위치기반 서비스에서 개인 정보 보호를 위한 그리드를 이용한 Cloaking 영역 생성 알고리즘

(An Algorithm for generating Cloaking Region Using Grids for Privacy Protection in Location-Based Services)

엄 정 호* 김 지 희** 장 재 우***
(Jung-Ho Um) (Ji-Hee Kim) (Jae-Woo Chang)

요 약 위치기반 서비스(Location-Based Service) 에서는 위치-기반 질의를 요청하는 사용자가 자신의 정확한 위치 정보를 데이터베이스 서버로 보내기 때문에, 사용자의 개인 정보가 상대방에게 노출될 수 있다. 따라서 사용자가 안전하게 위치기반 서비스를 사용할 수 있기 위해서는 개인 정보 보호 방법이 요구된다. 따라서 본 논문에서는 위치기반 서비스에서 개인정보 보호를 위한 새로운 클로킹(cloaking) 영역 생성 알고리즘을 제안한다. 제안하는 기법은 그리드를 이용하여 사용자가 요구하는 L개의 건물을 탐색한 후, K명의 사용자를 탐색하는 K-anonymity를 수행하여 최소 크기의 클로킹 영역을 생성한다. 이를 위해 그리드 기반의 색인 구조 및 효과적인 가지치기 방법을 사용한다. 마지막으로 성능평가를 통해 본 논문에서 제안하는 클로킹 영역 생성 알고리즘이 클로킹 영역의 크기 측면에서 기존의 연구보다 우수함을 보인다.

키워드 : 개인 정보 보호, 위치기반 서비스, cloaking 영역 생성 알고리즘

Abstract In Location-Based Services (LBSs), users requesting a location-based query send their exact location to a database server and thus the location information of the users can be misused by adversaries. Therefore, a privacy protection method is required for using LBS in a safe way. In this paper, we propose a new cloaking region generation algorithm using grids for privacy protection in LBSs. The proposed algorithm creates a minimum cloaking region by finding L buildings and then performs K-anonymity to search K users. For this, we make use of not only a grid-based index structure, but also an efficient pruning techniques. Finally, we show from a performance analysis that our cloaking region generation algorithm outperforms the existing algorithm in term of the size of cloaking region.

Keywords : Privacy Protection, Location-based Services, Cloaking Region Generation Algorithm

1. 서론

위치기반 서비스(Location-Based Service)[1]에서 사용자가 요청한 위치-기반 질의는 데이터베이스 서버에 전송된다. 즉, 모바일 사용자가 교통 정보, 사람 찾기, 인접한 POI(Point Of Interest) 찾기, 현재 위치의 날씨 정보 등의 서비스를 요청하면, 모바일 기기는 사용자가 요청한 서비스 내용과 사용자의 위치정보를 무선 네트워크를 통해 데이터베이스 서버로 전송한다. 그러나 이러한 서비스는 서비스를 요청하는 사용자가 자신의 정확한 위치 정보를 데이터베이스 서버에 보내기 때문에, 사용자의 개인 정보가 노출될 수 있는 취약성을 지닌다. 이 때문에, 상대방(adversary)은 서비스 이용자들이 어떤 장소에 자주 방문하는지, 또한 이러한 방문이 어떤 시간대에 주로 이루어지는지를 파악하여, 생활스타일, 질병 정보, 종교 등의 개인 정보를 획득할 수 있다. 실제로 국외의 경우, 위치 기반 서비스를 이용한 스토킹 피해 사례가 빈번히 발생하고 있다[2, 3]. 따라서 모바일 사용자의 안전한 위치기반 서비스 사용을 위해서는 개인 정보 보호 방법이 요구된다.

이러한 개인 정보 보호 방법의 대표적인 연구로는 클로

† 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2009-0059417)
† 본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임

* 전북대학교 전기전자컴퓨터공학부 컴퓨터공학전공 박사과정, jhum@chonbuk.ac.kr
** 롯데정보통신 마트 IS팀 사원, jiheekim@lotte.net
*** 전북대학교 전기전자컴퓨터공학부 컴퓨터공학전공 교수, jwchang@chonbuk.ac.kr(교신저자)

논문접수 : 2009.02.16 수정일 : 1차 2009.04.28 / 2차 2009.06.12 심사완료 : 2009.06.18

킹(cloaking) 기법에 대한 연구가 있다[4-11]. 클로킹 기법이란, 사용자가 데이터베이스 서버에 질의를 요청할 때, 사용자의 좌표정보 대신 K-anonymity를 만족하는 최소 넓이의 질의 영역(이하 클로킹 영역)을 설정하는 것을 말한다. 여기서 K-anonymity는 질의를 요청한 사용자 및 인접한 K-1명의 다른 사용자의 위치 정보를 클로킹 영역에 포함하는 것이다. 그러나 K-anonymity만을 만족하는 클로킹 기법은, 한 건물 내에 클로킹 영역이 설정될 경우 쉽게 사용자의 위치 추정이 가능한 문제점이 존재한다. 따라서, K-anonymity를 만족하는 동시에 L-diversity를 만족하는 클로킹 영역을 설정하여, 질의를 요청한 사용자의 위치 정보 노출 확률을 감소시켜야 한다. 여기서 L-diversity는 클로킹 영역 내에 L개의 다른 장소(우편물이 배달될 수 있는 주소지 기준)를 포함하는 것을 의미한다. 현재 cloaking 기법에 L-diversity를 적용시킨 기존 연구로는 B. Bamba 와 L. Liu 의 연구 [4] 및 김지희 등의 연구[5]가 존재한다. B. Bamba 와 L. Liu 의 연구는 그리드 기반의 cloaking 영역 생성 방식을 최초로 사용하였지만, L-diversity의 특성을 충분히 고려하지 못하여 cloaking 영역의 넓이가 증가하는 문제점을 가지고 있다. 한편 김지희등의 연구는 B. Bamba 와 L. Liu 의 연구에서 cloaking 영역의 넓이가 증가하는 문제점을 해결하기 위해 제안되었다. 그러나 2개의 R-tree 를 사용함에 따라, cloaking 영역 생성 시간이 매우 느려지는 단점을 지니고 있다. 따라서 본 논문에서는 위치기반 서비스에서 사용자의 위치 정보를 보호하기 위하여 그리드를 이용한 클로킹 영역생성 알고리즘을 제안한다. 이 기법은 먼저 사용자가 요구하는 L개의 건물을 탐색하는 L-diversity를 수행한 뒤, K명의 사용자를 탐색하는 K-anonymity를 수행하여 최소 크기를 가지는 클로킹 영역을 설정한다. 이를 위해, 그리드 기반의 색인 구조를 사용하며, 아울러 클로킹 영역의 빠른 설정을 위해 효과적인 가지치기 방법을 사용한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 K-anonymity 및 L-diversity를 고려한 기존 클로킹 기법들을 소개한다. 제 3장에서는 기존 연구의 문제점을 개선한 클로킹 영역 생성 알고리즘을 제안한다. 제 4장에서는 기존 연구와 본 논문에서 제안한 알고리즘과의 성능비교를 수행한다. 마지막으로 제 5장에서는 결론 및 향후 연구를 제시한다.

2. 관련 연구

최근 사용자의 위치 정보를 보호하기 위하여 연구된 대부분의 클로킹 기법들은 질의를 요청한 사용자의 좌표 정보를 K-anonymity를 만족하면서 최소 크기의 넓이를 가지는 클로킹 영역으로 변환하는 모델을 제안하고 있다. M. Gruteser et al. 연구[6]는 Quad-tree를 기반으로 K-anonymity를 이용한 클로킹 기법을 처음 제안하였다. Z. Xiao et al. 연구[7]는 공간 영역 탐색 방법을 사용하여 클로킹 영역을 설정하며, 이를 위해 서버에 존재하는 사용자들의 이웃(neighbor)관계를 동적 그래프로 유지하는 clique를 사용한다. C. Y. Chow et al. 연구[8]는 질의를 요청한 모바일 사용자가 자신의 통신 범위 내에 있는 K-1 명의 다른 사용자들을 찾아 클로킹 영역을 설정한다. M. F. Mokbel et al. 연구[9]는 그리드 기반 피라미드 구조(grid-based pyramid structure)를 사용하여 클로킹 영역을 설정하는 기법을 제안하였다. G. Ghinita et al. 연구[10]는 힐버트 곡선(Hilbert Curve)을 사용하여 구성한 계층적인 분산 B+-트리를 통해 클로킹 영역을 설정한다. 이 아름등의 연구[11]는 힐버트 곡선과 Chord[12] 프로토콜을 이용하여 클로킹 영역을 설정하는 기법을 제안하였다. 그러나 K-anonymity만을 만족하는 클로킹 기법은 클로킹 영역이 병원 등과 같이 한 장소 내에 설정될 경우, 사용자의 위치가 쉽게 추정될 수 있는 문제점을 가지고 있다. 이와 같은 문제점을 해결하기 위하여 클로킹 영역 내에 L개의 다른 장소를 포함시키는 L-diversity가 고려되었고, 이는 K-anonymity와는 보완적으로 사용된다.

한편, K-anonymity 및 L-diversity를 클로킹 기법에 사용하여 위치노출 확률을 감소시키는 연구는 두 가지가 존재한다. 첫째, 조지아 공과대학의 B. Bamba 와 L. Liu는 K-anonymity와 L-diversity를 동시에 고려한 클로킹 기법인 Privacy Grid [4]를 제안하였다. 이 기법은 그리드 셀을 이용하여 클로킹 영역을 설정하기 때문에 클로킹 설정 속도가 매우 빠른 장점을 가진다. 이를 위해 전체 영역을 같은 크기의 그리드 셀로 나누고, 각 그리드 셀 내에 위치한 사용자 수 및 건물 개수를 저장하는 Cell Object Count Map을 사용한다. 그림 1은 Privacy Grid의 색인 구조인 Cell Object Count Map을 나타낸다.

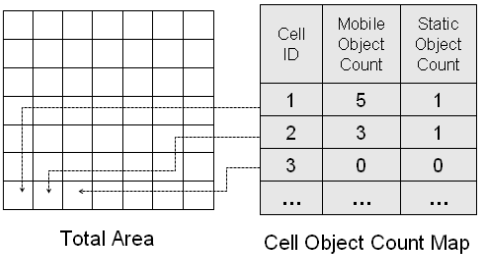

그림 1. Privacy Grid의 색인구조

Privacy Grid는 셀의 확장 방법에 따라 Quad-Grid, Bottom-Up, Top-down, Hybrid의 네 가지 방법으로 분류된다. Quad-Grid 방법은 전체 영역을 4등분의 그리드 셀로 분할하면서 클로킹 영역을 설정한다. Bottom-up 방법은 질의를 요청한 사용자가 위치한 셀을 기준으로 셀을 확장하면서 클로킹 영역을 설정하며, Top-down 방법은 사용자가 정의한 최대 클로킹 영역 크기에서 셀을 감소시키면서 클로킹 영역을 설정한다. 아울러, Hybrid 방법은 사용자가 요구한 K-anonymity와 최대 클로킹 영역의 크기에 따라 Bottom-up 과 Top-down 기법 중에서 하나를 선택하여 수행한다. 이 중 가장 최소 크기의 클로킹 영역을 설정하는 것은 Bottom-up 방법이다. 그림 2는 Bottom-

up 방법의 클로킹 영역 설정 과정을 나타내며, 각 그리드 셀에 표시된 숫자는 셀 내에 위치한 사용자 수를 나타낸다. 예를 들어 K=20 일 경우, 먼저 그림 2의 (a)와 같이 질의점이 위치한 셀을 중심으로 상하좌우 방향으로 셀을 탐색하여 가장 큰 값을 가지는 셀을 선택한다. 만약, 셀 내에 위치한 사용자 수가 같을 경우, 상하좌우 방향 중에서 교대로 선택한다. 검색된 사용자의 수 K′가 12로 요구한 K를 만족하지 않으므로 확장을 계속한다. 다음으로, 정사각형 모양의 클로킹 영역을 설정하기 위해 그림 2의 (b)와 같이 양 측면 셀을 두 개씩 탐색하여 사용자 수를 확인한다. 사용자 수가 많은 오른쪽 셀 선택을 통해, 결과적으로 그림 2의 (c)와 같이 K′=21을 만족하는 클로킹 영역이 설정된다.

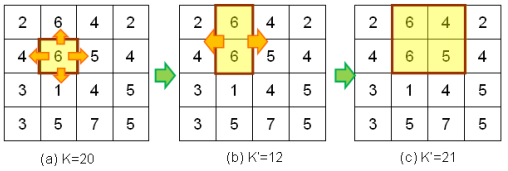

그림 2. Privacy Grid의 Bottom-up 클로킹 기법

둘째, 전북대학교의 김지희등의 연구[5]는 위치기반 서비스를 이용하는 사용자들의 정보 보호를 위한 클로킹 영역 생성 알고리즘을 제안하였다. 제안된 클로킹 영역 생성 알고리즘은 L-diversity를 먼저 고려하여 임시 클로킹 영역을 설정한 후, K-anonymity를 고려하는 클로킹 기법을 제안하였다. 한편, 건물의 위치 및 사용자의 위치 정보를 위하여 2개의 R*-트리를 사용한다. 이를 통해, 한 건물의 중복된 계산이 없는 정확한 L-diversity 알고리즘을 수행할 수 있으며, 사용자의 위치 정보를 통해 K-anonymity 알고리즘을 수행한다. 예를 들어 L=3 일 때, 질의를 요청한 사용자(user)가 건물 외부에 위치한 경우, R*-트리를 검색하여 q와 인접한 3개의 건물 L1, L2, L3를 검색한다(그림 3.a). 다음으로 각 건물 내에 서 질의 요청 사용자와 가장 인접한 사용자를 한 명씩 검색한 뒤, 이를 포함하는 최소경계사각형(Minimum Bounding Rectangle)을 임시 클로킹 영역으로 설정한다(그림 3.b). 아울러 임시 클로킹 영역 내에 포함된 사용자의 수를 확인하여, K-anonymity 알고리즘으로의 확장 또는 클로킹 알고리즘의 종료 여부를 선택한다. 부족한 사용자를 찾기 위한 K-anonymity 알고리즘은 임시 클로킹 영역을 기준으로 동서남북 각 방향에 대하여 인접한 사용자들을 탐색하여 최종적으로 K와 L을 만족하는 클로킹 영역을 설정한다.

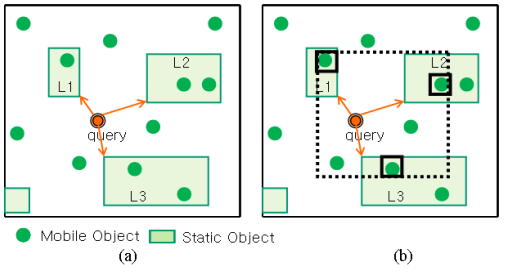

그림 3. 김지희등의 연구

3. 그리드를 이용한 클로킹 영역 생성 알고리즘

중앙 집중(Centralized) 방식의 클로킹 기법은 클로킹 영역을 설정하는 주체가 anonymizer 이다. anonymizer 란, 모바일 사용자와 LBS 서버 중간에 존재하는 신뢰할 수 있는 서버이다. 사용자는 anonymizer로 사용자의 위치 정보가 포함된 질의를 전송하고, anonymizer는 사용자의 위치 좌표를 숨기는 클로킹 영역을 설정한다. 설정된 클로킹 영역은 anonymizer에 의해 LBS 서버로 전송되며, LBS 서버는 클로킹 영역을 바탕으로 요청된 질의를 처리한다. 아울러 anonymizer는 LBS 서버에서 전송된 질의 수행 결과를 저장된 사용자의 위치 정보와 건물 정보를 바탕으로 필터링 한 뒤, 정확한 결과를 사용자에게 전송한다. 그림 4는 anonymizer를 사용하는 중앙 집중 방식을 나타낸다.

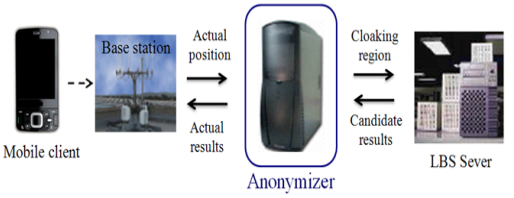

그림 4. 중앙 집중 방식

이러한 환경에서 L-diversity 를 고려하는 2가지 방법, 즉 Privacy Grid 및 김지희등의 연구가 수행되었다. Privacy Grid 는 L-diversity 특성을 충분히 고려하지 못하기 때문에 두 가지 문제점이 존재한다. 첫째, 하나의 건물을 여러 그리드 셀에서 중복적으로 계산하기 때문에, 설정된 클로킹 영역이 사용자가 요구하는 실제 L개의 건물을 만족하지 못한다. 예를 들어, 그림 5와 같이 건물 S3가 그리드 셀 C1, C2, C4, C5 에 위치했을 경우, 4개 셀 모두에서 객체 개수가 하나씩 증가된다. 만약 어떤 사용자가 건물 4개를 포함하는 클로킹 영역을 요구했을 때, 그 결과로 C1, C2, C4, C5 의 셀이 선택되었다면 이는 실제로는 하나의 건물이 포함된 클로킹 영역을 설정한 것이 된다. 따라서 L-diversity의 특성을 충분히 고려한 클로킹 영역을 설정하기 위해서는, 클로킹 영역 내에서 하나의 건물을 중복 계산하지 않아야 한다. 둘째, Privacy Grid는 클로킹 영역 내에 단순히 건물만 포함시키고, 건물 내에 위치한 사용자에 대해 고려하지 않아 질의를 요청한 사용자의 위치 노출 확률을 높인다. 이러한 문제점은 상대방(adversary)이 클로킹 영역을 요구한 사용자의 정확한 위치는 모르지만, 모든 사용자의 위치를 알고 있을 경우 발

생할 수 있다. 예를 들어 그림 5에서 클로킹 영역에 포함된 건물 S3 내에는 실제 어떠한 사용자도 위치하지 않는다. 이 경우 그 영역 내에 건물 하나가 포함되더라도, 클로킹 영역을 요구한 사용자(q)가 건물을 제외한 점선에 위치한다는 것을 쉽게 추측할 수 있다. 이는 L-diversity의 특성을 충분히 고려하지 못한 것으로, 한명 이상의 사용자가 위치한 건물을 클로킹 영역에 포함시키는 방법이 필요하다.

한편, 김지희 등이 제안한 R-tree based Cloaking 기법(이하 RTBC) 은 cloaking 영역 생성 시간이 현저히 느린 문제점을 지니고 있다. 즉, RTBC는 L-diversity를 만족하는 임시 클로킹 영역이 생성되면, K-anonymity를 만족하기 위한 나머지 사용자를 찾기 위해 R-tree를 사용하여 영역과 인접한 사용자를 탐색한다. 이러한 탐색은 각 방향마다 이루어지기 때문에, 인접 사용자를 찾기 위해 8번의 R-tree 탐색이 필요하다. 아울러, RTBC 기법에서는 클로킹 영역을 생성한 후, 찾아야 될 사용자가 많을 경우, 사용자 조합을 위해 선택되는 경우의 수가 증가되어 클로킹 영역 생성 성능을 저하시킨다.

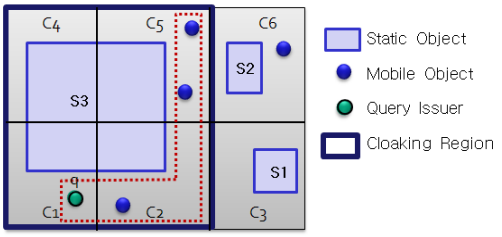

그림 5. Privacy Grid의 클로킹 기법

따라서 본 논문에서는 위치 노출 확률을 감소시켜주는 L-diversity 및 K-anonymity의 특성을 효과적으로 고려한 그리드 기반의 효율적인 클로킹 영역 생성 알고리즘을 제안한다. 한편 K-anonymity와 L-diversity를 클로킹 알고리즘에 적용하는 방법은 크게 세 가지가 존재한다. 첫째, K-anonymity와 L-diversity를 동시에 고려하는 방법이 있다. 이는 Privacy Grid에서 사용된 방법으로, 색인 구조에 저장된 그리드 셀 내의 사용자 수와 건물 수를 동시에 확인하면서 그리드 셀을 확장하여 클로킹 영역을 설정하는 방식이다. 그러나 그리드를 이용할 경우 K와 L을 동시에 만족하지 않을 때 그리드 셀 묶음 단위로 클로킹 영역이 확장되기 때문에 넓이가 증가하는 문제점이 있다. 둘째, K-anonymity를 먼저 고려한 후 L-diversity를 고려하는 방법이 있다. 그러나 일반적으로 클로킹 영역 내에 포함될 건물 수 L보다 사용자 수 K가 많기 때문에, 이 방법은 K명의 사용자를 먼저 찾는데 많은 시간이 소요된다. 마지막으로 L-diversity를 먼저 고려한 후 K-anonymity를 고려하는 방법이 있다. 이 방법은, 클로킹 영역이 L-diversity만 만족하더라도 그 안에 여러 개의 건물이 포함되어 다수의 사용자가 위치할 수 있기 때문에, K-anonymity 를 만족하여 클로킹 영역 생성 시간을 단축시킬 수 있다. 따라서 제안하는 클로킹 영역 생성 알고리즘은 먼저 사용자가 요구하는 L개의 건물을 탐색하는 L-diversity를 수행한 뒤, K명의 사용자를 탐색하는 K-anonymity를 수행한다. 이에 따라 제안하는 클로킹 영역 생성 알고리즘은 L-diversity를 먼저 고려하여 임시 클로킹 영역을 설정하는 L-diversity를 만족하는 클로킹 영역 설정 알고리즘 및 K-anonymity를 고려하여 영역을 확장하는 K-anonymity 만족하는 영역 확장 알고리즘으로 구성된다.

3.1 L-diversity를 만족하는 영역 설정 알고리즘

기존 연구인 Privacy Grid는 건물 내에 사용자가 위치한 상황을 고려하지 않고 인접한 건물을 선택하지만, 김지희 등의 연구는 건물 내에 사용자가 위치한 상황을 고려하고 건물을 포함하는 클로킹 영역을 설정한다. 따라서 L-diversity 알고리즘을 건물 내에 사용자가 위치한 경우와 그렇지 않은 경우로 나누어 제안한다.

3.1.1 건물만 고려한 L-diversity 알고리즘

본 절에서는 건물 내 사용자에 대해 고려하지 않고, 건물의 포함 여부에 따라 L-diversity를 만족하는 클로킹 영역 설정 알고리즘을 제안한다. 먼저, 수행단계 1은 사용자가 요구하는 건물 수가 2개 이상일 때, L-diversity를 수행하기 위하여 질의를 요청한 사용자 q와 인접한 L개의 건물을 선택하는 방법을 기술한다.

수행단계 1. 사용자가 요구한 L개의 건물 선택 (L≥2)

질의를 요청한 사용자가 위치한 그리드 셀을 중심으로, 사방으로 한 셀씩 확장된 영역에 위치한 셀을 탐색한다. 탐색 중 가장 인접한 건물이 포함된 셀을 선택하며, 요구한 건물 수를 만족할 때까지 영역을 확장하며 셀 탐색을 반복한다.

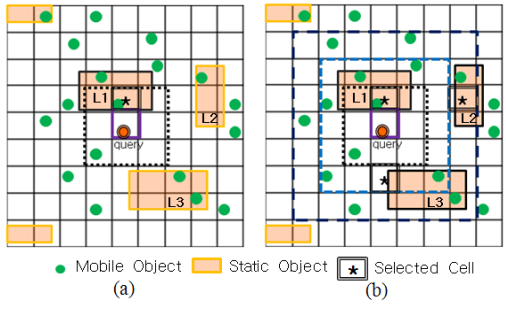

그림 6. 사용자가 요구한 L개의 건물 선택 방법

예를 들어 L=3 일 때, 그림 6의 (a)와 같이 먼저 사용자(query)가 위치한 셀을 찾고, 그것을 중심으로 한 셀씩 확장된 영역(점선 사각형)을 탐색하여 인접한 건물 L1이 포

함된 셀(*)을 선택한다. 계속해서 2개의 다른 건물을 추가적으로 찾기 위해, 그림 6의 (b)와 같이 두 번의 확장을 더 수행하여 질의를 요청한 사용자와 가장 가까운 건물이 위치한 두 개의 셀을 선택한다. 이와 같은 확장 방법을 사용하여 주변 셀을 탐색하기 때문에, 가장 인접한 그리드 셀을 빠짐없이 탐색할 수 있다.

사용자가 요구한 L개의 건물을 선택하면, 그것을 포함하는 클로킹 영역은 2개 이상의 건물을 포함하는 영역이 된다. 따라서 선택된 L개의 건물을 포함하는 임시 클로킹 영역을 설정하여, 그 영역 내에 위치한 사용자들의 수를 확인한다. 수행단계 2는 임시 클로킹 영역을 설정하는 방법을 기술한다.

수행단계 2. 임시 클로킹 영역 설정

셀 탐색을 통해 선택된 그리드 셀을 포함하는 최소경계 사각형을 임시 클로킹 영역으로 설정한다.

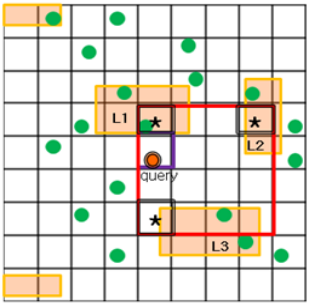

그림 7. 임시 클로킹 영역 설정

예를 들어 L=3 일 때, 그림 7은 그림 6에서 선택된 세 개의 셀을 포함하는 최소경계사각형을 임시 클로킹 영역으로 설정한 모습을 보여준다. 이로써 임시 클로킹 영역은 L-diversity를 만족하는 그리드 셀로 이루어진 최소 크기의 사각형이 된다.

수행단계 3. 임시 클로킹 영역 내의 사용자 수에 따른 클로킹 영역 확장

임시 클로킹 영역 내에 위치한 사용자의 수 K′가 클로킹 영역 내에 포함되기를 원하는 사용자의 수 K보다 많거나 같을 경우, 임시 클로킹 영역을 결과 클로킹 영역으로 설정한다. 만약, K보다 작을 경우 부족한 (K-K′)명의 사용자를 찾기 위한 확장 알고리즘을 수행한다.

1) K′≥K 인 경우, 클로킹 알고리즘 종료
   : 임시 클로킹 영역을 결과로 반환
2) K′< K 인 경우, 클로킹 영역 확장
   : K-anonymity를 만족하는 클로킹 영역 확장 알고리즘 수행

예를 들어, 그림 7에서 K=10 일 경우, 셀 탐색을 통해 임시 클로킹 영역 내에 3명의 사용자가 확인된다. 이것은 K′가 K보다 7명이 적은 경우로, 이때 부족한 7명의 사용자를 찾기 위한 확장 알고리즘을 수행한다.

만약 임시 클로킹 영역 내의 사용자의 수가 K를 만족하지 못하고, 임시 클로킹 영역 내의 가로 및 세로의 셀 개수가 2 이하일 경우, 확장 알고리즘이 비효율적으로 수행된다. 수행단계 4는 이러한 조건에서 효율적인 클로킹 영역 확장을 위한 알고리즘을 기술한다.

수행단계 4. 임시 클로킹 영역의 가로 및 세로의 셀 개수가 2 이하일 경우

셀 개수가 2이하인 경우, 임시 클로킹 영역의 개수를 양쪽으로 α개씩 (0<α≤가로 및 세로 최대 셀 개수) 증가시켜 영역의 범위를 확장한 뒤 위치한 사용자의 탐색을 재수행한다.

수행단계 4를 통해, 건물만 고려한 L-diversity 알고리즘은 그림 8과 같다. 먼저 클로킹 영역을 요청하는 사용자의 위치정보, 클로킹 영역 내에 포함되는 건물 수 L, 사용자 수 K를 입력받는다. 다음으로 질의를 요청한 사용자가 위치한 셀을 검색하여, 그것을 중심으로 한 셀씩 확장한 영역을 탐색한다. L개의 인접한 건물이 포함된 셀을 탐색한 뒤, 그들과 질의 요청자가 위치한 셀을 포함하는 최소경계 사각형을 임시 클로킹 영역으로 설정한다. 마지막으로 임시 클로킹 영역 내에 위치한 사용자 수가 요구된 K보다 많거나 같을 경우, 그리드 셀로 이루어진 임시 클로킹 영역을 결과 클로킹 영역으로 반환한다. 그러나 그 수가 요구된 K보다 적을 경우, K를 만족시키기 위해 확장 알고리즘을 수행한다.

```
Grid-Based Cloaking Algorithm (<qx, qy>, K, L)
Input:<qx,qy> //질의 요청 사용자의 좌표 정보,
      L //cloaking 영역 내에 포함시킬 건물의 수,
      K //cloaking 영역 내에 포함시킬 사용자 수
Output: L과 K를 만족하는 cloaking 영역

 1. cid=get_cell(qx,qy)
 2. clist=LnnSearch(cid)
 3. tempRegion=createTemporalCloakingRegion(clist)
 4. K'=checkUsers(tempRegion)
 5. if(K'≥K) return tempRegion
 6. else if(NumberOfCellsinRow(tempRegion)≤2
      and NumberOfCellsinCol(tempRegion)≤2)
 7. tempCloak=Resize(tempRegion,α)
 8. goto step 4
 9. else
10.      resRegion=
         CloakingRegionExpansion(tempRegion,K-K')
11.          return resRegion
End Algorithm
```

그림 8. 건물만 고려한 L-diversity 알고리즘

그림 8은 수행단계 1~4를 기반으로 L개의 건물을 포함하는 클로킹 영역 생성 알고리즘을 보여준다. 첫째, 사용자가 위치기반 서비스를 이용하기 위하여 자신의 위치 보

호 요청을 하면, 알고리즘은 사용자가 위치한 셀을 검색한다 (line 1). 둘째, 사용자와 인접한 L개의 건물을 포함하는 셀들(clist)을 선택한다(line 2). 셋째, 선택된 셀들과 질의가 발생한 셀을 포함하는 최소경계사각형을 임시 클로킹 영역(tempCloak)으로 설정한다(line 3). 넷째, 임시 클로킹 영역에 위치한 사용자 수 K′를 검사한다(line 4). 마지막으로, K′가 K 이상이면 알고리즘은 종료되고, K′가 K 이상이면 임시 클로킹 영역의 크기를 계산하여 가로 및 세로의 셀 개수가 2 이하일 경우 양쪽으로 α개를 확장시켜 사용자 수 K′를 재검사한다(line 5-8). 만일 셀의 개수가 2 이상이면, 3.1.2절에서 기술하는 Cloaking 영역 확장 알고리즘을 호출한다(line 9-10).

3.1.2 사용자를 포함하는 건물만 고려한 L-diversity 알고리즘

본 절에서는 상대방(adversary)이 모든 사용자의 위치를 알고 있을 경우, 위치노출 확률을 감소시키기 위해 사용자를 포함하는 건물만을 L-diversity를 만족하는 것으로 간주하는 클로킹 영역 설정 알고리즘을 기술한다. 따라서 수행단계 1-(a)는 L-diversity를 수행하기 위해 질의를 요청한 사용자 q와 인접한 L개의 건물을 선택하는 방법을 기술한다.

수행단계 1-(a). 사용자가 요구한 L개의 건물 선택(L≥2)

질의를 요청한 사용자가 위치한 그리드 셀에서 사방으로 셀씩 확장된 영역의 셀을 탐색한다. 탐색 중 사용자가 위치한 건물이 포함된 셀 중 질의셀과 가장 인접한 셀을 선택한다. 요구한 건물 수를 만족할 때까지 영역을 확장하며 셀 탐색을 반복한다.

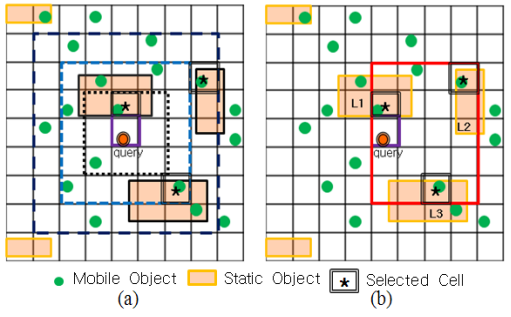

그림 9. 사용자를 포함하는 건물만 고려한 L-diversity 알고리즘

예를 들어 그림 9의 (a)는 L=3 일 때, 질의셀 부터 한 셀씩 추가하는 확장 방법을 통해 셀을 탐색하며, 사용자가 포함된 인접한 3개의 건물을 선택한 모습을 나타낸다. 아울러, 그림 9의 (b)와 같이 질의셀과 선택된 셀들을 포함하는 최소경계사각형을 임시 클로킹 영역으로 설정한다. 한편, 수행단계 2, 3, 4는 건물만 고려한 L-diversity 알고리즘의 수행단계와 동일하다.

3.2 K-anonymity를 만족하는 클로킹 영역 확장 알고리즘

본 절에서는 L-diversity를 만족하는 임시 클로킹 영역이 K-anonymity를 만족하지 않을 경우, 부족한 K-K′(이하 R로 명명)명의 사용자를 찾는 확장 알고리즘을 위한 수행 단계를 제시한다. 먼저, 부족한 R명의 사용자를 찾기 위하여 임시 클로킹 영역을 기준으로 동, 서, 남, 북 각 방향에 대해 R명 이상의 인접한 사용자들을 포함하는 그리드 셀을 탐색하여 가지치기(pruning)에 사용될 경계(threshold) 값을 구한다. 수행단계 5는 최대 확장 가능한 클로킹 영역의 탐색 방법을 기술한다.

수행단계 5. 각 방향으로 최대 확장 가능한 클로킹 영역 탐색

임시 클로킹 영역을 기준으로 동, 서, 남, 북 각 방향에 대해 R명 이상의 사용자를 포함시키면서, 최대로 확장 가능한 그리드 셀을 탐색한다. 셀을 임시 클로킹 영역의 가로 또는 세로의 크기만큼 증가시키면서 탐색을 수행한다. 셀 묶음 단위의 확장 영역이 R명 이상의 사용자를 포함할 때 탐색을 멈춘다.

수행단계 5와 같이 각 방향으로 최대 확장 가능한 영역을 탐색하여 경계 값을 설정한다. 아울러 이와 같이 경계 값을 설정하는 이유는 다음과 같다. 일반적으로 사각형의 넓이가 가장 작게 증가하는 경우는 그림 10의 (a)와 같이 한쪽 방향으로만 넓이를 확장시킨 경우이다. 그 다음은 (b)와 같이 한쪽의 대각선 방향으로 늘어나는 경우이고, (c)와 같이 세면이 늘어나는 경우, 마지막으로 (d)와 같이 네 면이 동시에 늘어나는 경우의 순서로 확장 넓이가 증가하게 된다. 따라서 처음에 각 방향으로 확장 영역을 탐색하여 설정된 경계 값이 최소 크기의 클로킹 영역에 가장 근사하기 때문에, 수행 단계 5와 같은 방법을 사용한다. 예를 들어, 그림 11은 수행단계 5에 따라 임시 클로킹 영역을 기준으로 각 방향에 대해, R명 이상을 포함하는 최대 확장 가능한 영역을 나타낸다.

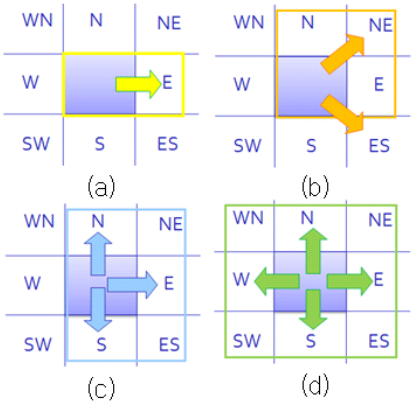

그림 10. 일반적인 넓이의 확장

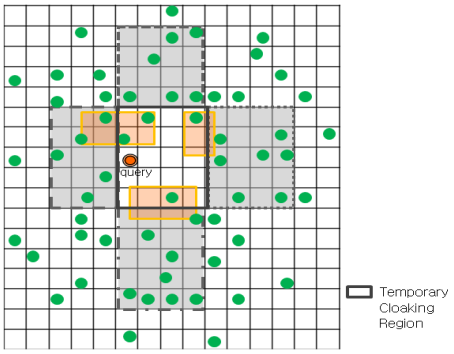

그림 11. 각 방향에 대해 5명이상의 사용자를 포함하는 영역 탐색

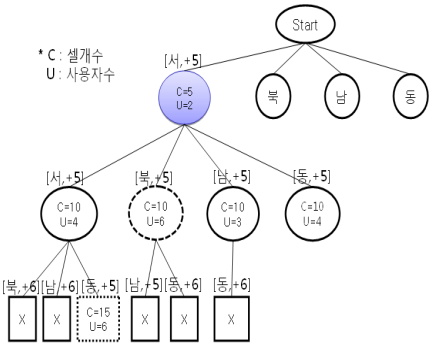

그림 12. 그림 11의 서쪽 방향을 기준으로 영역을 확장 가능한 모든 경우

수행단계 6. 영역 확장을 위한 경계 값 설정

각 방향으로 확장 가능한 영역이 구해지면 각 영역을 이루고 있는 셀 개수 C와 각 영역 내에 포함된 사용자 수 U를 구한다. 경계 값의 우선순위는 사용자 수가 R을 만족할 때, 셀 개수가 적을수록 높다. 아울러 셀 개수가 같은 경우, 사용자의 수가 많을수록 우선순위가 높다. 우선순위가 제일 높은 값을 초기 경계 값으로 설정하여, 확장 시 가치치기에 이용한다.

수행 단계 6의 우선순위에서 사용자 수가 R을 만족할 때 셀 개수가 적은 것을 선택하는 것은, 클로킹 영역에 대한 질의 처리 후 결과 후보 집합이 불필요하게 많아지는 것을 방지하기 위함이다. 아울러, 셀 개수가 같을 때 사용자의 수가 많은 것이 우선순위가 높은 이유는, 클로킹 영역 내에 가능한 많은 사용자를 포함시켜 개인 정보의 노출 확률을 낮추기 위함이다. 예를 들어, 그림 11에서 각 방향의 경계 값은 <동쪽 C=20, U=8>, <서쪽 C=15, U=5>, <남쪽 C=20, U=8>, <북쪽의 C=16, U=6>이 된다. 이에 수행단계 6을 이용하여 우선순위를 정렬하게 되면 서->북->남->동 방향이 된다. 따라서 초기 한계 값은 셀 개수 15개, 사용자 수 5명으로 설정되며, 이것을 바탕으로 다음의 확장 단계에서 가지치기를 수행하여 최적의 영역을 빠르게 탐색할 수 있다.

수행단계 7. 최소 크기를 가지는 그리드 셀조합을 구하는 방법

먼저, 우선순위가 제일 높은 쪽을 기준으로 각 방향의 셀 묶음 단위로 영역을 확장시킨다. 한쪽 방향에서 선택하는 셀 묶음을 한 개 증가시키고, 대신 중복되지 않는 다른 쪽 방향의 셀 묶음을 하나씩 증가시켜 순열을 구한다. 기준 방향으로는 최대 셀 묶음 단위보다 하나 적은 영역까지 확장이 가능하며, 각 방향으로 선택할 수 있는 모든 경우를 수행한다.

예를 들어, 그림 12는 한계 값으로 설정된 서쪽 방향을 기준으로 영역을 확장하며, 확장 가능한 모든 경우를 나타낸다. 서쪽방향의 묶음 셀의 최대 개수는 3개로, 최대 3개의 다른 방향을 가진 순열을 생성할 수 있다. 각 노드 안에 값은 각 상태에서의 셀 개수와 사용자 수를 나타내며, 각 노드 위의 [서, +5] 값은 서쪽으로 한 묶음 확장했을 경우 5개의 셀이 늘어나는 것을 나타낸다. 노드를 증가시킬 때마다 늘어나는 값이 다른 이유는 클로킹 영역을 직사각형으로 설정하면서 각 방향들 사이의 공간에 위치한 그리드 셀을 포함시키기 때문이다. 트리를 깊이 우선 탐색하며 확장이 진행되고, 최초에 지정된 한계 값인 셀 개수 15와 사용자 수 5를 바탕으로 가지치기가 수행된다. 점선으로 표시된 노드들은 한계 값이 갱신됨을 나타낸다. 한계 값을 이용한 트리 탐색을 통해 많은 셀 조합들의 계산이 가지치기 되는 모습을 확인할 수 있다. 그림 13은 그림 12의 트리에서 마지막으로 한계 값이 갱신된 경우의 영역을 나타낸다. 한 방향의 확장이 종료된 후, 다른 방향도 이와 같은 방식으로 수행하며 중복된 경우는 허용하지 않는다.

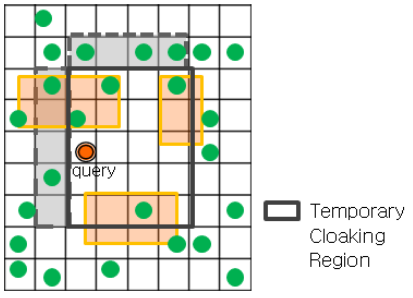

그림 13. 서쪽 방향을 기준으로 서쪽 한 묶음, 북쪽 한 묶음 영역을 확장한 경우

수행단계 5, 6, 7을 바탕으로 설계한 K-anonymity를 만족하는 클로킹 영역 확장 알고리즘은 그림 14와 같다. 첫째, 임시 클로킹 영역과 부족한 사용자 수 R을 기반으로 각 방향에 최대 확장 가능 영역을 탐색하여 가지치기에 사용될 초기 경계 값을 설정한다(line 1). 둘째, 각 방향의 그리드 셀 묶음을 이용하여 선택될 수 있는 모든 경우를 비교한다.(line 2) 마지막으로, 최소한의 확장 넓이를 가지는 최적의 그리드 셀을 계산한다(line 3).

```
CloakingRegionExpansion(TempCloak, R)
Input: R = K-K'
Output: 최소한의 넓이를 가지는 cloaking 영역
1.  upper=SetUpperBound(R, TempCloak)
2.  res=Combination(R, TempArea, upper)
3.  return res
End Algorithm
```

그림 14. K-anonymity를 만족하는 cloaking 영역 확장 알고리즘

4. 구현 및 성능평가

본 절에서는 건물만을 고려한 L-diversity를 이용한 제안하는 클로킹 알고리즘과 사용자를 포함한 건물만을 고려한 L-diversity를 이용한 제안하는 클로킹 알고리즘을 구현하여 성능 평가를 수행한다. 이하 각각의 알고리즘을 성능평가 상에서 GBC(Grid-Based 클로킹 region generation algorithm), GBC-O(Grid-Based 클로킹 region generation algorithm considering Object)라 표기한다. 성능 비교 대상은 Privacy Grid와 김지희등의 연구이다. 김지희등.의 연구는 RTBC(R*-Tree Based 클로킹 method)라 표기한다. 한편, Privacy Grid는 성능이 가장 우수한 Bottom-up 방법을 사용한다. GBC는 건물만을 고려한 L-diversity 알고리즘을 사용하므로 Privacy Grid와 비교하며, GBC-O는 사용자를 포함한 건물만을 고려한 L-diversity 알고리즘을 사용하므로 RTBC와 비교한다. 성능 평가 항목은 클로킹 영역 크기, 클로킹 영역 설정 시간, 클로킹 영역에 따른 질의처리 결과 후보 집합 개수, 클로킹 영역에 따른 질의처리 수행 시간이다.

표 1. 실험 환경

항목	성 능
CPU	Intel Core2 Quad CPU Q6600 2.40GHz
Memory	2GB
OS	Windows XP professional
Compiler	Microsoft Visual Studio.NET 2003

K-anonymity 및 L-diversity를 지원하는 클로킹 기법의 성능 평가 환경은 표 1과 같다. 아울러, 성능평가에 사용된 데이터는 GSTD(Generate Spatio Temporal Data) 알고리즘[10]을 이용하였으며, 클로킹 영역 설정을 위하여 1,000개의 건물과 10,000명의 포인트를 생성하였으며, 질의처리를 수행하기 위하여 500,000개의 POI(Point Of Interest)를 생성하였다. GSTD 알고리즘은 가로 및 세로의 범위가 1인 사각형 내에 객체를 생성하기 때문에, 그리드 크기는 가로 및 세로 크기를 0.01로 고정하여 성능 평가를 수행한다. 아울러 임시 클로킹 영역을 확장할 때 사용하는 그리드 셀의 개수 α는 실험을 통해 1개로 설정한다. 성능평가 항목은 K를 20으로 고정시키고, L을 2, 4, 6, 8, 10로 증가시켜가면서 L에 따른 성능을 측정한다.

4.1 클로킹 영역 크기 성능평가

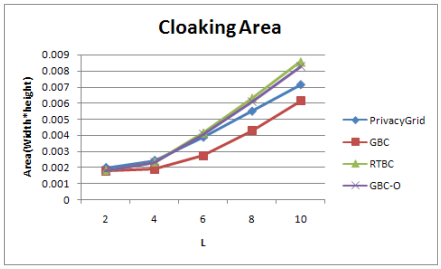

그림 15. 클로킹 영역 크기 비교

그림 15는 본 논문에서 제안한 알고리즘인 GBC, GBC-O와 기존 연구인 Privacy Grid와 RTBC의 클로킹 영역 크기를 비교한 것이다. GBC와 Privacy Grid 알고리즘의 경우, 두 기법 모두 L이 커질수록 클로킹 영역 크기가 증가하는 모습을 보인다. 결과적으로 GBC가 Privacy Grid 기법보다 작은 크기의 클로킹 영역을 생성함을 알 수 있다. 이는 GBC의 경우 효율적인 가지치기 방법을 사용하여 클로킹 영역을 설정하기 때문이다. 아울러 Privacy Grid는 K와 L을 만족하는 영역을 찾기 위해, 셀 개수의 배수씩 영역을 확장하기 때문에 클로킹 영역이 커진다. 특히, L이 6일 때, GBC의 평균 클로킹 영역 크기는 0.0027로, Privacy Gird의 0.0039보다 약 1.5배 작게 나타난다. 한편, GBC-O와 RTBC 알고리즘은 주변 영역을 탐색하기 위해 서로 다른 색인 구조를 사용하지만, 질의 요청자가 위치한 셀을 중심으로 인접한 건물을 찾는 L-diversity 알고리즘을 사용하기 때문에 클로킹 영역의 크기가 거의 유사하게 나타난다. 그러나 L이 증가할수록 RTBC 기법의 클로킹 영역 크기가 약 0.0003 크기만큼 증가한다. 이는 먼저 R*-트리를 통해 인접한 건물을 탐색하고 건물 내에 위치한 사용자를 고려하기 때문에, 질의 셀과 인접하지 않는 사용자를 탐색하는 경우가 존재하기 때문이다.

4.2 클로킹 영역 설정 시간 성능평가

그림 16은 제안한 기법인 GBC, GBC-O와 기존 연구 RTBC와 Privacy Grid의 클로킹 영역 설정 시간을 나타낸다. 제안하는 알고리즘인 GBC와 GBC-O 알고리즘 모두 그리드를 사용하여 클로킹 영역을 설정하기 때문에 빠른 속도를 보인며, L의 값이 증가할수록 영역 생성 시간이 감소함을 알 수 있다. 이는 제안하는 알고리즘의 경우 L의 값이 증가할수록 임시 클로킹 영역 내에 많은 건물이 포함되므로, K′≥K를 만족하여 확장 알고리즘을 수행하지 않고 알고리즘이 종료되기 때문이다. GBC와 GBC-O 알고리즘은 전체적으로 0.0005초 이내의 빠른 영역 설정

시간을 나타내지만, Privacy Grid 보다 최적의 영역을 탐색하기 때문에 시간이 더 소요된다. 그러나 클로킹 시간은 전체시간의 3% 정도밖에 차지하지 않기 때문에, 전체시간 측면에서 볼 때 이는 매우 미세한 차이를 가진다. 한편 RTBC 의 경우, 제안하는 GBC와 GBC-O 기법이나 Privacy Grid 에 비해 전체적으로 약 10배 정도 성능이 저하함을 알 수 있다. 그 이유는 RTBC는 임시 cloaking 영역이 생성되면, 나머지 사용자를 찾기 위해 R-tree를 사용하여 영역과 인접한 사용자를 탐색하며, 이것은 각 방향마다 인접 사용자를 찾기 위해 8번의 R-tree 탐색이 필요하기 때문이다. 아울러, RTBC 기법에서는 cloaking 영역을 생성한 후, 찾아야 될 사용자가 많을 경우, 사용자 조합을 위해 선택되는 경우의 수가 증가되어 cloaking 영역 생성 성능을 저하시킨다.

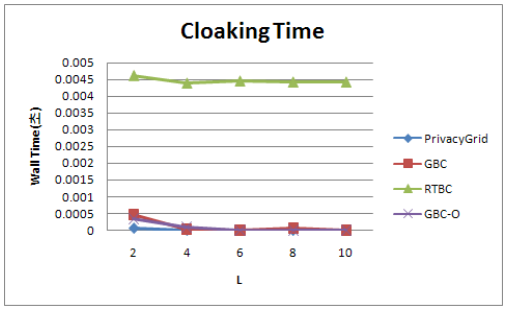

그림 16. 클로킹 영역 설정 시간 비교

4.3 질의 처리 수행 시간 성능평가

설정된 클로킹 영역 크기로 질의 처리 수행 시간에 미치는 영향을 측정하기 위해, 설정된 클로킹 영역에 대한 범위 질의(Range Search)를 수행하였다. 범위의 크기는 클로킹 영역의 가로 및 세로 길이를 10%씩 확장시켜 수행하였다.

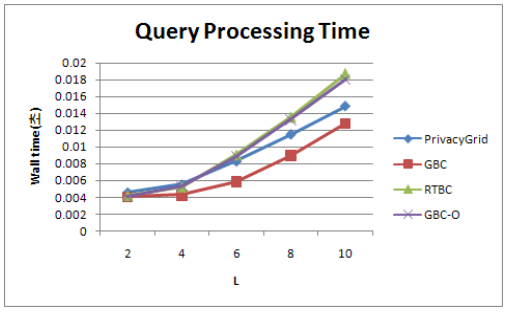

그림 17. 질의 처리 수행 시간 비교

그림 17은 제안한 기법인 GBC, GBC-O와 기존 연구 RTBC와 Privacy Grid의 클로킹 영역에 대한 범위 질의 처리 수행 시간을 나타낸다. 모든 알고리즘은 L이 증가할수록 클로킹 영역의 크기가 커지기 때문에, 시간이 증가함을 알 수 있다. GBC 알고리즘은 Privacy Grid보다 평균적으로 작은 크기의 클로킹 영역을 설정하기 때문에, 질의 처리 수행 시간이 적게 소요됨을 알 수 있다. 특히, L이 6인 경우, GBC의 평균 질의처리 시간이 0.005초로 Privacy의 0.008초보다 약 1.5배 빠르게 나타난다. 한편, 제안한 GBC-O와 RTBC의 클로킹 영역에 대한 질의 처리 수행 시간은 생성되는 클로킹 영역의 차이가 미세하기 때문에 질의 처리 시간 또한 유사한 성능을 보인다.

4.4 질의 처리 결과 집합 비교

그림 18은 제안한 기법인 GBC, GBC-O와 기존 연구 RTBC와 Privacy Grid의 질의 처리 결과 집합의 개수를 비교한 것이다. Privacy Grid의 경우 GBC보다 크기가 큰 클로킹 영역을 설정하기 때문에, 범위 질의 결과보다 많은 후보 집합이 검색됨을 알 수 있다. 실험 환경과 같이 POI의 밀도가 높은 도심지의 경우, 전송된 클로킹 영역의 작은 차이가 중요한 변수로 작용함을 알 수 있다. 특히, 후보 집합이 많을수록 질의 서버에서 anonymizer로 전송하기 위한 통신비용이 증가하고, anonymizer에서 필터링을 위한 오버헤드가 증가하기 때문에, K와 L을 만족하는 최소 크기의 클로킹 영역을 설정하는 것이 중요하다. 아울러 제안한 GBC-O와 RTBC의 질의 처리 결과 집합의 개수는 두 알고리즘 모두 유사한 크기의 클로킹 영역을 설정하기 때문에, 결과 집합 개수가 유사함을 알 수 있다. L이 8이상인 경우, RTBC의 클로킹 영역 크기의 증가와 더불어 결과 집합 개수가 약간 상승함을 알 수 있다.

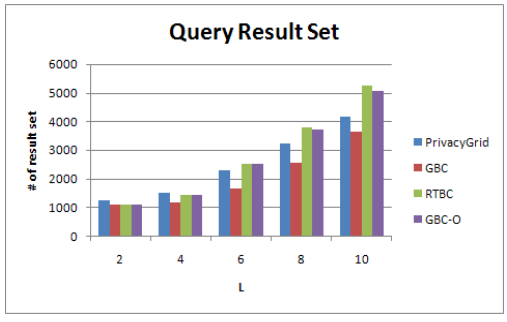

그림 18. 질의 처리 결과 집합 비교

4.5 전체 수행시간 성능평가

전체 수행시간은 anonymizer에서 사용자의 위치를 바탕으로 클로킹 영역을 설정하는 시간 및 생성된 클로킹 영역으로 질의를 처리한 시간을 합한 것으로, 사용자 질의에 대한 응답시간을 나타낸다. 그림 19는 전체 수행 시간의 성능평가 결과를 보여준다. GBC, GBC-O, Privacy Grid 알고리즘은 클로킹 영역 생성 시간이 매우 작아, 전체 수행 시간에서 질의 처리 시간이 큰 비중을 차지함을 알 수 있다. GBC 알고리즘은 클로킹 영역 생성 시간이 L이 4 이하일 때 Privacy Grid보다 느리지만, 질의 처리 수행 시간과 함께 고려했을 때 작은 크기의 클로킹 영역

을 설정한 GBC가 빠른 속도를 보임을 알 수 있다. 아울러, RTBC는 전체 수행 시간 측면에서 가장 저하된 성능을 보이고 있는데, 이는 다른 알고리즘에 비해 cloaking 영역 생성 시간이 매우 크기 때문이다.

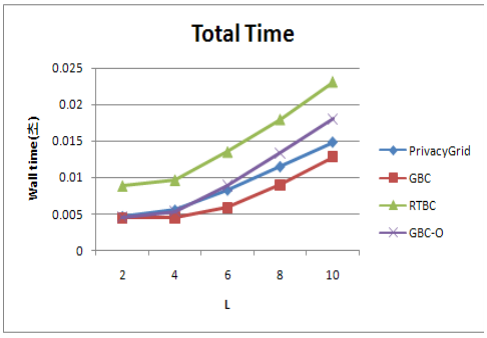

그림 19. 전체 수행 시간 비교

5. 결론

위치기반 서비스(Location-Based Service) 에서는 위치-기반 질의를 요청하는 사용자가 자신의 정확한 위치 정보를 데이터베이스 서버로 보내기 때문에, 사용자의 개인 정보가 상대방에게 노출될 수 있다. 따라서 사용자가 안전하게 위치기반 서비스를 사용할 수 있기 위해서는 개인 정보 보호 방법이 요구된다.

이를 위해 본 논문에서는 그리드를 이용한 효율적인 클로킹 영역생성 알고리즘을 제안하였다. 제안한 알고리즘은 기존의 Privacy Grid의 한 건물의 중복 계산 및 사용자가 포함된 건물을 고려하지 않는 문제점을 해결하였고, 아울러 RTBC 기법에서 R*-tree 색인 구조를 사용하여 영역을 계산함으로써 발생하는 클로킹 속도 저하문제를 해결하였다. 제안한 기법의 알고리즘은 먼저 사용자가 요구하는 L개의 건물을 탐색하는 L-diversity를 수행한 뒤, K명의 사용자를 탐색하는 K-anonymity를 통해 최소 크기를 가지는 클로킹 영역을 생성한다. 한편, 정확한 성능 비교를 위해 L-diversity를 건물만 고려한 방법과 사용자가 포함된 건물만 고려한 방법으로 나누어 제안하였다. 이를 위해, 그리드 기반 색인구조를 사용하며, 최소 크기의 클로킹 영역 설정 알고리즘을 위한 효과적인 가지치기 방법을 사용하였다. 기존 클로킹 기법과의 성능비교를 통해, 제안한 알고리즘이 K-anonymity와 L-diversity를 만족하는 최소 크기의 클로킹 영역을 설정하며, 빠른 질의 처리 시간을 달성함을 보였다. 아울러, 제안하는 알고리즘은 사용자의 위치를 보호하면서 사용자가 요구하는 서비스를 빠르게 제공할 수 있기 때문에 실제 텔레매틱스, 키오스크 등의 위치 기반 서비스에 적용 가능하다.

향후 연구로는 분산 환경에서 L-diversity 및 K-anonymity를 고려한 클로킹 기법을 연구하는 것이다.

참 고 문 헌

[1] 이준석, 김서균, "위치기반서비스(LBS)의 기술 동향 및 국내외 산업 동향 분석," 정보통신연구진흥원 계간 제 5권 제 2호(통권 16호), 2003.

[2] http://www.abc15.com/content/news/northerna-rizona/stoy.aspx?content_id=1cd715e3-76a0-4d15-81d7-f4aacdaae8f7

[3] http://www.hdrjapan.com/index.php?option=com_myblog&show=Accused-stalker-uses-hidden-GPS-to-track-woman.html&Itemid=67

[4] B. Bamba and L. Liu, "PRIVACYGRID: Supporting Anonymous Location Queries in Mobile Environments,", Research report in National Technical Information Service, 2007.

[5] 김지희, 이아름, 김용기, 엄정호, 장재우, "위치기반 서비스에서 개인 정보 보호를 위한 K-anonymity 및 L-diversity를 지원하는 Cloaking 기법," 한국 공간정보시스템 학회 논문지, 제10권, 제4호, 2008, pp. 1-10.

[6] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking," In Proc. of the International Conference on Mobile Systems, Applications and Services, 2003, pp. 31 - 42.

[7] Z. Xiao, X. Meng and J. Xu, "Quality Aware Privacy Protection for Location-based Services," In Proc. of Database Systems for Advanced Applications, vol.4443, April 2007, pp. 434-446.

[8] C. Y. Chow, M. F. Mokbel, and X. Liu. A, "Peer-to-Peer Spatial Cloaking Algorithm for Anonymous Location-based Services," In Proc. of the ACM International Symposium on Advances in Geographic Information Systems, November 2006, pp. 171 - 178.

[9] M. F. Mokbel, C. Chow, and W. Aref, "The New Casper : Query Processing for Location Services without Compromising Privacy," In Proc. of the International Conference on Very Large Data Bases, September 2006, pp. 763 - 774.

[10] G. Ghinita, P. Kalnis and S. Skiadopoulos, "PRIVE : Anonymous Location-Based Queries in Distributed Mobile Systems," In Proc. of World Wide Web, May 2007, pp. 237-246.

[11] 이아름, 엄정호, 장재우" 분산 그리드 환경에서 힐버트 커브를 이용한 효율적인 Cloaking 영역 설정 기법"한국 공간정보시스템 학회 논문지, 제 10권, 제 4호, 2009, pp.115-127.

[12] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek and H. Balakrishnan, "Chord: A Scalable Peer-to-peer Lookup Service for Internet

Application," In Proc. of IEEE/ACM TON, Vol.11 No.1, 2003, pp.17-32.

[12] Yannis Theodoridis, Jefferson R. O. Silva, and Mario A. Nascimento, "On the Generation of Spatiotemporal Datasets," 6th Int'l Symposium on Large Spatial Databases (SSD), 1999.

엄 정 호

2004년 전북대학교 컴퓨터공학과(공학사)
2004년~2006년 전북대학교 컴퓨터공학과 (공학석사)
2006년~현재 전북대학교 전기전자컴퓨터공학부 컴퓨터공학전공 박사과정
관심분야 : 공간 데이터베이스, 공간 색인 구조, GIS, 위치정보보안

김 지 희

2008년 전북대학교 컴퓨터공학과(공학사)
2008년~2009년 전북대학교 컴퓨터공학과 석사과정
2009년~현재 롯데 정보 통신 마트 IS팀 사원
관심분야 : 공간 데이터베이스, 질의처리, 위치 보안을 위한 클로킹

장 재 우

1984년 서울대학교 전자계산기공학과 (공학사)
1986년 한국과학기술원 전산학과 (공학석사)
1991년 한국과학기술원 전산학과 (공학박사)
1996년~1997년 Univ. of Minnesota, Visiting Scholar
2003년~2004년 Penn State Univ., Visiting Scholar.
1991년~현재 전북대학교 전기전자컴퓨터공학부 컴퓨터공학전공 교수
관심분야 : 공간 네트워크 데이터베이스, 하부저장구조, 센서네크워크