
복합질의의 효율적 수행을 위한 궤적질의 필터링 기법

반재훈* · 김종민*

A Filtering Method of Trajectory Query for Efficient Process of Combined Query

ChaeHoon Ban* · JongMin Kim*

요 약

영역 질의와 궤적 질의로 구성된 복합질의는 특정 영역에 위치하는 이동체의 궤적을 검색하는 질의이다. 궤적질의는 선행 궤적을 찾기 위해 점질의를 계속 반복하기 때문에 복합질의의 성능을 결정짓는 중요한 요소이다. 이 경우에 점질의는 인덱스의 노드를 중복해서 방문하기 때문에 질의 처리의 수행속도가 매우 느려진다.

이 논문은 영역 질의에 우수한 성능을 가진 R-tree 기반의 이동체 색인을 대상으로 궤적 질의를 효율적으로 처리함으로써 복합 질의의 성능을 향상시키기 위한 방법을 제시한다. 기존 연구의 문제점인 노드의 중복 방문을 방지하기 위해 최소 공통 탐색 선분을 정의하여 단일 방향으로만 탐색하며 궤적을 예측하고 예측된 선분을 이용하여 연속된 궤적을 동시에 탐색하는 예측기반 여과 방법을 제안한다.

ABSTRACT

The combined query which consists of the region and trajectory query finds trajectories of moving objects which locate in a certain region. The trajectory query is very informant factor to determine query performance because it processes a point query continuously to find predecessors. This results in bad performance due to revisiting nodes in an index.

This paper suggests an efficient method for the combined query based on the 3-dimensional R-tree which has good performance of the region query. The basic idea is that we define the least common search line which enables to search single path and a filtering method based on prediction without revisiting nodes

키워드

이동체, 3차원 R-tree, 복합질의, 궤적, moving objects, 3D R-tree, combined query, trajectory

I. 서론

무선 통신 서비스의 핵심 응용인 위치 기반 서비스는 이용자의 현재 위치를 기반으로 주변 정보를 입수하거나 타인의 위치를 파악해 제공하는 등의 다양한 부가 서비스를 일컫는다. 위치 기반 서비스의 핵심 기술은 연속적으로 이동하는 이동체를 고려한 위치 기반 질의의 효율적인

처리로서, 이동체의 위치 정보를 관리하고 빠른 검색을 제공하기 위해 이동체 데이터베이스를 이용한다.

위치 기반 서비스를 위한 이동체 데이터베이스의 질의는 영역 질의, 궤적 질의 및 복합 질의로 분류된다. 영역 질의는 3차원 시공간 원도우에 속하는 이동체를 검색하는 질의이며 궤적 질의는 특정한 이동체의 궤적을 검색하는 질의이다. 복합 질의는 영역 질의와 궤적 질의가

복합된 질의로서, 인구 밀집 지역의 교통량을 최적화하는 문제 등 특정 영역에 위치하는 이동체의 궤적을 검색하여 분석하는 이동 데이터베이스에서 필수적이다. 질의의 효율적인 처리를 위해 색인에 관한 연구가 진행되었는데, R-tree 기반의 색인들이 영역 질의에 우수한 성능을 가지며, 이동체의 궤적을 보존한 TB-tree 색인이 궤적 질의에서 우수한 성능을 가진다[1][2][3].

이 논문은 영역 질의에 우수한 성능을 가진 R-tree 기반의 이동체 색인을 대상으로 궤적 질의를 효율적으로 처리함으로써 복합 질의의 성능을 향상시키기 위한 방법을 제시한다. 먼저 기존 연구에서 제시한 복합 질의 처리 방법의 문제점인 노드의 중복 방문 문제를 분류하고 설명한다. 문제 해결을 위해 최소 공통 탐색 선분을 정의하여 단일 방향으로만 탐색하며 연결성과 근접성을 고려해 연속 궤적을 탐색하는 예측기반 여과 방법을 제안한다.

이 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고 3장에서는 문제 정의와 기존 연구의 문제점을 제시하며 복합 질의 처리 방법의 기본 아이디어를 소개한다. 4장에서는 단일 방향 탐색을 위하여 최소 공통 탐색 선분을 정의하고 5장에서는 예측기반 여과 방법을 사용하여 궤적을 탐색하는 방법을 제안한다. 마지막으로 6장에서는 결론 및 향후 연구를 기술한다.

II. 관련연구

이동체의 과거 데이터를 검색하기 위한 질의는 영역 질의와 궤적 질의로 분류되며 질의에 따라 적합한 색인이 연구되었다. 영역 질의를 위한 색인은 이동체의 과거 시간뿐만 아니라 공간에 대한 영역을 검색하는 것으로 R-tree를 기반으로 한다. 3D R-tree[1]은 2차원 공간에 시간 차원을 추가하여 데이터를 3차원으로 표현한 색인이다. 이 색인은 3차원 영역 질의에 효율적이지만 궤적 질의를 시공간 조인으로 처리해야 하는 문제점을 가진다.

궤적 질의를 위한 색인은 이동체의 과거 궤적을 효율적으로 검색하기 위한 색인으로서, STR-tree[2]와 TB-tree[3]이 대표적이다. STR-tree는 효율적인 궤적 질의 처리를 목적으로 R-tree를 확장한 것이다. R-tree의 삽입과 분할 알고리즘을 수정하여 단말 노드의 사장 공간 증가와 궤적이 보호되지 않은 문제를 해결한다. 즉, 보존

매개변수(preservation parameter)를 사용하여 같은 객체의 궤적을 근접한 페이지에 저장하도록 유도하여 부분적으로 궤적을 보존한다.

TB-tree는 STR-tree보다 극단적으로 궤적을 보존하는 색인이다. 하나의 단말 노드는 오직 동일한 객체에 속하는 궤적들만 저장할 수 있다. 즉, 공간적으로 근접한 궤적들이라 할지라도 같은 객체의 궤적이 아니면 서로 다른 단말 노드에 저장된다. 따라서 중복이 증가하고 공간 구별성이 감소하여 영역 질의의 성능이 나쁘다. 특히, 이동체의 개수가 증가하거나 영역 질의의 일종인 타임슬라이스 질의를 수행하는 경우에는 기존 색인에 비해 매우 높은 검색 비용을 가진다.

영역 질의와 궤적 질의 두 가지 모두의 효율적인 처리를 위한 색인인 CR-tree[6]는 R-tree와 이동체별 연결 노드를 결합한 형태의 색인이다. 영역 질의 처리를 위해 R-tree를 사용하며, 궤적을 보존한 연결 노드를 사용하여 궤적 질의를 처리한다. 영역 질의와 궤적 질의에 좋은 성능을 가지나 연결 노드를 유지해야 하는 단점이 있다.

III. 문제정의

3.1 복합 질의

복합 질의는 영역 질의와 궤적 질의가 복합된 질의이다. 예를 들어 “오전 7:00시에서 8:00시 사이에 시청 앞을 지나간 차량의 다음 1시간 내의 이동 경로를 구하라”라는 질의에서 “오전 7:00시에서 8:00시 사이에 시청 앞”은 영역 질의이며 “다음 1시간 내의 이동 경로”는 궤적 질의이다.

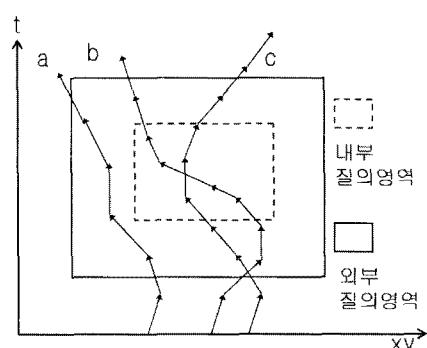


그림 1. 복합 질의
Fig. 1 Combined Query

그림 1은 복합 질의의 예로서, 객체 a, b, c의 궤적들 중에서 내부 질의영역에 포함되는 객체 b, c를 대상으로(영역 질의) 외부 질의영역에 포함된 모든 궤적들을 찾는 것이다(궤적 질의). 이 논문은 복합 질의를 아래와 같이 정의한다.

정의1: 복합질의

IR 를 내부질의영역, OR 를 외부질의영역, L_{ij} 를 객체 i 의 j 번째 궤적이라고 할 때, 복합 질의 CQ 는 $CQ(IR, OR) = \{i | \exists L_{ij} \cap IR \neq \emptyset\}$ 일 때

$$CQ(IR, OR) = \{L_{ij} | \forall L_{ij} \cap OR \neq \emptyset\} \text{ where } i \in OID_{LR}$$

복합 질의의 성능을 향상시키기 위해서는 영역 질의와 궤적 질의를 효율적으로 처리해야 한다. 이 논문은 영역 질의에 우수한 성능을 가진 R-tree 기반의 이동체색인을 대상으로 궤적 질의를 효율적으로 처리함으로써 복합 질의의 성능을 향상시키기 위한 방법을 제시한다.

3.2 기존 연구의 복합 질의 처리 방법

관련 연구 [3]은 R-tree 기반의 색인에서 3 단계의 알고리즘으로 구성된 복합 질의 처리 방법을 제시하였다. 1 단계는 `CombinedSearch()`로서, 그림 1의 내부 질의영역에 포함되는 모든 궤적들을 탐색하는 단계이다. 이것은 그림 2-①과 같이 색인에서 내부 질의영역으로 영역 질의를 수행한다.

2 단계는 `CombinedSearch()`에서 탐색된 모든 궤적들을 대상으로 호출되는 `DeterminTrajectory()`로서, 외부 질의영역에 포함된 연속 궤적들을 찾는 단계이다. 이것은 그림 2-②와 같이 탐색된 궤적이 위치한 단말 노드에서 연속된 다음 궤적을 탐색한다.

3 단계는 `FindConnectSegment()`로서, 2 단계 수행 중에 탐색하고 있는 단말 노드에 더 이상 연속 궤적이 존재하지 않는 경우에 호출된다. 이것은 그림 2-③과 같이 연속 궤적을 찾기 위해 색인의 루트부터 매번 점 질의를 수행한다.

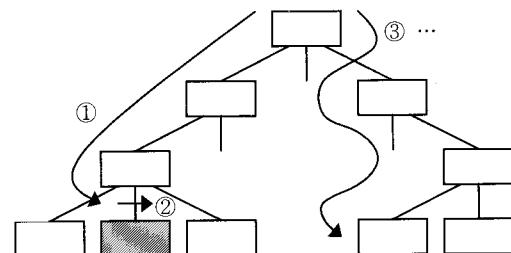


그림 2. R-tree에서의 복합 질의 처리 방법
Fig. 2 Combined Query Processing in the R-trees

3.3 기존 연구의 문제점 및 해결 방법

기존 연구 [3]에서 제안한 복합 질의 처리 방법의 문제점은 노드를 중복 방문하는 것이다. 노드를 중복 방문하는 원인은 3가지로 분류된다. 첫째, 내부 질의영역의 모든 궤적들을 대상으로 전/후방향으로 탐색하며 둘째, 연속 궤적을 점 질의로 탐색하기 때문이다.

3.3.1 내부 질의영역 궤적들의 전/후방향 탐색

기존 연구는 내부 질의영역과 교차하는 모든 궤적들을 대상으로 전/후방향으로 탐색하기 때문에 노드를 중복 방문한다. 그림 3과 같이 내부 질의영역과 교차하는 궤적들이 a_1, a_2, a_3, a_6, a_7 을 탐색한 후에 그들을 대상으로 나머지 궤적들을 탐색한다. 따라서 a_1 의 전방향(화살표방향)으로 궤적을 탐색하기 위하여 노드 LN_1, LN_2, LN_3 를 방문한 후에 a_2 의 전방향으로 다시 궤적을 탐색하기 때문에 방문한 노드들을 재방문한다. 또한 탐색은 전방향 후에 후방향(화살표반대방향)으로 수행되므로 a_7 의 후방향 탐색 시 이미 a_1, a_2 등의 전방향으로 탐색한 궤적들을 포함하는 노드들을 중복 방문하게 된다.

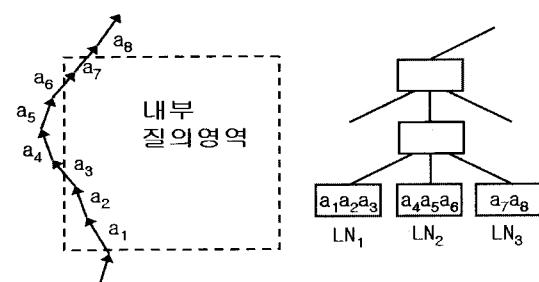


그림 3. 노드의 중복 방문 문제
Fig. 3 Problem of Revisiting Nodes

이 논문에서는 내부 질의영역에 걸치는 궤적들을 대상으로 단일 방향으로만 탐색하여 위의 문제를 해결한다. 그림 4와 같이 내부 질의영역과 교차하는 궤적 a_1, a_2, a_3, a_4 에 대하여 전/후방향으로 궤적을 탐색하지 않고 내부 질의영역에 걸치는(cross) a_1 은 전방향으로만, a_4 는 후방향으로만 궤적들을 탐색하여 노드의 중복 방문을 방지한다.

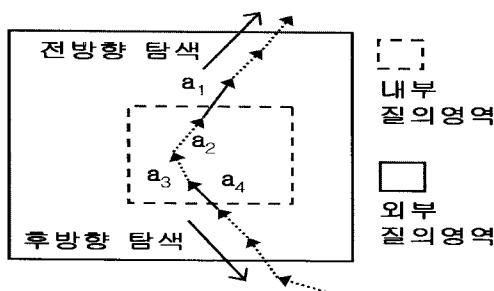


그림 4. 기본아이디어 - 중복 방문 방지
Fig. 4 Basic Idea - Prevention of Revisiting

3.3.2 연속 궤적의 점질의 탐색

기존 연구는 연속 궤적을 점 질의로 탐색하기 때문에 노드를 중복 방문한다. 그림 5와 같이 연속되는 다음 궤적이 현재 탐색된 단말 노드에 존재하지 않으면 색인의 루트부터 점 질의를 수행한다. 즉, 그림 5와 같이 궤적을 탐색하기 위하여 탐색 시작 선분부터 점질의를 수행하여 궤적을 탐색하는데 이를 위해 방문한 노드는 (N_1, N_2, N_6), (N_1, N_2, N_5, N_6), (N_1, N_2, N_4, N_5), (N_1, N_2, N_4, N_3, N_8), (N_1, N_3, N_8)의 순서로 총 19개의 노드를 방문하게 된다. 이 경우 N_1, N_2, N_3 은 부모노드로서 점질의 수행을 위해 중복 방문하게 되며 이로 인해 질의처리 시간이 매우 길어지게 된다.

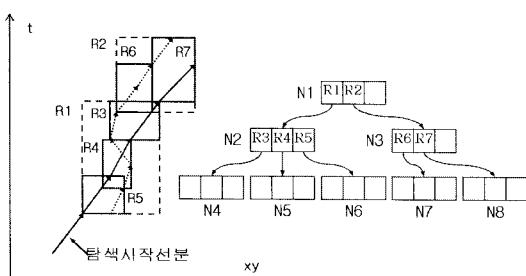


그림 5. 점질의에 의한 노드의 중복 방문
Fig. 5 Revisiting Nodes by Point Query

이동체의 궤적은 연결성(connectivity)과 근접성(proximity)을 가진다. 다음 궤적은 이전 궤적과 시간적으로 연속하므로 3차원 공간에서 가깝게 위치한다. 따라서 색인에서도 이들은 근접하므로 예측을 통해 다음 궤적을 순차적으로 찾지 않고 한꺼번에 찾아 노드의 중복 방문을 방지한다.

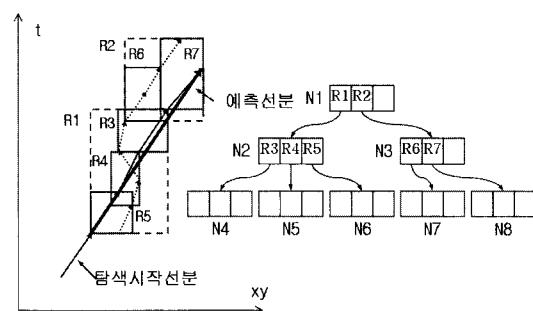


그림 6. 예측에 의한 궤적 탐색
Fig. 6 Trajectory Search by Prediction

그림 6과 같이 연속 궤적을 탐색하기 위해 궤적을 예측하고 예측한 궤적에 따라 질의를 처리하면 ($N_1, N_2, N_4, N_5, N_6, N_3, N_8$)의 순서로 총 7개의 노드만을 방문하여 질의를 처리할 수 있다. 이 논문에서는 이러한 궤적의 예측을 위하여 예측 함수를 제시하며 효율적인 예측 함수를 위한 여러 가지 요소 및 방법을 제시한다.

IV. 최소 공통 탐색 선분의 단일 방향 탐색

내부 질의영역과 CROSS 관계[4][5]를 가지는 객체의 궤적을 최소 공통 탐색 선분으로 정의하고 CL(minimum Common search Line)로 표기하며 CL에 대해서만 탐색을 수행하여 노드의 중복 방문을 방지한다.

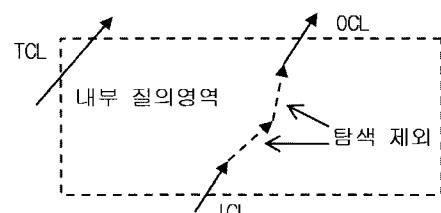


그림 7. CL의 종류 및 예
Fig. 7 Types and Examples of CL

그림 7과 같이 CL은 내부 질의영역과 CROSS 관계인 위상에 따라 영역을 빠져 나가는 OCL(Outward CL), 영역에 들어오는 ICL(Inward CL), 그리고 영역을 지나는 TCL(Traverse CL)로 분류하며 OCL은 전방향, ICL은 후방향, TCL은 전/후방향으로만 탐색하여 노드들의 중복 방문을 방지한다. 정의를 위하여 선분 X의 시작점을 X.start, 끝점을 X.end, 객체식별자를 X.oid로 표기하며 그 외의 표기는 관련 연구 [4], [5]를 따른다. 각각의 정의는 다음과 같다.

정의2: OCL, ICL, TCL

OCL 을 \vec{L} , ICL 을 \overleftarrow{L} , TCL 을 $\vec{\overleftarrow{L}}$ 라고 하자.
궤적 L 이 $(L^o \cap IR^o \neq \emptyset) \wedge (L^o \cap IR^- \neq \emptyset)$ 이면
다음 조건을 만족한다.

$$\begin{aligned} L = \vec{L} &\text{ iff } (L.start \in IR^o) \wedge (L.end \in IR^-) \\ L = \overleftarrow{L} &\text{ iff } (L.start \in IR^-) \wedge (L.end \in IR^o) \\ L = L &\text{ iff } (L.start \in IR^-) \wedge (L.end \in IR^-) \end{aligned}$$

V. 예측기반 여과 방법을 이용한 궤적의 점질의 탐색

기존 연구에서는 연속하는 궤적을 탐색하기 위하여 점 질의를 사용하여 궤적을 순차적으로 탐색한다. 이는 앞에서도 언급했듯이 색인의 부모 노드들을 중복 방문하여 질의처리가 느려지는 문제가 발생한다. 따라서 이 논문에서는 궤적을 예측하고 예측된 선분을 이용하여 연속된 궤적을 동시에 탐색하는 예측기반 여과 방법을 제안한다.

궤적을 예측하기 위해서는 다음의 사항을 고려해야 한다. 첫째 어떠한 예측 방법을 사용할 것인지를 결정해야 한다. 둘째 예측 방법을 사용할 때 무엇을 기준으로 할 것인지를 결정해야 한다. 셋째 얼마나 예측할 것인가를 결정해야 한다. 마지막으로 예측이 틀린 경우에 어떻게 보정할 것인지를 결정해야 한다. 이 각각의 요소들은 다음과 같다.

5.1 예측 방법

궤적을 예측하기 위하여 과거의 데이터를 사용하여 함수를 유도하고 이 함수에 따라 일정한 시간 뒤의 선분

을 예측하게 된다. 예측 방법은 선형 함수, 비선형 함수 등 다양하며 선형 함수를 이용한 예측 방법은 그림 8과 같이 실선으로 표시된 궤적이 있을 경우에 시작점과 끝점을 이용하여 선형함수를 유도한다. 이 선형 함수를 이용하여 실선과 같은 궤적을 예측한다.

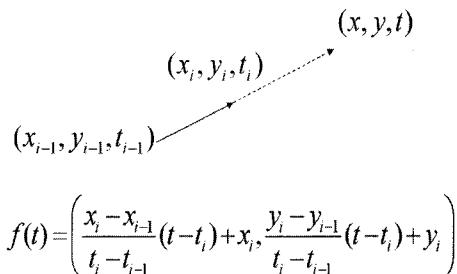


그림 8. 선형 함수를 이용한 궤적의 예측
Fig. 8 Prediction of Trajectory Using Linear Function

5.2 예측 기준

궤적을 예측하기 위해서는 예측 함수를 사용해야 되며 무엇을 기준으로 예측을 하느냐에 따라 예측한 궤적이 달라진다. 그림 9-(1)은 궤적을 예측하기 위하여 이전 선분 하나만을 이용하여 앞에서 언급한 선형함수를 이용하여 예측한 것이다. 즉, a1 선분을 기준으로 선형함수를 이용하여 예측을 하면 절선과 같은 예측 궤적이 나오게 된다.

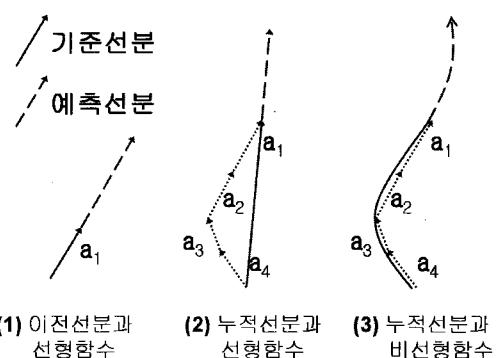


그림 9. 예측함수를 사용하기 위한 예측 기준
Fig. 9 Prediction Criteria for Using Prediction Function

그림 9-(2)는 누적선분과 선형함수를 이용하여 예측한 것으로 a1~a4까지의 선분을 기준으로 선형함수를 이용하여 예측한 것이다. 마지막으로 그림 9-(3)은 누적선

분을 이용하여 누적선분을 비선형으로 간주하고 비선형함수를 이용하여 예측한 결과이다. 이와 같이 예측 기준은 다양하며 보다 많은 과거 데이터를 사용하면 예측하는 것이 효과가 더 크다.

5.3 예측시간

케적을 예측하기 위해서는 얼마의 시간만큼 예측을 하느냐가 중요한 요소이다. 이를 위하여 먼저 다음과 같이 예측시간을 산출하기 위해 사용되는 기호를 표 1과 같이 정의한다. 이 경우에 다음과 같이 예측시간을 산출할 수 있다.

$$ls_{mo} = LS / MO$$

$$ls_{mohn} = ls_{mo} / LN = LS / MO / LN$$

$$t = (a * ls_{mohn} + b) * \Delta t \quad (\text{단 } a \geq 0, b \geq 0)$$

예를 들어 단말노드가 50개의 객체를 가지고 있을 수 있으며 객체의 종류가 5개라 하면 위 식에 의하여 단말노드에 들어가는 객체의 선분은 최대 10이 된다. 따라서 예측시간 $t = (a * 10 + b) * \Delta t$ 로 표기되어지며 실험에 의해 최적의 a, b 값을 산출해 낼 수 있다.

표 1. 예측시간 측정에 사용되는 기호
Table. 1 Symbols of Measuring Prediction Time

기호	의미
LS	전체 케적의 개수
MO	전체 이동 객체의 개수
LN	전체 단말노드의 개수
ls_{mo}	단일 객체를 구성하는 케적의 평균 개수
ls_{mohn}	단말노드에 포함된 동일 객체를 구성하는 케적의 평균 개수
Δt	한 선분의 시간 평균변화 값
t	예측할 시간 값

5.4 보정 방법

케적을 예측하고 예측된 선분을 이용하여 점 질의를 수행하는 경우에 예측이 잘못되었다면 보정이 필요하다. 이 경우에 어떠한 보정 방법을 사용할 것인가가 중요한 문제이다. 이 논문에서는 예측으로 임계 이하의 선분이 찾아진 경우에 예측 선분을 보정하여 탐색을 수행하는데 비어 있는 부분의 케적에 대하여 점 질의를 수행하여 탐색한다. 즉, 그림 10과 같이 기준 선분을 이용하여

예측을 한 결과 찾아진 선분이 이어지지 않았다면 예측을 잘못한 것이므로 이 경우 끝점을 이용해 점 질의를 수행하여 보정한다.

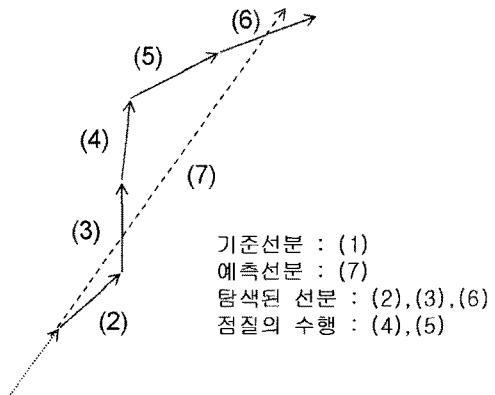


그림 10. 점질의에 의한 보정
Fig. 10 Correction by Point Query

보정하는 경우에 보정을 위한 최대 임계값은 다음과 같이 정의된다. 즉 보정 임계값은 최대 ls_{mohn} 값인 단말노드에 포함하는 객체의 선분수를 넘지 않아야 하며 실험을 통해 a 및 b 값을 결정해야 한다.

$$\text{보정 임계 값 } p: 1 \leq p \leq a * ls_{mohn} + b$$

$$(\text{단 } a \leq 1, 0 \leq b \leq ls_{mohn})$$

VI. 결론 및 향후 연구

이 논문은 영역 질의에 우수한 성능을 가진 R-tree 기반의 이동체 색인을 대상으로 케적 질의를 효율적으로 처리함으로써 복합 질의의 성능을 향상시키기 위한 방법을 제시한다. 먼저 내부 질의영역의 모든 케적들을 대상으로 전/후 방향으로 탐색하는 문제를 최소 공통 탐색 선분을 대상으로 단일 방향으로만 탐색하여 해결하였다. 또한 연속 케적을 점 질의로 탐색하는 문제는 연결성과 근접성을 고려하여 케적을 예측하는 예측기반 여과방법으로 해결하였다. 이 방법을 위해 예측을 위한 각각의 요소를 정의하고 각 요소에 사용될 수 있는 방법을 제시하였다.

향후 연구는 실제 데이터를 사용하여 다양한 요소를 가지고 실험을 수행하여 논문에서 제시한 알고리즘의 우수성을 증명하고 비용 모델을 제시하여 검증하는 것이다. 또한 복합 질의 성능의 보다 나은 향상을 위해 클러스터링 기반의 분할 정책을 이용해 새로운 색인 구조를 개발하는 것이다.

참고문헌

- [1] Y. Theoderidis, M. Vazirgiannis, T. Sellis, "Spatio-temporal indexing for large multimedia applications", In Proc. Multimedia Computing and Systems, p441-448, 1996
- [2] D. Pfoser, Y. Theodoridis, and C.S. Jensen, "Indexing Trajectories in Query Processing for Moving Objects", Chorochronos Technical Report, CH-99-3, 1999
- [3] D. Pfoser, C.S. Jensen, and Y. Theodoridis, "Novel Approaches in Query Processing for Moving Objects", In Proc. of the VLDB Conference, p395-406, 2000.
- [4] M.J. Egenhofer, and J.Herring, "Categorizing Binary Topological Relationship Between Regions, Lines and Points in Geographic Databases", Technical Report, Department of Surveying Engineering, University of Maine, 1991
- [5] E. Clementini, and P. Di Felice, "A Comparison of Methods for Representing Topological Relationships", Information Sciences 80, p1-34, 1994
- [6] 조형주, 정진완, "시공간 질의를 위한 인덱싱 기법", 한국정보과학회 한국데이터베이스 학술대회 논문집, 제18권 2호, p93-100, 2002

저자소개



반재훈(Ban ChaeHoon)

2006.2 부산대학교 공학박사
2008.9 ~ 고신대학교 인터넷비즈니스
학과 전임강사

※ 관심분야: 이동체 데이터베이스, RFID응용



김종민(Kim JongMin)

2000.2 부산대학교 공학박사
1996.3 ~ 고신대학교 인터넷비즈니스
학과 교수

※ 관심분야: 멀티미디어, 영상처리