

R-tree에서 GeoSpatial AR 응용을 위한 공간 필터링 기법

Spatial Filtering Techniques for Geospatial AR Applications in R-tree

박 장 유^{*} 이 성 호^{**} 남 광 우^{***}
Jang Yoo Park Seong Ho Lee Kwang Woo Nam

요 약 최근 스마트폰 기반의 공간정보를 서비스하는 증강현실 응용들이 보편화되고 있다. 이러한 공간정보 증강현실 응용들은 현재 위치를 기반으로 공간정보를 검색하게 되며, 이 때 카메라 뷰 앵글에 보이는 공간정보가 일차적인 중요성을 갖는다. R-tree 등 현재까지 공간 데이터베이스 분야에서 제안된 공간색인들 대부분은 최소경계사각형을 이용한 검색 알고리즘을 수행한다. 이는 전통적인 지리정보시스템, 위치기반서비스들에서는 유용하지만 증강현실 응용에서 사용되는 뷰 앵글 기반의 공간질의 특성을 반영하지 못한다. 즉, 증강현실 응용에서의 공간질의 비 단말 노드의 최소 경계 사각형과 질의 영역사이의 데드 스페이스 영역이 발생할 가능성이 높다. 제안된 알고리즘은 삼각형 노드 필터링 기법을 사용함으로써 이러한 데드 스페이스를 검색에서 배제한 공간 검색을 수행함으로써 검색의 효율성을 높였다. 이 논문에서는 제안된 알고리즘을 PostgreSQL/PostGIS 상에서 구현하였으며, 실제 POI 데이터베이스를 이용하여 실험하였다. 실험결과 제안된 알고리즘을 사용한 공간 검색이 최소 경계 사각형을 사용하는 검색 알고리즘보다 뚜렷한 성능 향상이 있었음을 확인할 수 있었다.

키워드 : 증강현실, R-tree 필터링 filtering, PostGIS

Abstract Recently, AR applications provide location-based spatial information by GPS. Also, the spatial information is displayed by the angle of the camera. So far, traditional spatial indexes in spatial database field retrieve and filter spatial information by the minimum bounding rectangle (MBR) algorithm.(ex. R-tree) MBR strategy is a useful technique in the geographic information systems and location-based services. But MBR technique doesn't reflect the characteristics of spatial queries in AR. Spatial queries of AR applications have high possibility of the dead space area between MBRs of non-leaf node and query area. We propose triangle node filtering algorithm that improved efficiency of spatial retrieval used the triangle node filtering techniques by exclusion the dead space. In this paper, the proposed algorithm has been implemented on PostgreSQL/PostGIS. Experimental results show the spatial retrieval that using the proposed algorithm better performance than the spatial retrieval that of the minimum bounding rectangle algorithm.

Keywords : Augmented Reality, R-tree Filtering, PostGIS

1. 서 론

최근 모바일과 관련한 정보통신 기술의 발전과 다양한 모바일 단말의 보급으로, 국내 인구 80% 이

상이 카메라가 달린 모바일 단말을 사용하고 있다 [21]. 또한 모바일 컴퓨팅 환경의 발달로 기존의 기술과 첨단 기술을 융합하여 새로운 서비스를 제공 하려는 시도가 계속되고 있다[22] 특히, 스마트폰의

[†] 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(No. 2009-0067958)이며, 국토해양부 첨단도시기술개발사업 - 지능형국토정보기술혁신사업과제의 연구비 지원(07국토정보C05)에 의해 수행되었습니다.

^{*} 군산대학교 컴퓨터정보공학과 연구원 parkstar82@naver.com

^{**} 한국전자통신연구원 융합기술연구부분 선임연구원 sholee@etri.re.kr

^{***} 군산대학교 컴퓨터정보공학과 부교수 kwnam@kunsan.ac.kr(교신저자)

보급으로 전통적인 모바일 서비스와 스마트폰에서 제공될 수 있는 높은 CPU성능을 기반으로 하는 기술들이 결합하여 다양한 콘텐츠들이 생산되고 있다.

전통적인 공간 정보 기술인 GIS(Geometry Information System)는 GPS(Global Positioning System)를 사용하는 단말과 결합하여 “나는 어디에 있나?”, “내 근처에 무엇이 있나?”, “그 곳까지 어떻게 갈 수 있나?”라는 질문에 대답할 수 있는 LBS(Location Based Service)를 제공할 수 있게 되었다. LBS가 제공하는 서비스는 다양하다. 보안 및 안전, 교통, 내비게이션, 위치를 기반으로 하는 정보 및 엔터테인먼트, SNS 등이 대표적이다. 보안 및 안전의 경우 어린 자녀들의 등하교 길의 위치를 알려주는 서비스 등이 인기를 얻고 있다[20].

한편, 증강현실(Augmented Reality)은 현실의 정보와 가상의 정보를 실시간으로 결합하여 더 많은 정보를 제공하는 기술이다[21]. 모바일 컴퓨팅 환경에서의 예를 들면 카메라가 음식점의 간판을 보고 있을 때 증강현실은 음식점의 간판 위에 음식점의 평가, 별점 등의 정보를 제공 하는 것이다. 증강현실은 카메라, 높은 그래픽 처리 능력, 빠른 무선 통신, GPS 등의 기능을 갖춘 단말기의 발달로 모바일 환경으로 진입하였다. 증강현실은 사용자의 정확한 위치, 바라보고 있는 방향, 자세, 동작 등을 파악하기 위하여 GPS와 센서가 필요하다. 또한, 현재 위치 주변의 정보도 필요하다. 이러한 증강현실의 필요조건 때문에 기존의 GIS, LBS와의 결합은 필수 가 되었다.

최근 스마트폰의 등장은 증강현실과 사용자 사이의 간극을 좁혀놓았다. 스마트폰의 빠른 그래픽 처리 능력, 카메라, 지자기 센서의 역할로 증강현실에 필요한 필수 요소들이 갖추어졌기 때문이다. 이러한 스마트폰을 위한 증강현실 기반의 위치기반 서비스 응용 프로그램들도 많이 등장하고 있다. 카메라로 주위 환경을 비추면 커피숍, 약국, 식당 등의 정보를 보여주는 레이어 등이 대표적이다.

모바일 단말기에서는 공간상에서 지속적으로 변화하는 사용자의 위치를 사용하기 때문에 공간데이터를 효율적으로 처리 하여야 한다. 그러나 이러한 위치 기반 서비스에 바탕을 둔 공간 데이터는 모바일 단말에 저장할 수 없기 때문에 대용량의 공간 데이터베이스 안에 저장된다. 마찬가지로 위치 기반 질의 또한 자주 발생하게 되어 질의응답 속도가 서

비스의 품질에 큰 영향을 미친다.

본 논문에서는 증강현실의 특성상 나타나는 삼각형 형태의 질의를 기존의 최소 경계 사각형 검색보다 효율적으로 처리하기 위한 알고리즘을 제안한다. 제안된 알고리즘은 최소 경계 사각형 검색에서 데드 스페이스를 검색에서 배제하기 때문에 기존의 R-tree[7] 종류의 최소 경계 사각형 기반 검색보다 응답 속도가 빠르다.

본 논문은 다음과 같이 구성되어 있다. 2장 관련 연구에서는 증강현실과 위치 기반 서비스에 관하여 알아본다. 또, GIS에서 일반적으로 많이 사용하고 있는 색인 구조인 R-tree, R+-tree[14], R*-tree[2]에 대해 살펴보고 비 MBR 인덱싱 기법을 사용하는 다른 색인 구조에 대해서 알아본다. 3장에서는 증강현실에서 나올 수 있는 질의와 문제를 정의한다. 4장에서는 본 논문에서 해결하고자 하는 문제점을 정의하고 증강현실에서의 범위질의를 효율적으로 처리하기 위한 알고리즘을 제안한다. 5장에서는 4장에서 제안한 알고리즘을 기반으로 실험한 결과를 기술한다. 마지막으로 6장에서는 향후 연구 및 결론을 맺는다.

2. 관련 연구

2.1 R-tree 공간 색인

공간 객체를 표현하는 대표적인 색인 구조는 R-tree 계열이 있다. R-tree 계열에는 대표적으로 R+-tree와 R*-tree등이 많이 쓰이고 있다. R-tree는 B-tree 계열의 균형 트리이다. 모든 단말 노드는 같은 레벨에 존재한다. 트리의 중간 노드는 하위 노드들의 영역을 포함하는 최소 경계 사각형(Minimum Bounding Rectangle)으로 구성된다. 단말노드에는 공간 객체가 저장되고 비 단말 노드에는 자식 노드에 대한 포인터와 자식 노드들의 영역을 포함하는 MBR이 저장된다. 비 단말 노드의 형태는(pointer, MBR)로 나타낸다. 단말 노드의 형태는 공간 객체에 대한 포인터와 공간 객체의 MBR의 형태로 나타내고(oid, MBR)로 표현한다. oid는 공간 객체가 저장된 디스크 페이지에 대한 포인터이다. 그림 1은 R-tree의 예를 보여준다. 이 그림에서 R1~R7은 R-tree의 중간 노드이며 하위 노드들의 영역을 포함하는 MBR을 나타내고 있다. R8부터 R19는 공간 객체를 저장한 MBR이다.

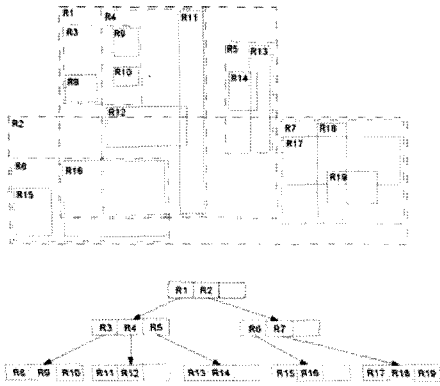


그림 1. R-tree 계층 구조의 예

R-tree에서는 비 단말 노드 MBR의 겹침이 생긴다. 따라서 R16에 대한 공간 질의를 수행할 시 R16에 겹침이 발생한 모든 노드에 대한 방문이 이루어져야 한다. 때문에 R-tree에서 검색 수행 시 접근해야 하는 노드의 수는 MBR의 겹침이 많을수록 증가하게 된다. 접근하는 노드가 많을수록 검색 효율은 떨어진다. 이러한 겹침으로 인해 증가된 검색 비용을 줄이기 위해 R*-tree, R+-tree등의 색인 구조가 제안되었다. 이 논문에서는 GiST기반 R-tree를 대상으로 GeoSpatial AR의 질의에 최적화된 MBR 필터링을 제안하고 있다.

2.2 GiST 공간 색인

보통 B-tree 이후 모든 데이터베이스 탐색 트리들은 B-tree로부터 파생되었다. 이 점을 이용해서 GiST(Generalized Search Tree)라고 부르는 새로운 종류의 인덱스를 개발했다. GiST(Generalized Search Tree)는 B-tree로부터 파생된 모든 tree들이 기본적으로 가지고 있어야 할 기능들을 제공하는 것과 개발자들이 임의의 데이터에 대한 인덱스를 개발할 수 있도록 확장 가능한 데이터 구조를 제공하는 것이다. 결국 이 GiST는 R-tree, B+-tree, TV-tree, Ch-tree등과 같은 유명한 탐색 트리들을 하나의 데이터 구조로 통합할 수 있게 된다.

GiST에서 기본적으로 구현해야할 메소드는 Consistent, Union, Penalty, PickSplit, Compress, Decompress 등의 6개 메소드이다. GiST는 이 6개 메소드만을 구현함으로써 간단히 트리형 색인을 구축할 수 있게 한다. 각 메소드는 다음과 같다.

Consistent 메소드는 사용자 질의 q 와 트리에 대

한 키 p 가 주어지면, Consistent는 q 와 p 를 비교하여 올바른 p 가 주어진 질의 q 에 만족하는지의 여부를 돌려준다. Union 메소드는 트리에서 특정 엔트리에 대한 상위 정보를 정리한다. 즉, 엔트리의 세트 S 가 주어지면 Union은 S 아래에 있는 데이터 아이템을 가리키는 key p 를 돌려준다. Penalty 메소드는 새로운 데이터 아이템이 삽입 될 때 루트로부터 서브트리로 적절한 노드를 선택할 수 있게 한다. Penalty 반환하는 값은 삽입하게 되었을 때 트리의 상태가 얼마나 나빠지느냐의 수치를 가리킨다. 결과적으로 아이템들은 트리에서 Penalty가 가장 작은 경로로 삽입되게 하는 것이 목적이다. PickSplit 메소드는 B-tree에서처럼, 때때로 GiST에서 페이지들은 새로운 데이터 아이템이 삽입 될 때 분할할 필요가 반드시 있다. PickSplit는 어떤 아이템들이 새로운 페이지에 들어가야 하는지, 어떤 아이템들이 기존 페이지에 들어가야 하는지를 결정하는 역할을 한다. Compress 메소드는 데이터 항목을 인덱스 페이지 내의 물리적인 기억장치에 적절한 형식으로 변환한다. Decompress 메소드는 compress 메소드의 역으로서 데이터 항목의 인덱스를 데이터베이스에 의해 처리될 수 있는 형식으로 변환한다.

이 논문에서는 GiST 기반의 R-tree MBR 필터링을 구현하기 위하여 consistent 메소드를 수정하는 방법을 사용하였다.

3. GeoSpatial AR 질의와 문제의 정의

3.1 GeoSpatial AR에서의 공간 검색 질의

증강현실은 사용자가 바라보는 방향에 위치한 지리 정보 객체에 덧입혀 부가적인 정보를 제공한다. 따라서 사용자의 정확한 위치정보와 함께 사용자의 시선 방향에 따라 질의를 요청하게 된다.

그림 2는 증강현실에서의 위치 질의 영역을 보여주고 있다. 초기의 증강현실을 이용한 위치 기반 서비스들에서는 사용자의 위치에서 검색 반경의 공간 객체 정보를 가져온다. 가져온 공간 객체 정보를 3D 구의 형태로 사용자 화면에 표시하게 되어 사용자의 시선 방향에 따라 어떤 공간 객체가 있는지 알려주었다. 하지만 이 방식의 공간검색은 두 가지 단점이 있다. 첫 째로, 검색 반경이 넓을수록 가져와야 하는 공간 객체의 정보가 많아진다. 모바일 단

말기에서는 화면에 표시할 수 있는 객체 정보의 양이 적기 때문에 필요한 만큼의 공간객체만을 가져오는 것이 좋다. 둘째로, 검색 반경이 넓거나 찾고자 하는 공간객체가 한 곳에 밀집해 있을 때 공간검색의 효율성이 낮아질 수 있다. 이는 공간검색의 속도를 저하시키는 요인이 된다. 따라서 증강현실에서의 공간 질의는 사용자의 위치와 사용자가 바라보는 방향을 함께 정해주어야 한다.

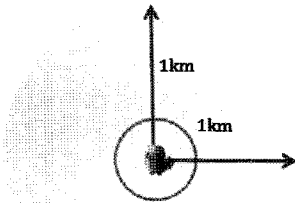


그림 2. 기존 AR의 공간 검색 방법

공간 AR 응용들이 일반적인 GIS 응용들의 질의와 같은 차이점은 그림 3과 같이 현재 AR 뷰어상에 보이는 영역을 대상으로 한 공간검색이 일반적이다. 즉, 일반적인 GIS에서의 범위 검색 형태의 질의와 다르게 증강 현실에서는 지자기 센서나 가속도 센서를 기반으로 현재의 위치에서 어떤 방향을 보고 있는지 알 수 있으며, 이를 기반으로 내 위치에 가까운 주요 공간 객체 중에서 화면 뷰앵글상에 현재 보이는 공간 객체 찾는 질의가 가장 많이 사용된다.

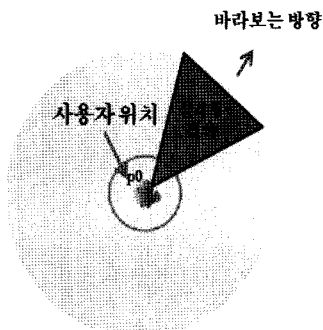


그림 3. AR에서의 공간질의 특성

3.2 문제의 정의

앞에서 살펴본바와 같이 증강현실에서의 가장 특징적인 질의 유형은 사용자가 바라보는 방향을 가진 단순형태의 질의가 된다. 단순형태의 질의는 기존 공간색인 기법으로 많이 사용되고 있는 R-tree 계열의 MBR 기반 검색에서는 데드 스페이스가 발생하여 검색 효율성의 저하를 가져 왔다.

그림 3에서 보는바와 같이 증강현실에서 질의하는 질의 영역은 R-tree와 같은 공간색인에서 사용하는 MBR 기반의 검색 영역과의 사이에 필연적으로 데드 스페이스가 발생하게 된다. 이러한 데드 스페이스는 비 단말 노드의 MBR과 겹침이 발생하게 되어 하위 노드로의 검색 횟수와 단말 노드의 디스크 접근 횟수를 증가 시키게 된다.

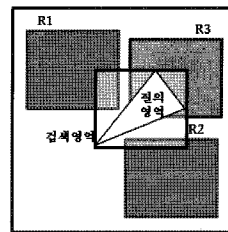


그림 4. AR에서 MBR 검색의 비효율성

그림 4는 GeoSpatial AR에서 사각형의 MBR 검색을 사용함으로써 발생할 수 있는 비효율성을 보이고 있다. 즉, GeoSpatial AR에서 주로 사용되는 뷰앵글을 기반으로 한 필터링을 사용하면 R3만 검색되지만, 일반적인 공간검색에서 사용되는 사각형 MBR을 이용했을 경우에는 추가적으로 R1, R2 까지 검색된다. 이러한 공간 필터상의 데드 스페이스, 즉 질의 영역과 검색 영역 사이에서 발생하는 데드 스페이스를 줄여 데드 스페이스와 겹침이 발생하는 비 단말 노드에 대한 검색을 수행하지 않는다면 검색 효율은 좋아질 것이다. 이 문제를 해결하기 위한 방법으로 R-tree 계열의 MBR 기반 검색 시에 증강현실을 위한 효율적인 공간 검색 필터링 기법을 사용한 arR-tree를 제안한다.

4. arR-tree 필터링 알고리즘

이 논문에서 제안하는 arR-tree는 세 가지 판별식으로 구성되어 있다. 질의 영역과 비 단말 노드의

MBR이 서로 교차하는지에 대한 판별식, MBR에 질의 영역이 포함되는지에 대한 판별식, 질의 영역이 MBR을 포함 하는지에 대한 판별식이 있다. 세 가지의 판별식 중 어느 하나라도 참이라면 해당 노드에 대한 검색을 수행하여야 한다. 그렇지 않은 경우 최소 경계 사각형에서 테드스페이스의 경우 이거나 질의 영역을 포함하지 않는 노드이다. 세 가지 판별식은 PostGIS에서 사용하는 GiST 인덱스 중 Consistent를 통하여 구현하였다.

4.1 arR-tree의 consistent 메소드 알고리즘

GiST에서 consistent 메서드는 GiST에서 인덱스의 탐색을 위해 사용되는데 사용자 질의 q와 tree page에 대한 key p가 주어지면, consistent 메서드는 q와 p를 비교하여 올바른 p가 주어진 질의 q에 만족하는지의 여부를 돌려준다. 따라서 본 논문에서 제안하는 검색 알고리즘을 consistent 메소드에 구현하였다.

```

Algorithm consistent(mbr, query )
Input : mbr=비단말 노드의 MBR, query = 질의 영역
Output : mbr과 query가 Intersect 하는지의 여부
{
    queryMbr = getMbr( query );
    if( isMbrIntersect(mbr, queryMbr)
        if( isTriangle(query) )
            if( isTriangleIntersect(nodeMbr, queryPolygon)
                || isTriangleInMbr(nodeMbr, queryPolygon)
                || isMbrInTriangle(nodeMbr, queryPolygon) )
                return true;
            else return false;
        else return true;
    else return false;
}
    
```

그림 5. arR-tree의 consistent 알고리즘

증강현실의 특징이 반영된 질의는 삼각형의 형태를 가지고 있다. 따라서 우리는 삼각형 형태의 질의를 구분하여 이 논문에서 제안하는 알고리즘을 적용시키고 그 외의 질의에 대해서는 알고리즘을 적용하지 않았다.

4.2 Triangle 교차판별 알고리즘

isTriangleIntersect는 검색 영역과 비 단말노드의 MBR 겹침을 검사하여 테드스페이스를 제거하기 위한 선분 교차 판별 알고리즘이다.

그림 6은 isTriangleIntersect 알고리즘을 보이고 있다. 이 알고리즘은 선분이 하나라도 교차되었다면

해당 비 단말 노드는 질의 영역과 겹침이 발생하기 때문에 하위 노드로의 검색이 수행되어야 한다.

```

Algorithm isTriangleIntersect(mbr, query)
Input : mbr=비단말 노드의 MBR, query = 질의 영역
Output : mbr과 query의 선분이 교차하는지의 여부
{
    linePolygon[3] = getLine(query);
    lineMbr[4] = getMbrLine(mbr);
    For( i=0 ; i<4 ; i++ )
        For( j=0 ; j<3 ; j++ )
            if ( isCross(lineMbr[i], linePolygon[j]) )
                return true;
    return false;
}
    
```

그림 6. isTriangleIntersect 알고리즘

그림 7은 선분 교차 판별식을 보여 준다. 두꺼운 검은 선으로 표시된 검색영역은 기존 R-tree 계열에서 MBR을 통한 검색 시에 사용되는 검색 영역이다. 이 검색 영역은 질의 영역을 MBR로 변형한 것이다. 삼각형으로 표시된 질의 영역은 증강현실에서 사용자의 시선 방향에 따른 실제 질의 영역을 나타낸다. 두꺼운 고동색 선으로 표시된 MBR은 색인 구조에서의 비 단말 노드의 MBR을 나타낸다.

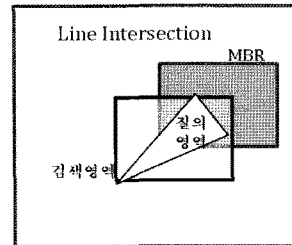


그림 7. 삼각형 교차 판별 알고리즘

isTriangleInMbr 함수는 질의 영역이 비 단말 노드의 MBR에 포함되는지에 대한 판별을 수행한다.

```

Algorithm isTriangleInMbr(mbr, query)
Input : mbr=비단말 노드의 MBR, query = 질의 영역
Output : mbr에 query가 포함되는지의 여부
{
    pointPolygon[3] = getPoint(queryPolygon);
    For( i=0 ; i < 3 ; i++ ) {
        if ( isPointInMBR(pointPolygon[i], mbr) ) {
            return true;
        }
    }
    return false;
}
    
```

그림 8. isTriangleInMbr 알고리즘

그림 9는 비 단말 노드의 MBR에 포함되는 질의 영역 판별식을 보여 준다. 질의 영역의 점이 한 개라도 비 단말 노드의 MBR 안에 포함되어 있다면 비 단말 노드의 하위 노드에 대한 검색이 수행되어야 한다.

사용자가 질의 영역의 지정을 크게 했을 경우나 실내 공간 객체처럼 공간 객체들이 밀집되어 있는 경우에는 질의 영역이 비 단말 노드의 MBR보다 커질 수 있는 상황이 될 수 있다.

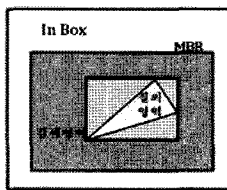


그림 9. MBR의 삼각형 포함 판단

그림 10은 삼각형이 MBR을 포함하는 예를 보이고 있다. 질의 영역에 비단말 노드의 MBR이 포함되는지에 대한 판별식은 다음과 같이 동작한다.

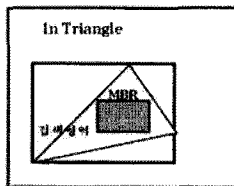


그림 10. 삼각형의 MBR 포함 판단

그림 11은 질의 영역에 비 단말 노드의 MBR이 포함되는 판별식을 보여 준다. 비 단말 노드 MBR의 점이 한 개라도 질의 영역 안에 포함되어 있다면 비 단말 노드의 하위 노드에 대한 검색이 수행되어야 한다.

```

Algorithm isMbrInTriangle(mbr, query)
Input :mbr=비단말 노드의 MBR, query = 질의 영역
Output : query안에 mbr이 포함되는지의 여부
{
    pointMbr[4] = getPoint(mbr);
    For( i=0; i<4; i++ ) {
        if(isPointInTriangle(pointMbr[i], QueryPolygon))
            return true;
    }
    return false;
}
    
```

그림 11. isMbrInTriangle 알고리즘

5. 구현 및 실험

이 절에서는 증강현실을 위한 효율적인 공간검색 알고리즘의 성능을 평가하기 위해 기존 R-tree 계열의 공간색인에서 사용하는 MBR 검색과 본 논문에서 제안하는 알고리즘의 검색 시간을 평가 기준으로 하여 실험을 수행하였다. 실험 결과 제안한 알고리즘의 효율성을 증명하였다.

5.1 실험 환경 및 실험 데이터

공간 객체에 대한 공간색인 및 공간 검색을 수행하기 위하여 공간 데이터베이스로 많이 활용되고 있는 PostGIS를 사용하여 Microsoft Windows XP 환경에서 실험하였다. PostGIS는 공간색인으로 R-tree를 사용하고 있다.

그림 12는 실험에 사용된 데이터를 보여주고 있다. 실험에 사용된 데이터는 전국 polygon 건물 데이터를 가지고 데이터를 구성하였다. 데이터는 약 30만개의 공간 객체를 가지고 있으며 실험 질의에 사용된 서울 지역에는 78900개의 공간 객체가 있다.



그림 12. 실험에 사용된 데이터

5.2 구현 시스템 구조

4절에서 제안한 arR-tree는 단말기에서 접근할 수 있도록 WebServer를 기반으로 하여 구현하였다. 단말기에서는 arR-tree 서버에서 공개하고 있는 Open API를 통해서 질의를 요청하고 결과를 응답 받는다.

arR-tree를 구현하기 위하여 PostGIS의 소스code 이 논문에서 제안하는 검색 알고리즘을 추가하였다. 그림 13은 arR-tree 시스템의 전체 구조를 보여준다.

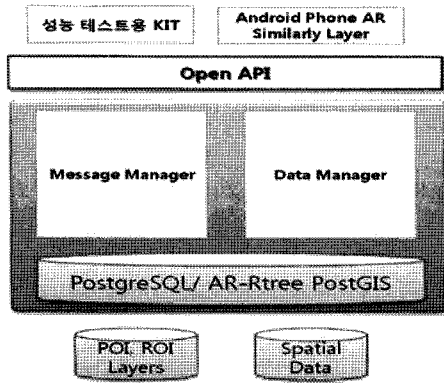


그림 13. arR-tree 시스템 구조도

5.3 실험 결과

실험에 사용된 질의는 3장에서 설명한 질의 유형 별로 준비하였다. 단순형태의 질의는 일반 카메라의 화각인 63°를 가지고 거리 500~1000미터 사이의 무작위 검색 범위를 갖는 질의를 서울 지역에 한하여 생성하였다. 이 질의를 2000개, 4000개, 6000개, 8000개, 10000개씩 생성하여 제안된 알고리즘을 적용한 검색과 알고리즘이 적용되지 않은 검색의 응답 시간을 비교하였다. 또, 화각이 63°이면서 거리가 1000미터인 질의를 생성하여 무작위로 거리가 설정된 질의와 비교하였다.

SightLine 형태의 질의는 화각을 2°로 설정하고 거리 500~1000미터 사이의 무작위 검색 범위를 갖는 질의를 서울 지역에 한하여 생성하였다. 이 질의를 2000개, 4000개, 6000개, 8000개, 10000개씩 생성하여 제안된 알고리즘을 적용한 검색과 알고리즘이 적용되지 않은 검색의 응답 시간을 비교하였다. 또, 화각이 2°이면서 거리가 1000미터인 질의를 생성하여 무작위로 거리가 설정된 질의와 비교하였다.

그림 14의 실험결과는 Radar형태의 질의를 실험하였다. 질의는 서울 지역 안에서 무작위로 선정한 점으로 하여 거리 100미터 안에 있는 공간 객체를 찾는 질의를 구성하였다.

그림 15의 실험결과는 단순형태의 질의 유형을 실험하였다. 실험에 사용된 질의는 500~1000미터 사이의 무작위 거리와 63°의 고정 화각을 설정하였다. 그래프에서 나타나듯이 2000개의 질의에서는 약 3초, 4000개의 질의에서는 약 2초, 6000개의 질의에서는 약 2초, 8000개의 질의에서는 약 13초, 10000개의 질의에서는 약 4초 정도의 검색 속도 차이가

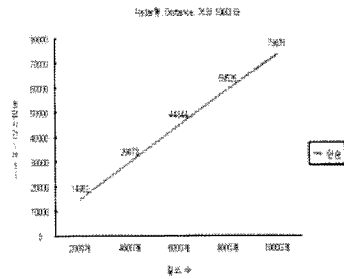


그림 14. Radar형(Distance=100m)

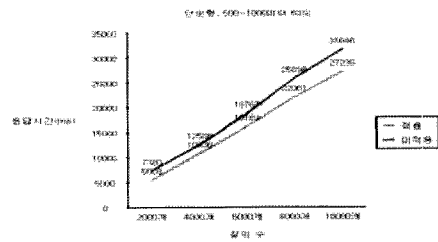


그림 15. 단순형(500~1000m, 63°)

발생하였다.

그림 16의 실험결과는 단순형태의 질의 유형을 실험하였다. 실험에 사용된 질의는 거리 1000미터 사이와 63°의 고정 화각을 설정하였다. 그래프에서 나타나듯이 2000개의 질의에서는 약 2초, 4000개의 질의에서는 약 3초, 6000개의 질의에서는 약 6초, 8000개의 질의에서는 약 8초, 10000개의 질의에서는 약 10초 정도의 검색 속도 차이가 발생하였다.

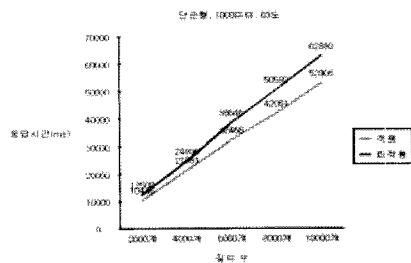


그림 16. 단순형(1000m, 63°)

그림 17의 실험결과는 Sightline 형태의 질의 유형을 실험하였다. 실험에 사용된 질의는 500~1000미터 사이의 무작위 거리와 2°의 고정 화각을 설정하였다.

그래프에서 나타나듯이 2000개의 질의에서는 약

2초, 4000개의 질의에서는 약 5초, 6000개의 질의에서는 약 8초, 8000개의 질의에서는 약 10초, 10000개의 질의에서는 약 13초 정도의 검색 속도 차이가 발생하였다.

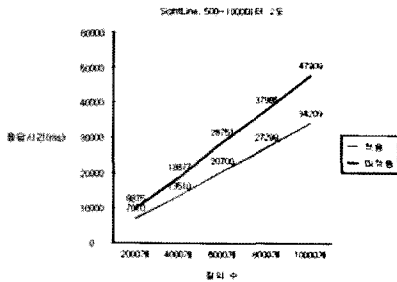


그림 17. Sightline형(500~1000m, 2°)

그림 18의 실험결과는 SightLine 형태의 질의 유형을 실험하였다. 실험에 사용된 질의는 거리 1000미터 사이와 2°의 고정 화각을 설정하였다. 그래프에서 나타나듯이 2000개의 질의에서는 약 7초, 4000개의 질의에서는 약 13초, 6000개의 질의에서는 약 20초, 8000개의 질의에서는 약 26초, 10000개의 질의에서는 약 31초 정도의 검색 속도 차이가 발생하였다.

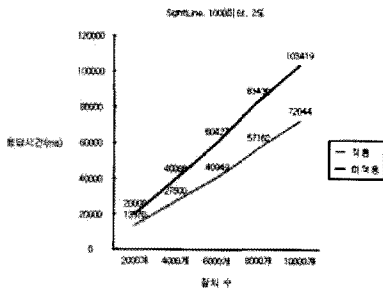


그림 18. Sightline형(1000m, 2°)

6. 결론

본 논문에서는 기존 증강현실에서 나타난 공간 검색의 특징을 이용하여 증강현실 공간 검색의 효율적인 처리를 위한 알고리즘을 제안하였다. 제안된 알고리즘은 증강현실 공간 검색의 특징을 고려하여 기존 최소 경계 사각형 검색에서의 레드 스페이스를 줄임으로써 검색 성능을 향상 시켜 검색 속도를

단축하였다.

제안된 알고리즘에 대한 구현은 공간 데이터베이스로 많이 활용되는 PostGIS를 통해 구현하였다. 실험결과 제안된 알고리즘을 통하여 공간 검색을 수행하는 것이 기존의 공간 검색보다 성능이 효율적임을 확인할 수 있었다. 본 논문에서 제안한 알고리즘은 증강현실을 이용한 LBS의 공간 검색 질의 요구가 많아짐에 따라 실제 유용하게 활용될 수 있다.

향후 증강 현실에서 나타날 수 있는 다양한 질의에 맞춰 검색 속도를 향상시킬 수 있도록 하는 연구가 필요하다.

참 고 문 헌

- [1] R. Azuma, Y. Baillet, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, 2001, "Recent Advances in Augmented Reality," IEEE Computer Graphics and Applications, Vol.21, No.6, pp. 34-47.
- [2] N. Beckmann, H. -P. Kriegel, R. Schneider, and B. Seeger, 1990, "The R*-tree: An efficient and robust access method for points and rectangles," in Proc. of ACM SIGMOD Conf., pp. 322-331.
- [3] C. Faloutsos, T. Sellis and N. Roussopoulos, 1987, "Analysis of object oriented spatial access methods," Proc. ACM SIGMOD Inf. Cont. on Management of Data., pp. 426-439.
- [4] C. Geiger, B. Kleinjohann, C. Reimann, and D. Stichling, 2001, "Mobile AR4ALL," IEEE and ACM International Symposium on Augmented Reality 2001, pp. 181-182.
- [5] D. Greene, 1990, "An Implementation and Performance Analysis of Spatial Data Access," Proc IEEE Data Engineering, pp. 606-615.
- [6] O. Guenther and A. Buchmann, 1990, "Research Issue is Spatial Databases," SIGMOD Record. Vol.19 No.4, pp. 61-64.
- [7] A. Guttman, 1984, "R-trees: A dynamic index structure for spatial searching," in Proc. of ACM SIGMOD Conf., pp. 47-57.
- [8] T. Hollerer, S. Feiner, T. Terauchi, G. Rashid, and D. Hallaway, 1999, "Exploring MARS: Developing

- Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System," IEEE Computers Graphics and Applications, Vol.23, No.6, pp. 779-785.
- [9] Norio Katayama, Shin'ichi Satoh, 1997, "The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries," the 1997 ACM SIGMOD International Conference on Management of Data Proceedings.
- [10] S.J.Oh, W.Lee, Y.Park, and W.Woo, "u-Contents :Realistic and Affective Contents in Ubiquitous Smart Space," Korea MultiMedia Society, Vol.10, No.2, pp. 73-83, 2006.
- [11] J. A. Orenstein, 1989, "Redundancy in spatial databases," Proc. ACM SIGMOD Conf.
- [12] M. Weiser, 1991, "The Computer for the 21st Century," Scientific American, Vol.265, No.3. pp. 94-104.
- [13] Robinson. J.T., 1981, "The K-D-B Tree: A Search Structure for Large Multidimensional Dynamic Indexes," Proc. ACM SIGMOD Conf.
- [14] T. Sellis, N. Roussopoulos, and C. Faloutsos, 1987, "The R+-tree: A dynamic index for multi-dimensional objects," in Proc. of the 13th VLDB Conf., pp. 507-518.
- [15] D. Sonnen, 2005, "Worldwide Spatial Information Management 2005-2009 Forecast and 2004 Vendor Shares," IDC Market Report.
- [16] Y. Suh, Y. Park, C. Shin, H. Yoon, Y. Chang, and W. Woo, 2007, "Context-Aware Mobile AR System for Personalization, Selective Sharing, and Interaction of u-Contents in u-Space," KHCI 2007, pp.598-605.
- [17] M. Wagner and G. Klinker, 2003, "An Architecture for Distributed Spatial Configuration of Context Aware Applications," 2nd International Conference on Mobile and Ubiquitous Multimedia.
- [18] D.A. White and R. Jain, 1996, "Similarity Indexing with the SS-tree," In proc. 12th Intl. Conf. On Data Engineering, New Orleans, pp. 516-523.
- [19] 김덕환, 박호현, 정진완, 1995, "GIS를 위한 공간색인 및 공간 질의 처리 기법," 정보과학회지 제13권 제3호., pp. 48-57.
- [20] 이성호, 민경욱, 김재철, 김주완, 박종현, 2005, "위치 기반 서비스 기술동향," ETRI 전자통신동향분석제20권제3호, pp. 33-42.
- [21] 장병태, 김주완, 김동현, 1997, "증강현실(Augmented Reality) 기술," 정보과학회지, pp. 14-19.
- [22] 홍동표, 우운택, 2008, "모바일 증강 현실 시스템에 대한 연구 동향," 정보과학회지, 제26권 제1호, pp. 88-97.
- [23] 조숙경, 김성희, 김종훈, 2009, "u-GIS 환경에서 플랫폼 독립을 지원하는 플렉스 기반 모바일 지리 정보 서비스 시스템의 설계 및 구현," 한국공간정보시스템학회 논문지, 제11권 제1호, pp. 87-96.
- [24] 심재민, 이용재, 주용완, 남광우, 류근호, 2009, "위치 기반 푸쉬서비스 플랫폼 설계 및 구현," 한국공간정보시스템학회 논문지, 제11권 제4호, pp. 47-55.

논문접수 : 2011.01.21

수정일 : 2011.02.24

심사완료 : 2011.02.26



박 장 유

2006년 군산대학교 컴퓨터정보과학과
이학사

2006년~2007년 포인트아이(주)

2010년 군산대학교 컴퓨터정보공학과
공학석사

관심분야는 LBS, 데이터베이스, 데이터스트림, GIS, 지
오센서 네트워크, GeoVideo



이 성 호

1997년 충북대학교 전자계산학과 이학
사

2000년 충북대학교 전자계산학과 이학
석사

2000년~현재 한국전자통신연구원 용
합기술연구부부 선임연구원

관심분야 데이터베이스, GIS, LBS, GeoWeb



남 광 우

1995년 충북대학교 전자계산학과 이학
사

1997년 충북대학교 전자계산학과 이학
석사

2001년 충북대학교 전자계산학과 이학

박사

2001년~2004년 한국전자통신연구원 텔레매틱스연구단

2004년~현재 군산대학교 컴퓨터정보공학과 부교수

관심분야는 데이터베이스, GIS, LBS 정책 및 기술, 데이
터스트림, 지오센서 네트워크