에는 정렬 방식으로 공간을 분할한다(단계 S803). 분할을 수행하고

더 이상 분할할 공간이 없는지 여부(리프노드에 해당하는지 여부)를 확인하고(단계 S804), 없는 경우 종료되고, 있는 경우에는 단계 S803으로 돌아가서 다시 정렬 방식으로 공간을 분할한다.

[0079]

일 실시예에서, 공간을 분할한다는 것은 공간을 분할하는 위치인 분할 면을 찾는 것이고 분할 면은 광선과 부딪치는 원시 데이터를 찾기 위한 값(예를 들어, 노드 방문 횟수, 원시 데이터와 교차 되는지 여부를 계산하는 횟수 등)이 최소가 되는 지점을 의미할 수 있다. SAH(Surface area heuristic) 계산 방식을 이용하여 분할 면을 찾고, SAH 방식에는 비닝 및 정렬 방식이 있다.

[0080]

상기에서는 본 출원의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 출원을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

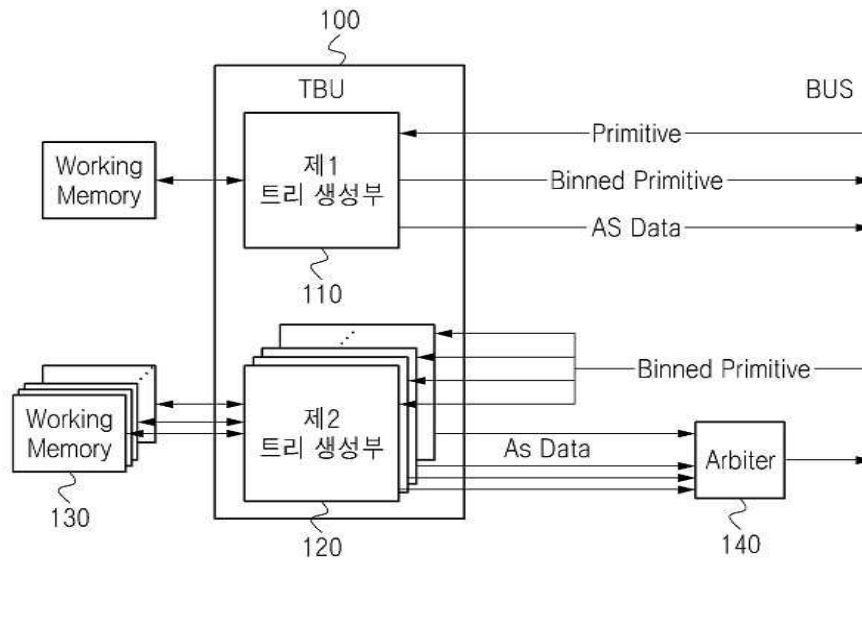
부호의 설명

[0081]

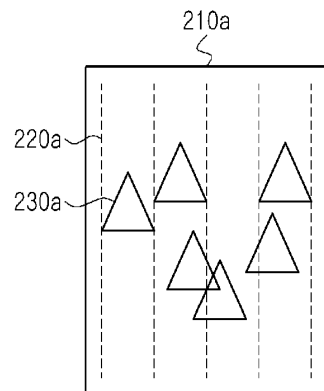
100: 트리 생성 유닛	110: 제1 트리 생성부
120: 제2 트리 생성부	130: 작업 메모리
210a: 바운딩 박스	220a: 분할 면
230a: 원시데이터	210b: 바운딩 박스
220b: 분할 면	230b: 원시 데이터
300: 레이 트레이싱 장치	310: CPU(Central Processing Unit)
320: 시스템 메모리	330: DRTX(Dynamic Ray Tracing Accelerator)
340: 외부 메모리	
410: 원시 데이터	420: 원시 데이터 인덱스
430: 임시 공간	440: 가속 구조 데이터
510: 노드 스케줄러	520: 노드 메모리
530: 버스트 노드 엔트리	
610: 카메라	620: 물체
631: 물체	632: 물체
640: 물체	650: 라이트
710: 박스 노드	720: 내부 노드
730: 리프노드	

도면

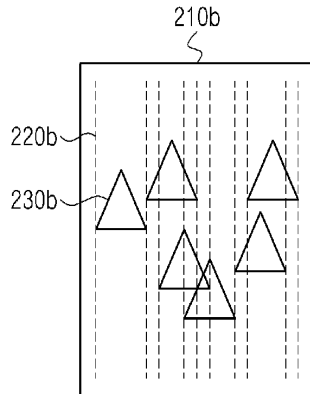
도면1



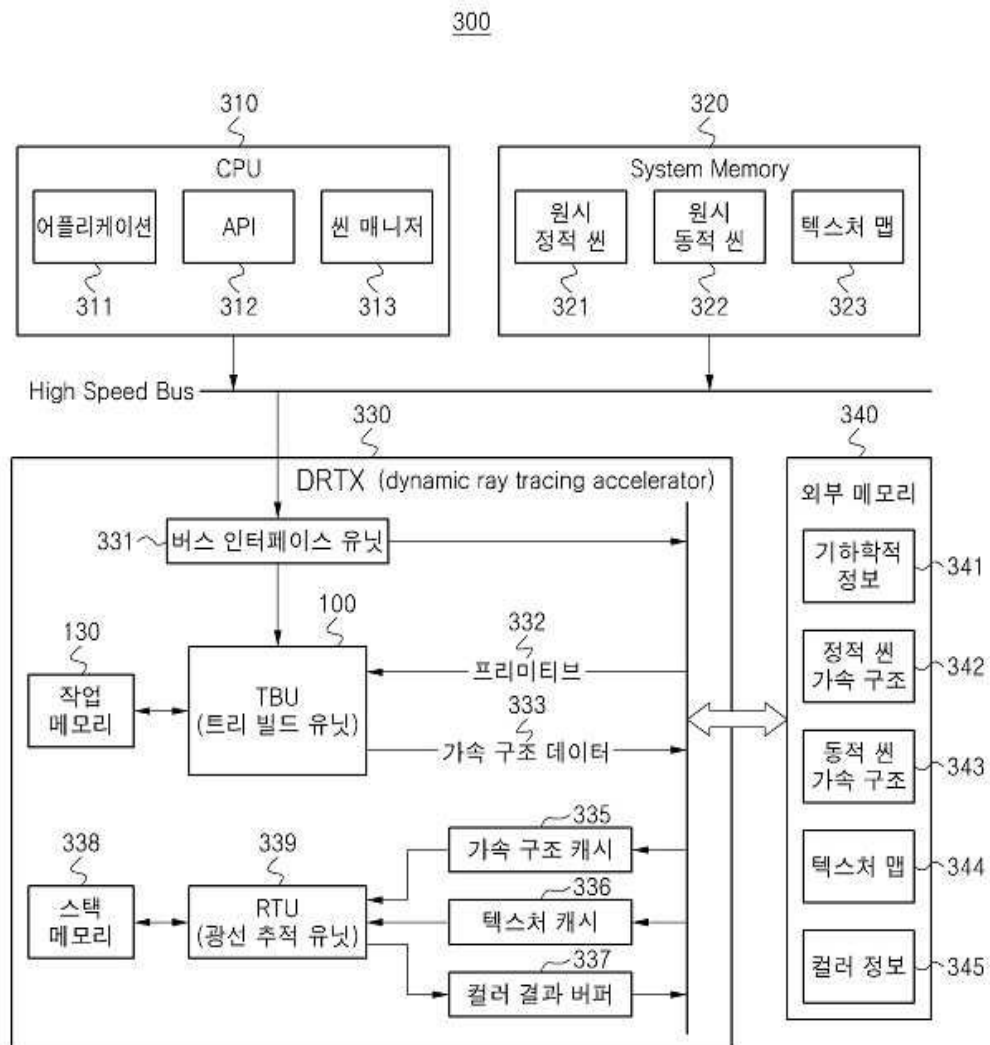
도면2a



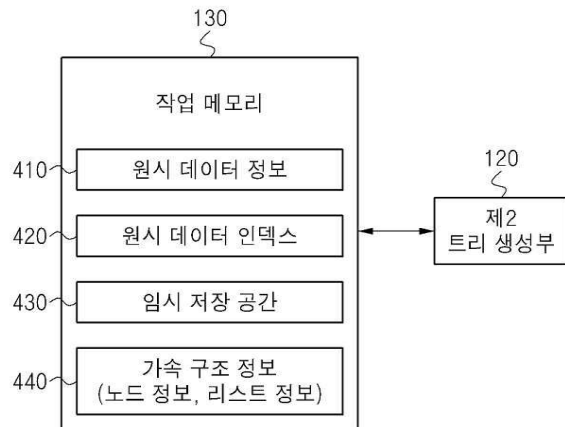
도면2b



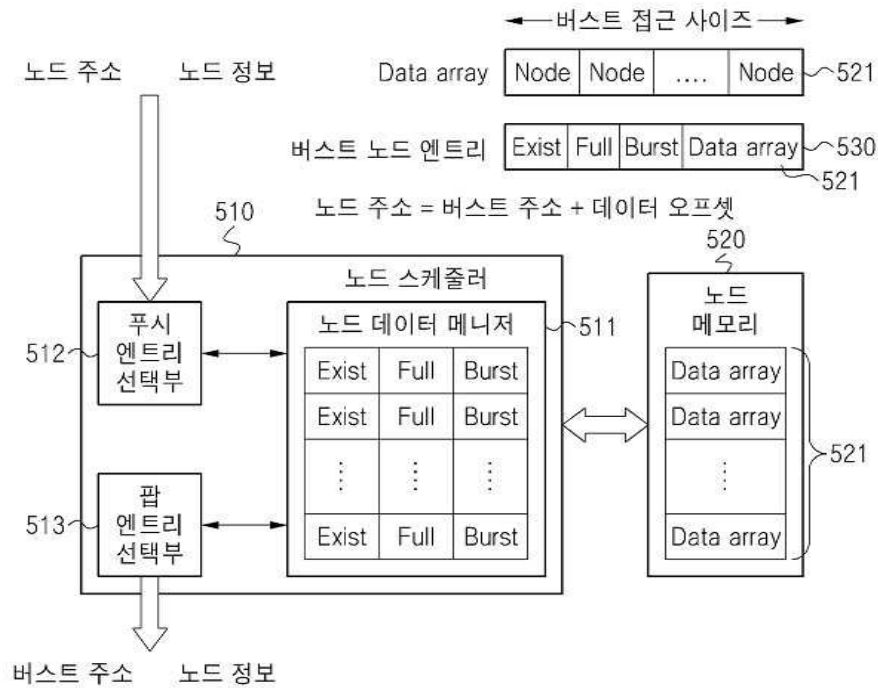
도면3



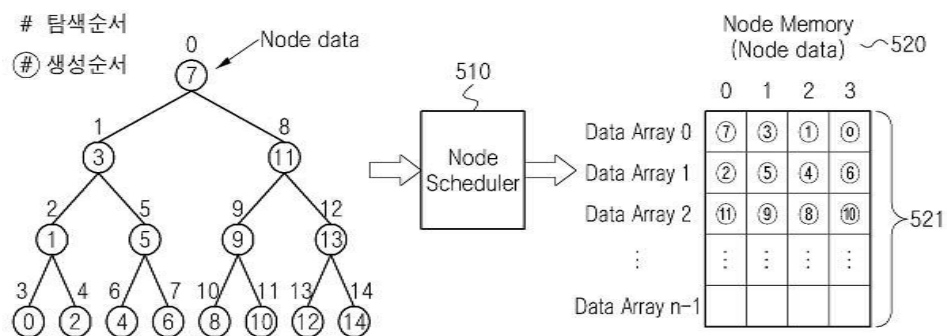
도면4



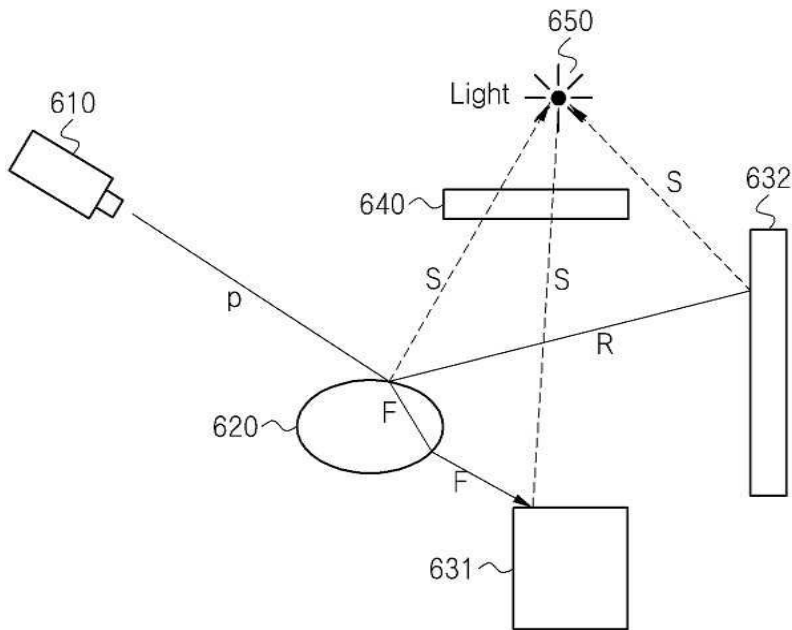
도면5a



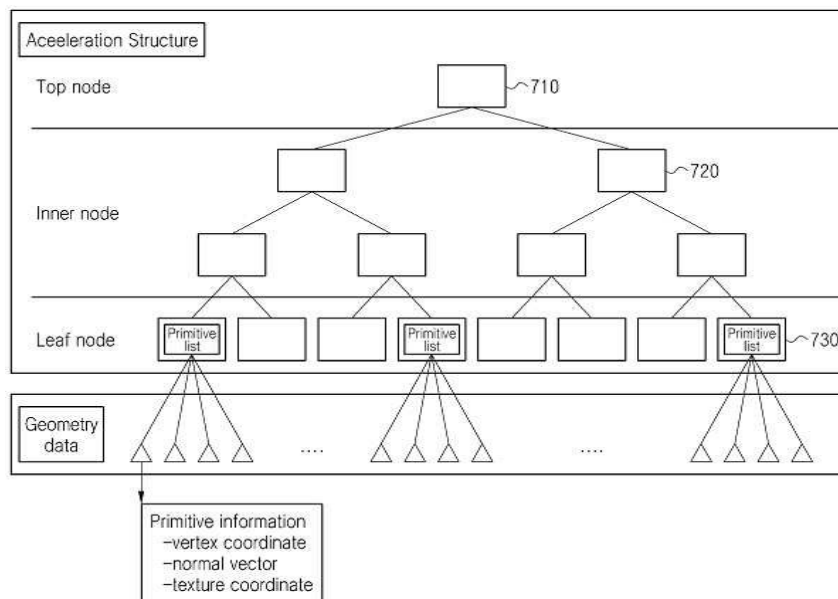
도면5b



도면6



도면7



도면8

