



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0082666
(43) 공개일자 2018년07월19일

(51) 국제특허분류(Int. Cl.)

G06F 17/30 (2006.01)

(52) CPC특허분류

G06F 17/30109 (2013.01)

G06F 17/30247 (2013.01)

(21) 출원번호 10-2017-0002906

(22) 출원일자 2017년01월09일

심사청구일자 2017년01월09일

(71) 출원인

경희대학교 산학협력단

경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)

(72) 발명자

이영구

대전광역시 서구 청사서로 11, 106동 302호(월평동)

박기성

경기도 부천시 원미로 129, E동 1층 4호 (원미동, 배송연립)

상의정

경기도 수원시 권선구 장다리로 14, 305호

(74) 대리인

김홍석, 김동용

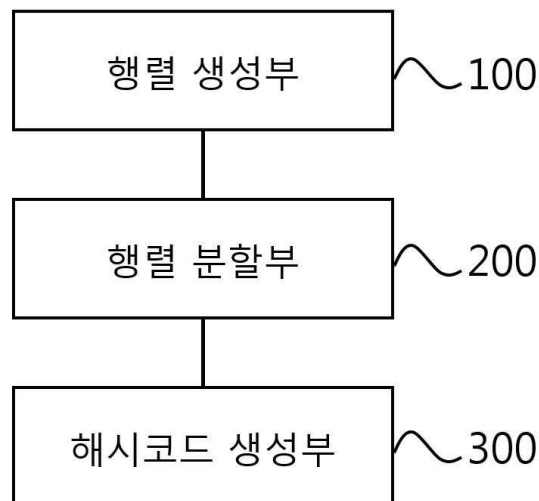
전체 청구항 수 : 총 8 항

(54) 발명의 명칭 유사이미지 검색을 위한 해시코드 생성장치 및 방법

(57) 요약

본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치는 GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성장치로서, 해시코드 생성에 사용되는 이미지 개수에 의해서 생성된 유사도 행렬과 기 설정된 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 행렬 생성부, 상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 행렬 분할부 및 상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 해시코드 생성부를 포함한다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호 R0613161203

부처명 미래창조과학부

연구관리전문기관 정보통신산업진흥원

연구사업명 서울어코드 활성화 사업

연구과제명 경희대학교 컴퓨터공학과 교육선진화프로그램

기 여 율 1/1

주관기관 경희대학교 산학협력단

연구기간 2016.03.01 ~ 2017.02.28

명세서

청구범위

청구항 1

GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성장치로서,

해시코드 생성에 사용되는 이미지 개수에 의해서 생성된 유사도 행렬과 기 설정된 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 행렬 생성부;

상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 행렬 분할부; 및

상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 해시코드 생성부를 포함하는 것을 특징으로 하는 유사이미지 검색을 위한 해시코드 생성장치.

청구항 2

제1항에 있어서,

상기 GPU 메모리는 GPU 공유메모리이며, 상기 행렬 분할부는 상기 GPU 공유메모리에 올릴 수 있는 블록의 크기로 상기 유사도 행렬과 해시비트행렬을 분할하는 것을 특징으로 하는 유사이미지 검색을 위한 해시코드 생성장치.

청구항 3

제1항에 있어서,

상기 GPU 메모리 내에 형성되는 블록에는 해시코드생성을 위한 병렬연산을 수행하는 복수 개의 스레드를 포함하는 것을 특징으로 하는 유사이미지 검색을 위한 해시코드 생성장치.

청구항 4

제3항에 있어서,

상기 해시코드 생성부는 상기 스레드 내에서 GPU 커널함수를 이용하여 병렬연산을 수행하는 것을 특징으로 하는 유사이미지 검색을 위한 해시코드 생성장치.

청구항 5

GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성방법으로서,

이미지 개수를 이용하여 유사도 행렬을 생성하는 단계;

상기 이미지 개수와 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 단계;

상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 단계; 및

상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 단계를 포함하는 것을 특징으로 하는 해시코드 생성방법.

청구항 6

제5항에 있어서,

상기 분할하는 단계는 상기 GPU 메모리 용량은 GPU 공유메모리 용량이며, 상기 GPU 공유메모리 가용용량에 따라 상기 유사도행렬과 상기 해시비트행렬을 분할하는 단계를 포함하는 것을 특징으로 하는 해시코드 생성방법.

청구항 7

제6항에 있어서,

상기 해시코드를 생성하는 단계는 상기 GPU 공유메모리 내에 형성되는 복수개의 스레드에서 행렬 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 단계를 포함하는 것을 특징으로 하는 해시코드 생성방법.

청구항 8

제5항 내지 제7항 중 어느 한 방법을 구현하는 프로그램이 저장되는 컴퓨터로 판독가능한 기록매체.

발명의 설명

기술 분야

[0001] 본 발명은 유사 이미지를 검색하기 위한 해시코드를 생성하는 장치 및 방법에 관한 것이다.

배경 기술

[0002] 소셜 네트워킹 웹사이트나 시스템은 회원이 다른 회원들과 연결하고 통신할 수 있는 동적 환경을 제공한다. 이런 웹사이트는 흔히 회원들이 기존의 소셜 네트워크 내에서 상호작용할 뿐 아니라 새로운 소셜 네트워크를 생성할 수 있게 해주는 온라인 메커니즘을 제공할 수 있다.

[0003] 회원들은 임의의 개인 또는 가령 기관이나 사업체와 같은 다른 엔티티를 포함할 수 있다. 다른 속성들 중에서, 소셜 네트워킹 웹사이트는 회원들이 관련 정보를 소셜 네트워크로 효과적이고 효율적으로 통신할 수 있게 해준다. 예컨대, 회원은 개인 정보, 직업 정보, 신상 정보 및 다른 개인화된 콘텐츠를 회원 전용 웹사이트의 영역에 게시할 수 있다. 다른 회원들은 회원 프로필을 브라우징하거나 전용 검색을 수행함으로써 게시된 콘텐츠에 액세스할 수 있다.

[0004] 소셜 네트워킹 웹사이트에 게시되는 하나의 일반적인 타입의 콘텐츠는 이미지, 예컨대 사진이나 다른 종류의 그림들인데, 이런 이미지는 대개 디지털 파일이거나 디지털화된 형태이다. 회원은 종종 이미지를 게시하여 개인정보를 소셜 네트워크로 통신한다. 예컨대, 게시된 이미지는 가족, 친구, 위치, 아이템 및 회원이 관심을 가질 수 있는 임의의 다른 내용을 나타낼 수 있다. 개인 정보의 공유가 회원 간의 통신을 촉진함으로써 웹사이트의 잠재성을 구현하는데 도움을 주기 때문에 소셜 네트워킹 웹사이트는 대개 이미지의 게시를 장려한다.

[0005] 특정 이미지는 소셜 네트워킹 웹사이트가 많은 이유로 특수한 관심을 가지는 콘텐츠를 포함할 수 있다. 예컨대, 소셜 네트워킹 웹사이트는 웹사이트의 이용 약관이나 다른 가이드라인을 위반하여 동일한 사진이 다른 회원들에 의해 반복적으로 게시될 때 개입하고자 할 수 있다. 또 다른 예로서, 소셜 네트워킹 웹사이트는 부적절한 콘텐츠를 포함하는 웹사이트에 공지된 이미지의 게시를 제한하고자 할 수 있다. 이런 종류의 부당한 이미지를 식별하기 위해, 이미지는 부당한 것으로 알려진 이미지들과의 유사성이나 동일성에 대하여 비교될 수 있다. 부당한 이미지와 유사한 이미지가 식별되는 경우, 소셜 네트워킹 웹사이트는 그들에 대해 적절한 조치를 취할 수 있다.

[0006] 이처럼 소셜 네트워킹 웹사이트가 발전할수록 이미지의 유사성을 판별하는 기술의 중요성은 커지고 있으며, 특히 유사 이미지인지를 판정하는 정확도, 뿐만 아니라 수많은 이미지에 대한 빠른 판별에 대한 욕구가 커지고 있다.

[0007] 유사 이미지 검색은 이처럼 부당한 이미지 사용 뿐만 아니라 검색포털의 검색서비스, 이미지 분류 등에도 활용될 수 있다.

[0008] 유사 이미지를 검색하는 방법은 콘텐츠 기반, 이미지 해싱 등의 방법을 활용되고 있으며, 해싱을 사용하는 이미지 탐색은 대용량 이미지에 대한 탐색이 가능한 기술로 많이 활용되고 있다.

[0009] 그러나 이미지 개수와 해시 코드의 비트수가 커지면 행렬의 크기와 연산 횟수가 늘어나게 된다. 그로인해 메모리에 행렬을 모두 올리지 못하거나 연산횟수가 과도하게 많아짐으로 인해 오랜 수행시간이 소요된다.

선행기술문헌

특허문헌

[0010] (특허문헌 0001) 대한민국 등록특허 제10-1533349호(2015.06.26. 공개)

발명의 내용

해결하려는 과제

[0011] 본 발명의 목적은 비교적 대용량의 이미지를 탐색함에도 부하를 줄이는 이미지 검색을 수행하기 위해서 해시코드를 생성하는 장치 및 방법을 제공하는데 있다.

[0012] 또한, 본 발명의 목적은 이미지 개수와 해시코드 비트 수에 따라 증가하는 데이터의 크기와 연산수에 따른 메모리 사용량과 수행시간을 개선하는 장치 및 방법을 제공하는데 있다.

과제의 해결 수단

[0013] 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치는 GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성장치로서, 해시코드 생성에 사용되는 이미지 개수에 의해서 생성된 유사도 행렬과 기 설정된 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 행렬 생성부, 상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 행렬 분할부 및 상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 해시코드 생성부를 포함한다.

[0014] 여기서, 상기 GPU 메모리는 GPU 공유메모리이며, 상기 행렬 분할부는 상기 GPU 공유메모리에 올릴 수 있는 블록의 크기로 상기 유사도 행렬과 해시비트행렬을 분할한다.

[0015] 여기서, 상기 GPU 메모리 내에 형성되는 블록에는 해시코드생성을 위한 병렬연산을 수행하는 복수 개의 스레드를 포함한다.

[0016] 여기서, 상기 해시코드 생성부는 상기 스레드 내에서 GPU 커널함수를 이용하여 병렬연산을 수행한다.

[0017] 본 발명의 실시예에 따른 해시코드 생성방법은 GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성방법으로서, 이미지 개수를 이용하여 유사도 행렬을 생성하는 단계, 상기 이미지 개수와 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 단계, 상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 단계 및 상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 단계를 포함한다.

[0018] 여기서, 상기 분할하는 단계는 상기 GPU 메모리 용량은 GPU 공유메모리 용량이며, 상기 GPU 공유메모리 가용용량에 따라 상기 유사도행렬과 상기 해시비트행렬을 분할하는 단계를 포함한다.

발명의 효과

[0019] 본 발명의 실시예에 따른 해시코드 생성장치 및 방법에 의하면 유사이미지를 검색하기 위한 메모리 사용량과 수행시간이 개선된다.

[0020] 본 발명의 실시예에 따른 해시코드 생성장치 및 방법에 의하면, 이미지 개수와 해시코드의 비트수가 많아지더라도 효율적으로 유사이미지를 검출할 수 있다.

도면의 간단한 설명

[0021] 도 1은 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치의 기능 블록도이다.

도 2는 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치의 동작 개념도이다.

도 3은 본 발명의 실시예에 따른 GPU 커널 함수를 사용한 해시코드 생성 알고리즘이다.

도 4는 본 발명의 실시예에 따른 해시코드 생성방법의 순서도이다.

도 5는 연산 블록 크기에 따른 수행시간 비교 그래프이다.

발명을 실시하기 위한 구체적인 내용

- [0022] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.
- [0023] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에 상세하게 설명하고자 한다. 그러나 이는 본 발명의 개념에 따른 실시예들을 특정한 개시형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 변경, 균등물, 또는 대체물을 포함한다.
- [0024] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만, 예를 들어 본 발명의 개념에 따른 권리 범위로부터 이탈되지 않은 채, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.
- [0025] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 표현들, 예를 들어 "~사이에"와 "바로~사이에" 또는 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0026] 본 명세서에서 사용한 용어는 단지 특정한 실시예들을 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함으로 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0027] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0029] 이하, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 그러나 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.
- [0030] 이하 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치 및 방법에 관해서 도면을 통해서 상세히 설명한다.
- [0031] 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치 및 방법은 기존 CPU기반의 알고리즘을 활용하되, GPU(Graphic Processing Unit)기반으로 연산을 분할하여 해시코드 생성을 위한 연산을 병렬적으로 수행하는 장치 및 방법에 관한 것이다. 이러한 본 발명의 실시예는 이미지 개수와 해시코드 비트수에 따라서 증가하는 행렬의 크기와 연산횟수에 따라 부하가 걸리는 기존 메모리의 사용량과 연산속도를 개선시키기 위해서 도출되었다.
- [0032] 본 발명의 전반에서 언급되고 있는 해시코드(hash code)는 해시함수에 의해서 얻어지는 값이다. 해시 함수(hash function)는 임의의 길이의 데이터를 고정된 길이의 데이터로 매핑하는 함수이다. 해시 함수에 의해 얻어지는 값은 해시 값, 해시 코드, 해시 체크섬 또는 해시라고 한다. 그 용도 중 하나는 해시 테이블이라는 자료구조에 사용되며, 매우 빠른 데이터 검색을 위한 컴퓨터 소프트웨어에 널리 사용된다.
- [0033] 해시 함수는 큰 파일에서 중복되는 레코드를 찾을 수 있기 때문에 데이터베이스 검색이나 테이블 검색의 속도를 가속할 수 있다. 예를 들어서, DNA sequence, 이미지 프로세싱을 위한 이미지 데이터에서 유사한 패턴을 찾는 데 사용될 수도 있다.
- [0034] 또한, 암호학에서도 사용된다. 암호용 해시 함수는 매핑된 해싱 값만을 알아가지고는 원래 입력 값을 알아내기 힘들다는 사실에 의해 사용될 수 있다. 또한, 전송된 데이터의 무결성을 확인해주는 데 사용되기도 하는데, 메

시지가 누구에게서 온 것인지 입증해주는 HMAC를 구성하는 블록으로 사용된다.

- [0035] 해시 함수는 결정론적으로 작동해야 하며, 따라서 두 해시 코드가 다르다면 그 해시코드에 대한 원래 데이터도 달라야 한다. 다만, 그 역은 성립하지 않는다.
- [0036] 해시 함수의 질은 입력 영역에서의 해시 충돌 확률로 결정되는데, 해시 충돌의 확률이 높을수록 서로 다른 데이터를 구별하기 어려워지고 검색하는 비용이 증가하게 된다.
- [0037] 또한, 유사도 행렬은 이미지간의 유사도를 표현한 행렬을 의미한다. 유사이미지를 탐색하는 기법으로 이미지 해싱 기법중에는 지도학습 알고리즘을 활용하는 방식이 있다. 이러한 지도학습 알고리즘은 이미지간의 유사도에 따라 1과 -1로 표현된 행렬인 유사도 행렬을 이용한다.
- [0039] 도 1은 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치의 기능 블록도이다.
- [0040] 도 2는 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치의 동작 개념도이다.
- [0041] 도 1과 도 2에 도시된 바와 같이 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치는 GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성장치로서, 해시코드 생성에 사용되는 이미지 개수에 의해서 생성된 유사도 행렬과 기 설정된 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 행렬 생성부(100), 상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 행렬 분할부(200) 및 상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 해시코드 생성부(300)를 포함한다.
- [0043] 도 1과 도 2에 도시된 바와 같이 행렬 생성부(100)는 유사도 행렬과 해시비트행렬을 생성한다. 유사도 행렬은 해시코드 생성에 사용되는 이미지 개수로 결정되며, 해시비트행렬은 이미지 개수와 해시코드의 비트수로 인해서 결정된다.
- [0044] 도 2에 도시된 유사도 행렬은 해시코드 생성에 사용되는 이미지 개수가 n 개인 경우에 $n \times n$ 행렬이고, 해시비트행렬은 해시코드의 비트수가 q 인 경우 $q \times n$ 행렬이 된다.
- [0045] 행렬 분할부(200)는 앞서 행렬 생성부(100)에서 생성된 유사도 행렬과 해시비트행렬을 GPU메모리 용량에 따라서 분할한다. 보다 상세하게는 이 행렬들을 GPU 공유 메모리에 올릴 수 있도록 블록크기를 지정해주고, 그 블록크기를 갖는 행렬들로 분할하게 된다.
- [0046] 도 2에 도시된 GPU 공유메모리 상의 블록크기는 b 이고 유사도 행렬과 해시비트행렬은 블록크기 b 로 분할된다. 또한, GPU 메모리 내에 형성되는 블록에는 해시코드생성을 위한 병렬연산을 수행하는 복수 개의 스레드를 포함한다.
- [0047] 도 1에 도시된 해시코드 생성부(300)는 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성한다. 특히 GPU 공유메모리의 블록내에 형성되어 있는 복수개의 스레드에서 최종 결과값인 해시코드를 생성한다.
- [0048] 기존 해시코드를 생성하기 위해서는 유사도 행렬과 해시비트 행렬을 분할하지 않은 상태에서 해시코드 생성 알고리즘을 수행하여 해시코드를 생성하였고, 이러한 경우 이미지 개수와 해시코드의 길이인 해시코드 비트수에 따라서 연산량이 늘어나게 된다.
- [0049] 그러나 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치에 의하면 두가지 관점에서 보다 빠른 연산이 수행되어질 수 있다.
- [0050] 첫째는 유사도 행렬과 해시비트행렬을 GPU 공유메모리의 블록의 크기에 따라서 분할하여 병렬연산을 수행하도록 한다는 점이다.
- [0051] 둘째는 GPU는 공유메모리와 글로벌 메모리를 포함하고 있는데, 본 발명의 실시예에 따른 해시코드 생성은 GPU의 공유메모리에서 담당한다. 공유메모리는 GPU 하나의 블록 안에 속해 있는 여러 스레드가 동시에 접근할 수 있는 메모리라는 점에서 글로벌 메모리에 비해 연산속도가 빠르다. 또한, 공유 메모리의 단점이라고 할 수 있는 글로벌 메모리에 비해 사용할 수 있는 용량이 한정적이라는 점은 유사도 행렬과 해시비트행렬의 크기가 공유 메모리의 가용용량에 비해 큰 경우, 앞서 설명한 바와 같이 일정 크기로 분할하여 연산블록을 구성한다는 점으로 해소

가 가능하다

- [0052] 본 발명의 실시예에 따른 해시코드 생성부(300)는 해시코드 생성 알고리즘을 수행하기 위해 분할된 연산블록들에서 기존 해시코드 생성 알고리즘을 수행한다.
- [0053] 또한, 연산을 위해 커널 함수는 각각의 스레드에서 행렬원소들에 대한 연산을 병렬적으로 수행하여 최종결과 값을 행렬의 형태로 나타낸다. GPU는 행렬 연산과 같이 병렬적으로 수행되는 연산에 대해서는 매우 빠른 수행속도를 갖게 되어 GPU를 사용하는 경우 해시 코드 생성 시간을 단축시키는 것이 가능하다.
- [0054] 분할된 연산블록들은 알고리즘에 필요한 연산을 담당하는 커널 함수들에 들어간다. 해시코드 생성에 사용되는 커널함수는 해시코드를 구하는 연산에 필요한 행렬을 만드는 함수, 개선된 해시코드를 생성하는 함수, 해시코드 행렬 H를 갱신하는 함수 세가지로 구성될 수 있다. 모든 커널 함수에서 공통적으로 사용되는 행렬들은 GPU의 글로벌 메모리를 사용하고 각 커널 함수에서 실제로 연산된 결과가 저장되는 행렬은 각 함수의 공유 메모리를 사용한다.
- [0055] 도 3은 본 발명의 실시예에 따른 GPU 커널 함수를 사용한 해시코드 생성 알고리즘이다.
- [0056] 도 3에 도시된 바와 같이 모든 커널함수들은 해시코드 생성 알고리즘을 GPU의 병렬연산을 사용하여 모든 행렬 원소에 대해 동시에 수행되도록 설계될 수 있다. 도 3의 커널함수 (1)은 해시코드 갱신에 필요한 행렬인 dL을 계산하는 함수로서 GPU에서 수학식 1과 같은 연산을 병렬적으로 수행한다.
- [0057] [수학식 1]
- [0058]
$$L(dS, dH, q) = dHdH^T - qdS$$
- [0059] 수학식 1을 통해 구해진 dL은 행렬값 $dH_{(i,j)}$ 을 갱신하기 위하여 필요한 값을 구하는 도 3의 커널함수 (2), (3)에서 사용된다. dH의 모든 행렬 원소 $dH_{(i,j)}$ 에 대해서 수학식 2, 수학식 3을 병렬적으로 수행한다.
- [0060] [수학식 2]
- [0061]
$$G'_{(i,j)}(dL, dH) = 4dL_i dH_j$$
- [0062] [수학식 3]
- [0063]
$$G'_{(i,j)}(dL, dH) = 4(dH_j^T dH_j + dH_{2ij} + dL_{ij})$$
- [0064] 마지막으로 $dH_{(i,j)}$ 의 값을 앞서 구한 $dG_{1(i,j)}$, $dG_{2(i,j)}$ 을 사용하여 전체 행렬 dH에 대해서 병렬적으로 갱신한다.
- [0065] 기존 CPU를 활용한 알고리즘에서 행렬의 각 원소에 대해서 순차적으로 이루어진 행렬 연산이 GPU에서의 병렬처리를 통해 동시에 이루어짐으로 인해 수행속도가 개선된다. 또한, 모든 함수의 행렬연산은 공유 메모리 안에서 이루어지기 때문에 GPU의 글로벌 메모리를 사용한 경우보다 더 빠른 성능 향상을 보이게 된다.
- [0066] 이상 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치에 대해서 살펴보았다. 이하 본 발명의 또 다른 양태에 따른 유사이미지 검색을 위한 해시코드 생성방법에 대해서 설명한다. 앞선 실시예와 중복되는 내용에 대한 설명은 생략한다.
- [0067] 도 4는 본 발명의 실시예에 따른 해시코드 생성방법의 순서도이다.
- [0068] 도 4에 도시된 바와 같이 본 발명의 실시예에 따른 해시코드 생성방법은 GPU(Graphic Processing Unit)를 이용한 유사이미지 검색을 위한 해시코드 생성방법으로서, 이미지 개수를 이용하여 유사도 행렬을 생성하는 단계(S100), 상기 이미지 개수와 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 단계(S200), 상기 유사도 행렬과 상기 해시비트행렬을 상기 GPU 메모리 용량에 따라 분할하는 단계(S300) 및 상기 GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 단계(S400)를 포함한다.
- [0069] 이미지 개수를 이용하여 유사도 행렬을 생성하는 단계(S100)는 이미지의 유사도에 따라 1과 -1로 표현되는 유사도 행렬을 생성하되, 이미지 개수가 n인 경우 $n \times n$ 행렬을 생성하는 단계이다.
- [0070] 이미지 개수와 해시코드 비트수를 이용하여 해시비트행렬을 생성하는 단계(S200)는 앞서 설명한 바와 같이 이미

지 개수 n , 해시비트행렬 q 인 경우 $q \times n$ 행렬을 생성하는 단계이다.

[0071] 유사도 행렬과 해시비트행렬을 GPU메모리 용량에 따라 분할하는 단계(S300)에서 GPU 메모리 용량은 GPU 공유메모리 용량이며, GPU 공유메모리 가용용량에 따라 유사도행렬과 해시비트행렬을 분할한다.

[0072] GPU 메모리 내에서 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 연산을 병렬적으로 수행하여 해시코드를 생성하는 단계(S400)는 분할된 유사도 행렬과 분할된 해시비트행렬의 원소들에 대한 커널 함수를 이용한 연산을 스레드에서 병렬적으로 수행하는 단계이다.

[0073] <실험예>

[0074] 본 발명의 실시예에 따른 유사이미지 검색을 위한 해시코드 생성장치 및 방법에 의한 해시코드 생성시 수행시간을 비교하였다.

[0075] 실험환경은 Intel(R) Core(TM) i7-5820K CPU @3.30GHz, VGA NVIDIA GeForce GTX 90, RAM 16GB, OS Window 10의 환경에서 수행되었다

표 1

이미지 개수	CPU(초)	GPU(초)
500	3.345	0.181
1000	20.664	0.245
1500	74.119	0.348

[0077] 표 1은 해시코드 생성에 사용되는 이미지 개수에 따른 수행시간 변화를 나타낸 값이다. CPU를 사용한 해시코드 생성의 경우 이미지 개수가 늘어남에 따라 행렬의 크기가 커져 순차적으로 수행되어야 할 연산의 개수가 크게 증가하였다. 반면 GPU를 사용한 경우 지정된 블록의 크기에 따라 연산을 수행하기 때문에 동일한 수행시간을 갖는 블록 개수가 증가함에 따른 수행시간 증가만이 있는 것을 확인했다.

표 2

해시비트수	CPU	GPU
3	20.822	0.206
5	34.73	0.233
8	55.694	0.246

[0081] 표 2는 해시코드의 비트수에 따른 수행시간 변화를 나타낸 표이다. 해시코드의 비트수가 증가할 수록 해시코드 행렬 H의 크기가 증가하게 됨에 따라 수행시간이 증가하게 된다. GPU는 연산이 병렬로 이루어짐에 따라 큰 수행시간 증가는 보이지 않는다.

표 3

반복회수	CPU	GPU
3	6.906	0.198
5	11.501	0.213
8	18.378	0.214

[0084] 표 3은 전체 알고리즘 반복 수행 횟수에 따른 수행시간 비교이다. 반복 횟수에 따라서 수행시간이 증가하는 것을 확인할 수 있다. 해시코드의 에러율이 임계값보다 낮아지게 되면 반복 수행을 중단하게 되어 다른 실험에 비해 수행시간의 증가가 적게 일어나는 것을 알 수 있다.

[0086] 도 5는 연산 블록 크기에 따른 수행시간 비교 그래프이다.

[0087] 도 5에 도시된 바에 따르면 GPU에서 해시코드 생성시 수행단위인 연산블록의 크기에 따른 수행시간을 비교한 도면이다. 연산블록의 크기가 커질수록 병렬적으로 수행하는 행렬의 크기가 커져 수행시간이 감소하는 모습을 볼 수 있다.

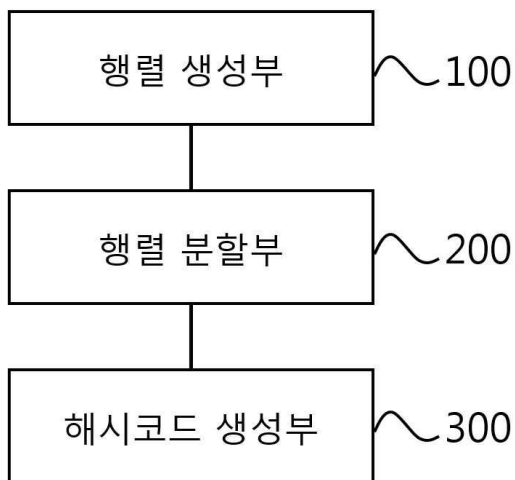
[0089] 본 발명은 도면에 도시된 실시 예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

부호의 설명

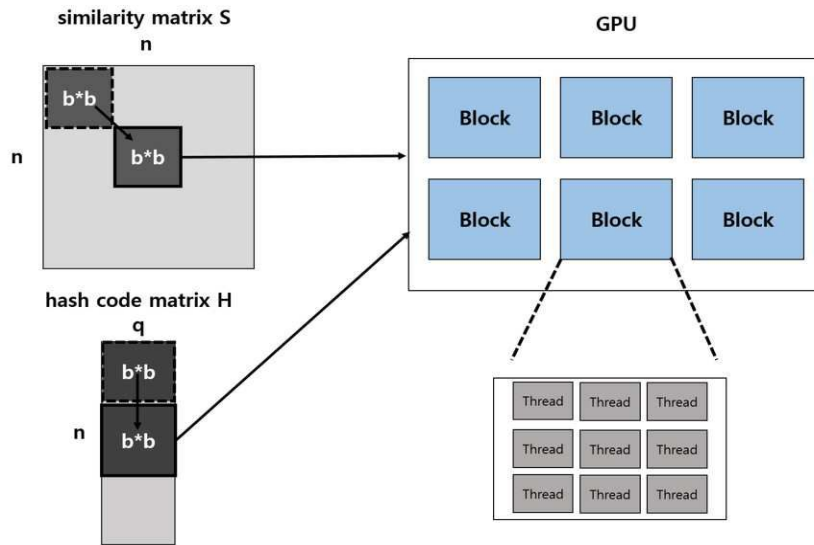
- [0090]
- 100 행렬 생성부
 - 200 행렬 분할부
 - 300 해시코드 생성부

도면

도면1



도면2



도면3

parallel_hash_bit_learning(S, H, b)

input: similarity matrix S , hash bit matrix H ,
submatrix size b , hash bit q

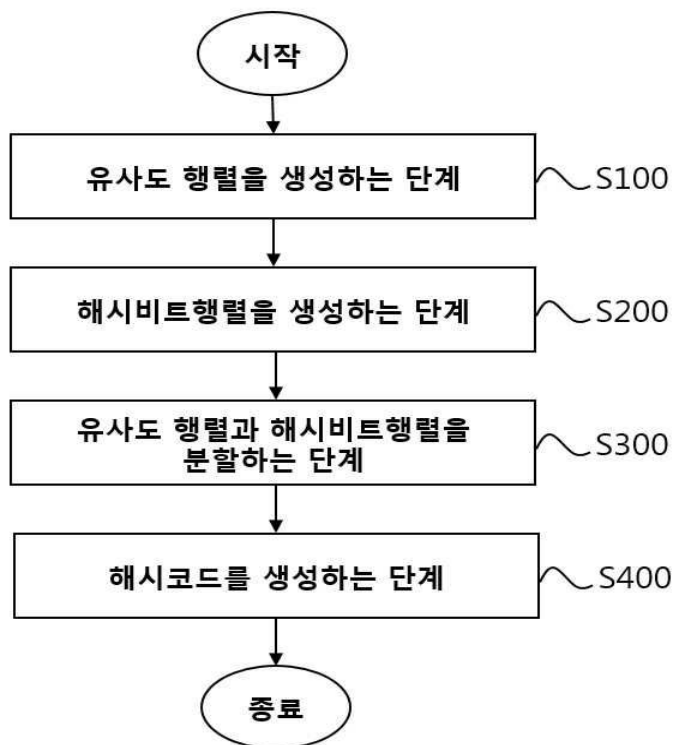
```

1: for  $t = 1, \dots, b$  do
2:   Initialize:  $dS \leftarrow S$ 의 부분행렬,
       $dH \leftarrow H$ 의 부분행렬
3:   행렬 연산  $dL \leftarrow L(dS, dH, q)$  식 (1)
4:   행렬 연산  $dG_1 \leftarrow G'(dL, dH)$  식 (2)
5:   행렬 연산  $dG_2 \leftarrow G''(dS, dH, b)$  식 (3)
6:   for  $dH_{(i,j)}$ 의 모든 원소
      ( $i = 1, \dots, b, j = 1, \dots, q$ )
7:      $u = \max(-1 - dH_{ij}, \min(-\frac{q'(dH_{ij})}{q''(dH_{ij})}, 1 - dH_{ij}))$ 
8:      $dH_{(i,j)} \leftarrow dH_{ij} + u$ 
9:   end for
10:   $S$ 를 사용하여  $dS$  갱신
11: end for

```

output: approximate hash code matrix H

도면4



도면5

