

# 공간 네트워크에서 이동 객체를 위한 그리드 기반 유사 궤적 검색<sup>†</sup>

## Grid-based Similar Trajectory Search for Moving Objects on Road Network

김영창\* / Young-Chang Kim, 장재우\*\* / Jae-Woo Chang

### 요약

최근, 이동 단말기의 확산과 통신 기술의 발달로 인하여 이동 객체들의 과거 궤적 데이터에서 이동 객체의 이동 패턴을 이용하는 응용 서비스의 활용이 점점 증대되고 있다. 특히, 대중교통의 노선 설계나 새로운 도시를 위한 도로 네트워크 설계에 활용하기 위하여, 도로나 철도와 같은 공간 네트워크 상에서 이동하는 이동 객체의 궤적들의 유사 패턴을 활용할 수 있다. 본 논문에서는 공간 네트워크에서 이동 객체 궤적을 위한 시공간 유사 궤적 검색 알고리즘을 제안한다. 이를 위하여 도로 네트워크 상에서 실제 도로 네트워크 거리에 기반한 시공간 유사도 측정방법을 정의하고, 효율적인 유사 궤적 검색을 위한 그리드 기반 색인 기법을 제안한다. 마지막으로 본 논문에서 제안하는 유사 궤적 검색 알고리즘의 효율성을 입증하기 위해 제안하는 알고리즘의 성능을 분석한다.

### Abstract

With the spread of mobile devices and advances in communication techknowledges, the needs of application which uses the movement patterns of moving objects in history trajectory data of moving objects gets increasing. Especially, to design public transportation route or road network of the new city, we can use the similar patterns in the trajectories of moving objects that move on the spatial network such as road and railway. In this paper, we propose a spatio-temporal similar trajectory search algorithm for moving objects on road network. For this, we define a spatio-temporal similarity measure based on the real road network distance and propose a grid-based index structure for similar trajectory search. Finally, we analyze the performance of the proposed similar trajectory search algorithm in order to show its efficiency.

**주요어** : 유사 궤적 검색, 이동 객체, 도로 네트워크 데이터베이스

**Keyword** : Similar trajectory search, moving objects, road network database

† 이 논문은 교육인적자원부, 산업자원부, 노동부의 출연금으로 수행한 최우수실험실지원사업의 연구결과임.

■ 논문접수 : 2008.1.9      ■ 심사완료 : 2008.3.4

\* 전북대학교 대학원 컴퓨터공학과 박사과정(yckim@dblab.chonbuk.ac.kr)

\*\* 교신저자 전북대학교 컴퓨터공학과 교수(jwchang@chonbuk.ac.kr)

## 1. 서론

최근 이동 단말기, 센서 네트워크, GPS 기술의 발전으로 대용량의 시공간 데이터의 수집이 가능하게 되었다. 그 결과로, 수집된 이동 객체의 궤적 데이터의 분석에 대한 관심이 증가하게 되었다[1, 2]. 이동 객체들은 도로, 철도와 같은 제한된 공간 네트워크 상을 이동하는데, 최근에는 이러한 이동 객체의 궤적 정보를 효율적으로 처리하여 적절한 서비스를 제공하기 위한 연구가 중요 이슈가 되고 있다[3, 4]. 예를 들어, 도시의 대중 교통 노선의 설계나 도로 계획과 같은 응용에서는 자주 정체되는 주요 도로 구간을 지난 궤적들의 데이터를 분석하여 이를 우회하는 노선을 추가하거나 우회 도로를 건설함으로써 교통량의 분산을 유도할 수 있다. 이를 위해서는 정체 도로 구간과 시간 및 공간상으로 유사한 궤적을 찾기 위한 시공간 유사도 측정 방법에 대한 정의가 필수적이다. 그러나 기존 유사도 측정 방법은 이동에 제약이 없는 유클리디언(Euclidean) 공간상에서의 연구가 대부분이며, 공간 네트워크 상에서의 시공간 유사도 측정 방법 연구는 거의 전무한 실정이다. 따라서 기존 유사검색 방법들[5-12]과 달리 공간 네트워크의 특징을 고려하여 유사 궤적을 검색하는 것이 필요하다. 기존의 방법들은 다음과 같은 단점을 가지고 있다. 첫째, 기존 연구들에서 가정한 유클리디언 거리 개념은 공간 네트워크상에서의 이동 객체의 특성을 반영하지 못한다. 궤적의 유사도는 궤적간의 거리에 매우 의존적이기 때문에, 공간 네트워크에 적합한 유사도 측정 방법과 이를 기반으로 하는 유사궤적 검색 방법이 정의되어야 한다. 둘째, 대부분의 기존 연구들은 궤적의 시공간 특성이 아닌 공간적 특성만을 고려한 연구가 대부분이다. 예를 들어, 다른 시간에 같은 공간을 지나는 궤적 또한 유사 궤적으로 간주한다. 하지만 이는 시공간 특성상 유사한 궤적이 아니다. 아울러, 이동 객체의 궤적의 움직임이 정반대일지라도 유사한 궤적으로 간주한다.

공간 네트워크 상에서 이동 객체 궤적의 유사성 비교를 위해, 본 논문에서는 새로운 유사도 측정방

법을 제안한다. 이를 위해, 첫째, 두 궤적 사이의 시공간 유사도 측정방법을 제안하고, 둘째, 효율적인 유사궤적 검색을 위한 그리드 기반 색인 기법을 제안한다. 마지막으로, 그리드 색인 기법을 이용한 새로운 시공간 유사 궤적 검색 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 관련 연구를 소개한다. 제 3장에서는 제안하는 시공간 유사도 측정 방법과 그리드 기반 색인구조, 새로운 유사 궤적 검색 알고리즘을 기술하고, 제 4장에서는 실험을 통한 성능 분석 결과를 제시한다. 마지막으로 제 5장에서는 결론 및 향후 연구를 제시한다.

## 2. 관련 연구

유사 궤적 검색을 위한 기존 연구는 크게 유클리디언 공간에서의 유사 궤적 검색 연구와 공간 네트워크 상에서의 유사 궤적 검색 연구로 나눌 수 있다. 첫째, 유클리디언 공간에 기반한 유사 궤적 검색 연구로는 [6, 8, 9, 10]이 존재한다.

먼저, [6]에서는 대용량의 이상값(outlier)를 포함하는 데이터에서 이동 객체의 시공간 궤적에 대한 분석 연구를 수행하였다. 이를 위해 LCSS(Longest Common Subsequence)에 기반한 비공간 거리 계산방법을 제안하였다. 하지만 유클리디언 거리에 기반한 연구이기 때문에, 공간 네트워크 상에서의 유사 궤적 검색에는 적합하지 않다. [8]에서는, 시간과 공간을 모두 고려한 궤적으로부터 각 객체의 이동 패턴을 추출하기 위한 연구를 수행하였다. 이를 위해 패턴 분석을 위한 “상점에서 고객이 이동한 경로와 유사한 형태의 궤적을 갖는 모든 객체를 검색하여라”와 같은 궤적기반 질의를 제안하고, 이를 사용하여 상점에서 궤적의 형태에 따라 고객들을 분류하였다. 또한, 모바일 데이터의 궤적을 표현하기 위한 데이터 모델과 시계열 데이터베이스에서 시간 유사도를 확장하여 궤적 사이의 거리에 기반한 질의를 제시하였다. 아울러, 효과적인 검색을 위한 색인 기법을 제안하였다. 하지만 이 방법 역시 유클리디언 거리에 기반한 방법으로 도로 네트워크에 적용할 수 없는 단점이 있다.

[9]의 연구는 시간을 고려하지 않은 이동 객체 궤적의 공간 형태에 초점을 맞추었다. 이들은 새로운 거리 개념인 OWD(one way distance)를 제안하고 이를 기반으로 단순하면서도 효과적으로 이동 객체의 궤적의 공간 형태를 비교하기 위한 방법을 제안하였다. 이를 이용하여 연속적인 궤적은 piece wise linear 방법을 통해, 비연속적인 경우는 그리드 표현을 통해 궤적사이의 OWD를 계산하여 효율적인 유사 궤적 검색 방법을 제안하였다. 그러나 궤적의 시간 특성을 고려하지 않았고, 유클리디언 공간상에서의 거리에 기반한 방법이기 때문에 실제로 도로네트워크에서 궤적사이의 거리를 계산하기에는 부적합하다. [10]의 연구에서는 분산 시공간 유사 검색 방법을 소개하였다. 주어진 궤적의 세그먼트가 다수의 노드에 분산 저장되었을 때, 사용자로부터 입력받은  $k$  개의 유사한 궤적을 검색하기 위해 LCSS에 기반한 UB-K (Upper Bound), UBLB-K (Upper Bound and Lower Bound)의 두 방법을 제안하였다. 제안된 방법은 시공간 특성을 모두 고려하였지만 유클리디언 공간을 가정하여 도로 네트워크 상에 적용할 수 없다는 단점이 있다. 이외에도 많은 유사 궤적 검색 연구들이 존재하지만 모두가 유클리디언 공간상에서 연구가 이루어져 도로 네트워크 상에 적용하기에는 부적합하다[7, 11, 5, 12].

도로 네트워크상에서 유사 궤적 검색을 위한 연구로는 [13, 14, 15]가 존재한다. [13]에서는 시공간 특성을 고려하여 도로 네트워크 상에서 유사 궤적 검색을 위해 두 궤적이 공통적으로 지나는 POI(Points of Interest) 사이의 실제 네트워크 거리 및 시간 거리 기반으로 유사 궤적 검색 알고리즘을 제안하였다. 아울러 불필요한 궤적 검색을 효과적으로 줄이기 위해, 첫째, 질의 궤적의 POI를 모두 포함하지 않는 궤적을 필터링한 후 공통 POI 사이의 시간 거리의 합 계산, 둘째, 질의 궤적과 같은 시간 범위를 갖는 궤적을 검색한 후 TOI(Time of Interest) 사이의 공간 거리의 합 계산, 그리고 시공간기반 필터링 후 두 궤적의 속력을 이용한 시공간 거리의 합 계산의 3가지 방법을 제시하였다. 하지만, 질의 궤적과 공통된 POI를 모두 포함한 궤

적만을 검색 대상으로 하기 때문에, 공간 거리상 매우 유사한 궤적일지라도 전체 POI를 포함하고 있지 않다면 검색대상에서 제외되어 정확한 유사 궤적 검색이 이루어지지 못하는 단점이 있다. [14]의 연구에서는 공간 유사도와 시간 유사도를 정의한 후 이를 결합한 시공간 유사도를 정의하였다. 또한 시그니처 기반 색인 구조를 통해 유사 궤적을 효과적으로 검색할 수 있는 알고리즘을 제안하였다. [14]에서는 먼저 질의 궤적의 모든 노드로부터 데이터 궤적의 가장 가까운 노드 쌍을 구한 후 이들의 평균 네트워크 거리로 공간 유사도를 정의하였다. 시간 유사도의 정의를 위하여 시간을 범위 거리, 날짜 거리, 주간 거리로 나누어 정의하고 이들의 합으로 시간 유사도를 정의하였다. 아울러 공간 유사도와 시간 유사도의 합 접근 방법과 곱 접근 방법을 이용하여 시공간 유사도를 정의하였다. 또한 효율적인 시공간 유사 궤적 검색을 위하여 시그니처 파일 기법을 이용한 유사 궤적 검색 알고리즘을 제안하였다. 그러나 질의 궤적과 적어도 하나 이상의 공유 노드를 포함한 데이터 궤적만을 검색 대상으로 하였기 때문에 공유 노드를 갖지 않는 유사한 데이터 궤적을 검색할 수 없다는 단점을 갖고 있다. 마지막으로, [15]의 연구는 시공간 특성을 모두 고려한 유사도 측정방법을 제안하고, 효과적인 유사 궤적 검색을 위한 M-tree를 이용한 유사 궤적 검색 알고리즘을 제안하였다. [15]에서는 공간 유사도를 궤적을 이루는 노드 사이의 최대 이동 거리 및 최소 이동 거리 비율로 정의하고, 시간 유사도를 두 궤적의 이동 속도의 최대 및 최소 속도 사이의 비율로 정의하였다. 또한 각 유사도에 대한 가중치를 이용하여 결합하고 이를 시공간 유사도로 정의하였다. 그러나 최대 및 최소 이동 거리 비율 기반 유사도는 양방향 통행이 가능한 일반적인 도로 네트워크 상에서는 대부분의 궤적이 모두 같은 공간 유사도를 갖게 되는 치명적인 단점이 있다. 아울러, 이동 객체의 속도에 기반한 시간 유사도는 비슷한 속력으로 이동한 이동 객체의 궤적을 유사한 시간 영역에서 이동한 이동 객체의 궤적보다 유사한 궤적으로 검색하는 단점이 존재한다.

이상에서 살펴본 바와 같이 기존 연구는 다음과 같은 단점을 가지고 있다. 첫째, 대부분의 기존연구 [5-12]는 공간상에서 객체 사이의 최단거리인 유클리디언 거리를 기반으로 하여 궤적 사이의 유사도를 측정하므로 공간 네트워크 상에서의 이동 객체 궤적의 공간 특성을 반영할 수 없다. 또한, 기존 공간 네트워크 상에서의 유사 궤적 검색 연구인 [13]은 질의궤적과 같은 노드를 지난 궤적만을 검색대상으로 하여, 다른 노드를 지난 궤적을 검색하지 못하며, [14]는 두 노드 사이의 최대 이동 비용 및 최소 이동 비용의 비율로 측정하여, 일방 통행이 아닌 도로를 지나는 궤적들은 모두 같은 공간 유사도를 갖게 된다. 둘째, 대부분의 기존 연구는 도로 네트워크 상에서 이동 객체 궤적의 공간 및 시간 특성 모두를 고려하지 않았다. [13]에서는 시공간 특성 모두를 고려하였지만 주어진 질의궤적과 정확히 일치하거나 질의궤적 전체를 포함하는 궤적만을 검색한다. 또한 [14]에서는 두 궤적의 이동 속도를 기반으로 유사도를 측정하여, 비슷한 시간영역에서 이동한 궤적 보다 다른 시간 영역에서 이동한 궤적을 유사한 궤적으로 검색한다.

### 3. 공간 네트워크에서 이동 객체 궤적을 위한 유사도 측정 방법

본 논문에서는 기존 관련 연구들의 단점을 개선하기 위해 다음 두 가지 사항을 고려하고자 한다. 첫째, 공간 네트워크상에서 시공간 특성을 모두 고려한 유사도 측정 방법을 설계하며, 둘째, 효율적인 유사 궤적 검색 알고리즘을 위한 그리드 기반 색인 기법을 설계하고, 마지막으로 시공간 유사 궤적 검색 알고리즘을 제안하고자 한다.

#### 3.1 시공간 유사도 측정 방법

일반적으로 공간 네트워크는 노드의 집합  $V$ , 에지의 집합  $E$ 로 표현되는 그래프  $G(V,E)$  로 표현된다. 따라서 공간 네트워크상에서 이동하는 이동 객체의 궤적  $T$ 는 다음과 같이 정의할 수 있다.

$$T = \{(v_1, t_1), (v_2, t_2), \dots, (v_n, t_n)\}$$

여기에서  $n$ 은 궤적의 길이를 나타내며,  $v$ 는 그래프로 표현되는 공간 네트워크의 노드를,  $t$ 는 이동 객체가 해당 노드에 위치한 시간을 나타낸다. 두 궤적 사이의 시공간 유사도 측정 방법을 설계하기 위하여, 첫째, 공간 네트워크 길이에 기반한 공간 유사도 측정 방법을 정의하고, 둘째, 시간 유사도 측정 방법을 정의하며, 마지막으로 이를 결합한 시공간 유사도 측정 방법을 정의한다. 먼저, 길이가 같은 두 궤적사이의 공간 유사도를 정의하고 이를 확장하여 다른 길이를 갖는 궤적의 공간 유사도로 확장한다.

시간 정보를 제외한 길이가  $m$ 인 두 궤적  $T_a, T_b$ 가 각각  $T_a=(v_{a1}, v_{a2}, \dots, v_{am}), T_b=(v_{b1}, v_{b2}, \dots, v_{bm})$ 이라 가정하자. 앞서 서론에서 언급한 대중 교통의 노선 설계나 도로 계획과 같은 응용에서는 질의궤적과 공간적으로 보다 가까운 거리에서 이동한 궤적을 유사한 궤적으로 검색할 수 있다. 공간 네트워크상에서는 이동 객체가 제한된 철도, 도로와 같은 네트워크를 따라서 이동하므로, 이들의 거리 또한 네트워크상의 거리로 측정하여야 한다. 따라서 두 궤적 사이의 공간 거리는 궤적을 구성하는 노드 사이의 거리는 최단 네트워크 거리로 정의할 수 있다. 그러나 단순히 두 노드 사이의 네트워크 최단 거리만을 계산하였을 경우, 이동 패턴이 유사한 같은 에지를 지나는 객체보다 이웃한 가까운 에지만을 지나는 이동 객체가 보다 유사한 궤적으로 계산될 수 있다. 이를 해결하기 위해 전체 궤적의 길이 중 공유되는 에지의 길이 비율이 클수록 보다 적은 평균 네트워크 거리를 갖도록 해야 한다. 따라서 두 궤적 사이의 공간 거리는 다음과 같이 정의 한다.

**정의 1.** 길이가  $m$ 인 두 궤적  $T_a, T_b$  사이의 공간 거리( $D_{sub}$ )는 두 궤적을 구성하는 노드 사이의 평균 네트워크 거리와 공유되는 에지의 길이의 비율의 곱으로 정의한다.

$$D_{sub}(T_a, T_b) = \frac{1}{m} \sum_{i=1}^m D_{network}(v_{a_i}, v_{b_i}) \cdot \left( \frac{1}{1 + \text{SharedEdgeRatio} * \alpha} \right)$$

이때,  $D_{network}$ 은 공간 네트워크 상에서 두 노드 사이의 네트워크 최단 거리를 나타낸다. 아울러 SharedEdgeRatio는 전체 궤적의 길이 중 공유되는 에지의 길이 비율을 나타내며,  $\alpha$ 는 SharedEdgeRatio에 대한 가중치를 나타낸다.

서로 길이가 다른 궤적 사이의 공간 유사도를 측정할 수 있도록 하기 위하여, 정의한 두 궤적 사이의 평균 네트워크 거리  $D_{sub}$ 를 이용하여 정의할 수 있다. 궤적을 일정한 크기의 부분 궤적으로 나누어 가장 작은  $D_{sub}$ 값을 갖는 부분 궤적 쌍을 구하고, 이들의 평균값으로 공간 유사도를 측정할 수 있다. 따라서 서로 다른 길이를 갖는 두 궤적 사이의 공간 유사도 SS(spatial similarity)는 다음과 같이 정의할 수 있다.

**정의 2.** 길이가 각각  $n, m$ 인 두 궤적  $T_a, T_b$  사이의 공간 유사도(SS)는 두 궤적의 가장 가까운 부분 궤적 쌍들의 공간 거리의 평균값으로 정의한다.

$$SS(T_a, T_b) = \frac{1}{n-m+1} \sum_{i=1}^{n-m+1} D_{sub}(T_{a_i}, T_{b_i})$$

이때,  $m$ 은 부분 궤적의 길이를 나타내며,  $1 < m \leq n$ 의 범위를 갖는다.

궤적의 시간 정보는 공간 네트워크 상에서 이동 객체의 이동 패턴을 나타내는 중요한 요소이다. 예를 들어, 도로 네트워크의 경우 아침과 저녁의 러시 아워(rush hour)에는 출퇴근을 위한 이동 패턴으로 교통 체증이 나타나며, 낮과 늦은 저녁 시간에는 이동 객체의 감소로 교통 체증 또한 줄어든다. 따라서 이러한 시간정보를 반영할 수 있도록 공간 유사도와 같은 방법으로 두 궤적사이의 시간 유사도를 정의할 수 있다. 대중 교통 노선 설계나 도로 계획과 같은 응용에서는 비슷한 시간에 근접한 도로를 이동한 이동 객체들의 궤적 정보가 중요하기 때문에, 두 궤적 사이의 시간 유사도는 각 궤적의 노드 사이의 평균 시간 거리로 측정할 수 있다. 따라서 두 궤적 시간 거리는 다음과 같이 정의할 수 있다.

**정의 3.** 공간 정보를 제외한 길이가  $m$ 인 두 궤적  $T_a, T_b$ 가 각각  $T_a=(t_{a1}, t_{a2}, \dots, t_{am}), T_b=(t_{b1}, t_{b2}, \dots, t_{bm})$ 이라 가정할 때, 두 궤적 사이의 시간 거리( $D_{time}$ )는 궤적을 구성하는 두 노드사이의 평균 시간 거리로 정의한다.

$$D_{time}(T_a, T_b) = \left| \frac{1}{m} \sum_{i=1}^m (t_{a_i} - t_{b_i}) \right|$$

서로 다른 길이를 갖는 두 궤적 사이의 시간 유사도 측정을 위해 공간 유사도와 같은 방법으로 확장할 수 있다. 즉, 서로 다른 길이를 갖는 두 궤적의 시간 유사도 TS(temporal similarity)는 다음과 같이 정의한다.

**정의 4.** 길이가 각각  $n, m$ 인 두 궤적  $T_a, T_b$  사이의 시간 유사도(TS)는 두 궤적의 가장 가까운 부분 궤적 쌍들의 시간 거리의 평균값으로 정의한다.

$$TS(T_a, T_b) = \frac{1}{n-m+1} \sum_{i=1}^{n-m+1} D_{time}(T_{a_i}, T_{b_i})$$

마지막으로 서로 다른 유사도인 공간 유사도와 시간 유사도를 결합하여 시공간 유사도를 정의한다. 응용에 따라 공간 유사도나 시간 유사도, 또는 두 유사도 모두 필요로 하거나 각각의 유사도에 대한 비중이 다를 수 있으므로, 각각에 대한 가중치인  $W_{net}$ 과  $W_{time}$ 을 이용하여 두 유사도를 결합한다. 즉, 두 궤적 사이의 시공간 유사도 STS (spatio-temporal similarity)는 다음과 같이 정의할 수 있다.

**정의 5.** 두 궤적  $T_a, T_b$  사이의 시공간 유사도는 두 궤적의 공간 유사도와 시간 유사도의 합으로 정의한다. 이때,  $W_{net}$ 과  $W_{time}$ 은 각각 공간 유사도와 시간 유사도를 위한 가중치이다.

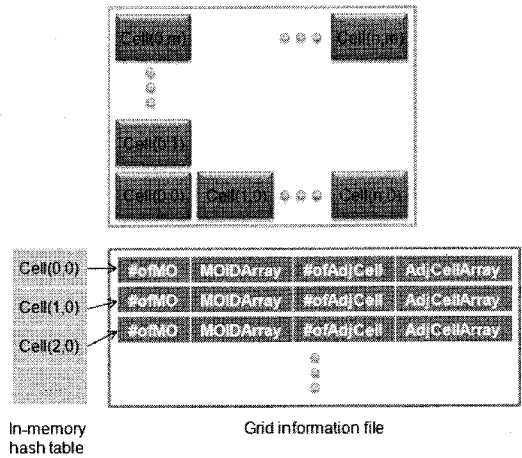
$$STS(T_a, T_b) = W_{net} * SS(T_a, T_b) + W_{time} * TS(T_a, T_b)$$

이때,  $W_{net} + W_{time} = 1$ 이다.

### 3.2 색인 구조

본 절에서는 앞서 정의한 시공간 유사도 계산을 기반으로 효율적인 유사 궤적 검색을 위한 그리드 기반 색인 구조를 설계한다. 일반적으로 많이 쓰이는 R-tree 기반 색인 기법들의 경우, 궤적의 길이가 길어지거나, 주요 도로와 같은 많은 이동 객체의 이동이 잦은 네트워크의 경우 잦은 삽입과 갱신에 따른 분할과 결합에 많은 비용이 소요되며, 궤적이 구성하는 MBR 사이의 중첩 영역이 증가하여 효율적인 관리가 어렵다. 또한, 궤적 사이의 시공간 유사도를 측정할 경우, 공간 유사도에서의 두 노드 간 최단 거리 계산은 많은 비용을 소모하게 되므로, 이러한 거리 계산을 효과적으로 줄일 수 있는 색인 구조 및 가지치기 방법이 필수적이다. 이를 위해 본 논문에서는 그리드 기반 색인 구조를 제안한다. 그리드 색인 구조의 전체적인 구조는 <그림 1>과 같다. 제안하는 그리드 색인 구조는 전체 공간 네트워크를 일정한 크기의 셀들로 나누고 셀 영역에 포함되는 공간 네트워크 정보를 Grid information 파일에 저장한다. 각 그리드 셀은 해당 셀을 지나간 궤적의 수와 궤적 ID정보를 저장하며, 그리드 셀 영역 내에 존재하는 노드의 인접 셀의 수와 인접 셀의 인덱스 정보를 관리한다. 그리드 색인 구조는 공간 네트워크의 노드와 에지의 정보를 이용하여 다음과 같이 생성한다. 첫째, 공간 네트워크의 노드의 좌표 정보를 이용하여 이를 포함하는 셀의 인덱스를 계산한다. 둘째, 노드의 인접 노드를 검색하여 인접 노드가 다른 셀 영역에 존재하면 #ofAdjCell 값을 증가시키고, AdjCellArray에 인접 셀의 인덱스 정보를 추가한다. 마지막으로, 위 두 과정을 네트워크의 모든 노드에 대하여 수행하여 그리드 색인 구조를 생성한다. 새로운 궤적 데이터의 삽입할 경우, 첫째, 해당 궤적이 지나가는 노드 정보를 이용하여 노드가 포함된 그리드 셀의 인덱스를 계산한다. 둘째, 궤적이 지나가는 모든 그리드 셀의 #ofMO 값을 증가시키고, MOIDArray에 삽입하고자 하는

궤적의 ID를 추가한다. 아울러 디스크에 저장된 셀 정보의 빠른 접근을 위해 셀의 페이지 정보를 메모리상에 hash table로 유지한다.

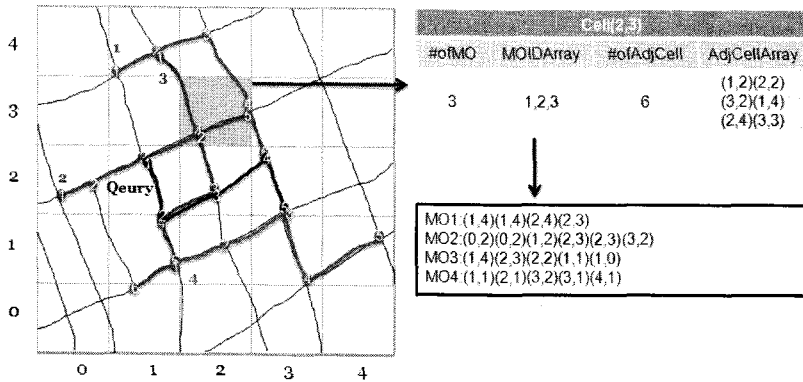


<그림 1> 그리드 색인 구조의 전체적인 구조

다음 <그림 2>는 그리드 색인 구조를 사용한 예를 나타낸다. 그림은 공간 네트워크가 5X5의 고정 크기 셀들로 이루어진 그리드를 나타낸다. 그림에서 나타난 바와 같이 이동 객체 MO1의 경우 (1,4), (1,4), (2,4), (2,3)의 셀 리스트로 표현되며, MO2의 경우 셀 (0,2), (0,2), (1,2), (2,3), (2,3), (3,2)의 셀 리스트로 표현됨을 알 수 있다. 또한, 그리드 셀 (2,3)의 경우 이동 객체 MO1, MO2, MO3가 해당 셀에 포함되는 노드를 지나가기 때문에 이들의 개수인 3과 이들의 식별자 1, 2, 3을 저장한다. 이때, 셀(2,3)은 (1,2), (2,2), (3,2), (1,4), (2,4), (3,3)의 6개의 인접 셀을 갖음을 알 수 있다.

### 3.3 시공간 유사 궤적 검색 알고리즘

제안하는 시공간 유사도 측정방법 및 그리드 색인 구조를 이용하여, 질의 궤적과 가장 가까운 k 개의 유사 궤적을 검색하는 알고리즘을 설계한다. 그리드 기반 색인 구조를 사용하여 효과적인 가지치기를 수행하기 위해, 이동 객체 궤적의 평균 셀 거리를 이용한다. 두 노드를 포함하는 그리드의 각 셀



<그림 2> 그리드 색인 구조의 예

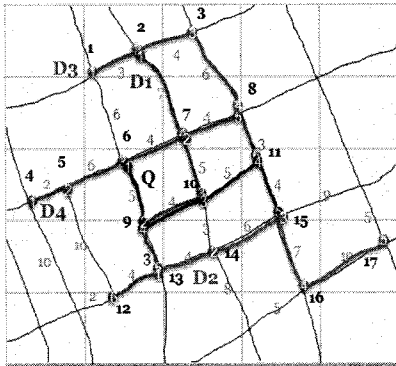
의 최소 거리는 이들이 포함하는 노드들 사이의 최소 네트워크 거리보다 항상 크거나 같기 때문에 검색된 이동 객체의 후보 중에서 가장 큰 시공간 유사도보다 큰 평균 셀 거리를 갖는 이동 객체 궤적은 실제 시공간 유사도를 계산할 필요없이 가지치기가 가능하다. 따라서 제안하는 시공간 유사 궤적 검색 알고리즘은, 첫째, 질의 궤적의 노드를 포함하는 그리드 셀 리스트를 계산하고, 둘째, 리스트에 포함된 각 셀을 지나는 이동 객체 궤적들을 후보 집합에 삽입한다. 이때, 이때, 검색된 궤적의 수가 k 보다 작을 경우 궤적의 수가 k 개가 될 때까지 셀 리스트의 인접 셀들을 차례로 확장하여 확장된

셀을 지나는 궤적을 검색한다. 셋째, 후보 집합에 포함된 이동 객체 궤적들을 데이터 파일로부터 검색한 후, 검색된 후보 궤적들과 질의 궤적 사이의 실제 시공간 유사도를 측정한다. 넷째, 질의 궤적이 지나는 각 셀로부터 k 번째 시공간 유사도를 갖는 궤적의 시공간 거리에 포함되는 모든 궤적들의 평균 그리드 셀 거리를 측정한다. 이때, 평균 그리드 셀 거리가 k 번째 유사 궤적의 시공간 거리보다 큰 궤적은 후보 집합에서 삭제하고, 그렇지 않은 경우 실제 시공간 유사도를 측정하여 결과 집합에 삽입하고 k 번째 시공간 유사 거리를 갱신한다. <그림 3>은 시공간 유사 궤적 검색 알고리즘을 나타낸다.

```

STsimilar Trajectory Search Algorithm ( Qtraj, k, α, Wnet, Wtime )
{
    1. get grid cells from Qtraj
    2. creates candidates and count #ofCand
    2. while( #ofCand < k ) {
        expand each grid cells
        updates candidates and increase #ofCand }
    3. Read MOs in candidates
    4. calculates STsimilarity and insert into result
    5. for every Qtraj's node
        get MO ID with in range k'th STsimilarity
    6. for every MO ID retrieved in step 5
        if (k-th STsimilarity > gridDistance(Q, MO ID))
            calculates STsimilarity and update result
    7. return result
}
    
```

<그림 3> 시공간 유사 궤적 검색 알고리즘



- D<sub>1</sub>: <(2,10),(7,18),(10,22),(9,26),(13,30),(12,35)>
- D<sub>2</sub>: <(13,40),(14,46),(15,53),(16,61),(17,75)>
- D<sub>3</sub>: <(1,15),(2,20),(3,25),(8,36)>
- D<sub>4</sub>: <(4,22),(5,26),(6,33),(7,38),(8,44),(11,48)>
- Q : <(6,13),(9,18),(10,23),(11,29),(15,33)>

<그림 4> 시공간 유사 궤적 검색의 예

예를 들어, <그림 4>와 같은 질의 궤적 Q와 데이터 궤적들이 주어지고, k는 2, 부분 궤적의 길이는 4라 가정하자. 그림에서 나타난바와 같이 데이터 궤적 D1의 경우 노드 2, 7, 10, 9, 13, 12를 지나며, 해당 노드에서의 타임스탬프는 각각 10, 18, 22, 26, 30, 35이다. 유사 궤적 검색 알고리즘은 먼저 질의 궤적이 지나간 그리드 셀의 리스트를 생성하고, 각 셀을 지나간 데이터 궤적들을 후보 궤적 대상으로 한다. 따라서 질의 궤적이 지나간 그리드 셀에 포함된 데이터 궤적 D1, D2, D4를 검색한다. 둘째, 검색된 후보 궤적이 k=2 보다 크기 때문에 그리드 셀을 확장하지 않고 데이터 파일로부터 후보 궤적들을 검색한다. 셋째, 질의 궤적과 후보 궤적사이의 시공간 유사도를 측정한다. 데이터 궤적 D1의 경우는 길이가 4인 부분 궤적 3개로 구성되며 질의 궤적은 2개의 부분 궤적으로 구성된다. 질의 궤적의 첫 번째 부분 궤적과 D1의 부분 궤적들 사이의 공간 거리는 각각  $D_{sub}(Q_{sub1}, D_{sub1}) = (9+9+0+9)/4=6.75$ ,  $D_{sub}(Q_{sub1}, D_{sub2}) = (4+4+4+12)/4=6$ ,  $D_{sub}(Q_{sub1}, D_{sub3}) = (9+0+7+16)/4=8$  이고 시간 거리는 각각  $D_{time}(Q_{sub1}, D_{sub1}) = (3+0+1+3)/4=1.75$ ,  $D_{time}(Q_{sub1}, D_{sub2}) = (5+4+3+1)/4=3.25$ ,  $D_{time}(Q_{sub1}, D_{sub3}) = (9+8+7+6)/4=7.5$  으로 계산된다. 따라서 질의 궤적의 첫 번째 부분 궤적과 데이터 궤적 D1의 부분 궤적들 사이의 시공간 유사도는  $W_{net}$ 과  $W_{time}$ 을 0.5라 가정했을 때 각각  $STS(Q_{sub1}, D_{sub1}) = 0.5*6.75 + 0.5*1.75 = 4.25$ ,

$STS(Q_{sub1}, D_{sub2}) = 0.5*6 + 0.5*3.25 = 4.625$ ,  $STS(Q_{sub1}, D_{sub3}) = 0.5*8 + 0.5*7.5 = 7.75$  가 되므로 데이터 궤적의 첫 번째 부분 궤적이 가장 작은 거리에 있음을 알 수 있다. 마찬가지로 질의 궤적의 두 번째 부분 궤적은 데이터 궤적의 두 번째 부분 궤적과 가장 작은 거리에 있으므로 질의 궤적과 데이터 궤적 D1의 시공간 유사도  $STS(Q, D1) = 0.5*6.875 + 0.5*1.75 = 4.3125$  이다. 마찬가지로 방법으로 질의 궤적과 데이터 궤적 D2, D4사이의 시공간 유사도는  $STS(Q, D2) = 16.625$ ,  $STS(Q, D4) = 8.5625$ 로 계산할 수 있다. 넷째, 질의 궤적이 지나간 각 셀에서 k 번째 유사 궤적인 D2의 시공간 유사도를 이용하여 범위내에 존재하는 셀을 검색하여 리스트를 구축하고, 구축된 셀 리스트를 지나간 데이터 궤적을 검색한다. 검색된 데이터 궤적의 셀과 질의 궤적 사이의 셀 기반 유사도를 측정 한 후, k 번째 시공간 유사도보다 작을 경우 실제 시공간 유사도를 측정하여 결과 셋을 갱신한 후 결과 셋인 D1과 D2를 반환한다.

#### 4. 성능평가

제안하는 시공간 유사 궤적 검색 알고리즘의 성능 분석을 위하여 HP ML 150 G3 서버, 인텔 제온 3.0 GHz dual CPU, 2 GB 메모리, 250GB HDD, Windows 2003 서버 환경에서 Visual Studio .Net 을 이용하여 구현하였다. 성능 분석을



이동 객체 궤적	질의 궤적
4 124993552 23327 23328 2005.2.10_THU_16:37 2005.2.10_THU_16:43	4 124993552 23327 23328 2005.2.10_THU_16:15 2005.2.10_THU_16:22
4 124993408 23328 23323 2005.2.10_THU_16:43 2005.2.10_THU_16:56	4 124993408 23328 23323 2005.2.10_THU_16:22 2005.2.10_THU_16:33
4 124993407 23323 23322 2005.2.10_THU_16:56 2005.2.10_THU_16:59	4 124993407 23323 23322 2005.2.10_THU_16:33 2005.2.10_THU_16:38
4 124993402 23322 23316 2005.2.10_THU_16:59 2005.2.10_THU_17:7	4 124993403 23322 23321 2005.2.10_THU_16:38 2005.2.10_THU_16:44
4 124993400 23316 23315 2005.2.10_THU_17:7 2005.2.10_THU_17:11	4 124993397 23321 23319 2005.2.10_THU_16:44 2005.2.10_THU_16:53
4 124993482 23315 23744 2005.2.10_THU_17:11 2005.2.10_THU_17:21	4 124993371 23319 23318 2005.2.10_THU_16:53 2005.2.10_THU_16:59
4 124993470 23744 23745 2005.2.10_THU_17:21 2005.2.10_THU_17:27	4 124993393 23318 23317 2005.2.10_THU_16:59 2005.2.10_THU_17:7
4 124993466 23745 23771 2005.2.10_THU_17:27 2005.2.10_THU_17:34	4 124993401 23317 23314 2005.2.10_THU_17:7 2005.2.10_THU_17:12
4 124993465 23771 23288 2005.2.10_THU_17:34 2005.2.10_THU_17:38	4 124993392 23314 23313 2005.2.10_THU_17:12 2005.2.10_THU_17:21
4 124993463 23288 23772 2005.2.10_THU_17:38 2005.2.10_THU_17:44	4 124993469 23313 23745 2005.2.10_THU_17:21 2005.2.10_THU_17:27
4 124993433 23772 23780 2005.2.10_THU_17:44 2005.2.10_THU_17:59	4 124993466 23745 23771 2005.2.10_THU_17:27 2005.2.10_THU_17:31
	4 124993465 23771 23288 2005.2.10_THU_17:31 2005.2.10_THU_17:39
	4 124993463 23288 23772 2005.2.10_THU_17:39 2005.2.10_THU_17:48
	4 124993433 23772 23780 2005.2.10_THU_17:48 2005.2.10_THU_17:57

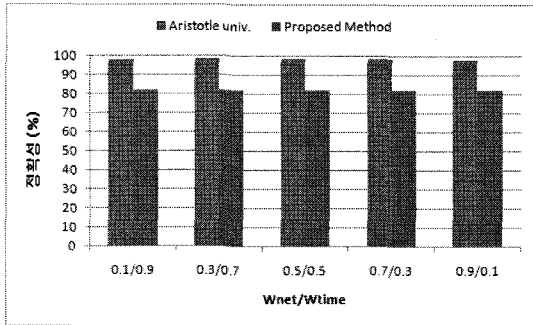
〈그림 5〉 생성된 질의 궤적의 예제

위해 사용된 데이터는 17만개의 노드와 22만개의 에지로 구성된 샌프란시스코 만의 도로 네트워크 데이터이며, Brinkhoff[16] 알고리즘을 사용하여 1000, 10000 개의 이동 객체 궤적을 생성 하였다. 유사 궤적 검색을 위한 그리드 색인 구조의 생성을 위해 그리드의 셀 크기는 사용된 네트워크 데이터의 평균 에지 길이인 1500으로 설정하였으며, 1208X946 개의 셀로 구성하였다. 질의 궤적은 데이터의 궤적 중에서 랜덤으로 1000개의 궤적을 추출하여 궤적의 10%~50%의 세그먼트를 대체하여 새로운 궤적을 생성하였다. 검색 성능 평가를 위해 데이터 궤적 생성과 동일한 방법으로 10~100개의 다양한 크기를 갖는 1,000개의 질의 궤적을 생성 하였다. 〈그림 5〉는 생성된 실제 질의 궤적의 예를 나타낸다. 〈그림 5〉에 나타난바와 같이 이동 객체 궤적을 변형하여 질의 궤적을 생성하였으며 표시된 부분은 대체된 구간을 나타낸다.

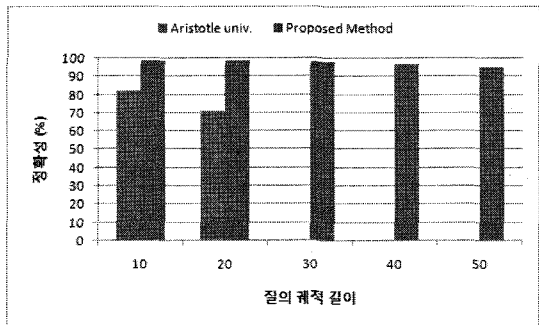
성능 분석은 첫째, 제안하는 방법의 정확성을 증명하기 위해 추출된 질의 궤적에 대해 검색된 데이터의 비교를 수행하고, 둘째, 공간 네트워크 환경에서의 유사 궤적 검색 연구인 [15]의 방법과 유사 궤적 검색 성능을 비교한다. [13, 14]는 질의 궤적과 일치하거나 적어도 하나의 같은 노드를 지난 데이터 궤적만을 검색 대상으로 하여 정확한 유사 궤적 검색이 이루어지지 않기 때문에 성능 평가 대상에서 제외하였다.

#### 4.1 정확성 평가

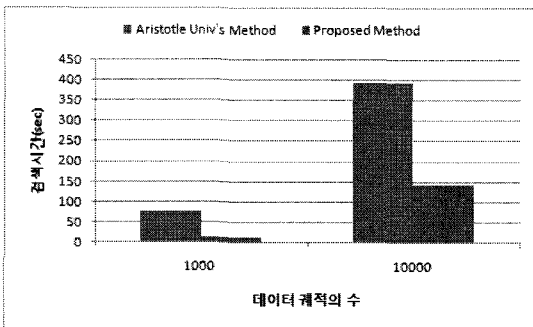
먼저, 제안하는 방법의 정확성을 증명하기 위해 데이터 궤적의 크기 및 세그먼트를 10%~50% 변형한 질의 궤적에 대한 검색 결과를 기존 연구 [15](이하 Aristotle 대학 방법)과 비교하였다. 정확성 비교는 변형된 질의 궤적에 대한 유사 궤적 검색 결과 변형하기 전의 원본 데이터 궤적이 검색되는 지를 평가하였다. 〈그림 6〉은 Wnet과 Wtime의 변화에 따른 유사 궤적 검색 정확성 비교를 나타낸다. 이때 질의 궤적의 길이는 부분 궤적의 길이와 같은 10으로 정하였다. 제안하는 방법과 기존 방법 모두 공간 유사도와 시간 유사도에 대한 가중치를 이용하여 시공간 유사도를 측정하기 때문에, 각 유사도에 대한 가중치 비율  $W_{net}/W_{time}$ 을 0.1/0.9, 0.3/0.7, 0.5/0.5, 0.7/0.3, 0.9/0.1로 변경하며 정확성을 측정하였다. 〈그림 6〉에 나타난바와 같이  $W_{net}/W_{time}$ 의 비율에 상관없이 유사 궤적 검색의 정확도가 일정함을 알 수 있다. 아울러 제안하는 방법은 약 99%, 기존 연구의 경우 약 82%를 나타내어 제안하는 방법이 보다 정확함을 알 수 있다. 이는 기존 연구의 공간 유사도가 양방향 공간 네트워크에서는 대부분 같은 유사도로 계산되어 유사 궤적 검색에서 궤적 사이의 유사도가 제대로 반영되지 못하기 때문이다. 〈그림 7〉은 질의 궤적 길이의 변경에 따른 유사 궤적 검색 정확성 비교를 나타낸다. 기존 연구의 경우 질의 궤적의 길이가 30 이상으로 증가하면 검색 시간이 급격히



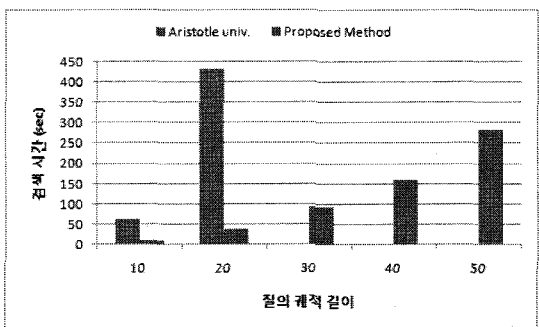
<그림 6> Wnet/Wtime 변화에 따른 유사 궤적 검색 정확성 비교



<그림 7> 질의 궤적 길이의 변화에 따른 유사 궤적 검색 정확성 비교



<그림 8> 데이터 궤적 수의 변화에 따른 시공간 유사 궤적 검색 시간 비교



<그림 9> 질의 궤적 길이의 변화에 따른 시공간 유사 궤적 검색 시간 비교

증가하여 측정할 수 없었다. 그림에 나타난바와 같이 기존 연구의 경우 궤적의 길이가 20일 때의 정확도는 약 71%를 나타내어 정확도가 급격히 감소하는 반면, 제안하는 방법의 경우 질의 궤적의 길이가 증가하여도 95% 이상의 정확도를 나타내고 있다. 따라서 제안하는 방법이 보다 정확함을 알 수 있다.

#### 4.2 검색 성능 평가

본 절에서는 제안하는 접근 방법과 기존 연구인 Aristotle 대학 방법과의 성능 비교를 수행하였다. 기존 연구와의 성능 비교를 위해 k 를 10, Wnet/Wtime을 0.5/0.5로 하여 가장 유사한 10개의 궤적을 검색한 시간을 측정하였다. <그림 7>은

데이터 궤적의 수가 각각 1000, 10000일 경우의 검색 시간을 나타낸다. 데이터 궤적의 수가 1000개인 경우 제안하는 방법은 약 10초대 반해, [14]의 방법의 경우 약 74초이며, 데이터 궤적의 수 10000개인 경우 제안하는 방법은 약 142초, Aristotle 대학 방법은 395초로 제안하는 방법이 매우 우수함을 나타낸다. 이는 제안하는 방법의 그리드 셀 거리를 통해 효과적인 가지치기를 하는 반면, Aristotle 대학 방법의 경우, 데이터 궤적을 정해진 크기의 부분 궤적으로 나누어 저장하기 때문에 상대적으로 많은 수의 부분 궤적의 수로 인해 성능 차이가 발생하였다. <그림 8>은 질의 궤적 길이의 변경에 따른 검색 시간을 나타낸다. [14]의 방법의 경우 질의 궤적의 길이가 증가함에 따라 검색 시간이 급격히 증가하여 궤적의 길이가 30이상

일 경우는 측정할 수 없었다. 반면 제안하는 방법의 경우 질의 궤적의 길이가 증가하여도 검색 성능이 완만히 감소함을 나타내어 제안하는 방법이 보다 우수함을 알 수 있다.

## 5. 결 론

유클리디언 공간 상에 존재하는 이동 객체의 유사 궤적 검색에 관련된 연구는 비교적 활발하게 수행된 반면, 공간 네트워크 상에서의 이동 객체 궤적의 시간 및 공간 특성 모듈을 고려한 연구는 거의 이루어지지 않았다. 이를 위하여, 본 논문에서는 공간 네트워크에서의 유사 궤적 검색을 위하여 유클리디언 거리 기반이 아닌 네트워크 거리에 기반한 새로운 시공간 유사도 측정 방법을 정의하였다. 또한, 효율적인 유사 궤적 검색을 위한 그리드 기반 색인 구조를 설계하고 마지막으로, 시공간 유사 궤적 검색 알고리즘을 제안하였다. 아울러, 제안하는 방법의 효율성을 입증하기 위해 기존 연구와의 성능 평가를 통해 제안하는 방법이 보다 우수함을 증명하였다.

향후 연구로는 도로 네트워크 상의 자동차의 움직임과 같은 실제 이동 객체의 궤적 데이터를 이용하여 제안하는 알고리즘의 성능을 입증하고, 이를 기반으로 데이터 마이닝 연구를 통해 대중 교통 노선 설계나 도시의 도로 네트워크 설계와 같은 응용에 적용하는 것이다.

## 참고문헌

1. Barbara, D., "Mobile computing and databases—a survey," IEEE TKDE, Vol.11 No.1, 1999, pp. 108-117.
2. Roddick, J.F., and Hornsby, K., "An Updated Bibliography of Temporal, spatial and spatio-temporal data mining Research," TSDM, 2000, pp. 147-164.
3. Pfoser, D., Jensen, C.S., and Theodoridis, Y., "Novel Approaches in Query Processing for Moving Object Trajectories," In Proc. of the 26th International Conference on Very Large Data Bases, 2000, pp. 395-406.
4. 윤대섭, 이수철, 권오천, 박종현, "차량기반 고객관계 고나리 기술과 상용차 텔레매틱스 기술의 응용서비스 동향", 전자통신동향분석, 21권, 3호, 2006, pp.109-116.
5. Vlachos, M., Kollios, G. and Gunopulos, D., "Discovering Similar Multidimensional Trajectories," In Proc. of the 18th International Conference on Data Engineering, 2002, pp. 673-684.
6. Vlachos, M., Gunopulos, D., and Kollios, G., "Robust Similarity Measures for Mobile Object Trajectories," In Proc. of the 13th International Workshop on Database and Expert Systems Applications, 2002, pp. 721-728.
7. Shim, C.B., and Chang, J.W., "Similar Sub-Trajectory Retrieval for Moving Objects in Spatio-temporal Databases," In Proc. of The 7th East European Conference on Advances In Databases and Informations Systems, 2003, pp. 308-322.
8. Yanagisawa, Y., Akahani, J.I., and Satoh, T., "Shape-Based Similarity Query for Trajectory of Mobile Objects," In Proc. of the 4th International Conference on Mobile Data Management, 2003, pp. 63-77.
9. Lin, B., and Su, J., "Shapes Based Trajectory Queries for Moving Objects," In Proc. of 13th ACM GIS, 2005, pp. 21-30.
10. Zeinalipour-Yazti, D., Lin, S., and Gunopulos, D., "Distributed Spatio-Temporal Similarity Search," In Proc. of

- 15th ACM CIKM, 2006, pp. 14-23.
11. Sakurai, Y., Yoshikawa, M., and Faloutsos, C., "FTW: Fast similarity search under the time warping distance," In Proc. PODS, 2005, pp. 326-337.
  12. Chen, L., Ozsu, M.T., and Oria, V., "Robust and fast similarity search for moving object trajectories," In Proc. ACM SIGMOD, 2005, pp. 491-502.
  13. Hwang, J.R., Kang, H.Y., and Li, K.J., "Spatio-temporal Similarity Analysis Between Trajectories on Road Networks," ER Workshops, 2005, pp. 280-289.
  14. 김영창, 라빈드라 비스타, 장재우, "도로 네트워크에서 이동 객체를 위한 시공간 유사 궤적 검색 알고리즘", 한국공간정보시스템학회 논문지, 제9권 제1호, 2007, pp. 59-77.
  15. E. Tiakas, A. N. Papadopoulos, A. Nanopoulos, Y. Manolopoulos, D. Stojanovic, S. Djordjevic-Kajan, "Trajectory Similarity Search in Spatial Networks", IDEAS 2006, pp. 185-192.
  16. Brinkhoff, T., "A Framework for Generating Network-Based Moving Objects," GeoInformatica, Vol.6 No.2, 2002, pp. 153-180.

**김영창**

2001년 전북대학교 컴퓨터공학과(공학사)  
2003년 전북대학교 대학원 컴퓨터공학과(공학석사)  
2004년~현재 전북대학교 대학원 컴퓨터공학과(박사과정)  
관심분야 : 데이터 마이닝, 공간 데이터베이스, 공간 색인 구조, 질의처리 알고리즘

**장재우**

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 컴퓨터공학과 교수  
관심분야 : 공간 네트워크 데이터베이스, 위치 기반 서비스, 상황인식, 하부저장구조