(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2021년06월07일
(11) 등록번호 10-2261257
(24) 등록일자 2021년06월01일

(51) 국제특허분류(Int. Cl.)
G06T 15/06 (2011.01) *G06T 1/20* (2018.01)
G06T 15/80 (2011.01)
(52) CPC특허분류
G06T 15/06 (2013.01)
G06T 1/20 (2013.01)
(21) 출원번호 10-2019-0159260
(22) 출원일자 2019년12월03일
심사청구일자 2019년12월03일
(56) 선행기술조사문헌
KR1020100077423 A*
KR1020100128493 A*
Jon Peddie, "SiliconArts new ray tracing chip and IP 4th generation path tracing engine", https://www.jonpeddie.com/news/siliconarts-new-ray-tracing-chip-and-ip, 2019.10.10.*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
세종대학교산학협력단
서울특별시 광진구 능동로 209 (군자동, 세종대학교)
(72) 발명자
박우찬
서울특별시 광진구 능동로 209, 대양AI센터 723호(군자동, 세종대학교)
정우남
서울특별시 광진구 능동로 209, 대양AI센터 827호(군자동, 세종대학교)
윤주원
서울특별시 광진구 능동로 209, 대양AI센터 712호(군자동, 세종대학교)
(74) 대리인
정부연

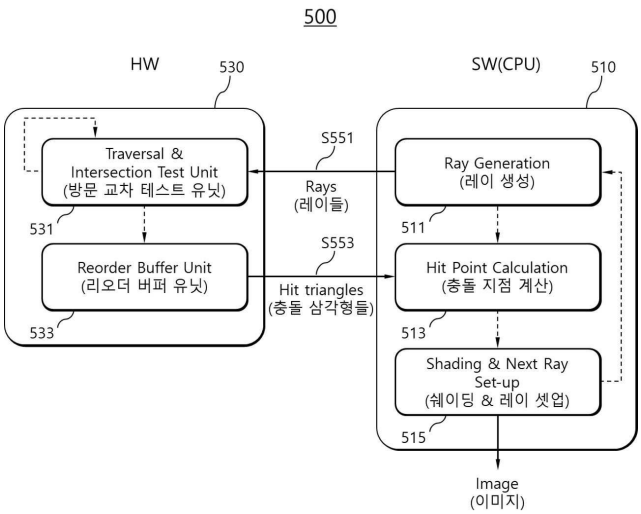
전체 청구항 수 : 총 11 항 심사관 : 박상철

(54) 발명의 명칭 **장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법**

(57) 요 약

본 발명은 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법에 관한 것으로, 상기 레이 트레이싱 장치는 소프트웨어를 통해, (a) 레이 트레이싱(ray tracing)을 위한 레이를 생성하는 레이 생성 연산과 (b) 상기 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 충돌 지점 계산 연산 및 (c) 상기 충돌 지점 상에서의 컬러 값을 산출하는 쉐이딩(shading) 연산을 각각 처리하는 소프트웨어 유닛부 및 상기 소프트웨어 유닛부과 연동하는 전용 하드웨어로서, 상기 레이를 기초로 방문 교차 테스트 연산을 수행하는 방문 교차 테스트 유닛과 상기 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하는 리오더 버퍼(reorder buffer) 유닛으로 구성된 전용 하드웨어 유닛부를 포함한다.

대 표 도 - 도5



(52) CPC특허분류
G06T 15/80 (2013.01)
G06T 2200/28 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711081077
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	ICT융합산업원천기술개발(R&D)
연구과제명	극사실적인 실시간 가상현실을 위한 모바일 GPU 하드웨어 개발
기 여 율	1/1
과제수행기관명	세종대학교 산학협력단
연구기간	2017.01.01 ~ 2021.12.31

명 세 서

청구범위

청구항 1

소프트웨어를 통해, (a) 레이 트레이싱(ray tracing)을 위한 레이를 생성하는 레이 생성 연산과 (b) 상기 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 충돌 지점 계산 연산 및 (c) 상기 충돌 지점 상에서의 컬러 값을 산출하는 쉐이딩(shading) 연산을 각각 처리하는 소프트웨어 유닛부; 및

상기 소프트웨어 유닛부과 연동하는 전용 하드웨어로서, 상기 레이를 기초로 방문 교차 테스트 연산을 수행하는 방문 교차 테스트 유닛과 상기 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하는 리오더 버퍼(reorder buffer) 유닛으로 구성된 전용 하드웨어 유닛부를 포함하고,

상기 소프트웨어 유닛부 및 상기 전용 하드웨어 유닛부 간에는 상기 레이와 상기 충돌 삼각형 정보의 일방향 전송만이 각각 수행되고,

상기 리오더 버퍼 유닛은 상기 방문 교차 테스트 유닛으로 입력되는 상기 레이를 저장하는 제1 단계, 상기 방문 교차 테스트 유닛으로부터 출력되는 상기 충돌 삼각형 정보를 수신하는 제2 단계 및 상기 충돌 삼각형 정보와 연관된 레이의 우선순위에 따라 해당 충돌 삼각형 정보를 출력하는 제3 단계를 순차적으로 수행함으로써 상기 소프트웨어 유닛부로부터 일방 전송되는 상기 레이의 순서에 대응하여 상기 충돌 삼각형 정보를 일방 전송하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치.

청구항 2

제1항에 있어서, 상기 소프트웨어 유닛부는

상기 쉐이딩 연산 단계에서 그림자(shadow) 레이 또는 세컨더리(secondary) 레이의 발생이 검출되면 상기 그림자 레이 또는 상기 세컨더리 레이에 관한 정보를 생성하여 상기 레이 생성 연산 단계로 전달하는 레이 셋업(Set-up) 연산을 더 처리하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치.

청구항 3

제1항에 있어서, 상기 소프트웨어 유닛부는

상기 레이 생성 연산 단계에서 프레임 내의 각 픽셀에 대해 시점 위치로부터 프라이머리 레이를 생성하여 해당 프라이머리 레이의 데이터 값을 상기 전용 하드웨어 유닛부로 일방 전송하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치.

청구항 4

제1항에 있어서, 상기 소프트웨어 유닛부는

상기 충돌 지점 계산 연산 단계에서 상기 충돌 삼각형 정보와 상기 레이를 기초로 충돌 삼각형과 상기 레이 간의 충돌 지점의 좌표를 산출하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치.

청구항 5

제1항에 있어서, 상기 방문 교차 테스트 유닛은

상기 레이를 기초로 가속 구조(Acceleration Structure, AS)를 탐색하여 상기 레이와 연관된 리프 노드를 결정하는 방문 연산과 상기 리프 노드에 포함된 모든 삼각형들에 관한 교차 테스트를 수행하여 상기 레이와 충돌하

는 충돌 삼각형을 결정하는 교차 테스트 연산을 각각 처리하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치.

청구항 6

삭제

청구항 7

소프트웨어 유닛부와 전용 하드웨어 유닛부가 상호 연동하는 레이 트레이싱(ray tracing) 장치에서 수행되는 방법에 있어서,

(a) 상기 소프트웨어 유닛부를 통해, 레이 트레이싱을 위한 레이를 생성하여 상기 전용 하드웨어 유닛부에게 일방 전송하는 단계;

(b) 상기 전용 하드웨어 유닛부를 통해, 상기 레이를 기초로 방문 교차 테스트 연산을 수행하는 단계;

(c) 상기 전용 하드웨어 유닛부를 통해, 상기 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하여 상기 소프트웨어 유닛부에게 일방 전송하는 단계;

(d) 상기 소프트웨어 유닛부를 통해, 상기 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 단계; 및

(e) 상기 소프트웨어 유닛부를 통해, 상기 충돌 지점 상에서의 컬러 값을 산출하는 단계를 포함하되,

상기 (c) 단계는 상기 방문 교차 테스트 유닛으로 입력되는 상기 레이를 저장하고, 상기 방문 교차 테스트 유닛으로부터 출력되는 상기 충돌 삼각형 정보를 수신하며, 상기 충돌 삼각형 정보와 연관된 레이의 우선순위에 따라 해당 충돌 삼각형 정보를 출력함으로써 상기 소프트웨어 유닛부로부터 일방 전송되는 상기 레이의 순서에 대응하여 상기 충돌 삼각형 정보를 일방 전송하는 단계를 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

청구항 8

제7항에 있어서, 상기 (a) 단계는

프레임 내의 각 픽셀에 대해 시점 위치로부터 프라이머리 레이를 생성하여 해당 프라이머리 레이의 데이터 값을 상기 전용 하드웨어 유닛부로 일방 전송하는 단계를 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

청구항 9

제7항에 있어서, 상기 (b) 단계는

상기 레이를 기초로 AS 구조를 탐색하여 상기 레이와 연관된 리프 노드를 결정하는 단계와 상기 리프 노드에 포함된 모든 삼각형들에 관한 교차 테스트를 수행하여 상기 레이와 충돌하는 충돌 삼각형을 결정하는 단계를 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

청구항 10

삭제

청구항 11

제7항에 있어서, 상기 (d) 단계는

상기 충돌 삼각형 정보와 상기 레이를 기초로 충돌 삼각형과 상기 레이 간의 충돌 지점의 좌표를 산출하는 단계

를 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

청구항 12

제7항에 있어서, 상기 (e) 단계는

그림자(shadow) 레이 또는 세컨더리(secondary) 레이의 발생이 검출되면 상기 그림자 레이 또는 상기 세컨더리 레이에 관한 정보를 생성하여 상기 (a) 단계로 전달하는 단계를 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

청구항 13

제7항에 있어서,

(f) 상기 소프트웨어 유닛부를 통해, 프레임 내의 모든 픽셀에 대한 컬러 값이 결정된 경우 상기 프레임에 관한 이미지를 출력하는 단계를 더 포함하는 것을 특징으로 하는 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 장치들 간의 대역폭을 줄이는 레이 트레이싱 기술에 관한 것으로, 더욱 상세하게는 장치들 간에 대규모의 데이터 전송을 줄여 레이 트레이싱의 성능을 향상시키는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법에 관한 것이다.

배 경 기 술

[0003] 레이 트레이싱(Ray Tracing)은 크게 레이 생성(ray generation), 트리 방문 & 교차 테스트(Tree Traversal & Intersection Test), 충돌 지점 계산(Hit Point Calculation), 쉐이딩(Shading)의 단계로 구성되며 반복적(recursive)으로 수행될 수 있다.

[0004] 최근에 영화나 게임에서 고품질의 CG 영상이 요구되면서 레이 트레이싱에서 계산량을 가장 많이 필요로 하는 트리 방문 & 교차 테스트와 충돌 지점 계산 부분을 전용 하드웨어(HW)에서 처리하고, 나머지 부분인 레이 생성과 쉐이딩은 소프트웨어(SW)적으로 처리하는 방식이 생겨나고 있다.

[0005] 특히, 쉐이딩은 고품질로 갈수록 다양한 조명 모델에 따른 복잡한 과정을 요구하기 때문에 모든 옵션을 하드웨어로 구현하기가 어려워 소프트웨어로 처리하기가 용이하다. 이러한 경우 두 개의 독립된 장치들이 병렬적으로 수행될 수 있기 때문에 성능 향상을 기대할 수 있으나, 두 개의 장치들 간에 대규모의 데이터 전송이 필요하게 되어 장치들 간에 대기시간이 늘어남으로 인해 성능이 저하될 수 있다.

선행기술문헌

특허문헌

[0007] (특허문헌 0001) 한국등록특허 제10-1076807(2011.10.19)호

발명의 내용

해결하려는 과제

[0008] 본 발명의 일 실시예는 장치들 간에 대규모의 데이터 전송을 줄여 레이 트레이싱의 성능을 향상시키는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법을 제공하고자 한다.

[0009] 본 발명의 일 실시예는 레이 트레이싱 과정에서 방문 교차 테스트는 전용 하드웨어서 처리하고 충돌 지점 계산

연산은 소프트웨어에서 처리함으로써 하드웨어와 소프트웨어 간의 데이터 전송을 최소화할 수 있는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법을 제공하고자 한다.

[0010] 본 발명의 일 실시예는 레이 트레이싱을 위한 전용 하드웨어 장치로서 리오더 버퍼 유닛을 포함함으로써 소프트웨어적 처리를 담당하는 소프트웨어 장치와의 연동을 효율적으로 수행할 수 있는 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법을 제공하고자 한다.

과제의 해결 수단

[0012] 실시예들 중에서, 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치는 소프트웨어를 통해, (a) 레이 트레이싱(ray tracing)을 위한 레이를 생성하는 레이 생성 연산과 (b) 상기 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 충돌 지점 계산 연산 및 (c) 상기 충돌 지점 상에서의 컬러 값을 산출하는 쉐이딩(shading) 연산을 각각 처리하는 소프트웨어 유닛부 및 상기 소프트웨어 유닛부과 연동하는 전용 하드웨어로서, 상기 레이를 기초로 방문 교차 테스트 연산을 수행하는 방문 교차 테스트 유닛과 상기 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하는 리오더 버퍼(reorder buffer) 유닛으로 구성된 전용 하드웨어 유닛부를 포함한다.

[0013] 상기 소프트웨어 유닛부는 상기 쉐이딩 연산 단계에서 그림자(shadow) 레이 또는 세컨더리(secondary) 레이의 발생이 검출되면 상기 그림자 레이 또는 상기 세컨더리 레이에 관한 정보를 생성하여 상기 레이 생성 연산 단계로 전달하는 레이 셋업(Set-up) 연산을 더 처리할 수 있다.

[0014] 상기 소프트웨어 유닛부는 상기 레이 생성 연산 단계에서 프레임 내의 각 픽셀에 대해 시점 위치로부터 프라이머리 레이를 생성하여 해당 프라이머리 레이의 데이터 값을 상기 전용 하드웨어 유닛부로 일방 전송할 수 있다.

[0015] 상기 소프트웨어 유닛부는 상기 충돌 지점 계산 연산 단계에서 상기 충돌 삼각형 정보와 상기 레이를 기초로 충돌 삼각형과 상기 레이 간의 충돌 지점의 좌표를 산출할 수 있다.

[0016] 상기 방문 교차 테스트 유닛은 상기 레이를 기초로 가속 구조(Acceleration Structure, AS)를 탐색하여 상기 레이와 연관된 리프 노드를 결정하는 방문 연산과 상기 리프 노드에 포함된 모든 삼각형들에 관한 교차 테스트를 수행하여 상기 레이와 충돌하는 충돌 삼각형을 결정하는 교차 테스트 연산을 각각 처리할 수 있다.

[0017] 상기 리오더 버퍼 유닛은 상기 방문 교차 테스트 유닛으로 입력되는 상기 레이를 저장하는 제1 단계, 상기 방문 교차 테스트 유닛으로부터 출력되는 상기 충돌 삼각형 정보를 수신하는 제2 단계 및 상기 충돌 삼각형 정보와 연관된 레이의 우선순위에 따라 해당 충돌 삼각형 정보를 출력하는 제3 단계를 순차적으로 수행함으로써 상기 소프트웨어 유닛부로부터 일방 전송되는 상기 레이의 순서에 대응하여 상기 충돌 삼각형 정보를 일방 전송할 수 있다.

[0018] 실시예들 중에서, 장치들 간의 대역폭을 줄이는 레이 트레이싱 방법은 (a) 상기 소프트웨어 유닛부를 통해, 레이 트레이싱을 위한 레이를 생성하여 상기 전용 하드웨어 유닛부에게 일방 전송하는 단계, (b) 상기 전용 하드웨어 유닛부를 통해, 상기 레이를 기초로 방문 교차 테스트 연산을 수행하는 단계, (c) 상기 전용 하드웨어 유닛부를 통해, 상기 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하여 상기 소프트웨어 유닛부에게 일방 전송하는 단계, (d) 상기 소프트웨어 유닛부를 통해, 상기 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 단계 및 (e) 상기 소프트웨어 유닛부를 통해, 상기 충돌 지점 상에서의 컬러 값을 산출하는 단계를 포함할 수 있다.

[0019] 상기 (a) 단계는 프레임 내의 각 픽셀에 대해 시점 위치로부터 프라이머리 레이를 생성하여 해당 프라이머리 레이의 데이터 값을 상기 전용 하드웨어 유닛부로 일방 전송하는 단계를 포함할 수 있다.

[0020] 상기 (b) 단계는 상기 레이를 기초로 AS 구조를 탐색하여 상기 레이와 연관된 리프 노드를 결정하는 단계와 상기 리프 노드에 포함된 모든 삼각형들에 관한 교차 테스트를 수행하여 상기 레이와 충돌하는 충돌 삼각형을 결정하는 단계를 포함할 수 있다.

[0021] 상기 (c) 단계는 상기 방문 교차 테스트 유닛으로 입력되는 상기 레이를 저장하고, 상기 방문 교차 테스트 유닛으로부터 출력되는 상기 충돌 삼각형 정보를 수신하며, 상기 충돌 삼각형 정보와 연관된 레이의 우선순위에 따라 해당 충돌 삼각형 정보를 출력함으로써 상기 소프트웨어 유닛부로부터 일방 전송되는 상기 레이의 순서에 대응하여 상기 충돌 삼각형 정보를 일방 전송하는 단계를 포함할 수 있다.

[0022] 상기 (d) 단계는 상기 충돌 삼각형 정보와 상기 레이를 기초로 충돌 삼각형과 상기 레이 간의 충돌 지점의 좌표

를 산출하는 단계를 포함할 수 있다.

[0023] 상기 (e) 단계는 그림자(shadow) 레이 또는 세컨더리(secondary) 레이의 발생이 검출되면 상기 그림자 레이 또는 상기 세컨더리 레이에 관한 정보를 생성하여 상기 (a) 단계로 전달하는 단계를 포함할 수 있다.

[0024] 상기 레이 트레이싱 방법은 (f) 상기 소프트웨어 유닛부를 통해, 프레임 내의 모든 픽셀에 대한 컬러 값이 결정된 경우 상기 프레임에 관한 이미지를 출력하는 단계를 더 포함할 수 있다.

발명의 효과

[0026] 개시된 기술은 다음의 효과를 가질 수 있다. 다만, 특정 실시예가 다음의 효과를 전부 포함하여야 한다거나 다음의 효과만을 포함하여야 한다는 의미는 아니므로, 개시된 기술의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.

[0027] 본 발명의 일 실시예에 따른 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법은 레이 트레이싱 과정에서 방문 교차 테스트는 전용 하드웨어서 처리하고 충돌 지점 계산 연산은 소프트웨어에서 처리함으로써 하드웨어와 소프트웨어 간의 데이터 전송을 최소화할 수 있다.

[0028] 본 발명의 일 실시예에 따른 장치들 간의 대역폭을 줄이는 레이 트레이싱 장치 및 방법은 레이 트레이싱을 위한 전용 하드웨어 장치로서 리오더 버퍼 유닛을 포함함으로써 소프트웨어적 처리를 담당하는 소프트웨어 장치와의 연동을 효율적으로 수행할 수 있다.

도면의 간단한 설명

[0030] 도 1은 레이 트레이싱 장치에 의해 수행되는 레이 트레이싱 과정의 일 실시예를 설명하는 도면이다.

도 2는 레이 트레이싱 과정에서 사용되는 가속 구조로서 KD 트리의 일 실시예를 설명하는 도면이다.

도 3은 레이 트레이싱 장치에서 수행되는 레이 트레이싱 과정의 일반적인 예를 설명하는 도면이다.

도 4는 SW 연산과 전용 HW의 연동을 통해 레이 트레이싱을 처리하는 일 실시예를 설명하는 도면이다.

도 5는 본 발명의 일 실시예에 따른 장치들 간에 대역폭을 줄이는 레이 트레이싱 장치를 설명하는 도면이다.

도 6은 도 5의 레이 트레이싱 장치에서 수행되는 장치들 간에 대역폭을 줄이는 레이 트레이싱 과정의 일 실시예를 설명하는 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0031] 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.

[0032] 한편, 본 출원에서 서술되는 용어의 의미는 다음과 같이 이해되어야 할 것이다.

[0033] "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.

[0034] 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.

[0035] 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 이

들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0036] 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.

[0037] 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, HDD, SSD 등이 있다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

[0038] 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.

[0040] 도 1은 레이 트레이싱 장치에 의해 수행되는 레이 트레이싱 과정의 일 실시예를 설명하는 도면이다.

[0041] 도 1을 참조하면, 레이 트레이싱 방식은 전역 조명(global illumination)에 따른 렌더링(rendering) 방식에 해당할 수 있다. 이는 다른 물체에서 반사되거나 굴절된 빛(Light)도 현재 물체의 영상에 영향을 준다는 것을 의미할 수 있다. 이로 인하여 반사, 굴절, 그림자 효과가 자연적으로 제공되기 때문에 현실감 있는 3D 영상을 생성할 수 있다.

[0042] 도 1에서, 레이 트레이싱 장치는 먼저 각 픽셀(pixel) 당 카메라(Camera) 위치로부터 프라이머리 레이(primary ray, P)를 생성하여 해당 레이와 만나는 물체를 찾기 위한 계산을 수행할 수 있다. 레이 트레이싱 장치는 레이와 만나게 된 물체가 반사나 굴절의 성질이 있으면 레이와 물체가 만난 위치에서 반사 효과를 위한 반사 레이(reflection ray, R)나 굴절 효과를 위한 굴절 레이(refraction ray, F)를 생성할 수 있고, 또한 그림자 효과를 위하여 빛(Light) 방향으로 그림자 레이(shadow ray, S)를 생성할 수 있다.

[0043] 이 때, 해당 빛(Light) 위치로 향한 그림자 레이와 어떤 물체가 만나면 그림자가 생성이 되며 그렇지 않을 경우는 그림자가 생성되지 않는다. 반사 레이와 굴절 레이는 세컨더리 레이(secondary ray)라고 불리며 레이 트레이싱 장치는 각각의 레이에 대해 해당 레이와 만나는 물체를 찾기 위한 계산을 수행할 수 있다. 이러한 과정은 레이 트레이싱 장치에 의해 반복적(recursive)으로 수행될 수 있다.

[0045] 도 2는 레이 트레이싱 과정에서 사용되는 가속 구조로서 KD 트리의 일 실시예를 설명하는 도면이다.

[0046] 도 2를 참조하면, 레이 트레이싱을 수행하기 위하여 전체 지오메트리 데이터(geometry data)(triangle의 좌표들로 구성)를 가지고 생성된 kd-트리(k-depth tree)나 BVH(Bounding Volume Hierarchy)와 같은 가속 구조(Acceleration Structure, AS)가 필수적으로 요구된다. 따라서, 레이 트레이싱을 수행하기 이전에 AS를 구축(build)할 필요가 있다. 이러한 가속 구조 구축(AS build)에는 연산량이 많이 필요하기 때문에 시간이 많이 소요될 수 있다.

[0047] 도 2에서, kd-트리의 전체 구성도를 설명하고 있다. kd-트리는 분할한 공간에 대하여 계층적(hierarchy) 구조를 갖는 이진 트리(binary tree)에 해당할 수 있다. kd-트리는 내부 노드(inner node)(top node 포함)와 리프 노드(leaf node)로 구성될 수 있으며, 리프 노드는 해당 노드와 교차(intersection)되는 객체(object)들을 포함하고 있는 공간에 대응될 수 있다. 즉, kd-트리는 공간 분할 트리(spatial partitioning tree)의 일종에 해당할 수 있다.

[0048] 반면, 내부 노드는 바운딩 박스(bounding box) 기반의 공간 영역을 가질 수 있으며 해당 공간 영역은 다시 2개의 영역들로 나뉘어서 두 개의 하위 노드에 할당될 수 있다. 결과적으로 내부 노드는 분할 평면과 이를 통해 분할된 두 개의 영역의 서브-트리(sub-tree)로 구성될 수 있고, 리프 노드는 일련의 삼각형(triangle)들 만을 포함할 수 있다.

[0049] 한편, 공간을 분할하는 위치 p는 임의의 레이와 충돌(hit)하는 삼각형을 찾기 위한 비용(cost)(노드 방문 횟수, 삼각형과 교차되는지 계산하는 횟수 등)이 최소가 되는 지점에 해당할 수 있고, 해당 위치 p를 찾기 위해 현재

가장 많이 사용되는 방법은 SAH(Surface area heuristic)에 해당할 수 있다.

[0051] 도 3은 레이 트레이싱 장치에서 수행되는 레이 트레이싱 과정의 일반적인 예를 설명하는 도면이다.

[0052] 도 3을 참조하면, 레이 트레이싱 과정은 재귀적으로 수행될 수 있고, (i) 레이 생성(Ray Generation) 단계(S310), (ii) 방문(Traversal) & 교차 테스트(Intersection Test) 단계(S330), (iii) 충돌 지점 계산(Hit Point Calculation) 단계(S350) 및 (iv) 쉐이딩(Shading) & 레이 셋업(Next Ray Set-up) 단계(S370)를 포함할 수 있다.

[0053] 먼저 레이 생성(ray generation) 단계에서는 각각의 픽셀(pixel)에 대하여 시점 위치로부터 프라이머리 레이(primary ray)를 생성할 수 있다(단계 S310). 다음 단계에서는 kd-트리(kd-tree)와 bounding volume hierarchy(BVH)와 같은 가속 구조(AS)를 탐색하여 레이와 만나는 리프 노드를 찾을 수 있다(단계 S330). 여기에서, 리프 노드에는 삼각형(triangle)의 정보들이 저장되어 있다.

[0054] 다음의 교차 테스트(intersection test) 단계에서는 만난 리프 노드에 있는 모든 삼각형에 대하여 레이와 만나는지를 테스트할 수 있다(단계 S330). 이러한 과정은 레이와 만나는 삼각형을 찾을 때까지 반복적으로 수행될 수 있다. 그 후, 레이와 만난 삼각형에 대하여 충돌 지점 계산(hit point calculation) 단계에서 충돌 지점(hit point)를 계산할 수 있다(단계 S350).

[0055] 다음 단계인 쉐이딩(shading) 단계에서 레이-삼각형 충돌 지점(ray-triangle hit point) 상에서의 컬러(color) 값을 계산할 수 있다(단계 S370). 만약 조명으로 인하여 그림자 레이(shadow ray)의 발생이 필요하거나 충돌된 삼각형의 재질로 인한 세컨더리 레이(secondary ray)의 발생이 필요하면 이러한 정보를 레이 셋업(next ray set-up) 단계에서 결정하여 레이 생성(ray generation) 단계로 전송될 수 있다. 레이 생성(ray generation) 단계에서는 이러한 정보를 기반으로 그림자 레이 및 세컨더리 레이를 발생시킬 수 있다(단계 S310).

[0057] 도 4는 SW 연산과 전용 HW의 연동을 통해 레이 트레이싱을 처리하는 일 실시예를 설명하는 도면이다.

[0058] 도 4를 참조하면, 레이 트레이싱 과정은 계산량을 가장 많이 필요로 하는 트리 방문 & 교차 테스트(tree traversal & intersection test)와 충돌 지점 계산(hit point calculation) 부분을 전용 하드웨어(HW)에서 처리하고, 나머지 부분인 레이 생성(ray generation)과 쉐이딩(shading)은 소프트웨어(SW)적으로 처리하는 방식으로 수행될 수 있다.

[0059] 먼저 소프트웨어 부분인 레이 생성 단계에서는 각각의 픽셀에 대하여 시점(view point) 위치로부터 프라이머리 레이를 생성할 수 있다. 이 레이들에 대한 데이터 값이 하드웨어(HW)로 전송될 수 있다. 이 값은 레이의 벡터(vector) 값, 레이의 종류(예를 들어, primary ray, secondary ray, shadow ray 등)를 구별하기 위한 레이 타입(ray type) 값 등을 포함할 수 있다.

[0060] 하드웨어 부분에는 트리 방문 & 교차 테스트(tree traversal & intersection test)와 충돌 지점 계산(hit point calculation)을 수행할 수 있다. 하드웨어 부분에서 수행된 결과인 충돌 지점(hit point) 정보와 충돌된 삼각형 정보가 소프트웨어 부분으로 전송될 수 있다. 이는 충돌 삼각형(hit triangle)이 입력된 레이의 순서대로 결정된다는 가정하에 유용하며, 만약 그렇지 않으면 추가적인 정보(예를 들어, 몇 번째 입력된 레이인지를 구별하기 위한 레이 식별자(ray id))가 더 필요할 수 있다.

[0061] 스프트웨어인 쉐이딩(shading) 단계에서는 레이-삼각형 충돌 지점(ray-triangle hit point) 상에서의 컬러(color) 값을 계산할 수 있다다. 만약 조명으로 인하여 그림자 레이(shadow ray)의 발생이 필요하거나 또는 충돌된 삼각형의 재질로 인한 세컨더리 레이(secondary ray)의 발생이 필요하면 이러한 정보를 레이 셋업(next ray set-up) 단계에서 결정하여 레이 생성(ray generation) 단계로 보내어 질 수 있다. 레이 생성(Ray generation) 단계에서는 이러한 정보를 기반으로 그림자 레이(shadow ray) 및 세컨더리 레이(secondary ray)를 발생시킬 수 있다. 발생된 레이에 대한 데이터 값은 다시 하드웨어 부분으로 보내질 수 있다.

[0062] 도 4에서, 하드웨어(HW) 부분과 소프트웨어(SW) 부분이 병렬적으로 수행될 수 있기 때문에 성능 향상을 기대할 수 있으나, 두 개의 장치들 간에 대규모의 데이터 전송이 필요하게 되어 장치들 간에 대기시간이 늘어남으로 인해 성능이 저하될 수 있다.

[0064] 도 5는 본 발명의 일 실시예에 따른 장치들 간에 대역폭을 줄이는 레이 트레이싱 장치를 설명하는 도면이다.

[0065] 도 5를 참조하면, 레이 트레이싱 장치(500)는 소프트웨어 유닛부(510) 및 전용 하드웨어 유닛부(530)를 포함할 수 있다. 소프트웨어 유닛부(510)는 레이 생성 모듈(511), 충돌 지점 계산 모듈(513) 및 쉐이딩 & 레이 셋업 모

듈(515)을 포함할 수 있다. 전용 하드웨어 유닛부(530)는 방문 교차 테스트 유닛(531) 및 리오더 버퍼 유닛(533)을 포함할 수 있다. 일 실시예에서, 소프트웨어 유닛부(510)는 중앙 처리 유닛(CPU)을 포함하여 구현될 수 있고, 전용 하드웨어 유닛부(530)는 그래픽 처리 유닛(GPU)을 포함하여 구현될 수 있다.

[0066] 소프트웨어 유닛부(510)는 소프트웨어를 통해, (a) 레이 트레이싱(ray tracing)을 위한 레이를 생성하는 레이 생성 연산과 (b) 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산하는 충돌 지점 계산 연산 및 (c) 충돌 지점 상에서의 컬러 값을 산출하는 쉐이딩(shading) 연산을 각각 처리할 수 있다. 즉, 소프트웨어 유닛부(510)에 포함된 각 모듈들에 의해서 각각의 연산이 독립적으로 처리될 수 있다.

[0067] 일 실시예에서, 소프트웨어 유닛부(510)는 쉐이딩 연산 단계에서 그림자(shadow) 레이 또는 세컨더리(secondary) 레이의 발생이 검출되면 그림자 레이 또는 세컨더리 레이에 관한 정보를 생성하여 레이 생성 연산 단계로 전달하는 레이 셋업(Set-up) 연산을 더 처리할 수 있다. 쉐이딩 연산과 레이 셋업 연산은 쉐이딩 & 레이 셋업 모듈(515)에 의해 수행될 수 있다.

[0068] 즉, 레이 트레이싱의 전 과정은 반복적(recursive)으로 수행될 수 있고, 이 과정에서 조명으로 인하여 그림자 레이의 발생이 필요하거나 또는 충돌된 삼각형의 재질로 인하여 세컨더리 레이의 발생이 필요한 경우 해당 정보들은 소프트웨어 유닛부(510)에 의해 레이 셋업 연산 단계에서 처리될 수 있다. 레이 셋업 연산 단계에서 처리된 정보들은 레이 생성 연산 단계로 전달될 수 있고, 레이 생성 연산 단계에서는 해당 정보를 기초로 그림자 레이 또는 세컨더리 레이를 발생시킬 수 있다.

[0069] 일 실시예에서, 소프트웨어 유닛부(510)는 레이 생성 연산 단계에서 프레임 내의 각 픽셀에 대해 시점 위치로부터 프라이머리 레이를 생성하여 해당 프라이머리 레이의 데이터 값을 전용 하드웨어 유닛부(530)로 일방 전송할 수 있다(S551). 레이 생성 연산은 레이 생성 모듈(511)에 의해 수행될 수 있다. 즉, 소프트웨어 유닛부(510)에서 전용 하드웨어 유닛부(530)로의 데이터 이동은 소프트웨어 유닛부(510)에 의해 생성된 레이에 관한 데이터 값으로 제한될 수 있다. 이 때, 레이에 관한 데이터 값은 레이의 벡터(vector) 값, 레이의 종류를 구별하기 위한 레이 타입(ray type) 값을 포함할 수 있다.

[0070] 일 실시예에서, 소프트웨어 유닛부(510)는 충돌 지점 계산 연산 단계에서 충돌 삼각형 정보와 레이를 기초로 충돌 삼각형과 레이 간의 충돌 지점의 좌표를 산출할 수 있다. 충돌 지점 계산 연산은 충돌 지점 계산 모듈(513)에 의해 수행될 수 있다. 충돌 지점 계산 연산은 트리 방문 & 교차 테스트에 비하여 연산량이 매우 작기 때문에 소프트웨어 유닛부(510)에 의해 소프트웨어적으로 처리되더라도 레이 트레이싱의 전체 성능에 영향을 미치지 않을 수 있다.

[0071] 전용 하드웨어 유닛부(530)는 소프트웨어 유닛부(510)과 연동하는 전용 하드웨어로서, 레이를 기초로 방문 교차 테스트 연산을 수행하는 방문 교차 테스트 유닛(531)과 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하는 리오더 버퍼(reorder buffer) 유닛(533)으로 구성될 수 있다.

[0072] 즉, 전용 하드웨어 유닛부(530)는 레이 트레이싱에서 계산량을 많이 필요로 하는 트리 방문 & 교차 테스트 연산만을 전용 하드웨어를 통해 처리함으로써 소프트웨어 유닛부(510)와의 데이터 전송량을 줄일 수 있다. 이 때, 전용 하드웨어 유닛부(530)에서 소프트웨어 유닛부(510)로 전송되는 데이터는 충돌된 삼각형들에 대한 정보만 해당되며, 충돌 지점에 관한 정보는 전송할 필요가 없다(S553).

[0073] 일 실시예에서, 방문 교차 테스트 유닛(531)은 레이를 기초로 가속 구조(Acceleration Structure, AS)를 탐색하여 상기 레이와 연관된 리프 노드를 결정하는 방문 연산과 상기 리프 노드에 포함된 모든 삼각형들에 관한 교차 테스트를 수행하여 레이와 충돌하는 충돌 삼각형을 결정하는 교차 테스트 연산을 각각 처리할 수 있다. 즉, 방문 교차 테스트 유닛(531)은 사전에 구축된 가속 구조를 이용하여 방문 연산과 교차 테스트 연산을 각각 독립적으로 처리할 수 있다.

[0074] 일 실시예에서, 리오더 버퍼 유닛(533)은 방문 교차 테스트 유닛(531)으로 입력되는 레이를 저장하는 제1 단계, 방문 교차 테스트 유닛(531)으로부터 출력되는 충돌 삼각형 정보를 수신하는 제2 단계 및 충돌 삼각형 정보와 연관된 레이의 우선순위에 따라 해당 충돌 삼각형 정보를 출력하는 제3 단계를 순차적으로 수행함으로써 소프트웨어 유닛부(510)로부터 일방 전송되는 레이의 순서에 대응하여 충돌 삼각형 정보를 일방 전송할 수 있다.

[0075] 보다 구체적으로, 리오더 버퍼 유닛(533)은 전용 하드웨어 유닛부(530)가 소프트웨어 유닛부(510)로부터 레이들을 수신하여 방문 교차 테스트를 수행하는 단계에서 방문 교차 테스트 유닛(531)으로 전달되는 레이에 관한 데이터를 할당받아 저장할 수 있다. 또한, 리오더 버퍼 유닛(533)은 방문 교차 테스트 유닛(531)으로부터 충돌 삼

각형 정보를 수신하여 연관된 레이에 대한 처리 동작이 완료되었음을 결정할 수 있다. 만약 처리 완료된 레이가 해당 리오더 버퍼 내에서 가장 오래된 레이(즉, 우선순위가 가장 높은 레이)에 해당하는 경우 리오더 버퍼 유닛(533)은 해당 레이를 출력 결과로서 제공할 수 있고, 전용 하드웨어 유닛부(530)에 의해 충돌 삼각형 정보만이 소프트웨어 유닛부(510)로 전송될 수 있다.

[0076] 일반적으로 전용 하드웨어 유닛부(530)의 방문 교차 테스트 유닛(531)에서 수행되는 방문 교차 테스트는 고성능 처리를 위하여 무순서(out-of-order)로 수행될 수 있다. 즉, 충돌 삼각형(hit triangle)이 입력된 레이의 순서대로 결정되지 않을 수 있다. 소프트웨어 유닛부(510)로부터 요청된 트리 방문 & 교차 테스트가 무순서(out-of-order)로 수행되는 반면, 인오더(in-order)로 완료되기 위하여 전용 하드웨어 유닛부(530)는 리오더 버퍼 유닛(533)을 포함하여 구현될 수 있다. 예를 들어, 리오더 버퍼 유닛(533)은 파이프라인 RISC 프로세서에서 많이 사용되는 리오더 버퍼(reorder buffer)로 구현될 수 있다.

[0078] 도 6은 도 5의 레이 트레이싱 장치에서 수행되는 장치들 간에 대역폭을 줄이는 레이 트레이싱 과정의 일 실시예를 설명하는 순서도이다.

[0079] 도 6을 참조하면, 레이 트레이싱 장치(500)는 소프트웨어 유닛부(510)를 통해 레이 트레이싱을 위한 레이를 생성하여 전용 하드웨어 유닛부(530)에게 일방 전송할 수 있다(단계 S610). 레이 트레이싱 장치(500)는 전용 하드웨어 유닛부(530)를 통해 레이를 기초로 방문 교차 테스트 연산을 수행할 수 있다(단계 S620). 레이 트레이싱 장치(500)는 전용 하드웨어 유닛부(530)를 통해 방문 교차 테스트 연산으로 출력되는 아웃오브오더(out-of-order) 결과를 인오더(in-order) 결과로 변경하여 소프트웨어 유닛부(510)에게 일방 전송할 수 있다(단계 S630).

[0080] 또한, 레이 트레이싱 장치(500)는 소프트웨어 유닛부(510)를 통해 레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산할 수 있다(단계 S640). 레이 트레이싱 장치(500)는 소프트웨어 유닛부(510)를 통해 충돌 지점 상에서의 컬러 값을 산출할 수 있다(단계 S650). 레이 트레이싱 장치(500)는 소프트웨어 유닛부(510)를 통해 프레임 내의 모든 픽셀에 대한 컬러 값이 결정된 경우 프레임에 관한 이미지를 출력할 수 있다(단계 S660).

[0082] 상기에서는 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

부호의 설명

[0084] 500: 레이 트레이싱 장치

510: 소프트웨어 유닛부 511: 레이 생성 모듈

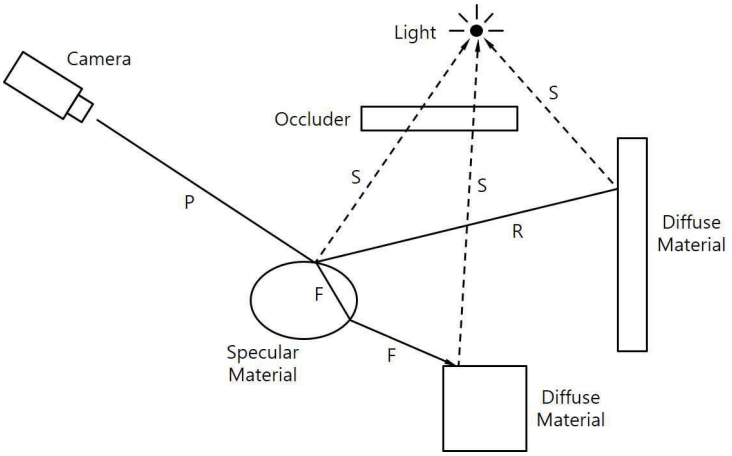
513: 충돌 지점 계산 모듈 515: 쉐이딩 & 레이 셋업 모듈

530: 전용 하드웨어 유닛부

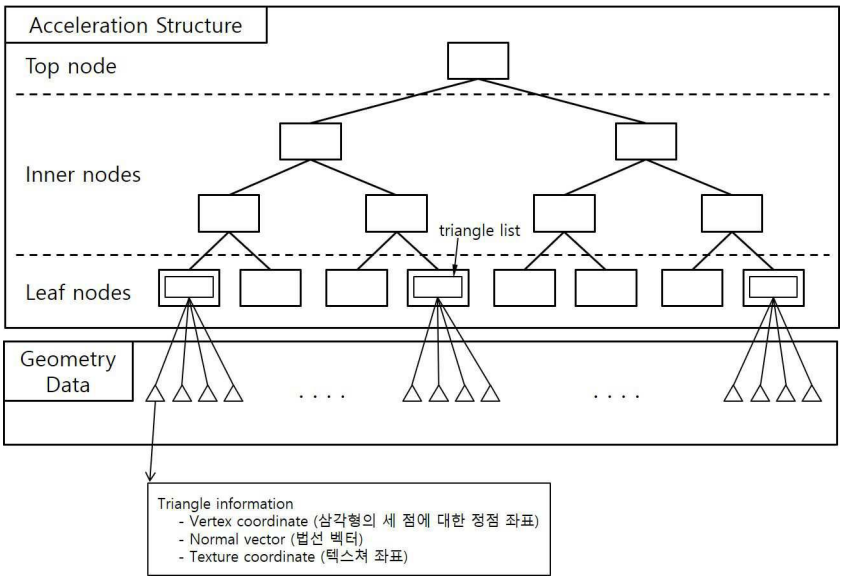
531: 방문 교차 테스트 유닛 533: 리오더 버퍼 유닛

도면

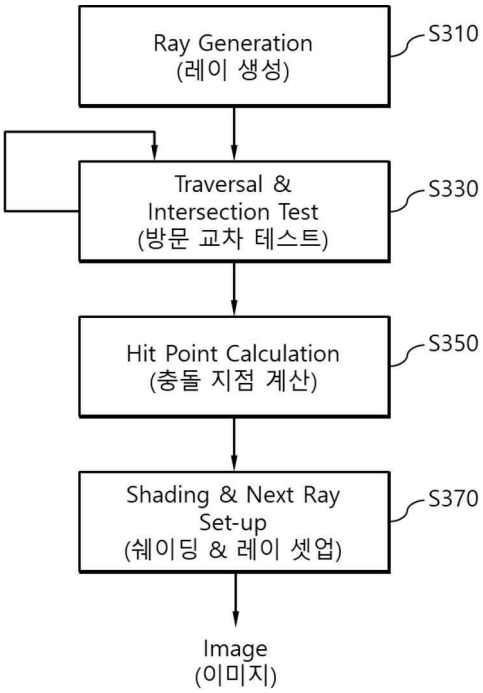
도면1


Light
Camera
Occluder
S
S
S
P
R
Diffuse
Material
F
F
Specular
Material
Diffuse
Material

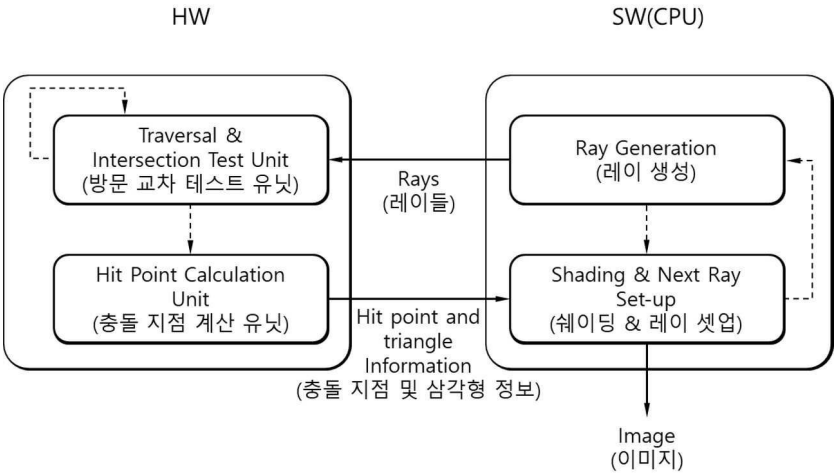
도면2


Acceleration Structure
Top node
Inner nodes
triangle list
Leaf nodes
Geometry
Data
Triangle information
- Vertex coordinate (삼각형의 세 점에 대한 정점 좌표)
- Normal vector (법선 벡터)
- Texture coordinate (텍스쳐 좌표)

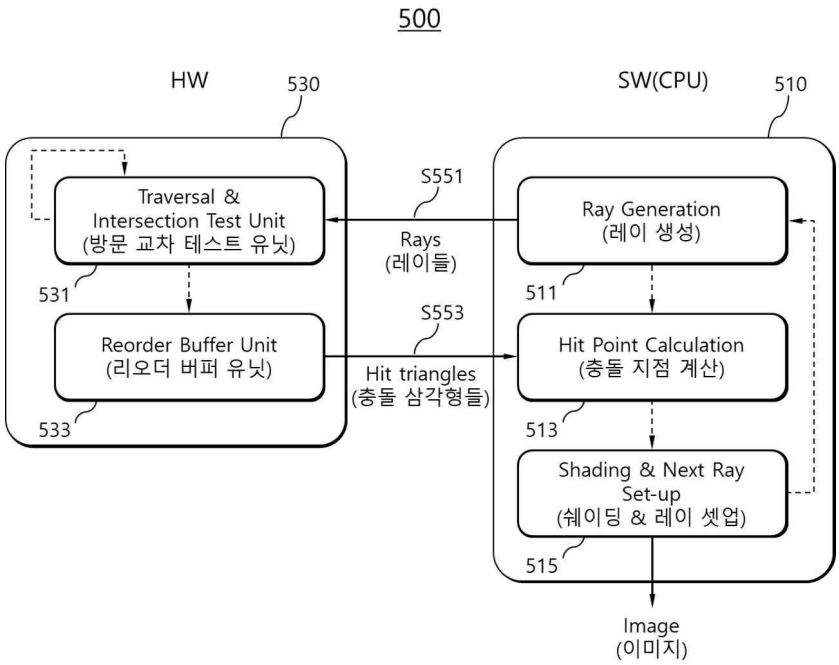
도면3


Ray Generation
(레이 생성)
S310
Traversal &
Intersection Test
(방문 교차 테스트)
S330
Hit Point Calculation
(충돌 지점 계산)
S350
Shading & Next Ray
Set-up
(쉐이딩 & 레이 셋업)
S370
Image
(이미지)

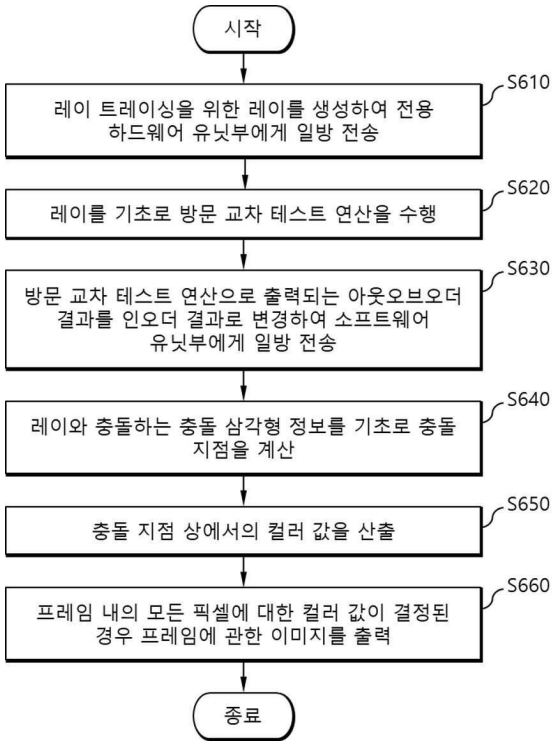
도면4


HW
SW(CPU)
Traversal &
Intersection Test Unit
(방문 교차 테스트 유닛)
Hit Point Calculation
Unit
(충돌 지점 계산 유닛)
Rays
(레이들)
Hit point and
triangle
Information
(충돌 지점 및 삼각형 정보)
Ray Generation
(레이 생성)
Shading & Next Ray
Set-up
(쉐이딩 & 레이 셋업)
Image
(이미지)

도면5


500
HW
530
SW(CPU)
510
Traversal &
Intersection Test Unit
(방문 교차 테스트 유닛)
531
Reorder Buffer Unit
(리오더 버퍼 유닛)
533
S551
Rays
(레이들)
S553
Hit triangles
(충돌 삼각형들)
Ray Generation
(레이 생성)
511
Hit Point Calculation
(충돌 지점 계산)
513
Shading & Next Ray
Set-up
(쉐이딩 & 레이 셋업)
515
Image
(이미지)

도면6


시작
레이 트레이싱을 위한 레이를 생성하여 전용 하드웨어 유닛부에게 일방 전송
S610
레이를 기초로 방문 교차 테스트 연산을 수행
S620
방문 교차 테스트 연산으로 출력되는 아웃오브오더 결과를 인오더 결과로 변경하여 소프트웨어 유닛부에게 일방 전송
S630
레이와 충돌하는 충돌 삼각형 정보를 기초로 충돌 지점을 계산
S640
충돌 지점 상에서의 컬러 값을 산출
S650
프레임 내의 모든 픽셀에 대한 컬러 값이 결정된 경우 프레임에 관한 이미지를 출력
S660
종료