

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)(11) 공개번호 10-2022-0099745
(43) 공개일자 2022년07월14일

(51) 국제특허분류(Int. Cl.)

G06F 16/22 (2019.01) G06F 16/182 (2019.01)
G06F 16/23 (2019.01) G06F 16/242 (2019.01)
G06F 16/2453 (2019.01) G06F 16/29 (2019.01)
G06F 16/909 (2019.01)

(52) CPC특허분류

G06F 16/2246 (2019.01)
G06F 16/182 (2019.01)

(21) 출원번호 10-2021-0002015

(22) 출원일자 2021년01월07일

심사청구일자 없음

(71) 출원인

서강대학교산학협력단

서울특별시 마포구 백범로 35 (신수동, 서강대학교)

(72) 발명자

정성원

서울특별시 양천구 오목로 300, 204동 604호(목동, 현대하이페리온2)

조범준

서울특별시 마포구 마포대로 53 (용강동, 마포트라팰리스)

(74) 대리인

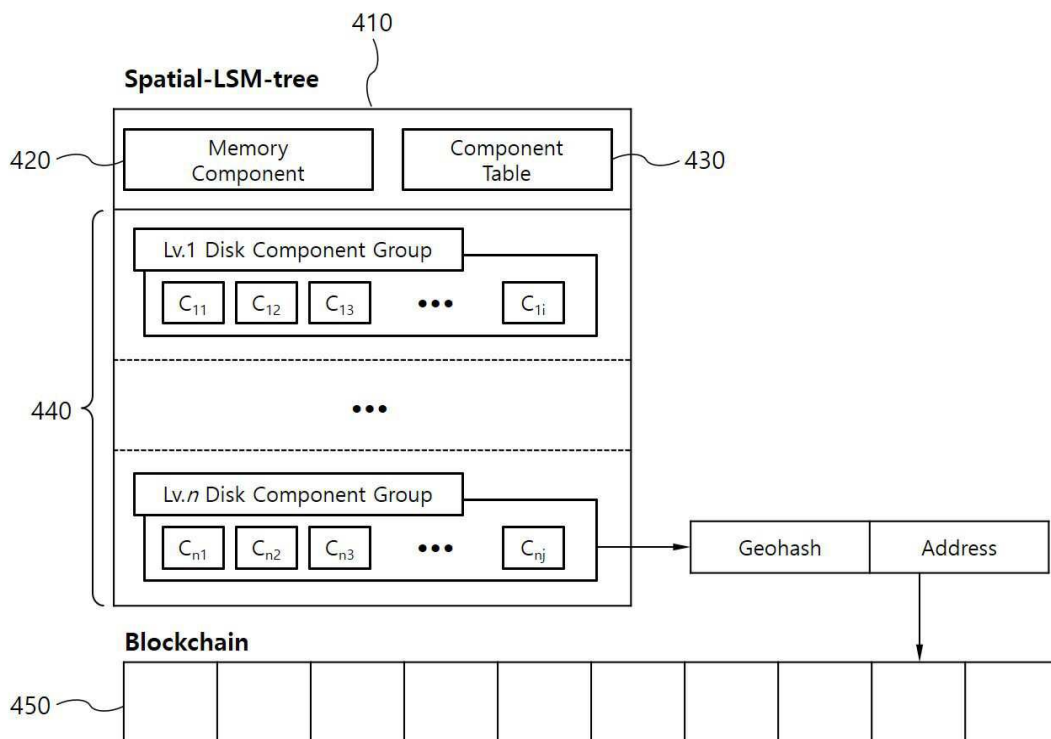
정부연

전체 청구항 수 : 총 13 항

(54) 발명의 명칭 **지리공간 블록체인 데이터 검색을 위한 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법 및 장치****(57) 요약**

본 발명은 지리공간 블록체인 데이터 검색을 위한 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법 및 장치에 관한 것으로, 상기 방법은 공간 데이터 객체를 수신하여 공간 블록체인에 저장하는 단계; 상기 공간 데이터 객체의 지오해시(geohash)값과 상기 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 해시-주소 쌍을 생성하는

(뒷면에 계속)

대표도 - 도4

단계; 상기 해시-주소 쌍을 메모리 컴포넌트(memory component)에 저장하는 단계; 상기 메모리 컴포넌트의 공간 점유율이 기 설정된 제1 임계값을 초과하는 경우 상기 메모리 컴포넌트에 저장된 제1 항목들을 순차적으로 디스크 컴포넌트(disk component)에 플러시(flush)하는 단계; 상기 디스크 컴포넌트의 공간 점유율이 기 설정된 제2 임계값을 초과하는 경우 상기 디스크 컴포넌트에 저장된 제2 항목들을 순차적으로 하위 레벨의 디스크 컴포넌트에 플러시하는 단계; 및 상기 메모리 컴포넌트 또는 상기 디스크 컴포넌트의 변경을 반영하여 컴포넌트 테이블(component table)을 갱신하는 단계를 포함한다.

(52) CPC특허분류

G06F 16/23 (2019.01)

G06F 16/242 (2019.01)

G06F 16/2453 (2019.01)

G06F 16/29 (2019.01)

G06F 16/909 (2019.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711108128
과제번호	2019R1F1A1043854
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	개인기초연구(과기정통부)(R&D)
연구과제명	블록체인 기반 지리공간 데이터베이스 기술 개발
기 여 율	1/2
과제수행기관명	서강대학교
연구기간	2020.03.01 ~ 2021.02.28

이 발명을 지원한 국가연구개발사업

과제고유번호	1711103334
과제번호	2017-0-01628-004
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성(R&D)
연구과제명	적응형 블록체인 플랫폼 기술 개발 및 전문 인력 양성
기 여 율	1/2
과제수행기관명	서강대학교산학협력단
연구기간	2020.01.01 ~ 2020.12.31

명세서

청구범위

청구항 1

공간 데이터 객체를 수신하여 공간 블록체인에 저장하는 단계;

상기 공간 데이터 객체의 지오해시(geohash)값과 상기 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 해시-주소 쌍을 생성하는 단계;

상기 해시-주소 쌍을 메모리 컴포넌트(memory component)에 저장하는 단계;

상기 메모리 컴포넌트의 공간 점유율이 기 설정된 제1 임계값을 초과하는 경우 상기 메모리 컴포넌트에 저장된 제1 항목들을 순차적으로 디스크 컴포넌트(disk component)에 플러시(flush)하는 단계;

상기 디스크 컴포넌트의 공간 점유율이 기 설정된 제2 임계값을 초과하는 경우 상기 디스크 컴포넌트에 저장된 제2 항목들을 순차적으로 하위 레벨의 디스크 컴포넌트에 플러시하는 단계; 및

상기 메모리 컴포넌트 또는 상기 디스크 컴포넌트의 변경을 반영하여 컴포넌트 테이블(component table)을 갱신하는 단계를 포함하고,

상기 컴포넌트 테이블은 인덱스 트리를 구성하는 컴포넌트들에 대한 정보로서 컴포넌트 레벨(component level), 블록 주소(block address), 키 범위(key range) 및 공간 필터(spatial filter)를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 2

제1항에 있어서,

상기 컴포넌트 테이블은 상기 컴포넌트들을 상기 컴포넌트 레벨 순서대로 정렬하여 저장하고,

상기 공간 필터는 해당 컴포넌트가 커버하는 영역에서 데이터가 분포하는 영역에 관한 정보를 포함하는 비트 스트링(bit string)으로 구현되는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 3

제2항에 있어서, 상기 공간 필터는

해당 컴포넌트가 커버하는 영역을 n (상기 n 은 자연수)개의 셀 영역들로 분할하는 단계;

상기 셀 영역들을 지오해시 순서로 정렬하는 단계;

j (상기 j 는 자연수)번째 셀 영역에 객체가 존재하는 경우 j 번째 비트(bit)를 '1'로 표시하고 그렇지 않은 경우 '0'으로 표시하는 단계; 및

모든 셀 영역에 대응되는 n 개의 비트들 각각의 값을 결정하여 비트 스트링을 생성하는 단계의 순차적인 실행을 통해 생성되는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 4

제1항에 있어서,

상기 메모리 컴포넌트에 저장하는 단계는

각각이 인덱스 블록(index block) 및 적어도 하나의 해시-주소 블록(geohash/address block)을 포함하는 복수

의 컴포넌트(component)들 중 어느 하나에 저장하는 단계를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 5

제4항에 있어서,

상기 인덱스 블록은 해당 컴포넌트에 저장된 해시-주소 쌍들에 대한 인덱스로서 상기 지오해시값에 관한 B+ 트리를 포함하며,

상기 적어도 하나의 해시-주소 블록은 상기 해시-주소 쌍을 상기 지오해시값에 대해 정렬된 상태로 저장하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 6

제1항에 있어서,

상기 디스크 컴포넌트에 플러시하는 단계들 각각은

동일 레벨의 컴포넌트들 중에서 상기 키 범위가 같은 컴포넌트들을 하나의 컴포넌트로 병합하는 단계; 및

상기 병합된 컴포넌트의 크기가 임계값을 초과하는 경우 하위 레벨의 새로운 컴포넌트를 생성하는 단계를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 7

제1항에 있어서,

질의 포인트 및 질의 범위에 관한 범위 질의를 수신하는 단계;

상기 컴포넌트 테이블을 스캔하여 상기 범위 질의에 관한 결과 집합을 생성하는 단계; 및

상기 결과 집합을 상기 범위 질의에 대한 응답으로 제공하는 단계를 더 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 8

제7항에 있어서,

상기 결과 집합을 생성하는 단계는

공간 필터와 동일한 크기의 비트 스트링으로 구현되는 질의 필터를 생성하는 단계;

상기 질의 필터를 이용하여 컴포넌트 테이블을 스캔하여 상기 범위 질의에 연관된 객체를 포함하는 적어도 하나의 질의 컴포넌트를 검출하는 단계; 및

상기 적어도 하나의 질의 컴포넌트에 대한 인덱스를 탐색하면서 상기 범위 질의에 관한 질의 결과를 생성하여 결과 집합에 추가하는 단계를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 9

제8항에 있어서,

상기 질의 필터를 생성하는 단계는

컴포넌트 레벨 별로 상기 공간 필터의 크기에 대응되는 상기 질의 필터를 생성하는 단계를 포함하고,

상기 질의 필터는 상기 질의 포인트와 상기 질의 범위에 의해 결정되는 질의 영역에 관한 정보를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 10

제9항에 있어서,

상기 적어도 하나의 질의 컴포넌트를 검출하는 단계는

특정 컴포넌트의 공간 필터와 상기 질의 필터 간의 논리곱(AND) 연산을 수행한 결과 0보다 큰 비트가 존재하는 경우 상기 특정 컴포넌트를 질의 컴포넌트로 결정하는 단계를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 11

제7항에 있어서,

상기 결과 집합을 생성하는 단계는

상기 메모리 컴포넌트를 스캔하여 상기 범위 질의와 연관되는 컴포넌트들을 후보 집합으로 생성하는 단계;

상기 질의 범위를 기초로 상기 컴포넌트 테이블을 스캔하면서 상기 질의 범위를 커버하는 컴포넌트들에 대해 상기 질의 포인트와의 거리를 계산하여 우선순위 큐에 삽입하는 단계;

상기 컴포넌트 테이블에 대한 스캔이 완료되면 상기 질의 포인트로부터 가장 가까운 거리를 갖는 루트 노드를 상기 우선순위 큐를 통해 선택하고 자식 노드들에 대한 탐색을 수행하는 단계;

상기 탐색이 완료되면 상기 후보 집합 및 상기 질의 반경을 갱신하는 단계; 및

상기 우선순위 큐를 통해 다음 노드를 선택하여 상기 탐색을 반복하는 과정에서 상기 다음 노드와 상기 질의 포인트 간의 거리가 상기 질의 반경보다 큰 경우 상기 반복을 종료하고 현재의 후보 집합을 상기 결과 집합으로 결정하는 단계를 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법.

청구항 12

공간 데이터 객체를 저장하는 공간 블록체인;

상기 공간 데이터 객체를 인덱싱하는 인덱스 트리; 및

상기 인덱스 트리를 생성하는 인덱스 트리 생성부를 포함하고,

상기 인덱스 트리는

상기 공간 데이터 객체의 지오해시(geohash)값과 상기 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 생성된 해시-주소 쌍을 저장하는 메모리 컴포넌트;

상기 메모리 컴포넌트와 연동하여 동작하고, 상기 해시-주소 쌍을 저장하는 복수의 컴포넌트들을 레벨 별로 그룹화 하여 저장하는 디스크 컴포넌트; 및

컴포넌트들에 대한 정보로서 컴포넌트 레벨(component level), 블록 주소(block address), 키 범위(key range) 및 공간 필터(spatial filter)를 저장하는 컴포넌트 테이블을 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 장치.

청구항 13

제12항에 있어서,

상기 인덱스 트리를 이용하여 상기 공간 데이터 객체에 관한 범위 질의를 처리하는 질의 처리부를 더 포함하는 것을 특징으로 하는 공간 분할 기반의 트리 인덱싱 및 질의어 처리 장치.

발명의 설명

기술 분야

[0001] 본 발명은 블록체인 환경에서 지리 공간 데이터를 효과적으로 처리하기 위한 기술에 관한 것으로, 보다 상세하게는 대량의 업데이트에 최적화된 인덱스 구조인 LSM tree 기반을 확장한 공간 데이터 인덱싱 기법과 이를 활용한 공간 질의 처리 기법에 관한 것이다.

배경 기술

[0003] 최근 블록체인 기술이 다양한 응용분야에 접목됨에 따라 블록체인 시스템 기반의 공간 정보 서비스가 많은 주목을 받고 있다. IBM에서는 소비자가 식품의 생산지부터 배송의 전 단계를 추적할 수 있는 블록체인 기반의 식품 유통 정보 시스템을 개발하였고, 공간 블록체인 시스템인 FOAM은 블록체인 주소와 지리 정보를 결합하여 위치 정보와 관련된 거래 서비스를 제공할 수 있는 프로토콜을 개발하였다. 이 밖에도 토지 거래, IoT, 헬스케어 등 정보의 신뢰성과 개방성을 필요로 하는 다양한 분야에서 블록체인은 기존의 분산 데이터베이스에 대한 대안으로 활용된다. 그러나, 블록체인 기반의 위치 정보 서비스에 대한 관심에 비하며, 블록체인에 저장된 지리 공간 데이터에 대한 검색 및 분석 기법은 지금까지 많은 연구가 진행되지 않았다.

[0004] 블록체인 시스템은 기존의 공간 데이터베이스에서 가정하고 있는 환경과는 다른 특징을 갖는다. 첫 번째로, 블록체인 시스템은 주기적으로 대량의 업데이트를 발생시키는 쓰기-집중적인(write-intensive)한 환경이다. 예를 들어, 대표적인 블록체인 시스템 중 하나인 이더리움(Ethereum)의 경우 트랜잭션 데이터는 블록 단위로 전송되며, 14~15초 간격으로 지속적인 업데이트가 발생한다. 다양한 데이터베이스 시스템에서 사용되는 R-tree와 같은 기법들은 인덱스의 유지 및 업데이트 비용이 크기 때문에, 이와 같이 업데이트가 잦은 환경에서는 병목현상을 유발할 수 있다. 두 번째 특징은 데이터의 불변성(immutability)이다. 블록체인 시스템에 저장된 데이터베이스는 삭제나 변경이 되지 않는 것을 원칙으로 한다. 따라서, 블록체인의 크기는 서비스가 지속됨에 따라 계속적으로 증가하게 되고, 데이터베이스의 인덱스를 메모리상에서 관리하는 것은 점차 어려워진다.

[0005] 현재 다수의 블록체인 시스템에서 공간 정보에 대한 검색은 NoSQL DBMS의 보조 인덱스에 의존하고 있다. 그러나, NoSQL의 보조 인덱스는 시스템의 주 인덱스 및 디스크에 대한 반복적인 접근을 필요로 하기 때문에 데이터베이스에 따라 충분한 성능 향상을 얻기 힘든 경우도 있으며, 또한 이들 역시 공간 데이터 전용 DBMS가 아니기 때문에 지원하는 데이터의 타입과 오퍼레이션의 종류 또한 충분하지 못하다.

선행기술문헌

특허문헌

[0007] (특허문헌 0001) 한국등록특허 제10-1117709(2012.02.10)호

발명의 내용

해결하려는 과제

[0008] 본 발명의 일 실시예는 블록체인 환경에서 지리 공간 데이터를 효과적으로 처리하기 위한 것으로, 대량의 업데이트에 최적화된 인덱스 구조인 LSM tree 기반을 확장한 공간 데이터 인덱싱 기법과 이를 활용한 공간 질의 처리 기법을 제공하고자 한다.

[0009] 본 발명의 일 실시예는 메모리상에서 데이터의 업데이트를 일괄 처리하고 디스크에 대한 순차 쓰기 작업을 통해 업데이트 결과를 저장함으로써 디스크에 대한 임의 접근 비용을 절감할 수 있는 지리공간 블록체인 데이터 검색을 위한 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법 및 장치를 제공하고자 한다.

과제의 해결 수단

- [0011] 실시예들 중에서, 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법은 공간 데이터 객체를 수신하여 공간 블록체인에 저장하는 단계; 상기 공간 데이터 객체의 지오해시(geohash)값과 상기 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 해시-주소 쌍을 생성하는 단계; 상기 해시-주소 쌍을 메모리 컴포넌트(memory component)에 저장하는 단계; 상기 메모리 컴포넌트의 공간 점유율이 기 설정된 제1 임계값을 초과하는 경우 상기 메모리 컴포넌트에 저장된 제1 항목들을 순차적으로 디스크 컴포넌트(disk component)에 플러시(flush)하는 단계; 상기 디스크 컴포넌트의 공간 점유율이 기 설정된 제2 임계값을 초과하는 경우 상기 디스크 컴포넌트에 저장된 제2 항목들을 순차적으로 하위 레벨의 디스크 컴포넌트에 플러시하는 단계; 및 상기 메모리 컴포넌트 또는 상기 디스크 컴포넌트의 변경을 반영하여 컴포넌트 테이블(component table)을 갱신하는 단계를 포함한다.
- [0012] 상기 컴포넌트 테이블은 인덱스 트리를 구성하는 컴포넌트들에 대한 정보로서 컴포넌트 레벨(component level), 블록 주소(block address), 키 범위(key range) 및 공간 필터(spatial filter)를 포함할 수 있다.
- [0013] 상기 컴포넌트 테이블은 상기 컴포넌트들을 상기 컴포넌트 레벨 순서대로 정렬하여 저장하고, 상기 공간 필터는 해당 컴포넌트가 커버하는 영역에서 데이터가 분포하는 영역에 관한 정보를 포함하는 비트 스트링(bit string)으로 구현될 수 있다.
- [0014] 상기 공간 필터는 해당 컴포넌트가 커버하는 영역을 n (상기 n 은 자연수)개의 셀 영역들로 분할하는 단계; 상기 셀 영역들을 지오해시 순서로 정렬하는 단계; j (상기 j 는 자연수)번째 셀 영역에 객체가 존재하는 경우 j 번째 비트(bit)를 '1'로 표시하고 그렇지 않은 경우 '0'으로 표시하는 단계; 및 모든 셀 영역에 대응되는 n 개의 비트들 각각의 값을 결정하여 비트 스트링을 생성하는 단계의 순차적인 실행을 통해 생성될 수 있다.
- [0015] 상기 메모리 컴포넌트에 저장하는 단계는 각각이 인덱스 블록(index block) 및 적어도 하나의 해시-주소 블록(geohash/address block)을 포함하는 복수의 컴포넌트(component)들 중 어느 하나에 저장하는 단계를 포함할 수 있다.
- [0016] 상기 인덱스 블록은 해당 컴포넌트에 저장된 해시-주소 쌍들에 대한 인덱스로서 상기 지오해시값에 관한 B+ 트리를 포함하며, 상기 적어도 하나의 해시-주소 블록은 상기 해시-주소 쌍을 상기 지오해시값에 대해 정렬된 상태로 저장할 수 있다.
- [0017] 상기 디스크 컴포넌트에 플러시하는 단계들 각각은 동일 레벨의 컴포넌트들 중에서 상기 키 범위가 같은 컴포넌트들을 하나의 컴포넌트로 병합하는 단계; 및 상기 병합된 컴포넌트의 크기가 임계값을 초과하는 경우 하위 레벨의 새로운 컴포넌트를 생성하는 단계를 포함할 수 있다.
- [0018] 상기 방법은 질의 포인트 및 질의 범위에 관한 범위 질의를 수신하는 단계; 상기 컴포넌트 테이블을 스캔하여 상기 범위 질의에 관한 결과 집합을 생성하는 단계; 및 상기 결과 집합을 상기 범위 질의에 대한 응답으로 제공하는 단계를 더 포함할 수 있다.
- [0019] 상기 결과 집합을 생성하는 단계는 공간 필터와 동일한 크기의 비트 스트링으로 구현되는 질의 필터를 생성하는 단계; 상기 질의 필터를 이용하여 컴포넌트 테이블을 스캔하여 상기 범위 질의에 연관된 객체를 포함하는 적어도 하나의 질의 컴포넌트를 검출하는 단계; 및 상기 적어도 하나의 질의 컴포넌트에 대한 인덱스를 탐색하면서 상기 범위 질의에 관한 질의 결과를 생성하여 결과 집합에 추가하는 단계를 포함할 수 있다.
- [0020] 상기 질의 필터를 생성하는 단계는 컴포넌트 레벨 별로 상기 공간 필터의 크기에 대응되는 상기 질의 필터를 생성하는 단계를 포함하고, 상기 질의 필터는 상기 질의 포인트와 상기 질의 범위에 의해 결정되는 질의 영역에 관한 정보를 포함할 수 있다.
- [0021] 상기 적어도 하나의 질의 컴포넌트를 검출하는 단계는 특정 컴포넌트의 공간 필터와 상기 질의 필터 간의 논리 곱(AND) 연산을 수행한 결과 0보다 큰 비트가 존재하는 경우 상기 특정 컴포넌트를 질의 컴포넌트로 결정하는 단계를 포함할 수 있다.

- [0022] 상기 결과 집합을 생성하는 단계는 상기 메모리 컴포넌트를 스캔하여 상기 범위 질의와 연관되는 컴포넌트들을 후보 집합으로 생성하는 단계; 상기 질의 범위를 기초로 상기 컴포넌트 테이블을 스캔하면서 상기 질의 범위를 커버하는 컴포넌트들에 대해 상기 질의 포인트와의 거리를 계산하여 우선순위 큐에 삽입하는 단계; 상기 컴포넌트 테이블에 대한 스캔이 완료되면 상기 질의 포인트로부터 가장 가까운 거리를 갖는 루트 노드를 상기 우선순위 큐를 통해 선택하고 자식 노드들에 대한 탐색을 수행하는 단계; 상기 탐색이 완료되면 상기 후보 집합 및 상기 질의 반경을 갱신하는 단계; 및 상기 우선순위 큐를 통해 다음 노드를 선택하여 상기 탐색을 반복하는 과정에서 상기 다음 노드와 상기 질의 포인트 간의 거리가 상기 질의 반경보다 큰 경우 상기 반복을 종료하고 현재의 후보 집합을 상기 결과 집합으로 결정하는 단계를 포함할 수 있다.
- [0023] 실시예들 중에서, 공간 분할 기반의 트리 인덱싱 및 질의어 처리 장치는 공간 데이터 객체를 저장하는 공간 블록체인; 상기 공간 데이터 객체를 인덱싱하는 인덱스 트리; 및 상기 인덱스 트리를 생성하는 인덱스 트리 생성부를 포함한다.
- [0024] 상기 인덱스 트리는 상기 공간 데이터 객체의 지오해시(geohash)값과 상기 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 생성된 해시-주소 쌍을 저장하는 메모리 컴포넌트; 상기 메모리 컴포넌트와 연동하여 동작하고, 상기 해시-주소 쌍을 저장하는 복수의 컴포넌트들을 레벨 별로 그룹화 하여 저장하는 디스크 컴포넌트; 및 컴포넌트들에 대한 정보로서 컴포넌트 레벨(component level), 블록 주소(block address), 키 범위(key range) 및 공간 필터(spatial filter)를 저장하는 컴포넌트 테이블을 포함할 수 있다.
- [0025] 상기 장치는 상기 인덱스 트리를 이용하여 상기 공간 데이터 객체에 관한 범위 질의를 처리하는 질의 처리부를 더 포함할 수 있다.

발명의 효과

- [0027] 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.
- [0028] 본 발명의 일 실시예에 따른 지리공간 블록체인 데이터 검색을 위한 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법 및 장치는 블록체인 환경에서 지리 공간 데이터를 효과적으로 처리하기 위한 것으로, 대량의 업데이트에 최적화된 인덱스 구조인 LSM tree 기반을 확장한 공간 데이터 인덱싱 기법과 이를 활용한 공간 질의 처리 기법을 제공할 수 있다.
- [0029] 본 발명의 일 실시예에 따른 지리공간 블록체인 데이터 검색을 위한 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법 및 장치는 메모리상에서 데이터의 업데이트를 일괄 처리하고 디스크에 대한 순차 쓰기 작업을 통해 업데이트 결과를 저장함으로써 디스크에 대한 임의 접근 비용을 절감할 수 있다.

도면의 간단한 설명

- [0031] 도 1은 본 발명에 따른 질의 처리 시스템을 설명하는 블록도이다.
- 도 2는 도 1에 있는 질의 처리 장치의 물리적 구성을 설명하는 도면이다.
- 도 3은 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법을 설명하는 순서도이다.
- 도 4는 본 발명에 따른 공간 블록체인에 대한 공간 LSM tree 구조를 설명하는 도면이다.
- 도 5는 본 발명에 따른 컴포넌트 구조를 설명하는 도면이다.
- 도 6은 본 발명에 따른 컴포넌트 테이블의 일 실시예를 설명하는 도면이다.
- 도 7은 본 발명에 따른 디스크 컴포넌트의 병합 및 플러시의 일 실시예를 설명하는 도면이다.
- 도 8은 본 발명에 따른 컴포넌트의 공간 필터를 설명하는 도면이다.
- 도 9는 본 발명에 따른 범위 질의 필터와 컴포넌트 필터링을 설명하는 도면이다.

도 10은 본 발명에 따른 kNN 질의 처리의 일 실시예를 설명하는 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0032] 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.
- [0033] 한편, 본 출원에서 서술되는 용어의 의미는 다음과 같이 이해되어야 할 것이다.
- [0034] "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.
- [0035] 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.
- [0036] 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.
- [0037] 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.
- [0038] 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 있다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.
- [0039] 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.
- [0041] 도 1은 본 발명에 따른 질의 처리 시스템을 설명하는 블록도이다.
- [0042] 도 1을 참조하면, 질의 처리 시스템(100)은 사용자 단말(110) 및 질의 처리 장치(130)를 포함할 수 있다.
- [0043] 사용자 단말(110)은 공간 데이터 객체에 관한 범위 질의를 요청하여 관련 응답을 확인할 수 있는 컴퓨팅 장치에 해당할 수 있고, 스마트폰, 노트북 또는 컴퓨터로 구현될 수 있으며, 반드시 이에 한정되지 않고, 태블릿 PC 등 다양한 디바이스로도 구현될 수 있다. 사용자 단말(110)은 질의 처리 장치(130)와 네트워크를 통해 연결될 수 있고, 복수의 사용자 단말(110)들은 질의 처리 장치(130)와 동시에 연결될 수 있다.
- [0044] 질의 처리 장치(130)는 공간 데이터 객체를 공간 블록체인에 저장하고 공간 데이터 객체들에 관한 인덱스 트리를 생성하여 범위 질의를 효과적으로 처리할 수 있는 컴퓨터 또는 프로그램에 해당하는 서버로 구현될 수 있다. 질의 처리 장치(130)는 사용자 단말(110)과 블루투스, WiFi 등을 통해 무선으로 연결될 수 있고, 네트워크를 통

해 사용자 단말(110)과 데이터를 주고 받을 수 있다. 질의 처리 장치(130)는 독립적으로 구현된 데이터베이스(도 1에 미도시함)와 연동하여 인덱스 트리 구축 및 범위 질의 처리 과정에서 다양한 형태로 수집되거나 가공된 정보들을 저장할 수 있다.

- [0046] 도 2는 도 1에 있는 질의 처리 장치의 시스템 구성을 설명하는 도면이다.
- [0047] 도 2를 참조하면, 질의 처리 장치(130)는 프로세서(210), 메모리(230), 사용자 입출력부(250) 및 네트워크 입출력부(270)를 포함하여 구현될 수 있다.
- [0048] 프로세서(210)는 질의 처리 장치(130)가 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법을 통해 동작하는 과정에서의 각 단계들을 처리하는 프로시저를 실행할 수 있고, 그 과정 전반에서 읽혀지거나 작성되는 메모리(230)를 관리할 수 있으며, 메모리(230)에 있는 휘발성 메모리와 비휘발성 메모리 간의 동기화 시간을 스케줄할 수 있다. 프로세서(210)는 질의 처리 장치(130)의 동작 전반을 제어할 수 있고, 메모리(230), 사용자 입출력부(250) 및 네트워크 입출력부(270)와 전기적으로 연결되어 이들 간의 데이터 흐름을 제어할 수 있다. 프로세서(210)는 질의 처리 장치(130)의 CPU(Central Processing Unit)로 구현될 수 있다.
- [0049] 메모리(230)는 SSD(Solid State Drive) 또는 HDD(Hard Disk Drive)와 같은 비휘발성 메모리로 구현되어 질의 처리 장치(130)에 필요한 데이터 전반을 저장하는데 사용되는 보조기억장치를 포함할 수 있고, RAM(Random Access Memory)과 같은 휘발성 메모리로 구현된 주기억장치를 포함할 수 있다.
- [0050] 사용자 입출력부(250)는 사용자 입력을 수신하기 위한 환경 및 사용자에게 특정 정보를 출력하기 위한 환경을 포함할 수 있다. 예를 들어, 사용자 입출력부(250)는 터치 패드, 터치 스크린, 화상 키보드 또는 포인팅 장치와 같은 어댑터를 포함하는 입력장치 및 모니터 또는 터치스크린과 같은 어댑터를 포함하는 출력장치를 포함할 수 있다. 일 실시예에서, 사용자 입출력부(250)는 원격 접속을 통해 접속되는 컴퓨팅 장치에 해당할 수 있고, 그러한 경우, 질의 처리 장치(130)는 서버로서 수행될 수 있다.
- [0051] 네트워크 입출력부(270)은 네트워크를 통해 외부 장치 또는 시스템과 연결하기 위한 환경을 포함하고, 예를 들어, LAN(Local Area Network), MAN(Metropolitan Area Network), WAN(Wide Area Network) 및 VAN(Value Added Network) 등의 통신을 위한 어댑터를 포함할 수 있다.
- [0053] 도 3은 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법을 설명하는 순서도이다.
- [0054] 도 3을 참조하면, 질의 처리 장치(130)는 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법에 따라 동작할 수 있다. 보다 구체적으로, 질의 처리 장치(130)는 본 발명에 따른 공간 분할 기반의 트리 인덱싱 방법을 이용하여 인덱스 트리를 구축할 수 있다(단계 S310). 즉, 본 발명에 따른 트리 인덱싱 방법은 블록체인에 저장된 대용량 지리 공간 데이터에 대한 빠른 업데이트를 지원하는 인덱스 구조인 spatial LSM tree를 새롭게 적용하여 구현될 수 있다.
- [0055] 구체적으로, 본 발명에 따른 인덱스는 삽입된 객체를 메모리 컴포넌트에 저장하고 메모리에 저장된 데이터가 가용 용량을 초과하는 경우 디스크에 대한 순차 쓰기 작업을 통해 디스크에 대한 임의 접근 횟수를 최소화하는 방식으로 업데이트 결과를 저장할 수 있다. 또한, 본 발명에 따른 인덱스는 데이터를 저장하는 컴포넌트와 저장된 컴포넌트들에 대한 요약 정보를 저장하는 컴포넌트 테이블로 구성될 수 있고, 컴포넌트 테이블은 컴포넌트들의 레벨, 키 값의 범위, 시작 주소와 같은 메타 데이터를 저장하며 컴포넌트 필터링을 위한 공간 필터를 함께 저장할 수 있다.
- [0056] 또한, 질의 처리 장치(130)는 사용자 단말(110)로부터 범위 질의를 수신할 수 있다(단계 S330). 이때, 범위 질의는 공간 상의 위치 정보에 대응되는 질의 포인트와 공간 상의 영역 정보에 대응되는 질의 범위를 포함할 수 있다. 이후, 질의 처리 장치(130)는 기 구축된 인덱스 트리를 기초로 범위 질의에 대한 질의 응답을 생성하여 사용자 단말(110)에게 제공할 수 있다(단계 S350).
- [0057] 즉, 질의 처리 장치(130)는 본 발명에 따른 인덱스 구조를 활용한 공간 질의 처리 기법이 적용될 수 있다. 이때, 공간 질의 처리 알고리즘은 공간 질의인 범위 질의와 kNN 질의에 대한 알고리즘을 포함할 수 있다. 알고리즘은 메모리 컴포넌트와 컴포넌트 테이블을 스캔하여 질의 영역과 관련이 있는 컴포넌트들만을 선택적으로 검색할 수 있고, 질의 처리기는 개별 컴포넌트에 대한 검색 결과를 최종적으로 병합하여 질의의 결과를 도출할

수 있다.

- [0059] 도 4는 본 발명에 따른 공간 블록체인에 대한 공간 LSM tree 구조를 설명하는 도면이다.
- [0060] 도 4를 참조하면, 질의 처리 장치(130)는 공간 LSM tree 구조에 기초하여 트리 인덱싱을 수행할 수 있다. 이를 위한 구성으로서, 질의 처리 장치(130)는 공간 데이터 객체를 저장하는 공간 블록체인(450), 공간 데이터 객체를 인덱싱하는 인덱스 트리(410) 및 인덱스 트리(410)를 생성하는 인덱스 트리 생성부(도 4에 미도시됨)를 포함할 수 있다.
- [0061] 일 실시예에서, 인덱스 트리(410)는 메모리 컴포넌트(420), 컴포넌트 테이블(430) 및 디스크 컴포넌트(440)를 포함할 수 있다. 메모리 컴포넌트(420)는 공간 데이터 객체의 지오해시(geohash)값과 공간 블록체인 상의 블록 주소(blockaddress)를 기초로 생성된 해시-주소 쌍을 저장할 수 있다. 컴포넌트 테이블(430)은 컴포넌트들에 대한 정보로서 컴포넌트 레벨(component level), 블록 주소(block address), 키 범위(key range) 및 공간 필터(spatial filter)를 저장할 수 있다. 디스크 컴포넌트(440)는 메모리 컴포넌트(420)와 연동하여 동작하고, 해시-주소 쌍을 저장하는 복수의 컴포넌트들을 레벨 별로 그룹화 하여 저장할 수 있다.
- [0062] 보다 구체적으로, 질의 처리 장치(130)는 다음의 구체적 단계들을 통해 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법을 수행할 수 있다.
- [0063] 첫번째 단계에서, 본 발명에 따른 공간 분할 기반의 트리 인덱싱 및 질의어 처리 방법은 공간 데이터 객체를 수신하여 공간 블록체인(450)에 저장할 수 있다. 즉, 공간 데이터 객체는 공간 블록체인(450)을 구성하는 데이터 블록들 중 어느 하나에 저장될 수 있으며, 경우에 따라 새롭게 생성된 데이터 블록에 저장될 수도 있다.
- [0064] 두번째 단계에서, 트리 인덱싱 및 질의어 처리 방법은 공간 데이터 객체의 지오해시(geohash)값과 공간 블록체인(450) 상의 블록 주소(blockaddress)를 기초로 해시-주소 쌍을 생성할 수 있고, 해시-주소 쌍을 메모리 컴포넌트(memory component)(420)에 저장할 수 있다.
- [0065] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 각각이 인덱스 블록(index block) 및 적어도 하나의 해시-주소 블록(geohash/address block)을 포함하는 복수의 컴포넌트(component)들 중 어느 하나에 저장하는 단계를 포함할 수 있다.
- [0066] 이때, 인덱스 블록은 해당 컴포넌트에 저장된 해시-주소 쌍들에 대한 인덱스로서 지오해시값에 관한 B+ 트리를 포함할 수 있으며, 적어도 하나의 해시-주소 블록은 해시-주소 쌍을 지오해시값에 대해 정렬된 상태로 저장할 수 있다.
- [0067] 세번째 단계에서, 트리 인덱싱 및 질의어 처리 방법은 메모리 컴포넌트(420)의 공간 점유율이 기 설정된 제1 임계값을 초과하는 경우 메모리 컴포넌트(420)에 저장된 제1 항목들을 순차적으로 디스크 컴포넌트(disk component)(440)에 플러시(flush)할 수 있다.
- [0068] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 동일 레벨의 컴포넌트들 중에서 키 범위가 같은 컴포넌트들을 하나의 컴포넌트로 병합하는 단계, 및 병합된 컴포넌트의 크기가 임계값을 초과하는 경우 하위 레벨의 새로운 컴포넌트를 생성하는 단계를 포함할 수 있다.
- [0069] 네번째 단계에서, 트리 인덱싱 및 질의어 처리 방법은 디스크 컴포넌트(440)의 공간 점유율이 기 설정된 제2 임계값을 초과하는 경우 디스크 컴포넌트(440)에 저장된 제2 항목들을 순차적으로 하위 레벨의 디스크 컴포넌트(440)에 플러시할 수 있다.
- [0070] 다섯번째 단계에서, 트리 인덱싱 및 질의어 처리 방법은 메모리 컴포넌트(420) 또는 디스크 컴포넌트(440)의 변경을 반영하여 컴포넌트 테이블(component table)(430)을 갱신할 수 있다. 이때, 컴포넌트 테이블(430)은 컴포넌트들을 컴포넌트 레벨 순서에 따라 정렬하여 저장할 수 있다. 또한, 컴포넌트 테이블(430)에 저장되는 공간 필터는 해당 컴포넌트가 커버하는 영역에서 데이터가 분포하는 영역에 관한 정보를 포함하는 비트 스트링(bit string)으로 구현될 수 있다.
- [0071] 일 실시예에서, 공간 필터는 다음의 구체적 동작에 의해 생성될 수 있다. 즉, 공간 필터는 해당 컴포넌트가 커버하는 영역을 n (상기 n 은 자연수)개의 셀 영역들로 분할하는 단계, 셀 영역들을 지오해시 순서로 정렬하는 단계, j (상기 j 는 자연수)번째 셀 영역에 객체가 존재하는 경우 j 번째 비트(bit)를 '1'로 표시하고 그렇지 않은 경우 '0'으로 표시하는 단계, 및 모든 셀 영역에 대응되는 n 개의 비트들 각각의 값을 결정하여 비트 스트링을

생성하는 단계의 순차적인 실행을 통해 생성될 수 있다.

- [0072] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 질의 포인트 및 질의 범위에 관한 범위 질의를 수신하고, 컴포넌트 테이블(430)을 스캔하여 범위 질의에 관한 결과 집합을 생성하며, 결과 집합을 범위 질의에 대한 응답으로 제공할 수 있다.
- [0073] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 결과 집합을 생성하는 단계에서, 공간 필터와 동일한 크기의 비트 스트링으로 구현되는 질의 필터를 생성하는 단계, 질의 필터를 이용하여 컴포넌트 테이블을 스캔하여 범위 질의에 연관된 객체를 포함하는 적어도 하나의 질의 컴포넌트를 검출하는 단계, 및 적어도 하나의 질의 컴포넌트에 대한 인덱스를 탐색하면서 범위 질의에 관한 질의 결과를 생성하여 결과 집합에 추가하는 단계를 포함할 수 있다.
- [0074] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 질의 필터를 생성하는 단계에서, 컴포넌트 레벨 별로 공간 필터의 크기에 대응되는 질의 필터를 생성하는 단계를 포함할 수 있다. 이때, 질의 필터는 질의 포인트와 질의 범위에 의해 결정되는 질의 영역에 관한 정보를 포함할 수 있다.
- [0075] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 적어도 하나의 질의 컴포넌트를 검출하는 단계에서, 특정 컴포넌트의 공간 필터와 질의 필터 간의 논리곱(AND) 연산을 수행한 결과 0보다 큰 비트가 존재하는 경우 특정 컴포넌트를 질의 컴포넌트로 결정하는 단계를 포함할 수 있다.
- [0076] 일 실시예에서, 트리 인덱싱 및 질의어 처리 방법은 결과 집합을 생성하는 단계에서, 메모리 컴포넌트를 스캔하여 범위 질의와 연관되는 컴포넌트들을 후보 집합으로 생성하는 단계, 질의 범위를 기초로 컴포넌트 테이블을 스캔하면서 질의 범위를 커버하는 컴포넌트들에 대해 질의 포인트와의 거리를 계산하여 우선순위 큐에 삽입하는 단계, 컴포넌트 테이블에 대한 스캔이 완료되면 질의 포인트로부터 가장 가까운 거리를 갖는 루트 노드를 우선순위 큐를 통해 선택하고 자식 노드들에 대한 탐색을 수행하는 단계, 탐색이 완료되면 후보 집합 및 질의 반경을 갱신하는 단계, 및 우선순위 큐를 통해 다음 노드를 선택하여 탐색을 반복하는 과정에서 다음 노드와 질의 포인트 간의 거리가 질의 반경보다 큰 경우 반복을 종료하고 현재의 후보 집합을 결과 집합으로 결정하는 단계를 포함할 수 있다.
- [0077] 도 4에서, Spatial LSM tree는 메모리 상에서 데이터의 업데이트를 일괄 처리하고, 디스크에 대한 순차 쓰기 작업을 통해 업데이트 결과를 저장함으로써 디스크에 대한 임의 접근 비용을 절감할 수 있다. 공간 블록체인(450)에 저장된 공간 데이터 객체는 객체의 좌표로부터 계산된 지오해시(Geohash)값과 데이터가 저장된 블록의 주소를 결합한 해시-주소(Geohash-address) 쌍을 통해 식별 및 접근될 수 있다. Spatial LSM tree는 해시-주소(Geohash-address) 쌍에 대한 공간 인덱스에 해당할 수 있다. spatial LSM tree는 메모리 컴포넌트(420)와 디스크 컴포넌트(440), 컴포넌트들에 대한 요약 정보를 저장하는 컴포넌트 테이블(430)로 구성되며 컴포넌트들은 계층적인 구조를 형성할 수 있다.
- [0078] 또한, 메모리 컴포넌트(420)는 삽입된 (Geohash, BlockAddress)를 우선적으로 저장하는 컴포넌트에 해당할 수 있다. 만약 메모리 컴포넌트(420)의 공간 점유율이 지정된 임계값(제1 임계값)을 초과하는 경우, 메모리 컴포넌트(420)에 저장되어 있는 항목(제1 항목)들은 순차적으로 디스크 컴포넌트(440)에 플러시될 수 있다. 데이터베이스의 크기가 증가하면서 디스크 컴포넌트(440)의 크기 또한 임계값(제2 임계값)을 초과하게 될 경우 디스크 컴포넌트(440)는 항목(제2 항목)들을 하위 레벨의 디스크 컴포넌트(440)로 플러시할 수 있다.
- [0080] 도 5는 본 발명에 따른 컴포넌트 구조를 설명하는 도면이다.
- [0081] 도 5를 참조하면, 컴포넌트(C_{ij})(510)는 인덱스 블록(Index block)(511)과 적어도 하나의 해시-주소 블록(Geohash/address block)(513)을 포함할 수 있다.
- [0082] 해시-주소(Geohash/address) 블록(513)은 (Geohash, BlockAddress) 쌍을 지오해시(Geohash)값에 대해 정렬된 상태로 저장할 수 있다. 인덱스 세그먼트(Index block)는 컴포넌트에 포함된 (Geohash, BlockAddress) 쌍에 대한 인덱스로 Geohash 값에 대한 B+ 트리(B+ tree)를 저장할 수 있다. Spatial LSM tree는 컴포넌트가 가질 수 있는 Geohash 값의 범위를 제한하여 컴포넌트에 공간적 속성을 부여할 수 있다. 메모리 컴포넌트가 가질 수 있는 Geohash 값의 범위는 제한이 없으며, 하위 레벨의 컴포넌트는 상위 레벨 컴포넌트가 커버하는 영역을 사분할한 사분면에 대한 Geohash 값의 범위를 키 값의 범위로 포함할 수 있다.

- [0084] 도 6은 본 발명에 따른 컴포넌트 테이블의 일 실시예를 설명하는 도면이다.
- [0085] 도 6을 참조하면, 컴포넌트 테이블(430)은 Spatial LSM tree를 구성하는 컴포넌트들에 대한 정보를 저장할 수 있다. 컴포넌트 테이블(430)은 컴포넌트의 레벨(Component Lv.)과 인덱스 주소(Block address), 키 값의 범위(Key range), 컴포넌트 필터(Spatial filter)(610)를 저장하고 있으며, 컴포넌트가 속한 레벨 순서대로 정렬하여 저장할 수 있다. 공간 필터(610)는 컴포넌트가 커버하는 영역에서 데이터가 분포하는 영역을 대략적으로 나타내는 bit string에 해당할 수 있다.
- [0087] 도 7은 본 발명에 따른 디스크 컴포넌트의 병합 및 플러시의 일 실시예를 설명하는 도면이다.
- [0088] 보다 구체적으로, 그림 (a)는 Level 1 컴포넌트에 대한 병합 작업의 일 실시예를 나타내고, 그림 (b)는 Level 컴포넌트에 대한 플러시 작업의 일 실시예에 해당할 수 있다.
- [0089] 도 7을 참조하면, 동일한 레벨에 속하는 컴포넌트들 중에서 키 값의 범위가 같은 컴포넌트들은 하나의 컴포넌트로 병합될 수 있고, 병합된 컴포넌트의 크기가 임계값을 초과할 경우 하위 레벨에 새로운 컴포넌트를 생성하는 방법을 통해 플러시 될 수 있다. 그림 (a)에서 레벨 1의 키 값의 범위가 0-15인 컴포넌트와 32-47인 컴포넌트가 각각 2개씩 존재하므로 해당 컴포넌트들은 하나의 컴포넌트로 병합될 수 있다. 또한, 그림 (b)에서 키 값의 범위가 0-15인 컴포넌트들의 병합으로 그 크기가 임계값을 초과하면 레벨 2에 새로운 컴포넌트가 생성될 수 있고 항목들이 순서대로 플러시 될 수 있다.
- [0091] 도 8은 본 발명에 따른 컴포넌트의 공간 필터를 설명하는 도면이다.
- [0092] 도 8을 참조하면, 공간 필터는 컴포넌트에 포함된 Geohash 쌍들의 공간적 분포를 n개의 비트로 이루어진 비트 스트링으로 표현될 수 있다. 공간 필터를 생성하는 방법은 다음의 단계들을 통해 수행될 수 있다. 먼저 filter를 생성하고자 하는 컴포넌트가 커버하는 영역을 n개의 셀로 분할할 수 있다. 분할된 공간은 Geohash 순서로 정렬될 수 있으며, i 번째 셀이 커버하는 영역에 객체(810)가 존재할 경우 i 번째 bit을 1로, 존재하지 않는 경우 0으로 표시할 수 있다. 그림 (a)에서 컴포넌트 A가 커버하는 영역은 4×4개의 셀들로 분할될 수 있고, 7개의 오브젝트(객체)(810) $O_1 \sim O_7$ 가 분포할 수 있다. 주어진 공간에 대하여 분할된 셀들을 Geohash 순서로 정렬한 후 오브젝트를 포함하고 있는 셀은 회색으로 표시한 결과는 그림 (b)에 해당할 수 있다. 컴포넌트 A는 1, 4, 5, 10, 13, 14 번째 셀에 오브젝트(810)를 가지고 있기 때문에 공간 필터(830)는 그림 (c)와 같이 생성될 수 있다.
- [0094] 도 9는 본 발명에 따른 범위 질의 필터와 컴포넌트 필터링을 설명하는 도면이다.
- [0095] 도 9를 참조하면, 본 발명에 따른 Spatial LSM tree는 범위 질의와 kNN질의 처리를 지원할 수 있다. 범위 질의는 질의 포인트 q와 질의 범위 r을 입력받고 질의 포인트로부터의 거리가 r 이내에 위치한 객체들을 검색하는 질의에 해당할 수 있다. Spatial LSM tree 기반의 범위 질의 처리 알고리즘은 먼저 주어진 질의 조건을 바탕으로 각 레벨의 필터 분할 크기에 맞춰 질의 필터(910)를 생성할 수 있다. 질의 필터(910)는 컴포넌트 테이블을 스캔할 때 컴포넌트의 공간 필터(830)에 대한 AND 연산을 수행하여 해당 컴포넌트가 질의 영역과 관련된 객체를 포함하고 있는지 여부를 보다 정확히 알려줄 수 있다.
- [0096] 만약 AND 연산의 결과로 반환된 bit string에서 0보다 큰 비트가 존재한다면, 해당 컴포넌트는 질의 범위와 오버랩되는 영역에 대한 데이터를 포함할 수 있음을 나타낼 수 있다. 따라서, 해당 컴포넌트의 인덱스를 탐색하면서 범위 질의를 수행하고, 반환된 결과를 결과 집합에 추가할 수 있다.
- [0097] 한편, 아래의 알고리즘 1은 컴포넌트 필터링과 컴포넌트 탐색 단계로 이루어진 범위 질의 처리 알고리즘에 해당할 수 있다. 알고리즘 1에서 개별 컴포넌트 내부의 범위 질의를 처리하고 결과 집합을 반환하는 프로세스는 COMPONENT_RANGEQUERY 함수로 간략하게 표시될 수 있다. 해당 프로세스는 잘 알려진 B+트리 기반의 범위 질의 처리 알고리즘을 바탕으로 수행될 수 있다.

Algorithm 1. Range query processing algorithm**Input :** query point q , range r **Output :** A set R of objects located within query range**Initialize result set** $R \leftarrow \emptyset$ **For** each component c_i in Component table C Transform query point q and range r to query filter F_i **if** (c_i .filter AND $F_i > 0$) **then** $R = R \cup \text{COMPONENT_RANGEQUERY}(c_i, q, r)$

[0098]

[0100]

또한, 아래의 알고리즘 2는 kNN 질의 처리 알고리즘에 해당할 수 있다. 메모리 컴포넌트가 커버하는 영역은 데이터가 분포할 수 있는 전체 영역이기 때문에 메모리 컴포넌트에 대한 탐색은 최우선적으로 수행될 수 있다. 탐색이 완료되면 메모리 컴포넌트에서 찾아낸 후보군과 질의 범위 r 을 활용하여 컴포넌트 테이블을 스캔하면서 질의 범위를 커버하는 컴포넌트들은 질의 포인트와의 거리를 계산하여 우선순위 큐인 Q 에 삽입하고, 아닌 컴포넌트들은 필터링할 수 있다. 컴포넌트 테이블에 대한 스캔이 완료되면 질의 포인트로부터 가장 가까운 거리를 갖는 루트 노드를 선택하고 자식 노드들에 대한 검사를 수행할 수 있다. 컴포넌트에 대한 검사가 끝나면 후보군 및 질의 반경을 업데이트한 후 Q 에서 다음으로 가까운 컴포넌트 인덱스를 선택하고 마찬가지로 후보군 탐색을 수행할 수 있다. 탐색은 Q 에 속한 컴포넌트 인덱스의 루트가 현재까지의 질의 반경 r 보다 먼 경우 종료되며, 후보군 S 를 정답으로 반환할 수 있다.

Algorithm 2. Third kNN query processing algorithm**Input :** query point q , number of neighbor k **Output :** k nearest neighbors to q Initialize priority queue Q, S Candidate set $S \leftarrow$ kNN objects of the memory componentrange $r \leftarrow$ distance from query point q to k^{th} candidate/*Scan component table C */**For** each c_i in C Transform query range r to query filter F_i **if** (c_i .filter AND F_i) **then** $Q.\text{enqueue}(\langle \text{root node of } c_i.\text{index}, \text{MINDIST}(q, \text{root node of } c_i.\text{index}) \rangle)$ **while** (! $Q.\text{empty}()$) $\langle n, d \rangle = Q.\text{pop}()$ **if** ($d > r$) **then****break****else****if** (n is leaf)**for** each entry e in n $S.\text{enqueue}(e, \text{DIST}(q, e))$ **UPDATE** $r \leftarrow \text{DIST}(q, \text{farthest neighbor in } S)$ **else****for** each child d of n $N.\text{enqueue}(d, \text{MINDIST}(q, d))$ **while** (! $N.\text{empty}()$) $\langle n, d \rangle = N.\text{pop}()$ **if** ($d > r$) **then****break****if** (n is leaf)**for** each entry e in n $S.\text{enqueue}(e, \text{DIST}(q, e))$ **UPDATE** $r \leftarrow \text{DIST}(q, \text{farthest neighbor in } S)$ **else****for** each child d of n $Q.\text{enqueue}(d, \text{MINDIST}(q, d))$ **return** S

[0101]

[0103] 도 10은 본 발명에 따른 kNN 질의 처리의 일 실시예를 설명하는 도면이다.

[0104] 도 10을 참조하면, 상기의 알고리즘 2의 컴포넌트 탐색 순서를 확인할 수 있다. 질의 포인트는 3의 Geohash 값을 갖기 때문에 포인트를 완전히 커버하는 컴포넌트들은 최소 거리 0을 갖게 되고 우선적으로 탐색될 수 있다. 그림 (b)에서 탐색은 0-63의 범위를 갖는 메모리 컴포넌트, 0-15 범위를 갖는 Level 1 컴포넌트, 0-3 범위를 갖는 Level 2 컴포넌트 순서로 진행될 수 있다. 프로세서는 각각의 컴포넌트를 탐색할 때마다 kNN의 후보군과 범위 r을 업데이트할 수 있다. 탐색 과정에서 질의 반경에 대한 업데이트가 크게 일어나지 않은 경우에는 큐에 남아 있는 컴포넌트들을 최소 거리가 작은 순서대로 탐색할 수 있다. 결과적으로, 컴포넌트의 탐색은 그림 (b)에서 화살표로 표시된 순서대로 수행될 수 있으며, 탐색 과정에서 우선순위 큐 N에 남아있는 컴포넌트 중 현재까지 검색된 kNN 후보보다 거리가 먼 컴포넌트가 발견될 경우 탐색은 종료될 수 있다.

[0106] 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

[0108] 100: 질의 처리 시스템

110: 사용자 단말 130: 질의 처리 장치

210: 프로세서 230: 메모리

250: 사용자 입출력부 270: 네트워크 입출력부

410: 인덱스 트리 420: 메모리 컴포넌트

430: 컴포넌트 테이블 440: 디스크 컴포넌트

450: 공간 블록체인

510: 컴포넌트 511: 인덱스 블록

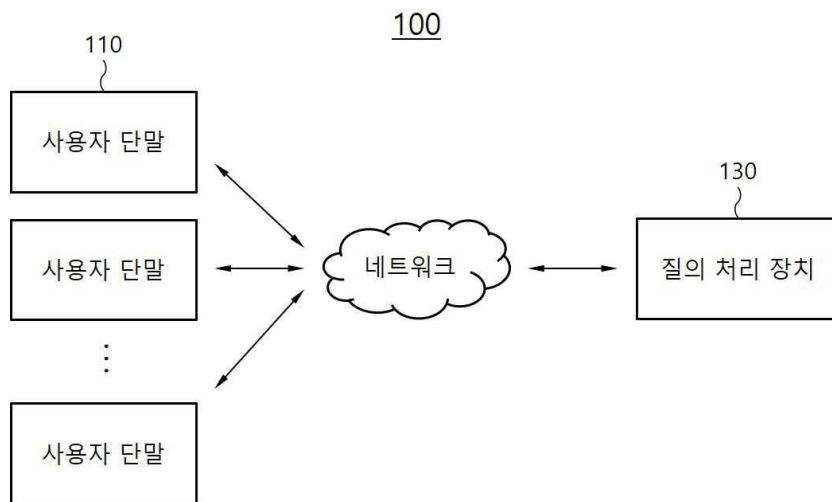
513: 해시-주소 블록 610: 공간 필터

810: 공간 데이터 객체 830: 공간 필터

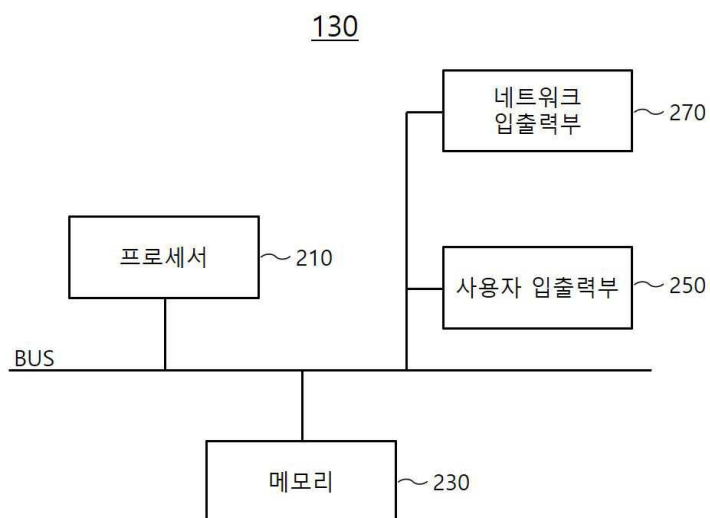
910: 질의 필터

도면

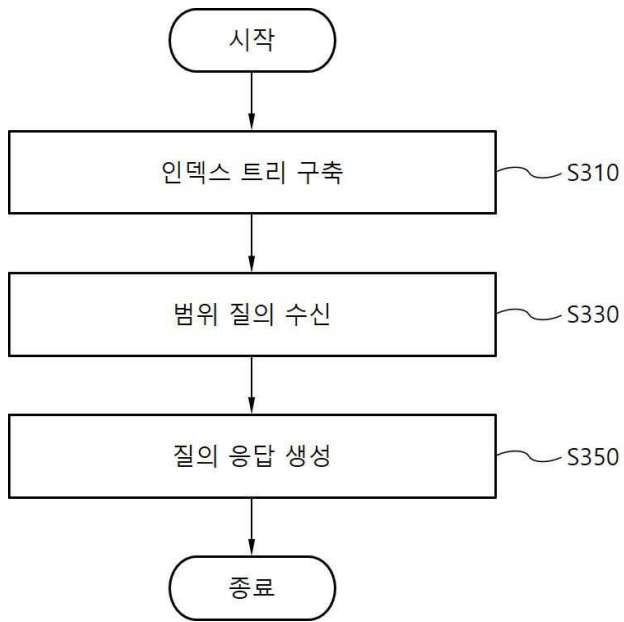
도면1



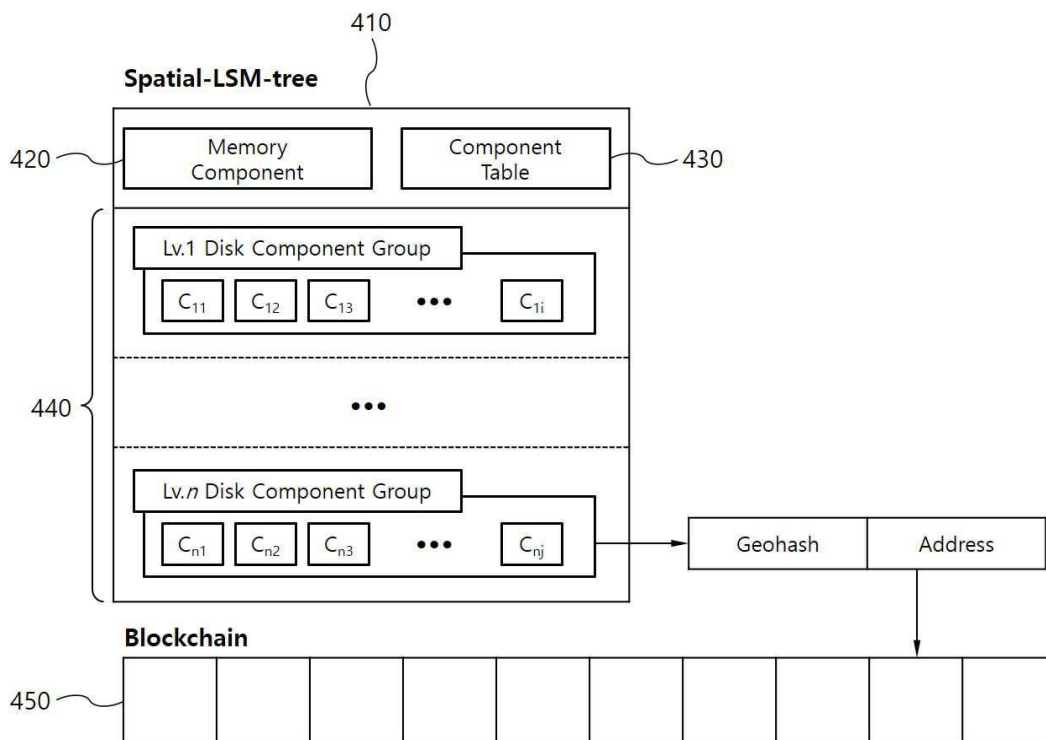
도면2



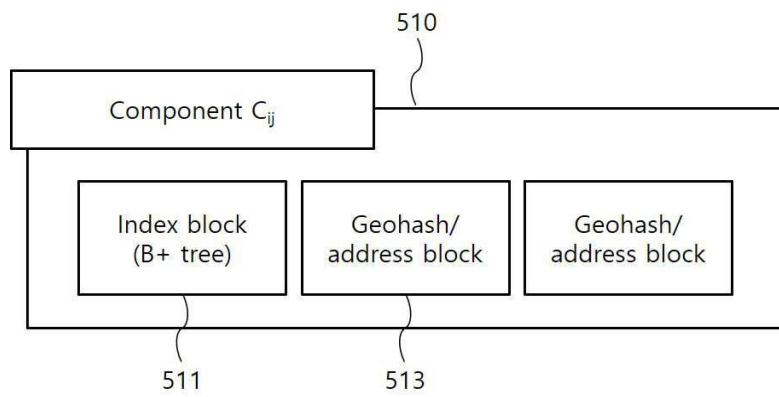
도면3



도면4



도면5

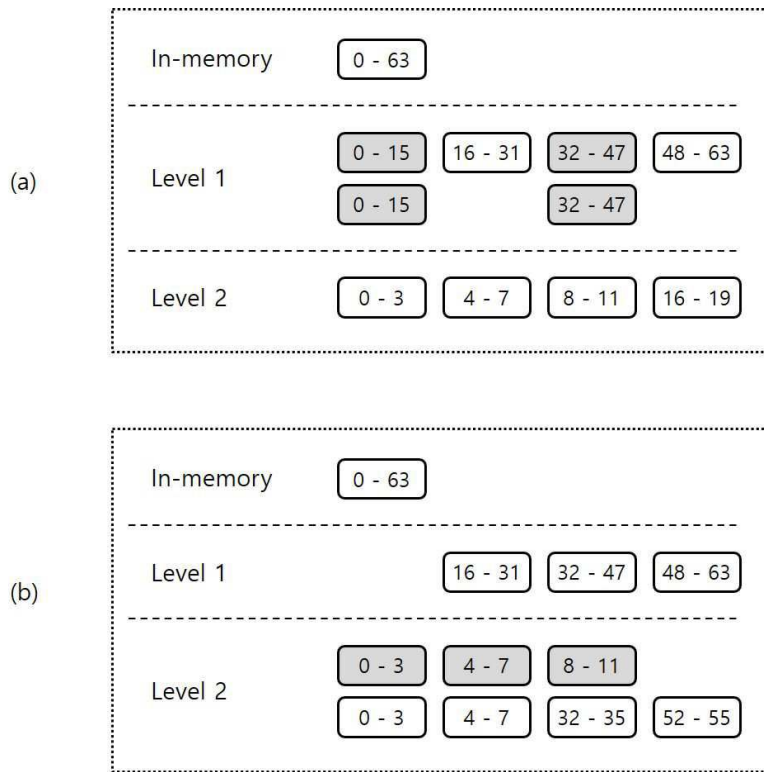


도면6

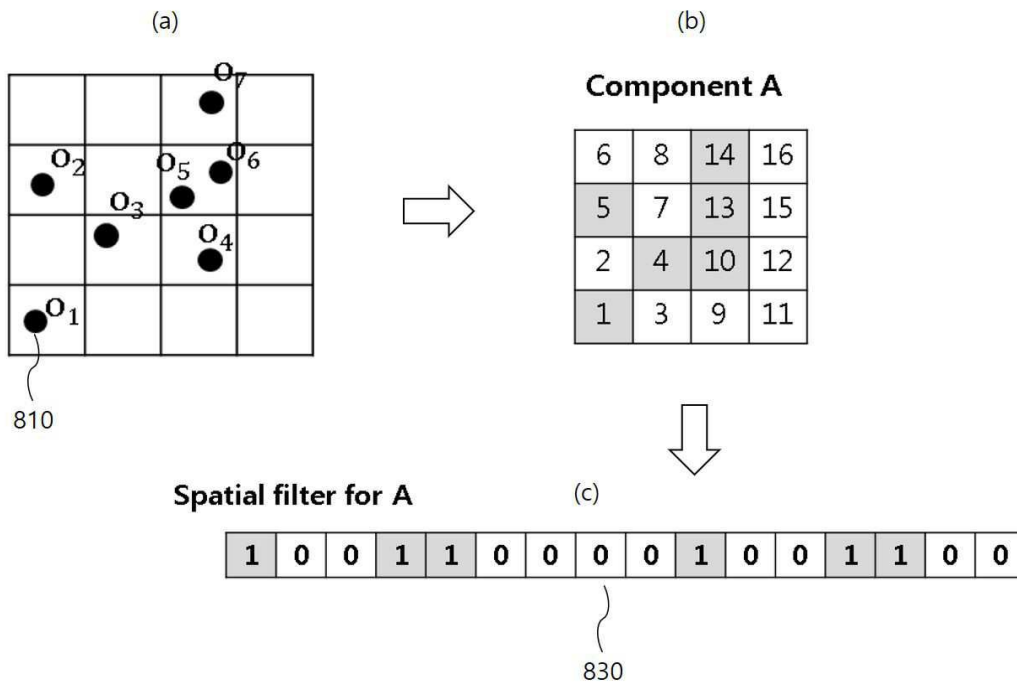
430

Component Lv.	Key range	Spatial filter	Block address
0	0-31	1101	...
1	0-7	1001	...
1	8-15	1011	...
2	0-3	1001	...
2	4-7	1011	...
2	8-11	0011	...

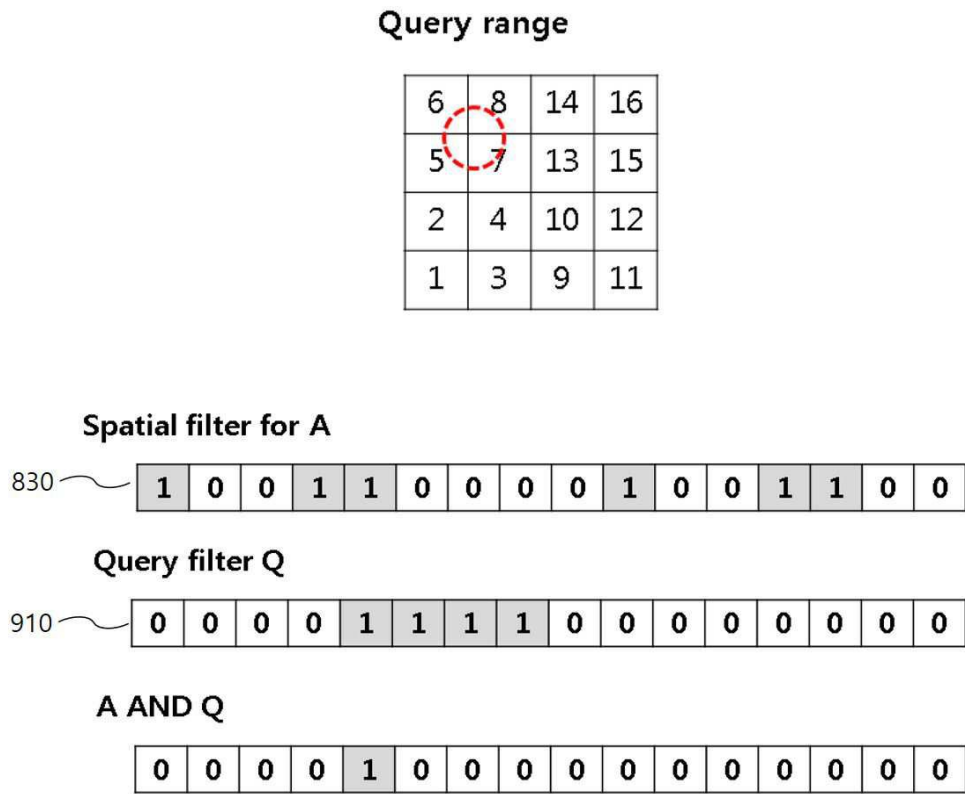
도면7



도면8



도면9



도면10

