

의미 패턴 추출을 위한 이동 객체의 위치 일반화

이연식^{1*}, 고 현²

¹군산대학교 컴퓨터정보공학과, ²한국항공우주연구원

Location Generalization of Moving Objects for the Extraction of Significant Patterns

Yonsik Lee^{1*} and Hyun Ko²

¹Dept. of Computer Information Engineering, Kunsan National University

²Korea Aerospace Research Institute

요약 최적 이동 경로 탐색이나 스케줄링 경로 예측 등 최적의 위치 기반 서비스 제공을 위하여 이동 객체의 위치 이력 데이터들에 대한 시공간적 속성들을 고려한 이동 객체들의 의미 있는 이동 패턴 추출 기법이 요구된다. 이에 본 논문에서는 의미 있는 이동 패턴 추출을 위하여 이동 객체의 연속적 위치 변화를 보다 효과적으로 패턴화하고 2차원 공간 영역으로의 인덱싱을 위한 R*-tree 기반의 이동 객체 위치 속성 일반화 기법을 제안한다. 제안한 위치 일반화 방법은 이동 객체들의 각 공간 영역 데이터를 이용하여 영역 간의 시간 간격에 대한 제약을 만족하는 이동 시퀀스를 생성하며, 생성된 이동 시퀀스들을 통하여 의미 있는 이동 패턴들을 추출한다. 추출된 이동 패턴들은 최적의 위치 기반 서비스를 위한 이동 객체의 시간 패턴 마이닝이나 객체의 이동 추이 분석 등에 효과적으로 이용될 수 있다.

Abstract In order to provide the optimal location based services such as the optimal moving path search or the scheduling pattern prediction, the extraction of significant moving pattern which is considered the temporal and spatial properties of the location-based historical data of the moving objects is essential. In this paper, for the extraction of significant moving pattern we propose the location generalization method which translates the location attributes of moving object into the spatial scope information based on R*-tree for more efficient patterning the continuous changes of the location of moving objects and for indexing to the 2-dimensional spatial scope. The proposed method generates the moving sequences which is satisfied the constraints of the time interval between the spatial scopes using the generalized spatial data, and extracts the significant moving patterns using them. And it can be an efficient method for the temporal pattern mining or the analysis of moving transition of the moving objects to provide the optimal location based services.

Key Words : Moving Object, Moving Pattern, Location Generalization, Extraction of Significant Moving Pattern

1. 서론

이동 객체의 위치 데이터를 이용해 효과적인 위치 기반 서비스를 제공하기 위한 기술들로 이동 객체의 시간에 따른 위치 변화를 고려하여 객체의 위치 및 궤적을 추적하는 방법들[1,2,3]과 객체의 현재 위치, 속도, 방향 정보를 이용하여 미래의 위치를 예측하기 위한 방법들[4,5]

이 제안되었으나, 이들은 단순한 객체의 이동 상태 정보를 기반으로 한정적 이력 정보를 추출하는 방법들만을 제시하였을 뿐 지속적으로 변화하는 위치 데이터들 간의 관계에 내포된 의미 있는 지식 추출을 위한 패턴 탐사에는 한계를 보인다. 따라서 위치 기반 서비스 이용자들이나 각 활용 분야의 특성에 맞게 개인화되고 세분화된 응용 서비스나 콘텐츠를 제공하기 위해서는 방대한 이동

본 논문은 2009년도 정부의 재원으로 한국학술진흥재단(No. 2009-0074891)의 지원과 2010년도 중소기업청(No. 00039599)의 산학연공동기술개발 지원사업으로 수행된 연구결과임.

*교신저자 : 이연식(yslee@kunsan.ac.kr)

접수일 10년 09월 16일 수정일 (1차 10년 11월 22일, 2차 10년 11월 24일, 3차 11년 01월 03일) 게재확정일 11년 01월 13일

객체의 위치 이력 데이터로부터 의미 있는 지식인 유용한 패턴을 탐색하기 위한 시간 패턴 탐사가 필요하다. 기존의 시간 패턴 탐사 기법들은 시간과 공간 속성을 동시에 고려해야 하는 이동 객체 데이터의 패턴 탐사에 있어 공간 속성들의 변화를 충분히 고려하지 않아 이동 패턴 탐사 문제에 적합하지 않다. 이에 대한 해결 방법으로 이동 객체에 대한 시간 패턴 탐사를 위해서 Apriori를 확장한 이동 패턴 탐사 기법인 MP(Moving Pattern mining) 알고리즘들[4,6,7,8]이 제안되었으나, 이들은 이동 객체의 공간 속성을 고려하였지만 제약을 가진 공간 정보의 연속적인 변화에 대한 패턴 탐사에는 적용하기는 어렵기 때문에 특정 영역 내에서의 최적 경로 탐색 문제나 단위 기간 동안 이동 객체가 순회해야할 지점들에 대한 스케줄링 경로 예측 문제 등에 적합하지 않다[6,7,8]. 따라서, 이동 객체의 위치 이력 데이터들에 대한 시공간적 속성들을 고려하여 최적 이동 경로나 스케줄링 경로 예측 등을 위한 새로운 시간 패턴 탐사 기법의 개발을 위해서는 상세 수준의 위치 이력 데이터들을 의미 있는 공간 영역 정보 형태로 변환함으로써 지식화 가능한 패턴 탐사를 통해 보다 효율적인 위치 기반 서비스를 제공할 수 있는 방법이 요구된다. 이동 패턴 마이닝의 대상이 되는 이동 객체의 위치 이력 데이터들은 낮은 수준의 정보형태(x, y 좌표 값)로 표현되므로 이들로부터 직접적으로 지식화 가능한 패턴들을 탐색하는 것은 매우 어려운 일이기 때문에 상위 개념 수준으로 추상화할 필요가 있다[12]. 상위 개념 수준에서는 데이터의 집합이 요약되고 더 일반적인 정보로 표현되어진다.

본 논문에서는 이동 객체와 이들의 데이터베이스 내의 표현을 정의하고, 위치 일반화 접근을 위한 공간 영역으로의 인덱싱을 위하여 공간 개념 계층을 행정구역 중심으로 추상화한 R*-Tree 기반의 이동 객체 위치 속성에 대한 공간영역으로의 일반화 알고리즘들을 제시하고 이를 이용한 의미 패턴 추출 방법을 제안한다. 제시한 위치 일반화 알고리즘들은 공간 질의 처리 오버헤드를 줄이기 위하여 방향, 거리 및 위상 관계 등으로 분류되는 공간 관계들 중 위상 관계만을 고려하여 두 객체 간의 관계를 검사한다. 또한, 알고리즘들의 실효성을 위하여 서울시의 도로 네트워크 상에서 교차점이나 분기점, 종료점 등을 노드로 설정하고 각 노드에서 택시들의 운행 기록을 측위하여 이력 데이터베이스를 생성하고, 이를 정렬한 후 위치 일반화를 수행하여 이동 시퀀스로 생성되는 의미 패턴 추출 결과를 보인다. 추출된 이동 패턴들은 최적의 위치기반 서비스를 위한 이동 객체의 시간 패턴 마이닝이나 객체의 이동 추이 분석 등에 효과적으로 이용될 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 이동 객체의 표현과 위치 일반화 접근 방법을 설명하고, 3장에서는 의미 패턴 추출을 위한 위치 일반화 연산들을 제시한다. 4장에서는 실제 이동 객체의 이력을 데이터베이스화하고 위치 일반화를 실행하여 의미 있는 이동 패턴인 이동 시퀀스를 구하는 과정을 실험을 통하여 보이고, 결론 및 향후 연구를 제시한다.

2. 이동 객체의 표현과 위치 일반화

2.1 이동 객체 정의

이동 객체는 시간의 흐름에 따라 객체가 이동하면서 위치 및 모양이 연속적으로 변경되는 특성을 가지는 시공간 데이터로서 이동 점 객체(Moving Point Object)와 이동 영역 객체(Moving Region Object)로 구분할 수 있다. 이동 객체의 위치 변화는 이산적 또는 연속적으로 발생할 수 있으며, 시점(time points) 또는 시간 구간(time periods)으로 기록될 수 있다. 이러한 이동 객체를 표현하는 데이터 모델은 크게 이동 객체를 무한한 점들의 집합으로 표현하는 연속적 모델과 이동 객체를 유한한 점들의 집합으로 표현하는 이산적 모델이 있다. 이동 객체의 위치 속성은 x, y 를 축으로 하는 2차원 공간상에서 연속적으로 변화한다. 이동 점 객체의 공간 속성은 이동 객체의 평면상의 x, y 좌표로써 표현된다. 본 논문에서는 이동 점 객체 *Mpoint*와 의미 있는 이동 패턴을 다음과 같이 정의한다.

정의 1- 이동 점 객체

$$Mpoint = oid, \{(VT_1, L_1), (VT_2, L_2), \dots, (VT_n, L_n)\}$$

oid : 유일한 특성을 갖는 객체의 식별자

VT_i : 유효시간

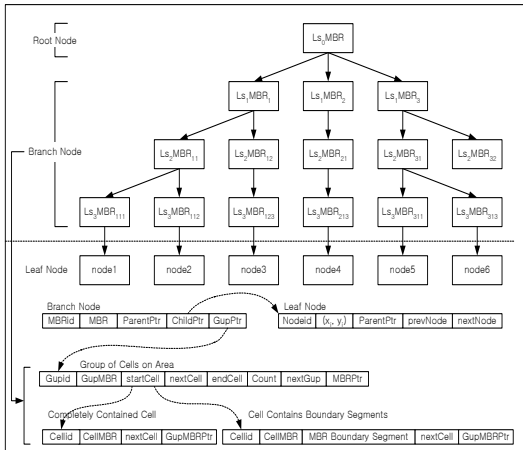
L_i : VT_i 에 샘플링된 객체의 위치 (x_i, y_i)

정의 2- 의미 패턴

이동 객체 데이터베이스 MD, 사용자가 지정한 특정지점들 S와 F(공간제약), 시간기간 T1과 T2(시간제약)이 주어졌을 때, 방대한 이동 객체의 이력 데이터들로부터 공간 및 시간제약을 만족하고 공간 개념에 따라 추상화된 영역으로 일반화되어 나타나는 S에서 F까지의 부분 이동 시퀀스이다.

2.2 위치 일반화 접근법

이동 객체는 정의 1에서와 보인 바와 같이 객체 식별자(OID)와 객체의 위치를 샘플링하기 위한 유효시간(VTi), 유효시간 상에서의 위치(Li)로 표현된다. 실세계에서 이동 객체는 끊임없이 공간을 이동하며 무한한 위치 정보를 발생시키기 때문에 상세화된 위치 정보를 가지고 의미 있는 패턴을 탐색하기에는 많은 어려움이 따른다. 따라서 의미 있는 지식을 획득하기 위해서 이동 객체의 상세한 위치 정보를 공간상의 상위 개념으로 추상화할 필요가 있다. 이동 객체의 위치 속성은 공간상의 특정 지역 수준의 의미로 일반화되기 위하여 공간 지역에 대한 공간 개념 계층이 요구된다. 다음 그림 1은 5 레벨의 공간 개념 계층을 공간 탐색 트리와 AGHT(Area Grid Hash Tree)를 이용하여 표현한 예이다. 그림 1의 단말 수준의 노드는 도로 상의 교차점, 분기점, 커브지점, 도로종료점 등을 표현하고 각 레벨별 공간 범위는 포함관계를 통해 상위 레벨 영역으로 일반화되며, R-Tree와 해시 테이블은 각 셀에 대한 이동 점의 포함 여부를 검사하기 위한 연산 시 사용한다.



[그림 1] 공간 개념 계층의 탐색 트리 표현

이동 객체의 위치 정보에 대한 공간 일반화는 x, y 좌표 값으로 표현된 이동 객체의 공간 정보를 좌표가 포함되는 영역으로 나타나는 방법이다. 특정 시간 기간 VT_i 에서 VT_n 동안의 이동 객체의 위치 정보는 이동 객체의 공간 위치 속성을 나타내는 좌표의 유한집합 $L = \{l_1, l_2, \dots, l_m\}$ 로 표현될 수 있다. $l_i = (x_i, y_i)$ 이며 각 x_i, y_i 는 평면상에서의 이동 객체의 좌표이다. $A = \{a_1, a_2, \dots, a_n\}$ 은 이동 객체의 공간 위치 속성 값으로 표현되는 영역의 집합으로, 각 $1 \leq j \leq n$ 에 대해 $a_j = (l_1, l_2, \dots, l_k)$ 이고 $l_k = (x_k, y_k)$ 이다.

이동 객체의 위치 일반화는 영역이 가지는 대표 좌표 값들을 사용하여 좌표 값으로 표현된 이동 객체의 공간 위치 속성을 공간연산을 통해 영역으로 변환한다[12].

이동 객체의 위치 속성은 일반화된 공간 영역으로 표현되며, 이동 객체의 이동 이력은 영역의 순차 리스트로 표현될 수 있다. 영역의 순차 리스트는 한 이동 객체에 대해 시간의 증가에 따라서 영역을 순서적으로 정렬한 이동 시퀀스(moving sequence)로 표현한다[13,14]. k 개의 길이를 가진 시퀀스, 즉 k 개의 영역으로 이루어진 시퀀스는 k -시퀀스로 표기되며, 하나의 영역은 한 시퀀스에서 여러 번 나타날 수 있다. a_k, b_k 등과 같이 이동 영역을 나타내는 식별자들로 구성된 두 이동 시퀀스에 대하여 시퀀스 $\langle a_1 a_2 \dots a_n \rangle$ 은 만약 $a_1 = b_{11}, a_2 = b_{12}, \dots, a_n = b_{1n}$ 인 정수 $i_1 < i_2 < \dots < i_n$ 이 존재한다면 다른 시퀀스 $\langle b_1 b_2 \dots b_m \rangle$ 의 부분 시퀀스이다. 하나의 시퀀스 $MS = \{s_1, s_2, \dots, s_k\}$ 로 표현할 수 있다. 이 때, $s_j = (a_j, a_i)$ 이며, a_j 는 이동 객체가 a_j 영역 내에 있을 때의 대표유효시간이고 $a_i \in A$ 이다. 특정 영역(a_i) 내 대표유효시간 AT_i 는 특정 영역(A_i)에 들어갔을 때 처음 샘플링된 시간($A_i Tim$)에서부터 나오기 전 샘플링된 시간($A_i Tout$)까지의 평균유효시간이다. 한편, 이동 객체의 위치 변화에 대한 영역 순차 리스트인 시퀀스를 형성하기 위해서는 트랜잭션 시간 간의 시간 간격인 시간제약조건을 적용한다. 시간제약조건은 시퀀스 내에 연속적인 이동으로 영역이 포함되기 위해 인접한 이동이 발생한 시간 $t_j - t_{j-1}$ 을 의미한다. 최대 시간간격은 max_gap 으로 $t_j - t_{j-1} \leq max_gap$ 이고, $2 \leq j \leq k$ 이다. 이동 시퀀스의 집합 $S = \{ms_1, ms_2, \dots, ms_m\}$ 이라 할 때, 각 ms_i 는 이동 시퀀스를 나타내며, 이 때 $1 \leq i \leq m$ 이다. 시퀀스 ms_i 가 다른 시퀀스 ms_j 의 부분 시퀀스이면 ms_i 는 ms_j 를 포함한다고 한다.

3. 위치 일반화를 위한 연산

위치 일반화 과정은 시간에 따라 지속적으로 변화하는 이동 객체의 위치 좌표와 공간 영역 간의 관계를 미리 파악하여 공간 관계에 관련된 질의 처리 시 오버헤드를 줄이기 위한 것으로, 두 객체의 공간 관계에 대한 분석을 통해 이동 객체의 위치를 공간 영역으로 일반화한다. 공간 관계는 크게 방향 관계(directional relationship), 거리 관계(distance relationship), 위상 관계(topological relationship)로 분류되며 이동 객체의 위치를 일반화하는데 있어서는 위상 관계만을 고려하여 두 객체 간의 관계를 검사한다. 본 논문에서는 위상 관계들 중 각각 서로의 역이되거나 점과 영역 간에서 발생할 수 없는 관계는 무시하고 유효한 대상이 되는 *meet*와 *properly-contain* 관

계만을 고려 대상으로 한다. 이동 객체의 위치 값에 대한 공간 영역으로의 일반화는 방향 및 위상 관계에 대한 공간 연산을 통해 객체의 x, y 좌표 점이 특정 영역에 포함되어 있는지를 검사하여 일정한 범위 값을 가지는 공간 영역으로 변환한다.

a. *Contains*

이동 객체의 위치 일반화는 객체의 위치 좌표 점이 공간 개념 계층에서 레벨 수준별 영역의 최소경계사각형 (MBR : Minimum Bounding Rectangle)에 포함되었는지를 검사하는 단계와 특정 영역들의 경계나 영역 내부에 포함되었는지를 검사하여 위치 속성을 영역 범위로 일반화하는 단계를 거친다. 이 두 과정은 이동 객체의 위치 이력을 공간 영역의 순차리스트로 표현함에 있어 이동 객체가 스캔되는 위치 좌표 점을 영역 범위로 변환하는 *Contains* 공간 연산을 통해 처리된다. *Contains* 연산을 위한 선행 작업으로는 사용자가 입력한 시공간 질의를 분석하여 공간 개념 계층의 레벨 수준을 결정해야 하는 과정이 필요하다. *Contains* 연산 처리 과정은 먼저, 시공간 질의 분석을 통해 결정된 공간 계층 레벨 수준(N)과 이동 객체의 위치 좌표 점(MP), 객체의 이전 위치 영역($PrevA$)에 대한 정보들을 입력받아 최상위 레벨 수준의 전체 공간 영역을 입력된 하위 레벨 수준까지 최소 영역 사각형인 MBR로 분할한다. MBR로 분할하는 이유는 사용자의 시공간 질의 수준에 맞게 공간 개념 계층의 레벨 수준으로 이동 시퀀스를 생성함으로써 지식화할 수 있는 이동 패턴을 탐색하기 위함이다. 한편 *Contains* 연산은 질의 분석을 통해 결정된 레벨 수준까지 공간 영역을 반복적으로 분할하며 *ContainedMBR* 함수를 통해 이동 객체의 위치 좌표 점(MP)이 레벨 수준별 영역 MBR에 포함되는지를 검사한다. 또한, *ContainedMBR* 함수를 통해 이동 객체의 위치 좌표 점이 포함된 모든 MBR들을 추출한 후 *ContainedArea* 함수를 이용하여 위치 좌표 점이 실제 어느 영역의 경계 또는 내부에 위치하는지를 검사한다. 이때, *ContainedArea* 함수의 결과로 반환되는 좌표 점을 포함한 영역은 이동 시퀀스를 생성하기 위한 일반화된 영역이다.

b. *ContainedMBR*

이동 객체의 위치 일반화 위한 두 단계 중 먼저, 첫 번째 단계는 공간 개념 계층의 레벨 수준에 따라 전체의 공간 영역을 레벨 수준별 영역의 MBR로 분할하여 각 MBR에 이동 객체의 위치 좌표 점이 포함되었는지를 검사하는 단계이다. 이 때, 시공간 질의 분석을 통해 공간 개념 계층의 레벨 수준이 어느 정도인지를 판단한 후 전

체 공간 영역을 해당 레벨 수준의 영역들로 분할하고 각 영역들의 동서남북 네 방향에 대한 최대 경계점을 기준으로 MBR을 생성한다. *ContainedMBR* 연산은 위의 과정을 처리하기 위한 알고리즘으로, 이동 객체의 위치 좌표 점(MP)과 각 레벨 수준별 영역 MBR 집합(L_i)을 입력받아 위치 좌표 점이 각 MBR에 포함되었는지를 검사하여 해당 MBR들의 집합(*ContMBR*)을 반환하는 공간 연산 함수이다. *ContainedMBR* 연산에서는 앞서 기술한 바와 같이 점 객체와 영역 객체 간의 관계에 있어 유효한 위상 관계인 *properly-contain* 과 *meet* 관계가 발생하는지에 대해서만 검사한다.

c. *ContainedArea*

이동 객체의 위치 일반화에서 두 번째 단계는 첫 번째 단계를 거쳐 추출한 레벨 수준별 영역 MBR들의 실제 영역에 대해서 이동 객체의 위치 좌표 점이 영역의 경계에 위치하는지 또는 영역 내부에 포함되는지를 검사하여 이동 객체의 위치 값을 영역 범위로 일반화하는 단계이다. 이 방법은 먼저, 두 개 이상의 영역들이 만나는 간선 (Edge) 상에 이동 객체의 위치 좌표 점이 위치하는지를 검사한다. 만약 영역 간선 상에 존재한다면 이동 객체의 궤적에서 이전에 위치한 영역이 어디인지를 판단하여 이전 영역에 포함된다고 간주한다. 이 외에도 특정 영역 내에 또 다른 영역을 완전 포함하는 형태의 영역 객체에 대한 점 객체의 포함 여부 검사는 내부 영역 간선들과의 교차는 모두 무시하고 외부 간선들과의 교차 수에 따라 포함 여부를 판별한다. *ContainedArea* 연산은 위치 좌표 점을 포함하는 MBR 집합, 이동 객체의 위치 좌표 점, 이동 객체의 이전 위치 포함 영역($PrevA$)에 대한 정보를 전달받아 각 영역 MBR에 대응하는 실제 영역들이 위치 좌표 점을 포함하는지 검사하여 해당 영역의 이름을 반환하는 공간 연산 함수이다. 특정 영역 내에 특정 좌표 점이 포함되는지를 검사하는 메커니즘은 위에서 설명한 바와 같이 임의의 좌표 점 P에서 시작하는 수평 단방향 직선 l 을 그린 후, l 과 교차하는 영역의 간선 개수가 짝수이면 점 P는 A의 외부 위치, 홀수이면 내부 위치 점으로 판별한다.

ContainedArea 연산의 처리 과정은 먼저, *isAreaEdge* 함수를 통해 특정 레벨 수준의 영역 MBR로 구성된 MBR 집합으로부터 각 MBR에 대한 실제 영역의 간선 정보들을 추출한다. 이 때 *isAreaEdge* 함수에서는 영역 내에 또 다른 영역을 포함하는 영역의 경우 외부 간선 정보만을 반환한다. 특정 영역에 대한 간선 정보가 추출되면, *onEdge* 함수는 각각의 간선 상에 이동 객체의 위치 좌표 점이 위치하는지를 검사한다. 만약 좌표 점 MP 가

간선 상에 존재한다면 이전 MP 가 위치했던 영역의 이름을 반환하고, 존재하지 않는다면 $Intersect$ 함수에 의해 교차 간선의 수를 계산한다. $Intersect$ 함수는 좌표 점 MP 에서부터 시작하는 수평 단방향 직선 l 이 영역 간선들과 교차하는지에 대한 여부를 판별하는 공간 연산 함수이다. $ContainedArea$ 연산의 마지막 과정은 $Intersect$ 함수에 의해 판별된 교차 간선의 총 개수가 짝수인지 혹은 홀수인지를 계산하여 홀수일 경우 좌표 점 MP 가 해당 영역에 포함된다고 간주하여 영역의 이름을 반환한다.

d. *Intersect*

Intersect 연산은 특정 영역을 구성하는 하나의 간선 E 와 직선 l 의 값을 입력받아 두 선이 교차하는지 또는 직선 l 과 간선 E 가 동일선 상에 위치하는지를 검사하는 공간 연산 함수이다. 이 때 수평 단방향 직선 l 은 이동 객체의 위치 좌표 점 MP 에서부터 특정 영역에 대한 최소 경계 사각형인 MBR의 우측 경계선을 구성하는 한 점까지의 직선으로 $l = \{MP, BP, \alpha \in \{above, right, below, right\}\}$ 로 표현된다. *Intersect* 연산은 특정 영역을 구성하는 간선 E 와 직선 l 이 교차하는지 또는 동일선 상에 위치하는지를 검사한다.

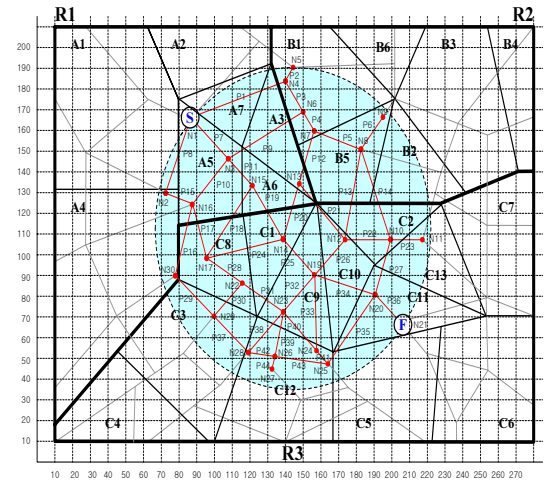
Intersect 연산 과정에서 *onEdge* 함수는 직선 l 과 점들의 집합 $\{P_1, P_2, P_3, \dots, P_n\}$ 으로 표현될 수 있는 영역 간선 E 의 한 점 P 를 입력받아 직선 l 상에 점 P 가 위치하는지 검사한다. 만약 영역 간선 E 의 시작점 P_1 과 마지막 끝점 P_n 이 직선 l 상에 모두 존재하지 않는다면 두 선을 구성하는 각각의 점이 서로 같은 지점에 위치하는지를 반복적으로 검사하여 같을 경우 True 값을 반환한다. 반대로 영역 간선 E 의 두 점 P_1 과 P_n 이 직선 l 상에 모두 존재한다면 직선 l 과 간선 E 는 동일선 상에 존재하므로 교차되지 않는다고 간주되어 False 값을 반환한다. 또한, 간선 E 의 두 점 P_1 과 P_n 중 하나의 점만이 직선 l 상에 존재할 경우 다른 점이 l 과의 교차점보다 위에 있으면 True 값을 아래에 있으면 False 값을 반환한다.

4. 실험 및 결론

4.1 실험

위치 일반화 방법은 데이터베이스 정렬 단계를 거친 후 위치 일반화를 수행하여 이동 시퀀스를 추출하게 된다. 이동 객체 데이터베이스는 데이터 마이닝을 위해 객체 식별자를 주기로, 트랜잭션 시간을 보조기로 정렬한다. 이 과정은 이동 패턴 탐사를 위해 객체별로 정렬하고 트랜잭션 시간에 따라 정렬하는 변환 과정이라 할 수 있

다. 실험을 위한 구현은 Windows XP 기반으로 Eclipse 3.2, JDK 1.6.0과 JBuilder X 프로그램을 이용하였으며, 데이터베이스는 Oracle 10g를 사용하였다. 실험을 위한 Geometry 데이터로 서울시 행정 구획 데이터와 도로 네트워크 데이터(그림 2)를 사용하였고, 이동 객체 데이터로 서울시의 도로 네트워크 상에서 택시들의 운행 기록을 측위하여 이력 데이터를 생성하였다.



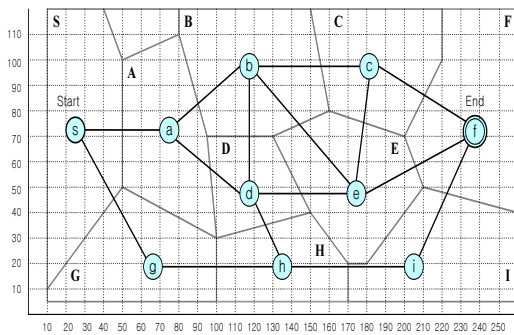
[그림 2] 공간지역과 도로 네트워크 예

이력 데이터는 출발점(서울시 중구 내 임의노드)을 중심으로 도로 네트워크 상에서의 교차점이나 분기점, 종로점 등을 노드로 설정하여 각 노드에서 이동 객체의 위치 정보를 샘플링 하였으며, 실험에 이용한 이력 데이터의 유형은 표 1과 같다.

[표 1] 이동 객체 표현 예

OID	VT	X	Y
15441032	2009/11/21/10/10/35	126.5101	37.1998
15441032	2009/11/21/10/14/04	126.5740	37.1997
15441032	2009/11/21/10/17/53	126.4778	37.1916

다음 그림 3은 실험을 위하여 데이터베이스를 정렬한 결과에 따라 이동 객체의 이동 경로가 x, y 좌표 상에 있다는 가정으로 표현한 것이며, 표 2는 이동 객체의 이력 데이터베이스와 일반화 결과를 보인다.



[그림 3] 이동 객체의 이동 경로 예

[표 2] 객체 식별자와 트랜잭션 시간에 따른 데이터베이스와 일반화 결과

(a) 이동 객체의 이력 데이터베이스

OID	VT	X	Y
1	2009/11/19/08/04	24	69
	2009/11/19/08/19	80	76
	2009/11/19/08/32	118	103
	2009/11/19/08/55	176	55
	2009/11/19/09/17	232	70
2	2009/11/21/09/30	21	78
	2009/11/21/09/38	77	75
	2009/11/21/09/45	122	95
	2009/11/21/09/56	119	45
	2009/11/21/10/04	133	19
	2009/11/21/10/31	203	23
3	2009/11/21/10/31	236	71
	2009/11/21/20/13	121	49
	2009/11/21/20/25	113	98
	2009/11/21/20/37	181	92
	2009/11/21/20/53	175	48
4	2009/11/21/21/28	233	71
	2009/11/25/18/24	22	71
	2009/11/25/18/56	63	14
	2009/11/25/19/18	135	23
	2009/11/25/19/26	121	44
	2009/11/30/10/05	121	44
	2009/11/30/19/16	173	51

(b) 위치 일반화 결과

OID	VT	Location
1	2009/11/19/08/04	s
	2009/11/19/08/19	a
	2009/11/19/08/32	b
	2009/11/19/08/55	e
	2009/11/19/09/17	f
2	2009/11/21/09/30	s
	2009/11/21/09/38	a
	2009/11/21/09/45	b
	2009/11/21/09/56	d
	2009/11/21/10/04	h

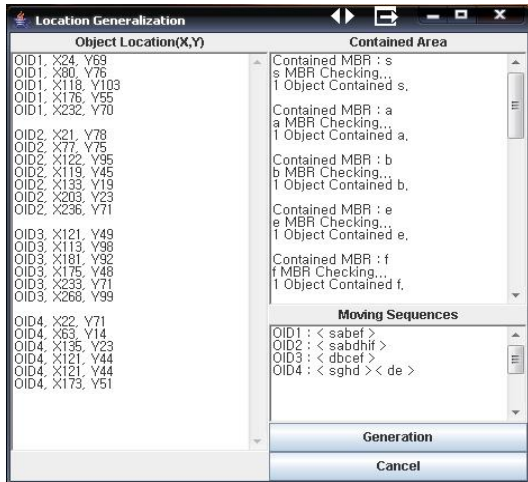
3	2009/11/21/10/31	i
	2009/11/21/11/01	f
	2009/11/21/20/13	d
	2009/11/21/20/25	b
	2009/11/21/20/37	c
4	2009/11/21/20/53	e
	2009/11/21/21/28	f
	2009/11/25/18/24	s
	2009/11/25/18/56	g
	2009/11/25/19/18	h
	2009/11/25/19/26	d
	2009/11/30/10/05	d
2009/11/30/19/16	e	

(c) 이동 시퀀스 (트랜잭션 데이터)

OID	Moving Sequence
1	<s a b e f>
2	<s a b d h i f>
3	<d b c e f>
4	<s g h d>, <d e>

표 2의 (b)는 (a)의 이력 데이터베이스로부터 각 이동 객체의 공간 값 속성을 ID별로 샘플링된 시간 순으로 정렬한 후 어떠한 영역에 포함되는지 *Contains* 공간 연산을 적용하여 일반화된 영역으로 변환한 예이다. 이 때, 이동 객체가 특정 영역에 머무르다가 다시 이동하는 경우 특정 영역에 도달한 공간 속성으로 위치를 축약한다. 예를 들면, 표 2에서 이동 객체 4는 영역 d에 2009년 11월 25일 17시 26분에 도달하였다가 다시 2009년 11월 30일 10시 05분 e를 향해 이동하게 되는데 이 두 공간 속성 값을 최초 영역 d에 도달한 값으로 축약한다. 표 2의 (b)와 같이 이동 객체의 위치 속성이 *Contains* 연산을 통해 공간 영역으로 일반화되면 이동 객체의 이동 변화를 시간에 따라 순차적으로 나열하여 이동 시퀀스를 생성할 수 있다. 이동 시퀀스의 생성은 객체의 이동 변화를 공간 연산을 적용한 위치 일반화를 통해 시간 순차의 지속기간 (duration) 및 특정 공간 범위로 제한된 영역의 순차리스트로 표현할 수 있으나, 이러한 영역의 순차리스트는 사건(영역 변화) 사이의 시간 간격이 고려되지 않아 패턴 마이닝을 위한 트랜잭션인 이동 시퀀스로 사용하기에는 문제가 있다. 일반적으로 패턴 탐사의 대상이 되는 하나의 시퀀스는 패턴에 있는 사건들 사이의 시간 간격을 만족해야만 하나의 트랜잭션, 즉 시퀀스로 생성될 수 있다. 따라서 이동 패턴 마이닝의 대상이 되는 이동 시퀀스를 생성하기 위해서는 시퀀스를 구성하는 영역 간에 최대 시간 제약조건을 두어 영역 변화 시 최대 시간 간격에 대

한 제약을 만족해야만 의미 있는 이동 시퀀스로 생성될 수 있도록 해야 한다. 즉, 이동 객체의 공간 속성에 대한 샘플링 시간을 검사하여 특정 영역에 머문 시간이 최대 시간 간격 max_gap 을 초과하면 초과 이전까지의 영역 순차리스트에 대한 이동 시퀀스와 초과 이후의 이동 시퀀스로 분리한다. 가령, 표 2의 (b)에서 이동 객체 4의 이동 경로는 영역 간을 이동한 시간 간격을 고려하지 않았을 경우 $s \rightarrow g \rightarrow h \rightarrow d \rightarrow e$ 로 표현될 수 있다. 만약 영역 간의 이동 시간 간격이 최대 1일이라면 영역 d에서 멈추었다가 다시 e를 향해 출발하는데 걸린 시간이 5일이기에 때문에 이동 경로 $s \rightarrow g \rightarrow h \rightarrow d \rightarrow e$ 는 $\langle s \ g \ h \ d \rangle$ 와 $\langle d \ e \rangle$ 로 분리된다. 표 2의 (c)는 (b)의 일반화된 영역 표현을 시간 간격을 고려하여 이동 시퀀스로 생성한 것이다. 다음 그림 4는 표 2의 이동 객체들의 이력 데이터베이스를 대상으로 위치 일반화를 통한 의미 패턴 추출 결과 화면이다.



[그림 4] 이동 객체의 의미 패턴 추출 결과

4.2 결론 및 향후 연구과제

이동 객체의 위치 이력 데이터들은 공간상에서 x, y 좌표 값으로 표현되기 때문에 낮은 수준의 상세한 정보 형태를 띠고 있다. 하위 개념 수준으로 표현된 위치 데이터들로부터 직접적으로 지식화 가능한 패턴들을 탐색하는 것은 매우 어려운 일이기 때문에 이동 객체의 연속적 위치 변화를 보다 효과적으로 패턴화하기 위한 공간영역으로의 일반화 접근법이 필요하다. 따라서 본 논문에서는 2차원 공간 영역으로의 인덱싱을 위해서 R^* -Tree에 기반한 이동 객체의 위치 속성에 대한 공간영역으로의 일반화 방법을 제시하였고, 이를 통해 일반화된 데이터 집합을 생성하여 이동 패턴 탐사를 보다 효과적으로 수행할 수 있도록 하였다. 공간 영역에 대한 정보는 MBR 간의

겹침의 최소화과 영역의 최소화에 기반한 최적화 인덱스로 R^* -Tree로 구성하였으며, 이동 객체의 위치 속성을 공간 영역으로의 변환을 위해 *Contain* 공간 연산(공간 개념 계층에서 레벨 수준별 영역의 MBR에 대한 포함 여부를 검사하는 *ContainedMBR* 연산과 특정 영역 내에 포함되었는지를 검사하는 *ContainedArea* 연산)를 제시하였으며, 좌표 점에서 부터 시작하는 수평 단방향 직선이 영역 간 선들과 교차하는지에 대한 여부를 판별하는 *Intersect* 공간 연산 함수를 제시하였다. 또한, 일반화된 각 공간 영역 데이터를 이용하여 영역 간에 최대 시간 제약조건을 두어 영역 변화 시 최대 시간 간격에 대한 제약을 만족하는 이동 시퀀스 생성 방법을 제안하였다. 제안된 방법으로 생성한 이동 시퀀스들은 의미 있는 지식 추출을 위한 이동 패턴 탐사에 이용될 수 있으며, 패턴 탐사를 통한 객체의 이동 추이 분석을 통해 다양한 형태의 서비스로 개발될 수 있다.

향후 연구과제로는 공간영역으로의 일반화 방법을 통해 생성된 이동 객체 데이터와 이동 시퀀스를 이용한 최적 이동 경로 마이닝 기법과 이를 이용하여 단위 시간동안 이동 객체가 순회해야 하는 지점들에 대한 스케줄링 경로 예측 등의 연구가 요구된다.

참고문헌

- [1] 안윤애, 김동호, 류근호, “차량위치 추적을 위한 이동 객체 관리 시스템의 설계”, 정보처리학회논문지D, 제 9-D권 제5호, 2002.
- [2] N. Mamoulis, H. Cao, G. Kollios, M. Hadjieleftheriou, Y. Tao and D. W. Cheung, “Mining, Indexing and Querying Historical Spatio-Temporal Data”, in proc. on the International Conference on Knowledge Discovery and Data Mining, 2004.
- [3] D. Pfoser, C. S. Jensen, and Y. Theodoridis, “Novel Approaches in Query Processing for Moving Objects”, Proc. of the Int'l Conf. on VLDB, 2000.
- [4] J. Gudmundsson, M. V. Kreveld and B. Speckmann, “Efficient Detection of Motion Patterns in Spatio-Temporal Data Sets”, in proc. on the 12th annual ACM international workshop on Geographic Information Systems(GIS), pp.250- 257, 2004.
- [5] Y. Huang, L. Zhang and P. Zhang, “Finding Sequential Patterns from a Massive Number of Spatio-Temporal Events”, SDM, SIAM, 2006.
- [6] J. D. Chung , O. H. Paek , J. W. Lee , K. H. Ryu, “Temporal Pattern Mining of Moving Objects for

Location-Based Service”, Proc. of the 13th International Conference on Database and Expert Systems Applications, p.331-340, September 02-06, 2002.

- [7] J. W. Lee, O. H. Paek, K. H. Ryu, “Temporal Moving Pattern Mining for Location-based Service”, The Journal of Systems and Software, Vol. 73. 2004.
- [8] H. Cao, N. Mamoulis and D. W. Cheung, “Mining Frequent Spatio-Temporal Sequential Patterns”, in proc. on the 5th IEEE International Conference on Data Mining(ICDM), pp.82-89, 2005.
- [9] D. O. Kim, H. K. Kang, D. S. Hong, J. K. Yun and K. J. Han, “STMPE : An Efficient Movement Pattern Extraction Algorithm for Spatio-temporal Data Mining”, in proc. on International Conference on Computational Science and Its Applications(ICCSA), pp.259-269, 2006.
- [10] 이준욱, “위치 기반 서비스를 위한 이동 객체의 시간 패턴 탐사”, 한국정보과학회 논문지, 제29권, 제5호, 2002.
- [11] 한선영, “시공간 이동 시퀀스 패턴 마이닝 기법”, 이화여자대학교 대학원, 석사학위논문, 2006.
- [12] 고현, 김광중, 이연식, “R* Tree와 Grid를 이용한 이동 객체의 위치 일반화 기법”, 한국컴퓨터정보학회 논문지, 제12권 2호, pp.231-242, 2007. 5
- [13] 고 현, 이연식, “이동 시퀀스의 빈발도를 이용한 최적 이동 패턴 탐사 기법”, 한국정보처리학회 논문지, 제16-D권, 제1호, 2009.
- [14] 이연식, 박성숙, “시퀀스 빈발도와 가중치를 이용한 최적 이동 패턴 탐사” 한국인터넷정보학회논문지, 제10권 제5호, pp.79-93, 2009.10

이 연 식(Yonsik Lee)

[정회원]



- 1982년 2월 : 전남대학교 전자계산학과 졸업(이학사)
- 1984년 2월 : 전남대학교 대학원 전자계산학과 졸업(이학석사)
- 1994년 2월 : 전북대학교 대학원 전산응용공학전공 졸업(공학박사)

- 1997년 ~ 1998년 : University of Missouri(Kansas City) 교환교수
- 2004년 ~ 2005년 : Ohio State University 교환교수
- 1986년 3월 ~ 현재 : 군산대학교 컴퓨터정보공학과 교수

<관심분야>

번역기 이론, 객체지향시스템, 능동시스템, 지능형 에이전트 미들웨어

고 현(Hyun Ko)

[정회원]



- 2001년 2월 : 군산대학교 컴퓨터정보과학과 졸업(이학사)
- 2003년 2월 : 군산대학교 대학원 컴퓨터정보과학과 졸업(이학석사)
- 2007년 2월 : 군산대학교 대학원 컴퓨터정보과학과 졸업(이학박사)
- 2008년 1월 ~ 현재 : 한국항공우주연구원 프로젝트 연구원

<관심분야>

에이전트 시스템, 이동객체 시스템, 시공간 데이터베이스, 데이터 마이닝