(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년08월30일
(11) 등록번호 10-1651827
(24) 등록일자 2016년08월23일

(51) 국제특허분류(Int. Cl.)
G06T 17/20 (2006.01)
(52) CPC특허분류
G06T 17/20 (2013.01)
(21) 출원번호 10-2015-0027596
(22) 출원일자 2015년02월26일
심사청구일자 2015년02월26일
(56) 선행기술조사문헌
KR1020030004017 A*
KR101467735 B1
다중해상도 복셀화 기법, 한국정보기술학회논문지 제10권제6호(2012.06.)*
GPU-based Scalable Volumetric Reconstruction for Multi-view Stereo, IVCNZ. 2011(2011.11.29.)*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
한발대학교 산학협력단
대전광역시 유성구 동서대로 125 (덕명동)
(72) 발명자
안은영
대전광역시 유성구 봉명로 48, 802동 903호 (원신흥동, 신안인스빌리베라)
이수열
대전광역시 동구 옥천로 38, 105동 2202호 (신흥동, 신흥마을아파트)
(74) 대리인
특허법인 웰

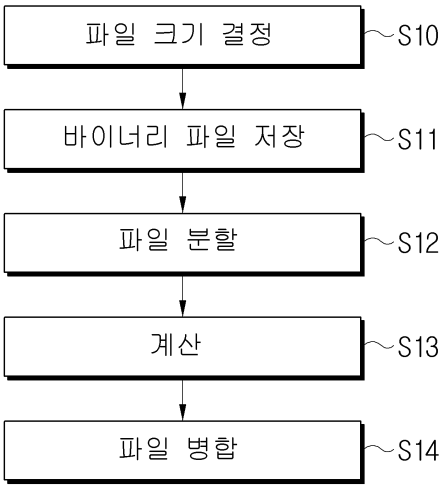
전체 청구항 수 : 총 6 항 심사관 : 이병우

(54) 발명의 명칭 **파일 및 병렬 처리를 이용한 객체의 복셀화 방법**

(57) 요 약

폴리곤으로 구성된 3차원 모델에 대해 표면 복셀화를 수행하여 3차원 모델의 표면 정보를 추출하고, 이 정보를 기초로 내부 복셀화를 수행하여 고해상도로 3차원 모델을 복셀화하는 방법에 관한 것으로, (a) 복셀화 계산 시에 추출되는 복셀 정보를 바이너리 파일로 저장하는 파일 처리 단계, (b) 상기 단계 (a)에서 저장된 바이너리 파일에 따른 복셀화 계산 시에 다중 스레드(Multi-Thread)를 이용하여 병렬로 처리하는 병렬 처리 단계를 포함하는 구성을 마련하여, 복셀화 과정에서 계산 시간을 단축하고, 메모리 사용량을 최소화할 수 있다.

대 표 도 - 도3



이 발명을 지원한 국가연구개발사업
과제고유번호 2015052427
부처명 미래창조과학부
연구관리전문기관 한국연구재단
연구사업명 기초연구사업
연구과제명 석조 문화재의 체계적 관리를 위한 노후화 예측 시뮬레이션 기술 개발
기 여 율 1/2
주관기관 한밭대학교 산학협력단
연구기간 2015.11.01 ~ 2016.10.31
이 발명을 지원한 국가연구개발사업
과제고유번호 s07201522022015
부처명 문화체육관광부
연구관리전문기관 스포츠산업기술진흥원
연구사업명 스포츠산업기술개발사업
연구과제명 스포츠과학과 첨단 HMD기술을 활용한 실감형 익스트림 스포츠 서비스 개발
기 여 율 1/2
주관기관 한밭대학교 산학협력단
연구기간 2015.12.30 ~ 2016.12.29

공지예외적용 : 있음

명 세 서

청구범위

청구항 1

3차원 모델의 삼각 폴리곤과 육면체의 교차 판정을 이용하여 표면 복셀화(Surface Voxelization)를 수행하고, 표면 복셀화 정보를 이용한 내부 공간을 채우는 내부 복셀화(Inner Voxelization)를 수행하여 고해상도로 3차원 모델을 복셀화하는 방법으로서,

(a) 복셀화 계산 시에 추출되는 복셀 정보를 바이너리 파일로 저장하는 파일 처리 단계,

(b) 상기 단계 (a)에서 저장된 바이너리 파일에 따른 복셀화 계산 시에 다중 스레드(Multi-Thread)를 이용하여 병렬로 처리하는 병렬 처리 단계를 포함하고,

상기 내부 복셀화에서 복셀 정보는 각 축 별로 내부 복셀화를 수행한 후에 3축(x, y, z)의 모든 방향에 대해 중복 검출되는 복셀에 대해서만 내부 복셀 정보로 간주하는 것을 특징으로 하는 복셀화 방법.

청구항 2

삭제

청구항 3

제1항에 있어서,

상기 단계 (a)에서 상기 바이너리 파일은 복셀 정보를 1 바이트(Byte)의 부울(Bool) 형태로 저장되는 것을 특징으로 하는 복셀화 방법.

청구항 4

제1항에 있어서,

상기 단계 (a)에서 파일 처리는 스레드 수만큼 파일을 분할하여 사용하고 계산 완료 후 병합하는 것을 특징으로 하는 복셀화 방법.

청구항 5

제4항에 있어서,

상기 단계 (a)에서 스레드 수로 분할되는 축(z-축)의 계산되는 해상도를 파일 생성 시에만 스레드의 배수로 설정하여 파일의 크기를 결정하고, 분할 시의 축 방향으로 동일하게 파일을 병합하는 것을 특징으로 하는 복셀화 방법.

청구항 6

제1항에 있어서,

상기 단계 (b)에서 사용되는 스레드 수에 대응하여 초기화된 바이너리 파일을 생성하고, 상기 스레드 수에 대응하는 계산 영역을 분할하여 각 영역의 복셀화 계산 시에 연동된 바이너리 파일을 사용하는 것을 특징으로 하는 복셀화 방법.

청구항 7

제6항에 있어서,

상기 표면 복셀화의 계산 시에 각 스레드에서는 3차원 모델의 폴리곤 정보를 읽어 들여 자신의 계산 영역의 폴리곤 정보인지 판별 후 표면 복셀화 작업을 수행하며, 교차 판별된 복셀 정보는 연동된 바이너리 파일에 적용하고,

상기 내부 복셀화의 계산 시에 각 스레드에서는 자신의 계산 영역의 정보를 병합된 표면 복셀화 파일에서 읽어들여 내부 저장 계산을 수행하며 저장으로 판별된 복셀 정보는 연동된 바이너리 파일에 적용하는 것을 특징으로 하는 복셀화 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 파일 처리 및 병렬 처리를 이용한 객체의 복셀화 방법에 관한 것으로, 특히 폴리곤으로 구성된 3차원 모델에 대해 표면 복셀화를 수행하여 3차원 모델의 표면 정보를 추출하고, 이 정보를 기초로 내부 복셀화를 수행하여 고해상도로 3차원 모델을 복셀화하는 방법에 관한 것이다.

배 경 기 술

[0002] 현재 3차원 그래픽스 분야에서는 이전에 목표로 했던 높은 정보 처리량뿐 아니라 더욱 세밀하고 실감있는 영상을 만들기 위한 노력이 계속 되고 있다. 그러나 아무리 다각형 정보의 개수를 증가시켜 실감영상을 구현한다 해도 물체의 표면 정보만을 갖고 있는 다각형 정보를 통해서는 물체 내부를 표현할 수 없다는 근본적인 문제가 있었다. 때문에 기존의 다각형 정보가 갖고 있는 한계를 극복하고 실감 영상을 구현하기 위해 영상 기반 렌더링이나 볼륨 렌더링 등 새로운 그래픽 정보를 사용한 렌더링 기법에 대한 많은 연구가 진행되기 시작했다.

[0003] 이 중 볼륨 렌더링 기법은 렌더링 기법 중의 하나로 복셀(voxel)을 기본 단위로 하는 3차원 스칼라 밀도값(scalar intensity)의 집합이 볼륨 정보를 사용하여 영상을 만들어내는 렌더링 기법이다. 최근에는 범용 그래픽 하드웨어에 내장된 텍스쳐 매핑 유닛을 통해 볼륨 정보의 처리가 가능해져서 기존의 의학용 영상 처리분야뿐 아니라 3차원 게임 등의 일반 분야에까지 그 적용 영역이 확대되고 있는 추세이다. 하지만 범용 그래픽 하드웨어를 사용하여 효과적으로 볼륨 렌더링을 구현하는 데에는 아직도 많은 문제점이 있다. 볼륨 정보가 3차원 다각형 정보에 비해 그 크기가 매우 크고 그 렌더링 기법 자체가 다각형 기반 렌더링과 많은 차이점이 있기 때문이다.

[0004] 렌더링 작업은 폴리곤 메쉬 방법을 이용하여 3차원 모델을 생성하였는지 복셀 볼륨 데이터 방법을 이용하여 3차원 모델을 생성하였는지에 따라 각각 다른 방식으로 진행된다. 따라서, 폴리곤 메쉬 방법을 이용하여 생성된 3차원 모델을, 복셀 볼륨 데이터를 사용하는 것이 유리한 애니메이션 작업 등에 사용하기 위해서는 이 폴리곤 메쉬 방법에 따른 3차원 모델을 복셀화(Voxelization)하는 과정이 반드시 수반되어야 한다. 복셀화란 각각의 3차원 좌표가 3차원 그리드 상에서 물체의 내부에 속하는지 또는 외부에 속하는지의 정보를 저장하는 것을 의미한다.

[0005] 최근, 컴퓨터 그래픽스 응용분야에서는 처리의 효율성을 높이기 위해 삼각 폴리곤으로 구성된 표면 모델(surface model)을 주로 사용하여 현실세계의 객체를 표현한다.

[0006] 그러나 객체의 외부만이 아니라 내부의 정보를 필요로 하는 과학적 시뮬레이션을 위해서는 표면모델이 아닌 솔리드 모델링(Solid Modeling)이 필요하다. 이 경우에 폴리곤(Polygon)으로 구성된 표면 모델로부터 볼륨 객체를 생성하기 위해 입체 화상(3D 화소)을 구성하는 복셀화 과정이 필요하다. 최근, 표면 모델링 객체를 대상으로 하는 복셀화에 관한 연구가 국내외적으로 다양하게 진행되어 왔다.

[0007] 지금까지 제시된 연구결과를 살펴보면, 객체의 모양이나 해상도에 따라 복셀화 과정에서 발생되는 다양한 문제로 인해 완벽하게 볼륨 객체를 생성하지 못하는 경우가 있음을 확인할 수 있었다. 기존의 제안된 방법에서는 고밀도의 폴리곤 모델에 관한 복셀화 과정이나 저해상도 복셀화 과정에서, 객체를 구성하는 폴리곤들 간의 간격보다 복셀의 크기가 큰 경우에는 내부 복셀화 과정에서 영역 채우기 오류가 발생할 수 있다. 그 외에도 고해상도의 복셀화 수행 시에는 해상도가 높아짐에 따라 계산량 및 처리시간의 증가가 기하급수적으로 일어나게 된다. 또한, 계산과정에서 필요한 정보를 저장하기 위한 메모리 사용량의 증가로 인해 범용 컴퓨터에서 고해상도의 복셀화 수행이 사실상 어렵다.

[0008] 이러한 기술의 일 예가 하기 문헌 1 및 2 등에 개시되어 있다.

[0009] 예를 들어, 하기 특허문헌 1에는 영상 데이터를 획득하는 영상 데이터 획득부, 상기 영상 데이터를 이용하여 볼륨 데이터를 생성하는 볼륨 데이터 생성부, 상기 볼륨 데이터를 슬라이싱하여 복수의 2차원 슬라이스로 분할하

는 볼륨 데이터 슬라이싱부, 그래픽 연산을 수행하는 그래픽 처리 유닛을 포함하되, 상기 그래픽 처리 유닛은 상기 복수의 2차원 슬라이스를 저장하는 텍스쳐 메모리, 2차원 텍스쳐 매핑 기능을 지원하며 상기 복수의 2차원 슬라이스를 2차원 텍스쳐로 간주하여 2차원 텍스쳐 샘플링을 수행하는 텍스쳐 매핑 유닛 및 상기 2차원 텍스쳐 샘플링의 샘플링값들을 이용하여 볼륨 렌더링을 수행하는 연산 프로세서로 이루어지는 영상 처리 장치에 대해 개시되어 있다.

[0010] 또 하기 특허문헌 2에는 각 셀의 좌표에 대응하는 배열의 좌표에 매칭 큐브 케이스를 저장하는 매칭 큐브 케이스 저장 단계, 매칭 큐브 케이스를 구분한 후 미리 정의된 테이블을 이용해서 추출될 삼각형이 몇 개의 인덱스가 필요한지를 저장하는 삼각형화 인덱스 카운트 저장 단계, 유일한 정점을 추출하기 위해 각 셀 당 3개의 모서리를 검사하여 셀의 좌표와 대응하는 배열의 좌표에 해당하는 유일한 정점 케이스를 저장하는 유일한 정점 케이스 저장 단계, 셀마다 추출해야 하는 유일한 정점의 수를 셀과 대응하는 배열의 좌표에 저장하는 유일한 정점 카운트 저장 단계를 포함하여 각 셀을 분류하는 셀 분류 단계, 삼각형화 인덱스의 총수와 유일한 정점의 총수를 저장하는 히스토피라미드 구축 단계, 정점의 총 수에 키 인덱스 값을 더한 값을 인덱스 배열에서 셀에 대응하는 위치에 저장하는 정점 보간 및 인덱스 할당 단계, 매칭 큐브 케이스를 참조하여 셀 내부 삼각형 인덱스를 가져오는 인덱스 계산 단계를 구비하는 메쉬 연결성을 유지하는 병렬적 등위면 추출 방법에 대해 개시되어 있다.

선행기술문헌

특허문헌

[0011] (특허문헌 0001) 대한민국 등록특허공보 제10-1353303호(2014.01.14 등록)

(특허문헌 0002) 대한민국 등록특허공보 제10-1017974호(2011.02.21 등록)

발명의 내용

해결하려는 과제

[0012] 그러나, 상술한 바와 같은 종래의 기술에서는 3차원 모델의 표면 객체에 대한 고해상도 복셀화 과정에서 복셀 입자의 크기를 설정하는 해상도에 따라 계산량 및 메모리 사용량이 클 수 있다. 즉, 계산량은 생성하고자 하는 복셀의 크기가 작아짐에 따라서 기하급수적으로 수행시간이 증가할 수 있으며, 메모리 사용량은 수행하는 컴퓨터의 사용 가능한 메모리를 초과하여 오류를 발생시킬 수 있다는 문제가 있었다.

[0013] 또 상기와 같은 종래의 기술에서는 내부가 비어있는 3차원 모델의 단면에 대해 내부 복셀화 과정에서 내부가 채워지는 문제가 발생하여 정확한 복셀화가 곤란하며, 이에 따른 처리 시간도 지연된다는 문제도 있었다.

[0014] 본 발명의 목적은 상술한 바와 같은 문제점을 해결하기 위해 이루어진 것으로서, 내부 복셀화 수행 시에는 각 축 방향으로 내부 복셀화 계산된 정보를 병합하여 사용함으로써 복잡한 형태의 3차원 모델을 다양한 해상도에서도 완벽하게 내부 복셀화를 수행할 수 있는 복셀화 방법을 제공하는 것이다.

[0015] 본 발명의 다른 목적은 복셀화 과정에서 계산 시간을 단축하고, 메모리 사용량을 최소화하는 복셀화 방법을 제공하는 것이다.

과제의 해결 수단

[0016] 상기 목적을 달성하기 위해 본 발명에 따른 복셀화 방법은 3차원 모델의 삼각 폴리곤과 육면체의 교차 판정을 이용하여 표면 복셀화(Surface Voxelization)를 수행하고, 표면 복셀화 정보를 이용한 내부 공간을 채우는 내부 복셀화(Inner Voxelization)를 수행하여 고해상도로 3차원 모델을 복셀화하는 방법으로서, (a) 복셀화 계산 시에 추출되는 복셀 정보를 바이너리 파일로 저장하는 파일 처리 단계, (b) 상기 단계 (a)에서 저장된 바이너리 파일에 따른 복셀화 계산 시에 다중 스레드(Multi-Thread)를 이용하여 병렬로 처리하는 병렬 처리 단계를 포함하는 것을 특징으로 한다.

[0017] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 내부 복셀화에서 복셀 정보는 각 축 별로 내부 복셀화를 수행한 후에 3축(x, y, z)의 모든 방향에 대해 중복 검출되는 복셀에 대해서만 내부 복셀 정보로 간주하는 것을 특징으로 한다.

[0018] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 단계 (a)에서 상기 바이너리 파일은 복셀 정보를 1 바이트(Byte)의 부울(Bool) 형태로 저장되는 것을 특징으로 한다.

[0019] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 단계 (a)에서 파일 처리는 스레드 수만큼 파일을 분할하여 사용하고 계산 완료 후 병합하는 것을 특징으로 한다.

[0020] 삭제

[0021] 삭제

[0022] 삭제

[0023] 삭제

[0024] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 단계 (a)에서 스레드 수로 분할되는 축(z-축)의 계산되는 해상도를 파일 생성 시에만 스레드의 배수로 설정하여 파일의 크기를 결정하고, 분할 시의 축 방향으로 동일하게 파일을 병합하는 것을 특징으로 한다.

[0025] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 단계 (b)에서 사용되는 스레드 수에 대응하여 초기화된 바이너리 파일을 생성하고, 상기 스레드 수에 대응하는 계산 영역을 분할하여 각 영역의 복셀화 계산 시에 연동된 바이너리 파일을 사용하는 것을 특징으로 한다.

[0026] 또 본 발명에 따른 복셀화 방법에 있어서, 상기 표면 복셀화의 계산 시에 각 스레드에서는 3차원 모델의 폴리곤 정보를 읽어 들여 자신의 계산 영역의 폴리곤 정보인지 판별 후 표면 복셀화 작업을 수행하며, 교차 판별된 복셀 정보는 연동된 바이너리 파일에 적용하고, 상기 내부 복셀화의 계산 시에 각 스레드에서는 자신의 계산 영역의 정보를 병합된 표면 복셀화 파일에서 읽어 들여 내부 저장 계산을 수행하며 저장으로 판별된 복셀 정보는 연동된 바이너리 파일에 적용하는 것을 특징으로 한다.

발명의 효과

[0027] 상술한 바와 같이, 본 발명에 따른 복셀화 방법에 의하면, 표면 및 내부 복셀화 계산 시에는 추출되는 복셀 정보를 바이너리 파일로 저장함으로써 컴퓨터의 메모리 사용량을 줄여 저사양 컴퓨터에서도 고해상도의 복셀화를 수행할 수 있다는 효과가 얻어진다.

[0028] 또, 본 발명에 따른 복셀화 방법에 의하면, 복셀화 계산 시에 다중 스레드를 이용한 병렬 처리를 적용하여 처리 시간을 단축할 수 있다는 효과도 얻어진다.

도면의 간단한 설명

[0029] 도 1은 삼각형과 육면체의 교차에 따라 표면 복셀을 추출하는 과정을 설명하기 위한 도면,

도 2는 3차원 모델의 단면에 대해 내부 복셀화 과정에서의 문제점을 제시하는 도면,

도 3은 본 발명에 따른 객체의 복셀화 방법에 적용되는 파일 처리의 과정을 나타내는 도면,

도 4는 표면과 내부 복셀화 계산 시에 분할되어 사용된 바이너리 파일을 하나의 파일로 병합하는 형태를 나타내는 도면,

도 5는 본 발명에 따른 객체의 복셀화 방법에 적용되는 병렬 처리의 과정을 나타내는 도면,

도 6은 본 발명에 따른 병렬 처리에 의해 저장되는 파일의 형태를 나타내는 도면,

도 7은 본 발명에 적용되는 모델의 형상의 일 예를 나타내는 도면,

도 8은 본 발명에 따른 시험 환경에서 구현된 프로그램을 이용하여 복셀화 수행시의 CPU 및 메모리 사용 현황을 나타낸 도면,

도 9는 내부 복셀화가 완료된 시험 모델들의 각 축별 중심 기준의 단면도를 나타낸 도면.

발명을 실시하기 위한 구체적인 내용

[0030] 본 발명의 상기 및 그 밖의 목적과 새로운 특징은 본 명세서의 기술 및 첨부 도면에 의해 더욱 명확하게 될 것이다.

[0031] 먼저 본 발명의 기본 개념에 대해 설명한다.

[0032] 본 발명에서는 표면 객체(surface object)에 대한 고해상도 복셀화 과정에서 발생할 수 있는 상술한 바와 같은 문제점들을 해결 즉, 복셀화 계산 시의 메모리 사용량 문제를 해결하기 위해 계산된 복셀 정보를 저장할 수 있는 바이너리(Binary) 형태의 파일을 복셀화를 위한 계산 전에 미리 생성하여 복셀화 과정 중 산출되는 정보를 저장한다.

[0033] 내부 복셀화 수행 시에 발생하는 영역 채우기 오류를 보완하기 위한 방안으로는 3차원의 각 축별로 생성된 내부 복셀화 정보를 병합하여 완벽한 내부 복셀화가 가능하도록 한다. 또한 처리시간을 단축시키기 위한 방안으로 파일을 처리함에 있어서 다중 스레드(Multi-Thread)를 이용한 병렬처리를 실행한다.

[0034] 본 발명에 적용되는 복셀화 기법으로서, 복셀화는 우선 3차원 모델의 삼각 폴리곤과 육면체의 교차 판정을 이용하여 표면 복셀화(Surface Voxelization)를 수행하고, 표면 복셀화 정보를 이용한 내부 공간을 채우는 내부 복셀화(Inner Voxelization)를 수행한다.

[0035] 즉, 폴리곤으로 구성된 3차원 모델을 내부 복셀화하기 위해서는 표면 복셀화를 우선적으로 수행한다. 이를 위해 3차원 모델의 가로, 세로, 높이에 해당하는 3개의 축 중에서 한 개를 기준 축으로 할당하여 해상도를 설정하며, 복셀의 크기는 식(1)과 같이 정의할 수 있다.

[0036] $V_{size} = \alpha_{Slect} / \varepsilon_{Resolution}$ … 식(1)

[0037] 여기서, V_{size}는 복셀의 크기, α_{Slect}는 선택된 축의 크기, $\varepsilon_{Resolution}$는 적용 해상도로 나타낸다.

[0038] 3차원 모델은 수많은 폴리곤으로 구성되며 일반적으로 한 개의 면은 삼각형 형태로 구성된다. 또한 복셀은 3차원 상에서 육면체로 구성된다. 따라서 표면에 대한 복셀화는 도 1에 도시된 바와 같이, 3차원 상의 삼각형과 육면체의 교차 여부를 판별하여 교차가 일어나는 위치에 객체 표면에 대한 복셀을 생성한다.

[0039] 도 1은 삼각형과 육면체의 교차에 따라 표면 복셀을 추출하는 과정을 설명하기 위한 도면이다.

[0040] 삼각형 형태의 폴리곤과 교차하는 복셀을 판별하기 위해서는 우선 삼각형을 구성하는 3개의 정점에 해당하는 x, y, z 축의 최대 최소값을 추출하여 사각 영역을 생성한다. 생성된 사각 영역을 복셀 크기로 분할하여 육면체를 만들고, 삼각형 형태의 폴리곤과 교차 판별 알고리즘을 적용하여 도 1에서와 같이 표면 복셀을 추출한다.

[0041] 다음에 표면 복셀화가 완료되면 복셀화된 객체의 내부를 채우는 내부 복셀화를 수행한다. 내부 복셀화 수행 시에는 3차원 상의 각 축 별로 주사선 알고리즘(Scan line algorithm)을 적용하여 내부를 복셀로 채우게 된다. 빠른 내부 복셀화를 위해 단방향으로 내부를 채우는 방법을 사용할 수 있지만, 이런 경우 복셀화 수행 시에 설정하는 해상도 또는 복셀화를 수행하는 3차원 모델의 형태에 따라 내부를 채우지 않아야 할 부분도 채워지는 경우가 발생한다.

[0042] 예를 들면, 도 2에서와 같이 내부가 비어있는 3차원 모델의 단면에 대해 내부 복셀화 과정에서 내부가 채워지는 문제가 발생할 수 있다.

[0043] 도 2는 3차원 모델의 단면에 대해 내부 복셀화 과정에서의 문제점을 제시하는 도면이다.

[0044] 이러한 경우를 방지하기 위해 본 발명에서는 각 축 별로 내부 복셀화를 수행한 후, 3축(x, y, z)의 모든 방향에 대해 중복 검출되는 복셀에 대해서만 내부 복셀로 간주한다.

[0045] 상술한 바와 같이, 3차원 모델을 복셀화 하는 경우 복셀 입자의 크기를 설정하는 해상도에 따라 계산량 및 메모리 사용량이 클 수 있다. 계산량은 생성하고자 하는 복셀의 크기가 작아짐에 따라서 기하급수적으로 수행시간이 증가할 수 있으며 메모리 사용량은 수행하는 컴퓨터의 사용 가능한 메모리를 초과하여 오류를 발생시킬 수 있다.

[0046] 본 발명에서는 이 두 가지 문제를 해결하기 위해 복셀화를 수행하는 컴퓨터의 스레드를 최대한 활용하는 병렬 처리로 계산 시간을 단축하고, 파일 처리를 통해 계산에 사용되는 데이터를 바이너리 파일로 저장하여 메모리 사용량을 최소화 한다.

[0047] 이하, 본 발명의 구성을 도면에 따라서 설명한다.

[0048] 도 3은 본 발명에 따른 객체의 복셀화 방법에 적용되는 파일 처리의 과정을 나타내는 도면이고, 도 4는 표면과 내부 복셀화 계산 시에 분할되어 사용된 바이너리 파일을 하나의 파일로 병합하는 형태를 나타내는 도면이고, 도 5는 본 발명에 따른 객체의 복셀화 방법에 적용되는 병렬 처리의 과정을 나타내는 도면이며, 도 6은 본 발명에 따른 병렬 처리에 의해 저장되는 파일의 형태를 나타내는 도면이다.

[0049] 본 발명에 적용되는 파일 처리로서, 도 3에 도시된 바와 같이, 먼저 복셀화 수행 시에 설정된 해상도 및 사용되는 스레드(Thread) 수에 따라 파일의 크기를 결정된다(S10).

[0050] 다음에 복셀화 과정에서 사용되는 바이너리 파일은 복셀 정보를 1 바이트(Byte)의 부울(Bool) 형태로 저장한다(S11). 복셀 정보를 비트 단위로 저장하면 파일 크기를 줄일 수 있겠지만 파일 위치 정보 색인 시의 처리 효율성을 위해 1 바이트 단위로 저장한다.

[0051] 한편, 병렬 처리를 위해 다중 스레드를 사용하는 경우에는 한 개의 파일에 대해 읽기는 다중 처리가 가능하고 쓰기는 다중 처리가 불가능하다. 이를 위해 본 발명에서는 사용되는 스레드 수만큼 파일을 분할하여 사용하고(S12), 계산 완료 후 병합한다(S13, S14).

[0052] 또한 파일의 분할 및 병합 그리고 파일 위치 색인 계산 시의 빠른 처리를 위해 스레드 수로 분할되는 축(z-축)의 계산되는 해상도를 파일 생성 시에만 스레드의 배수로 설정하여 파일의 크기를 결정한다.

[0053] 병렬 처리 시에 사용되는 분할 파일의 크기는 식 (2)와 같이 정의할 수 있으며, 최종적으로 병합된 파일의 크기는 식 (3)과 같이 정의할 수 있다.

[0054] $$P_{size} = ((\alpha_x \times \alpha_y \times \beta_z)/V_{Size})/T_{cnt}$$

[0055] $$\beta_z = (\alpha_z + (T_{cnt} - (\alpha_z/V_{Size}) \bmod T_{cnt})) \times V_{Size} \quad \cdots \text{식(2)}$$

[0056] $$F_{size} = \sum_{i=1}^{T_{cnt}} P_{size} \quad \cdots \text{식(3)}$$

[0057] 여기서, P_{size}은 병렬 처리 시에 사용되는 분할 바이너리 파일의 크기, α_x는 x축의 크기, α_y 는 y축의 크기, α_z는 z축의 크기, β_z는 스레드의 배수로 설정된 z축의 크기, T_{cnt}는 사용되는 스레드의 개수, F_{Size}는 병합된 파일의 크기로 나타내며, V_{Size}는 식(1)에서 정의한 복셀의 크기이다.

[0058] 도 4에 도시된 바와 같이, 본 발명에서는 표면과 내부 복셀화 계산 시에 분할되어 사용된 바이너리 파일을 하나의 파일로 병합하며, 분할 시의 축 방향으로 동일하게 병합하여 파일 위치 정보의 색인 처리 시에 효율성을 높였다.

[0059] 또한, 본 발명에 적용되는 병렬 처리로서, 도 5에 도시된 바와 같이, 복셀화를 병렬 처리할 경우에는 사용되는 스레드 수만큼 false(0x00)으로 초기화된 바이너리 파일을 생성한다(S20). 또 도 6에서와 같이 스레드 수만큼 계산 영역을 분할하여(S21) 각 영역의 복셀화 계산 시에 연동된 바이너리 파일을 사용하고, 복셀 정보로 판별된 위치 정보에 해당하는 연동된 바이너리 파일 위치에 true(0x01) 값을 덮어쓰기 한다(S22).

[0060] 표면 복셀화 계산 시에 각 스레드에서는 3차원 모델의 폴리곤 정보를 읽어 들여 자신의 계산 영역의 폴리곤 정보인지 판별 후 표면 복셀화 작업을 수행하며, 교차 판별된 복셀 정보는 연동된 바이너리 파일에 적용한다

(S23).

[0061] 또 내부 복셀화 계산 시에 각 스레드에서는 자신의 계산 영역의 정보를 병합된 표면 복셀화 파일에서 읽어 들여 내부 채우기 계산을 수행하며 채우기로 판별된 복셀 정보는 연동된 바이너리 파일에 적용한다(S24).

[0062] 각 축 별로 내부 복셀화 계산이 완료되면 최종 내부 복셀화 파일을 생성하기 이전과 동일한 방식으로 분할된 바이너리 파일을 생성하고, 각 스레드 영역에서 각 축별 내부 복셀화 병합 파일의 동일 위치 해당하는 값의 중복 여부를 판별하여 연동된 바이너리 파일에 적용하고, 각 스레드의 계산이 완료되면 분할된 바이너리 파일을 병합하여 최종 내부 복셀화 파일을 생성한다(S25).

[0063] 본 발명의 파일 및 병렬 처리를 이용한 표면 객체의 복셀화 시스템에 따른 구현 결과를 시험하였다.

[0064] 시험 환경은 CPU(Intel i7-2600 3.4Ghz, 8 Threads), 메모리(4GB), 하드 디스크(1TB, 7200 rpm)를 가진 개인용 컴퓨터에서 Visual C++로 내부 복셀화 프로그램을 구현하여 수행하였다. 시험 데이터는 Princeton Shape Benchmark(PSB)의 3차원 모델 중 5개를 선택하여 사용하였다. 시험에 사용된 객체에 대한 정보는 표 1과 같으며 모델의 형상은 도 7과 같다.

[0065] 표 1은 본 발명에 따른 시험에 사용된 객체에 대한 정보를 나타내고, 도 7은 본 발명에 적용되는 모델의 형상의 일 예를 나타내는 도면이다.

표 1

[0066]

Model Name		m23	m110	m329	m374	m1601
# of Point		462	453	6191	886	6727
# of Polygon		878	902	12378	1770	13450
Shape		Spider	Rabbit	Hand	Body	Chessman
Axial Size	x	0.450631	0.95	0.85366	0.526195	0.531126
	y	0.400936	0.44555	0.329297	0.282288	0.52981
	z	0.95	0.694399	0.95	0.95	0.95

[0067] 본 발명에 적용되는 프로그램을 이용한 복셀화 시험은 해상도를 결정하는 기준 축은 x축, 해상도는 250, 사용하는 스레드 수는 8개로 설정하여 수행하였다.

[0068] 표 2는 복셀화 수행 완료 후의 시험 모델별 정보를 나타낸다.

표 2

[0069]

Model Name		m23	m110	m329	m374	m1601
Axis Resolution	x	250	250	250	250	250
	y	552	528	641	462	249
	z	224	392	720	832	448
# of Voxel	Surface	127,936	398,523	638,473	205,191	282,077
	Inner	2,082,014	16,016,824	22,736,219	40,733,286	5,859,827
	Total	29,232,000	51,744,000	115,380,000	96,096,000	27,888,000
File Size(Byte)		29,232,032	51,744,032	115,380,032	96,096,032	27,888,032

[0070] 기준 축을 x축으로 설정하여 수행하였기 때문에 x축에 해당하는 해상도는 동일하며, y축과 z축은 크기에 비례하여 해상도가 설정된 것을 확인할 수 있다. 표면 복셀의 수는 외부에 노출되는 복셀의 수를 나타낸다. 내부 복셀 수는 표면 복셀 정보를 이용한 내부 복셀화 수행 후의 복셀의 수를 나타낸다. 전체 복셀 수는 전체 복셀 데이터 수를 나타내며 파일 크기는 생성되는 파일의 크기를 나타낸다. 또한 파일의 크기는 식(3)이 적용되었기 때문에 각 축별 해상도의 곱의 값과 다를 수 있다. 여기서, 전체 복셀 수와 파일 크기는 32만큼 차이가 발생하게 되는데 복셀화 파일 생성 시에 파일 끝에 각 축의 해상도 및 내부 복셀 수를 각 unsigned int(2 Byte) 형식으로 저장하여 복셀화 파일 사용 시에 활용할 수 있도록 하였다.

[0071] 도 8은 본 발명에 따른 시험 환경에서 구현된 프로그램을 이용하여 복셀화 수행시의 CPU 및 메모리 사용 현황을 나타낸다. CPU 사용 현황은 모든 스레드가 CPU 사용률 90% 이상임을 확인 할 수 있다. 또한 메모리 사용 현황은 복셀화 수행 전과 수행 중의 사용량의 변화가 대체로 없음을 확인할 수 있다.

[0072] 도 9는 내부 복셀화가 완료된 시험 모델들의 각 축별 중심 기준의 단면도를 나타낸다.

[0073] m110 모델과 m374 모델의 경우 각 축별 중심 단면의 형태가 복잡하지 않고 공백이 발생하는 부분의 없다. 하지만 m23 모델의 z축과 m329 모델의 x축의 경우 중심 단면의 형태가 복잡하고 단면에 나타나는 폴리곤들 간의 다수의 공백 발생으로 인한 단방향 복셀화 수행 시에 문제가 나타날 수 있는 부분들에 대해 내부 복셀화에 의한 채우기가 정확하게 수행된 것을 확인할 수 있다.

[0074] 또한 m1601 모델의 x축과 y축의 중심 단면의 결과를 통해 체스 말의 상단의 공백 처리가 정확하게 수행된 것을 확인할 수 있다. 다음의 결과를 통해 본 발명에서 제시한 내부 복셀화 방안을 이용한 시험에 사용된 모든 모델에 대한 내부 복셀화가 확실하게 수행된 것을 알 수 있다.

[0075] 상술한 바와 같이, 본 발명은 폴리곤으로 구성된 3차원 모델을 고해상도로 내부 복셀화하기 위해 표면 복셀화를 수행하여 3차원 모델의 표면 정보를 추출하고 이 정보를 기초로 내부 복셀화를 수행하였다. 내부 복셀화 수행 시에는 각 축 방향으로 내부 복셀화 계산된 정보를 병합하여 사용함으로써 복잡한 형태의 3차원 모델을 다양한 해상도에서도 완벽하게 내부 복셀화를 수행할 수 있게 되었다.

[0076] 표면 및 내부 복셀화 계산 시에는 추출되는 복셀 정보를 바이너리 파일로 저장함으로써 컴퓨터의 메모리 사용량을 줄여 저사양 컴퓨터에서도 고해상도의 복셀화를 수행할 수 있었다. 또한 복셀화 계산 시에 다중 스레드를 이용한 병렬 처리를 적용하여 처리시간을 단축시킬 수 있다.

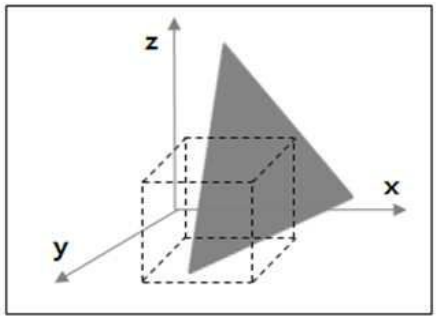
[0077] 이상 본 발명자에 의해서 이루어진 발명을 상기 실시 예에 따라 구체적으로 설명하였지만, 본 발명은 상기 실시 예에 한정되는 것은 아니고 그 요지를 이탈하지 않는 범위에서 여러 가지로 변경 가능한 것은 물론이다.

산업상 이용가능성

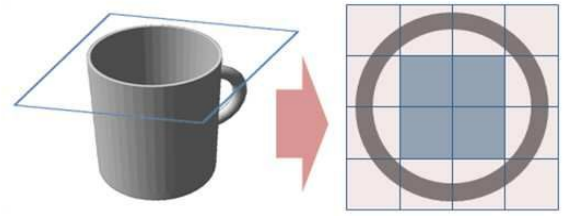
[0078] 본 발명에 따른 복셀화 방법을 사용하는 것에 의해 물체를 구성하는 물질을 복셀 정보로 설정하고 물체의 형태를 3차원 폴리곤 모델로 구현하여 복셀화를 수행하므로, 복셀화가 완료된 물체 정보를 이용하여 복셀(구성물질)의 위치에 따른 물성변화를 예측하는 분야, 즉 성능실험 시뮬레이터, 가상현실, 훈련용 시뮬레이터 등에 적용되는 기반기술로서 3차원 시각화를 다루는 모든 산업분야에서 활용이 가능하다.

도면

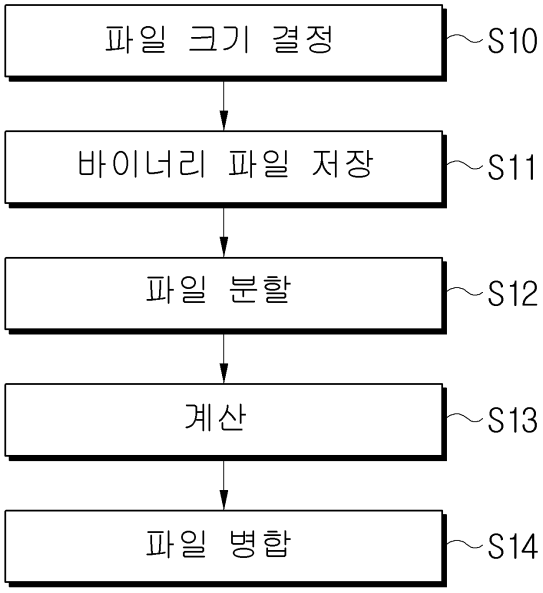
도면1



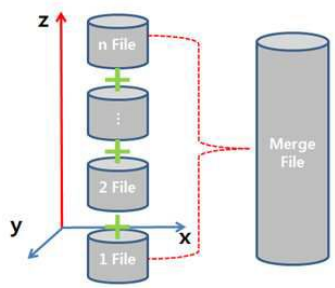
도면2



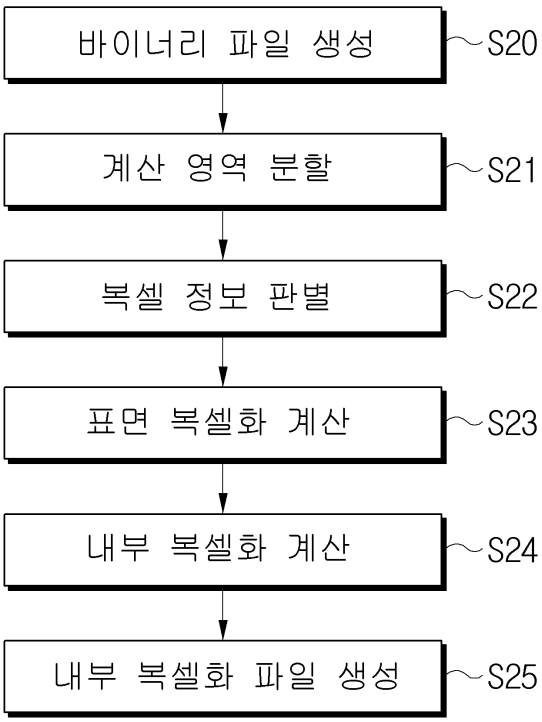
도면3



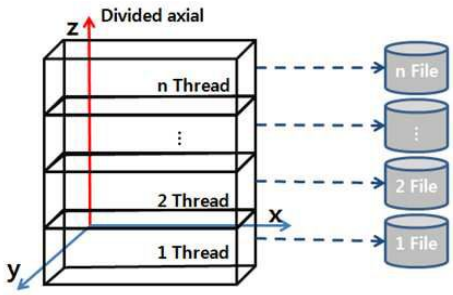
도면4


z
n File
2 File
1 File
y
x
Merge File

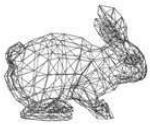
도면5


바이너리 파일 생성
S20
계산 영역 분할
S21
복셀 정보 판별
S22
표면 복셀화 계산
S23
내부 복셀화 계산
S24
내부 복셀화 파일 생성
S25

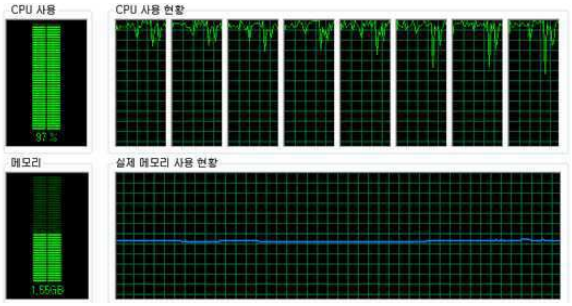
도면6


Divided axial
z
n Thread
2 Thread
1 Thread
x
y
n File
2 File
1 File

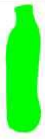
도면7

Model Name		m23	m110	m329	m374	m1601
Axis	x					
	y					
	z					

도면8



도면9

Model Name		m23	m110	m329	m374	m1601
Axial	x					
	y					
	z					