

도로 네트워크 상의 이동 객체 궤적의 간략화

Simplification of Moving Object Trajectory on Road Networks

황정래* / Jung-Rae Hwang, 강혜영** / Hye-Young Kang, 이기준*** / Ki-Joune Li

요약

도로 네트워크 상의 이동 객체 궤적을 분석하기 위해서는 그것의 표현이 명확하게 정의되어야 한다. 도로 네트워크 상의 이동 객체 궤적을 표현하는 기존의 대부분의 방법들은 이동 객체 궤적을 그 궤적이 통과한 위치와 시간의 집합으로 복잡하게 표현한다. 이것은 이동 객체 궤적의 검색 등과 같은 분석을 처리할 시 많은 시간이 요구된다. 본 논문에서는 도로 네트워크 상의 관심 있는 지점(POI: Points of Interest)들에 초점을 두어 도로 네트워크 상의 이동 객체 궤적을 효율적으로 간략화 하는 방법을 제시한다. 본 논문에서 제시하는 방법은 도로 네트워크 상의 이동 객체 궤적이 통과한 POI의 수를 줄임으로써 궤적을 간략화 하며, 궤적이 간략화 된 후에도 그 궤적의 형태를 그대로 유지하도록 한다.

Abstract

In order to analyze moving object trajectories on road networks, its representation needs to be defined correctly. The most previous methods representing moving object trajectories on road networks defined moving object trajectories as a set of passed location and its time. It is required much time in processing analysis such as retrieval for moving object trajectories. In this paper, we focus on POI(Points of Interest) on road networks and propose methods simplifying moving object trajectories based on it. Our method simplifies moving object trajectories by reducing the number of POIs that moving object trajectories passed and maintains its form after moving object trajectories were simplified.

주요어 : 이동 객체 궤적, 도로 네트워크 공간, POI, 간략화

Keyword : Moving Object Trajectories, Road Network Space, POI, Simplification

1. 서론

도로 네트워크 상에는 수많은 이동 객체들이 존재한다. 이러한 도로 네트워크 상의 이동 객체 궤적들을 분석하기 위해서는 그 궤적들을 정확하게 표현하는 것이 필요하다. 일반적으로, 도로 네트워크

상의 이러한 이동 객체들은 (x, y, t) 공간에서 위치와 그 위치를 통과한 시간의 집합으로 된 궤적으로 표현된다. 대부분의 궤적은 수많은 지점들을 통과하기 때문에 매우 많은 위치 정보로 표현된다. 심지어, 도로 네트워크 상의 이동 객체 궤적은 GPS에 의하여 획득된 데이터이기 때문에 표현된 궤적이

■ 논문접수 : 2007.10.17 ■ 심사완료 : 2007.11.26

* 교신저자 부산대학교 컴퓨터 및 정보통신 연구소 기금교수(jrhwang@pusan.ac.kr)

** 부산대학교 대학원 컴퓨터공학과 박사과정(hykang@isel.cs.pusan.ac.kr)

*** 부산대학교 정보컴퓨터공학과 교수(lik@pnu.edu)

통과한 위치는 도로 네트워크 상에서 의미 없는 위치 정보일지도 모른다. 이러한 궤적들을 가지고 유사한 궤적을 검색하는 등의 분석을 처리할 때 많은 시간이 요구되는 것은 당연하다. 만약 이러한 이동 객체 궤적을 간략화 시킬 수 있다면, 궤적 데이터를 이용한 질의 처리 시간이 감소하게 될 것이다. 예를 들어, 하나의 이동 객체 궤적이 20개의 위치 정보로 표현되었다고 가정하자. 이때, 이 궤적이 본래의 형태를 유지하면서 10개의 위치 정보를 가진 궤적으로 간략화가 가능하다면, 이것은 이동 객체 궤적 데이터를 분석할 때 매우 의미 있는 방법일 것이다. 본 논문에서는 도로 네트워크 상의 이동 객체 궤적을 실질적으로 의미를 가지는 지점이나 관심 있는 지점(POI: Points of Interest)으로 표현한다. 예를 들어, 교차로, 학교, 시청 등과 같은 공공기관 등이 POI가 될 수 있다. 특히, 교차로는 도로 네트워크 상의 매우 중요한 위치 정보를 가지는 POI가 될 수 있다. 따라서, 본 논문에서는 도로 네트워크 상의 이동 객체 궤적이 통과한 지점(POI)들을 줄임으로써 그 궤적을 간략화 하는 방법을 제시한다. 본 논문의 나머지 구성은 다음과 같다. 2 장에서는 본 논문과 관련된 연구들과 동기를 소개한다. 3 장에서는 도로 네트워크 상의 이동 객체 궤적을 표현하는 방법에 대하여 살펴보고, 4 장에서는 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법을 제시한다. 5 장에서는 본 논문에서 제시한 방법의 효율성을 실험을 통하여 보여주고, 6 장에서는 결론을 맺는다.

2. 관련 연구 및 동기

본 장에서는 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법과 관련된 연구들을 소개하고, 본 논문의 연구 동기를 제시한다.

2.1 관련 연구

최근에 도로 네트워크 상의 이동 객체 궤적을 분석하는 연구가 활발하게 진행되고 있다. 도로 네트

워크 상의 이동 객체 궤적을 표현하거나 검색하는 것과 관련된 주제는 대표적인 연구이다. 본 절에서는 본 논문에서 제시하는 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법과 관련된 연구들을 다음과 같이 정리하여 살펴보도록 한다.

- 도로 네트워크 상의 이동 객체 궤적의 표현
- 도로 네트워크 상의 이동 객체 궤적의 검색
- 이동 객체 궤적의 간략화

2.1.1 도로 네트워크 상의 이동 객체 궤적의 표현

도로 네트워크 공간은 제한된 네트워크를 기반으로 하기 때문에 도로 네트워크 공간에 존재하는 이동 객체들은 유클리디언 공간에 존재하는 이동 객체들과는 다르게 표현된다. 이러한 궤적의 표현은 이동 객체들의 궤적을 분석하기 위하여 가장 선행되어야 하는 연구이다. 따라서, 제한된 공간 네트워크 상의 이동 객체들의 궤적을 표현하는 연구가 많이 진행되고 있다[1] [2] [3] [4].

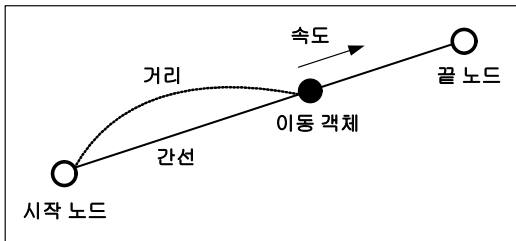
도로 네트워크 상의 이동 객체들의 표현과 질의를 위한 간결한 모델이 [3]에서 정의되었다. 여기서는 이동 객체 O 의 궤적 $\pi(O)$ 을 $[i, (x, y), t_i, b]$ 의 집합으로 표현하고 있다. 여기서, (x, y) 는 이동 객체 O 의 i 번째 중간점이고, t_i 는 그 곳의 시간이다. 그리고, b 는 부울 속성이다. 이 모델은 이동 객체들을 시공간적 측면으로 명확히 표현되었다는 점에서 의미가 크다.

도로 네트워크 정보와 속도 그래프를 이용하여 궤적을 표현하는 방법이 제시되었다[1]. [1]에서는 이동 객체의 궤적을 $\langle objid, x, y, t \rangle$ 의 집합으로 표현하고 있다. (x, y) 는 시간 t 에서의 위치이다. 이와 더불어, 도로 네트워크를 $(roadid, from, to, maxspeed)$ 로 간결하게 나타내었다. 여기서, $from$ 과 to 는 시작과 끝 노드를 나타내고, $maxspeed$ 는 그 도로 구간의 최대 허용 속도이다. 따라서, 이동 객체의 궤적에 이동 객체의 기본 속도 그래프를 적용시킴으로써 좀더 실질적인 궤적을 표

현하고자 하였다.

도로 네트워크 기반의 최근접 질의를 처리하는 방법을 제시하고, 효율적인 질의 처리를 위하여 네트워크와 이동 객체 위치 데이터 표현을 2 차원과 그래프 기반으로 표현하는 방법이 [2]에서 제안되었다. 2 차원 기반의 표현은 도로 네트워크를 선(Line)과 점(Vertex)의 연결로 보고 있다. 선은 $Line = (LineID, V_{start}, V_{end}, Length, Max.Speed)$ 로 구성하였고, 점은 $Vertex = (VertexId, X, Y)$ 로 표현하였다. 여기서, $Max.Speed$ 는 도로의 최대 허용 속도이다. 2 차원 표현에서 이동 객체의 점은 $Point = (PointID, Ref.Time, X, Y, Speed)$ 로 표현하였다. 여기서, $Ref.Time$ 은 제일 최근 기록된 시간이다. 그래프 표현은 네트워크를 노드와 간선으로 구성된 그래프로 보고, 하나의 간선은 $Edge = (EdgeID, N_{start}, N_{end}, W_{edge})$ 로 표현하였다. W_{edge} 는 간선의 길이이다. 그리고 이동 객체의 위치는 $Point = (PointID, Ref.Time, EdgeID, Init Position, Speed)$ 로 구성하였다.

<그림 1>은 [2]에서 제시된 네트워크 기반으로 데이터 위치를 표현하는 예를 보여주고 있다. <그림 1>에서 알 수 있듯이, 이동 객체의 위치 표현은 객체가 존재하는 도로 아이디와 시작 노드의 위치로부터의 거리, 시간 그리고 속도로 표현되고 있다.



<그림 1> [1]의 네트워크 기반의 데이터 위치 표현의 예

도로 네트워크 상의 이동 객체 궤적들의 특성을 고려하여 이동 객체 궤적을 표현하는 방법이 [4]에서 제시되었다. 다시 말해서, 도로 네트워크 상에는 교차로, 학교 그리고 공공기관 등과 같은 관심 있는 지점(POI: Points of Interest)들이 존재하며, 이동 객체들은 출근 및 퇴근 시간대 등과 같이 관심

있는 시간대(Times of Interest)가 존재한다. [4]에서는 이러한 POI를 기반으로 이동 객체 궤적을 표현하고 있다. [4]에서 제시된 방법은 도로 네트워크 상의 이동 객체 궤적 TR 을 다음과 같이 표현하고 있다.

$$TR = \{(p_1, t_1), (p_2, t_2), \dots, (p_n, t_n)\}$$

여기서, $t_1 < t_i < t_j < t_n, 1 < i < j$.

n 은 궤적 TR 이 통과한 POI의 개수이다. p_i 는 궤적 TR 이 통과한 도로 네트워크 상의 POI이고, t_i 는 p_i 를 통과한 시간이다.

[4]에서 제시된 이동 객체 궤적을 표현하는 방법은 도로 네트워크 상의 특적인 POI(Points of Interest)를 기반으로 표현되기 때문에 매우 실용적인 방법이라 할 수 있다.

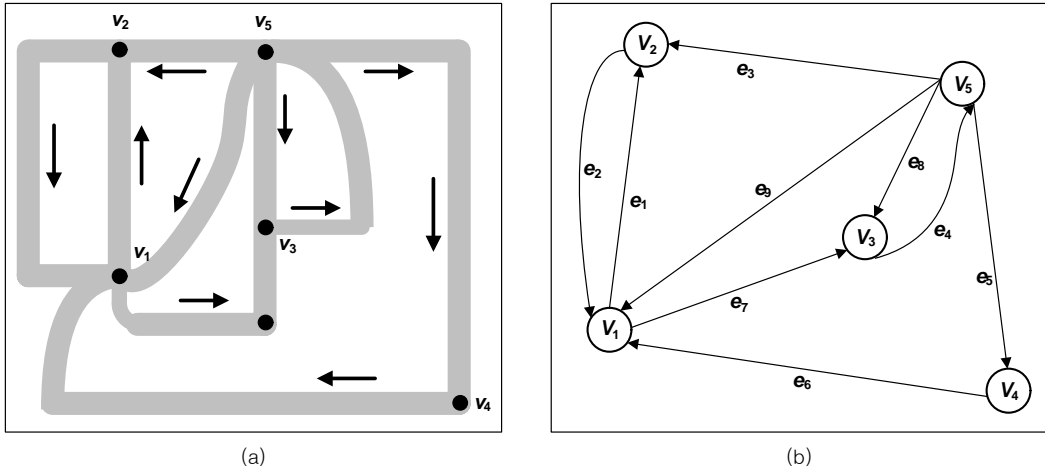
2.1.2 도로 네트워크 상의 이동 객체 궤적의 검색

도로 네트워크 상에서 이동 객체 궤적을 분석하는 대표적인 것은 궤적들 간의 유사성을 검색하는 것이다. 도로와 같이 제한적인 네트워크 상에서 이동 객체 궤적들 간의 유사성을 다루는 문제가 최근에 연구되고 있다 [4] [5] [6]. 먼저, 공간 네트워크 상의 이동 객체들을 그래프 위상에 의하여 결정된 특정한 경로들을 따르는 것으로 고려하여 두 가지 유사성 측정 - 네트워크 거리와 시간 거리 - 을 정의함으로써 궤적들의 유사성을 검색하는 방법이 제시되었다[6]. [6]에서는 <그림 2>와 같이 도로 네트워크를 노드와 선으로 이루어진 그래프 위상을 고려하여 궤적 T 를 노드 v_i 와 그에 따른 시간 t_i 로 다음과 같이 정의하고 있다.

$$T = ((v_1, t_1), (v_2, t_2), \dots, (v_m, t_m))$$

여기서, m 은 궤적의 길이이다.

[6]에서는 궤적들 간의 유사성을 표현하기 위하



<그림 2> [6]의 도로 네트워크와 그것의 그래프

여 두 궤적 T_a 와 T_b 간의 네트워크 거리 $D_{net}(T_a, T_b)$ 와 시간 거리 $D_{time}(T_a, T_b)$ 를 각각 다음과 같이 정의하였다.

$$D_{net}(T_a, T_b) = \frac{1}{m} \cdot \sum_{i=1}^m (d(v_{ai}, v_{bi}))$$

여기서, m 은 궤적의 길이이며, $d(v_{ai}, v_{bi})$ 는 그래프 상에서 두 노드 v_{ai} 와 v_{bi} 간의 거리이다.

$$D_{time}(T_a, T_b) = \frac{1}{m-1} \cdot \sum_{i=1}^{m-1} \frac{|(T_a[i+1]_t - T_a[i]_t) - (T_b[i+1]_t - T_b[i]_t)|}{\max(|T_a[i+1]_t - T_a[i]_t|), |T_b[i+1]_t - T_b[i]_t|}}$$

[6]에서 제시된 궤적들 간의 유사성 검색 방법은 이동 객체 궤적을 그래프의 노드와 그에 따른 시간으로 궤적의 근접성을 고려하여 표현하고 있기 때문에 실생활의 이동 객체 궤적을 정확하게 표현하지 못 한다. 왜냐하면, 실생활에서 도로 네트워크 상의 이동 객체들은 출발과 도착에 제한을 가지지 않을 뿐만 아니라 궤적의 길이 또한 매우 다양하기 때문이다. 예를 들어, <그림 2>의 (b)에서 이동 객체 궤적이 선 e_5 의 중간지점에서 출발하여 선 e_1 의 중간지점에 도착하였다고 하자. 이때, [6]에서 정의한 방법으로는 정확한 궤적을 표현하지 못한다.

도로 네트워크 상의 POI와 그것의 통과 시간을

고려하여 이동 객체 궤적을 표현하고, 그것들을 기반으로 궤적들 간의 공간적 유사성 $Sim_{POI}(TR_A, TR_B, P)$ 뿐만 아니라 시간적 유사성 $Sim_{TOI}(TR_A, TR_B, T)$ 을 정의하는 방법이 다음과 같이 [4]와 [5]에서 제시되었다.

$$Sim_{POI}(TR_A, TR_B, P) = \begin{cases} 1, & \text{if } \forall p \in P, p \text{ is on } TR_A \text{ and } TR_B \\ 0, & \text{otherwise} \end{cases}$$

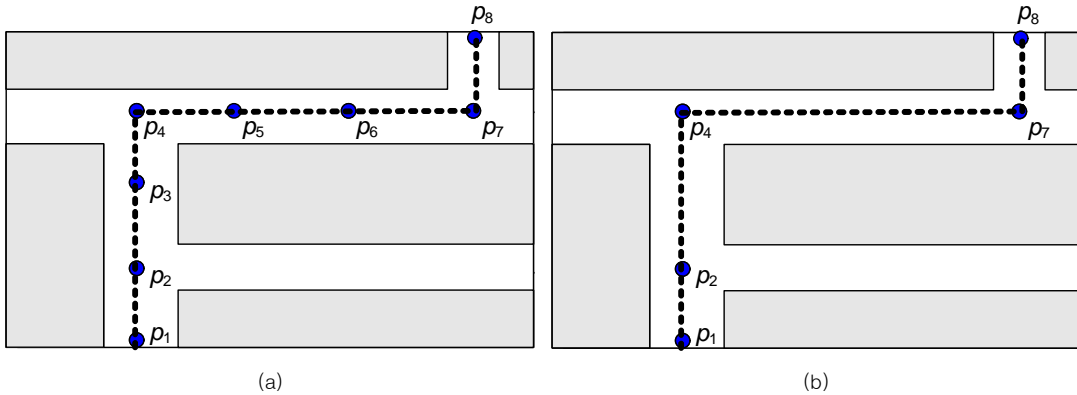
$$Sim_{TOI}(TR_A, TR_B, T) = \begin{cases} 1, & \text{if } \forall t \in T, \\ & t \in [t_s(TR_A), t_e(TR_A)] \text{ \& } t \in [t_s(TR_B), t_e(TR_B)] \\ 0, & \text{otherwise} \end{cases}$$

여기서, P 는 POI들의 집합이고, T 는 TOI들의 집합이다. 그리고, $t_s(TR_A)$ 는 궤적 TR_A 의 출발 시의 시간을 의미한다.

기존의 연구에서는 도로 네트워크 상의 모든 POI들을 동일하게 중요시 취급하고 있으며, 이동 객체 궤적들에 대한 모든 TOI들 역시 동등하게 취급하고 있다.

2.1.3 이동 객체 궤적의 간략화

복잡한 데이터를 줄이기 위한 방법은 다양한 분야에서 연구되고 있다[7] [8] [9]. 특히, 데이터의



<그림 3> 이동 객체 궤적의 간략화의 예

흐름을 그래프로 표현하고, 그 그래프를 줄이기 위한 다양한 규칙들을 적용함으로써 구조적으로 문제가 없는 간단한 그래프를 유지하는 방법이 [9]에서 소개되었다. 그리고, 센서 네트워크 상에서 대용량의 스트림 데이터를 수집하고 처리하고자 하는 활발한 연구에 따라, 이러한 데이터를 축소시키는 기법이 [10]에서 제시되었다. 기존의 데이터를 축소시키는 대부분의 방법들은 스트림 데이터를 기반으로 하며, 데이터 자체의 흐름을 축소시키고자 하였다. 하지만, 도로 네트워크 상의 이동 객체 궤적 데이터를 간단하게 표현하는 방법은 기존의 연구에서 찾아볼 수 없다.

2.2 동기

도로 네트워크 상에는 수많은 이동 객체들이 존재한다. 최근에, 이러한 이동 객체들의 움직임을 표현하고 분석하는 연구가 시도되고 있다. 일반적으로, 이동 객체 궤적은 그 궤적이 통과한 지점과 그 지점을 통과한 시간의 집합으로 표현되고 있다. 하지만, 대부분의 지점들은 GPS에 의하여 무작위로 획득된 지점들이기 때문에 이동 객체 궤적의 표현이 도로 네트워크를 반영하지 못 할 뿐만 아니라 쓸모 없는 지점들이 표현되게 된다. 따라서, 본 논문의 이전 연구 [4]에서, 우리는 도로 네트워크 상의 이동 객체 궤적을 POI와 그것을 통과한 시간의

집합으로 표현하였다. POI는 이동 객체 궤적을 매우 실질적인 방법으로 표현하기 위한 하나의 수단으로 사용되고 있다. POI를 기반으로 표현된 이동 객체 궤적은 도로 네트워크를 반영한다. 하지만, 이러한 표현 역시 도로 네트워크 상의 대부분의 이동 객체들은 많은 POI를 통과하며, 그것들의 궤적은 POI에 의하여 표현된다. 따라서, 이동 객체 궤적을 검색하는 등의 질의를 처리할 때, 이러한 POI들을 비교하고 처리하기 위해서는 많은 시간이 요구된다. 하지만, 많은 POI를 통과한 이동 객체 궤적을 간략화 시킬 수 있다면 질의를 처리하는 시간은 단축될 것이다.

<그림 3>은 이동 객체 궤적의 간략화의 예를 보여주고 있다. <그림 3> (a)는 궤적이 간략화 되기 전의 모습이고, <그림 3> (b)는 궤적이 간략화 된 후의 모습이다. 만약 궤적이 <그림 3> (b)처럼 간략화 된다면, 이동 객체 궤적들을 검색하는 데 매우 의미 있는 방법일 것이다. 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법은 기존의 연구에서는 전혀 찾아볼 수 없다. 따라서, 본 논문에서는 이러한 관찰을 통하여 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법을 제시한다.

3. 도로 네트워크 상의 이동 객체 궤적

도로 네트워크 상의 이동 객체 궤적을 표현하기

위한 방법은 다양하게 존재한다. 본 논문에서는 2장에서 언급하였듯이, [4]에서 다음과 같이 제시된 POI와 그 지점을 통과한 시간을 기반으로 이동 객체 궤적을 표현하는 방법을 사용한다.

$$TR = \{(p_1, t_1), (p_2, t_2), \dots, (p_n, t_n)\}$$

[4]에서 표현된 궤적은 모든 POI와 그것들의 통과 시간을 동등하게 취급하고 있다. 하지만, 실생활에서, 도로 네트워크 상의 POI는 사용자들마다 중요성이 다르게 다루어질 수 있다. 따라서, 본 논문에서는 이러한 관찰을 조사하여, 도로 네트워크 상의 POI에 대하여 각각 서로 다른 가중치를 부여함으로써 궤적의 표현을 좀더 실질적으로 다음과 같이 정의한다.

$$TR = \{((p_1, w_{p1}), t_1), ((p_2, w_{p2}), t_2), \dots, ((p_n, w_{pn}), t_n)\}$$

여기서, w_{pn} 은 궤적 TR이 통과한 n 번째 POI가 가지는 가중치를 의미한다.

도로 네트워크 상의 POI에 대해 가중치를 부여하는 방법은 다양하게 존재할 뿐만 아니라 응용 분야에 따라 다르게 적용될 수 있기 때문에 이러한 문제는 본 논문에서는 다루지 않기로 한다. 다만, 본 논문에서는 도로 네트워크 상의 POI에 대해 가중치를 부여한 방법은 이동 객체들이 많이 통과한 POI에 대해서는 높은 가중치를 부여하고 적게 통과한 POI에 대해서는 낮은 가중치를 부여하였다.

일반적으로 도로 네트워크 상의 이동 객체들은 많은 POI들을 통과한다. 따라서, 이러한 POI들을 가지고 이동 객체 궤적을 표현하는 것은 비효율적이다. 왜냐하면, 궤적들 간의 유사성 측정 등과 같이 이동 객체 궤적을 분석하는 데 많은 시간이 요구되기 때문이다. 만약 궤적의 표현이 간략화 된다면, 궤적을 분석하는 데 시간이 많이 줄어들 것이다. 예를 들어, 하나의 이동 객체가 10개의 POI를 통과하였고, 그 이동 객체의 궤적은 10개의 POI와 그것의 통과 시간의 집합으로 표현되었다고 가정하

자. 이때, 만약 이 궤적이 5개의 POI로도 충분히 표현이 가능하다면, 이 궤적을 이용한 분석 시간은 매우 줄어들 것이다. 따라서, 본 논문에서는 본래 이동 객체 궤적의 형태를 유지하면서 그 궤적을 간략화 하여 표현하는 방법을 제시한다. 본 논문에서 제시하는 궤적의 간략화 방법은 이동 객체 궤적이 통과한 POI의 수를 줄임과 동시에 그 형태 역시 그대로 유지하는 데 목적이 있다.

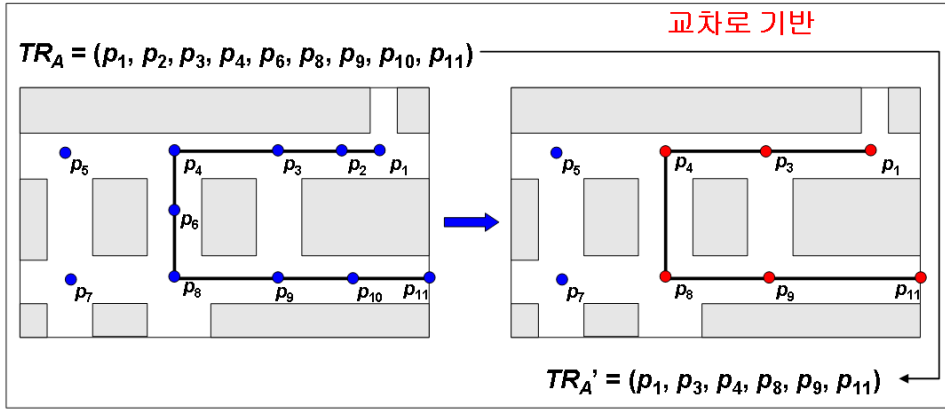
4. 이동 객체 궤적의 간략화

본 장에서는 도로 네트워크 상의 이동 객체 궤적을 간략화 하는 방법을 소개한다. 본 논문에서는 이동 객체 궤적을 간략화 하기 위하여 다음과 같이 세 가지 방법을 제시한다. 각 방법에 대한 자세한 내용은 각 절에서 살펴보도록 한다.

- 교차로 기반의 이동 객체 궤적의 간략화
 - 도로 네트워크 상에서 가장 중요시 되는 POI 중의 하나인 교차로를 기반으로 이동 객체 궤적을 간략화 하는 방법이다.
- POI의 가중치 기반의 이동 객체 궤적의 간략화
 - 도로 네트워크 상에서 중요하게 취급되는 POI 들만을 선택하여 그것을 기반으로 이동 객체 궤적을 간략화 하는 방법이다.
- 정의된 규칙 기반의 이동 객체 궤적의 간략화
 - 이동 객체 궤적이 통과한 POI들에 대하여 특정한 규칙을 정의하고, 그 규칙을 궤적들에게 적용하여 이동 객체 궤적을 간략화 하는 방법이다.

4.1 교차로 기반의 간략화 방법

본 절에서는 도로 네트워크 상의 교차로를 기반으로 궤적들의 POI를 줄임으로써 궤적을 간략화 하는 방법을 소개한다. 도로 네트워크 상의 수많은 POI들 중에 교차로는 매우 중요한 POI이다. 왜냐하면, 대부분의 이동 객체들의 방향이 교차로에 의하여 다음 도로 구간이 선택되어지기 때문이다.



<그림 4> 교차로 기반의 궤적 간략화의 예

<그림 4>는 교차로를 기반으로 한 방법을 적용하여 궤적의 POI를 간략화 한 예를 보여주고 있다. 그림에서 알 수 있듯이, 궤적 TR_A 가 POI를 간략화한 후 6 개의 POI를 통과한 궤적 TR'_A 로 간략화되었다.

알고리즘 1은 교차로 기반으로 이동 객체 궤적을 간략화 하는 과정을 요약하여 보여주고 있다. 알고리즘을 간단히 설명하면, 궤적의 간략화는 그 궤적의 시작 POI에서 시작하고 도착 POI에서 끝나며, 그 궤적의 시작 POI와 도착 POI는 항상 남는다. 그리고 그 궤적의 모든 교차로를 제외한 나머지 POI들은 제거된다. 단, 시작 POI와 도착 POI는 남게 된다.

Algorithm 1. 교차로 기반의 이동 객체 궤적의 간략화

Input : *OriginTR* (원본 궤적)

Output: *ReducedTR* (축소 궤적)

Begin

$i = 0;$

while ($i < \text{number of POIs on } OriginTR$) **do**

temp = poi in *OriginTR* pointed by i ;

If (temp is on a crossroad or i points the first POI or i points the last POI) **then**

ReducedTR.AddPOI(temp);

end if

$i++;$

end while

return *ReducedTR*;

end

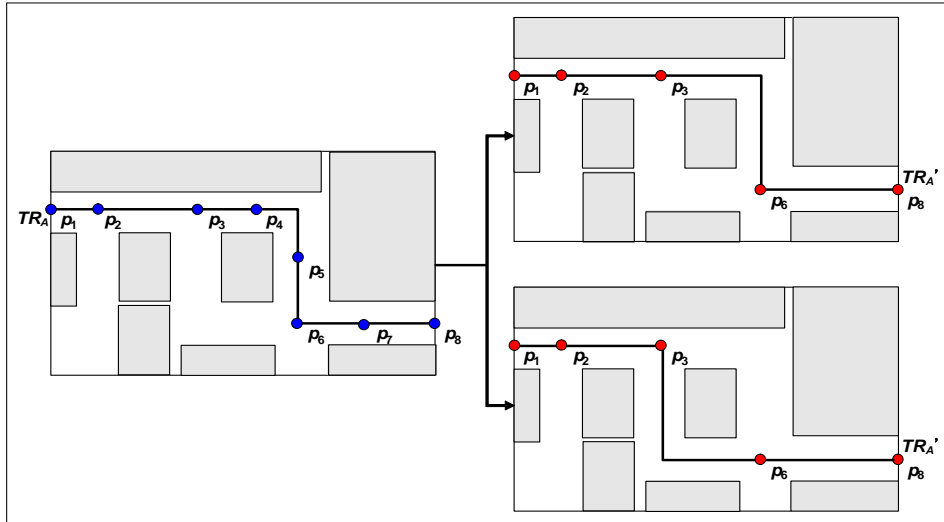
이 방법의 단점은 도로 네트워크 상에 존재하는 교차로라는 POI에 너무 의존적이기 때문에 궤적의 간략화가 본래 궤적과 구조가 다르게 나타날 수도 있다는 것이다.

<그림 5>는 본래 궤적 TR_A 가 도로 네트워크 상의 교차로를 기반으로 POI가 간략화 된 또 다른 예를 보여주고 있다. <그림 5>에서 알 수 있듯이, 궤적 TR_A 는 POI($p_1, p_2, p_3, p_4, p_6, p_8$)를 통과하였으며, 교차로를 기반으로 간략화 된 궤적 TR'_A 은 $= (p_1, p_2, p_3, p_6, p_8)$ 로 표현된다. 하지만, 간략화 된 궤적 TR'_A 는 <그림 5>와 같이 서로 다른 두 개의 구조를 가진 궤적으로 나타날 수 있다. 다시 말해서, 간략화 된 궤적 TR'_A 와 본래의 궤적 TR_A 가 동일한 구조를 가지지 않을 수도 있다는 것을 의미한다.

교차로 기반의 궤적의 간략화는 본래 궤적과 그것의 간략화 된 궤적이 완전히 동일한 형태는 아니지만, 본 논문에서 제시하는 POI 기반의 도로 네트워크 상의 궤적을 표현하는 데는 큰 영향을 미치지 않는다.

4.2 POI의 가중치 기반의 간략화

도로 네트워크 상에는 수많은 POI들이 존재하지만, 모든 POI가 동일하게 고려되지 않는다. 다시 말해서, 도로 네트워크 상에 중요하게 고려되는



<그림 5> 교차로 기반의 또 다른 궤적 간략화의 예

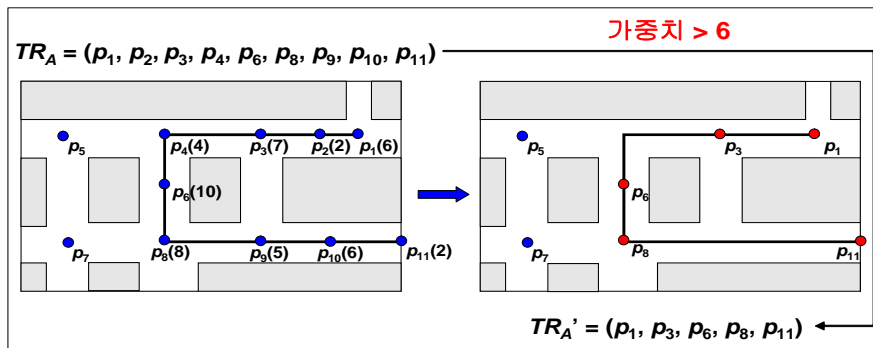
POI들이 있는 반면에, 그렇지 않은 POI들 역시 많이 존재한다. 예를 들어, 차량들이 많이 통과하지 않는 도로 상의 POI들은 이동 객체들에게는 그렇게 중요하게 고려되지 않을 것이다. 본

논문에서는 도로 네트워크 상의 모든 POI들에게 서로 다른 가중치를 부여하였으며, 이러한 점을 이용하여 POI의 가중치를 기반으로 궤적을 간략화하는 방법을 제시한다.

<그림 6>은 시작 POI와 끝 POI 그리고 POI의 가중치가 6 보다 큰 POI들만을 남기고 나머지 POI들은 제거된 모습을 보여주고 있다. 그림의 왼

쪽에서 궤적 TR_A 가 통과한 POI들에게 가중치가 부여되어 있다. 이때, 그림 오른쪽은 가중치가 6보다 큰 POI들만을 고려하여 궤적 TR_A 가 간략화 된 모습이다.

알고리즘 2는 POI의 가중치를 기반으로 이동 객체 궤적을 간략화 하는 과정을 요약하여 보여주고 있다. 간단히 설명하면, 궤적의 간략화는 그 궤적의 시작 POI에서 시작하고 도착 POI에서 끝난다. 그 궤적의 시작 POI와 도착 POI는 항상 남는다. 그리고, 그 궤적에서 주어진 가중치보다 작은 값을 가지는 POI들은 제거된다. 단, 시작 POI와 도착 POI



<그림 6> POI의 가중치를 기반으로 간략화 된 궤적의 예

는 남는다.

이 방법은 주어지는 POI의 가중치에 의하여 너무 민감하게 궤적들이 간략화 되는 경향을 가진다. 만약 주어진 가중치 값이 너무 높으면, 대부분의 POI가 제거되어 검색 시간은 줄어들지만 정확성은 낮을 것이다. 반면에, 주어진 가중치 값이 너무 낮으면, 대부분의 POI가 남게 되어 검색 시간은 많이 줄지 않지만 정확성은 높을 것이다. 따라서, POI의 가중치를 기반으로 이동 객체 궤적을 간략화 하는 것은 사용자의 목적에 따라 그 성향이 크게 달라질 수 있다.

Algorithm 2. POI의 가중치 기반의 간략화

Input : *OriginTR* (원본 궤적), *threshold* (임계값)

Output: *ReducedTR* (축소 궤적)

Begin

i = 0;

while (*i* < number of POIs on *OriginTR*) do

temp = poi in *OriginTR* pointed by *i* ;

weight = weight of poi in *OriginTR* pointed by *i* ;

If (weight is bigger than *threshold* or

i points the first POI or *i* points the last POI) then

ReducedTR.AddPOI(temp);

end if

i++;

end while

return *ReducedTR*;

end

4.3 정의된 규칙 기반의 간략화

도로 네트워크 상의 이동 객체 궤적들은 일반적으로 POI들을 순차적으로 통과한다. 이러한 관찰을 고려하여 도로 네트워크 상의 궤적에 대한 POI들을 줄이기 위한 특정한 규칙을 정의한다. 따라서, 본 절에서는 이러한 규칙을 다음과 같이 정의하고, 그 정의된 규칙을 모든 궤적에 적용함으로써 궤적들이 가지는 POI들을 줄이는 방법을 제시한다.

규칙 1. 궤적의 간략화는 그 궤적의 시작 POI에서 시작하고 도착 POI에서 끝난다.

규칙 2. 그 궤적의 시작 POI와 도착 POI는 항상 남는다.

규칙 3. 순서를 가진 세 개의 POI(*p*₁, *p*₂, *p*₃)에서 *p*₁에서 *p*₃까지 가는 경로가 도로 네트워크 상에 오직 하나일 경우, 항상 중간 POI *p*₂를 통과하게 된다. 이때, 그 중간 POI는 궤적으로부터 제거된다.

규칙 4. 순서를 가진 세 개의 POI(*p*₁, *p*₂, *p*₃)에서 *p*₁에서 *p*₃까지 가는 경로가 도로 네트워크 상에 두 가지 이상일 경우, 반드시 그 중간 POI를 통과하여야 한다. 이때, 그 중간 POI는 궤적에서 남게 된다.

규칙 5. 규칙 3과 4의 과정을 그 궤적의 마지막 POI까지 수행한다.

규칙 6. 간략화 된 궤적은 원래 궤적과 동일한 구조를 가져야 한다.

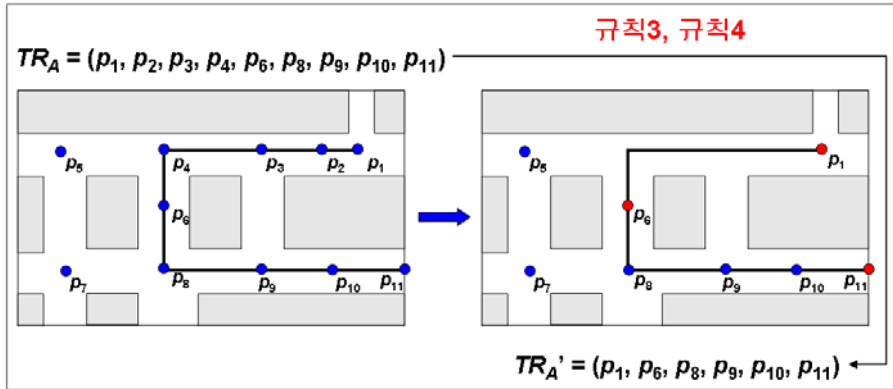
본 절에서 제시되는 궤적을 간략화 하는 방법은 위 규칙에서 규칙 3~5의 규칙들을 중점으로 이루어진다.

<그림 7>과 8은 본 절에서 제시된 방법에 의하여 본래 궤적 *TR_A*가 간략화 되는 모습을 보여주고 있다. 특히, <그림 7>은 규칙 3과 4를 적용함으로써 궤적 *TR_A*의 POI가 줄어드는 중간 과정을 보여주고 있다.

<그림 7>에서 알 수 있듯이, 궤적 *TR_A*은 규칙 3에 의하여 중간 POI(*p*₂, *p*₃, *p*₄)가 제거되었으며, 규칙 4에 의하여 중간 POI *p*₆이 남게 되었다. <그림 8>은 궤적 *TR_A*가 간략화 된 결과를 보여주고 있다. <그림 8>에서 알 수 있듯이, 9 개의 POI를 통과한 궤적 *TR_A*가 POI를 줄임으로써 3 개의 POI를 통과한 궤적 *TR_A'*로 변환되었다.

궤적의 간략화에서 중요한 점은 궤적에 대한 POI의 수를 줄이는 것뿐만 아니라 간략화 된 궤적과 본래 궤적 간의 구조가 변함없이 같아야 한다는 것이다. <그림 8> 서, 간략화 된 궤적 *TR_A'* = (*p*₁, *p*₆, *p*₁₁)이 본래 궤적을 충분히 표현하고 있음을 알 수 있다.

알고리즘 3은 정의된 규칙을 기반으로 이동 객체 궤적을 간략화 하는 과정을 요약하여 보여주고 있다.



<그림 7> 궤적에 대한 POI가 줄어드는 중간 과정

<그림 9>에서, 궤적 TR_A 은 <그림 7>의 본래 궤적과 같다고 가정하자. 이때, <그림 9>는 본래 궤적 TR_A 과 잘못 간략화 된 궤적 TR_A'' 간의 모습을 보여주고 있다. 잘못 간략화 된 궤적은 본래의 궤적과 다른 구조를 가질 수 있는 궤적을 의미한다. 예를 들어, 본래 궤적 TR_A 의 간략화 된 궤적 $TR_A'' = (p_1, p_8, p_{11})$ 와 같다고 하자. 이때, 이 간략화 된 궤적은 본래 궤적과 다른 구조를 가질 수 있다. <그림 9>에서 알 수 있듯이, 간략화 된 궤적 TR_A'' 은 p_8 을 통과하는 두 가지 경로를 가진다. 이것은 간략화 된 궤적 TR_A'' 과 본래 궤적 TR_A 간의 구조가 변함없이 같아야 한다는 규칙에 어긋남을 의미한다.

Algorithm 3. 정의된 규칙 기반의 간략화

Input : $OriginTR$ (원본 궤적), $threshold$ (임계값)

Output: $ReducedTR$ (축소 궤적)

Begin

$i = 0;$

$startpoi = poi$ in $OriginTR$ pointed by i ;

$ReducedTR.AddPOI(startpoi);$

while ($i < \text{number of POIs on } OriginTR$) **do**

$currentpoi = poi$ in $OriginTR$ pointed by $i+1$;

$nextpoi = poi$ in $OriginTR$ pointed by $i+2$;

If (there are multiple paths between $startpoi$

and $nextpoi$ or $nextpoi$ is null) **then**

$ReducedTR.AddPOI(currentpoi);$

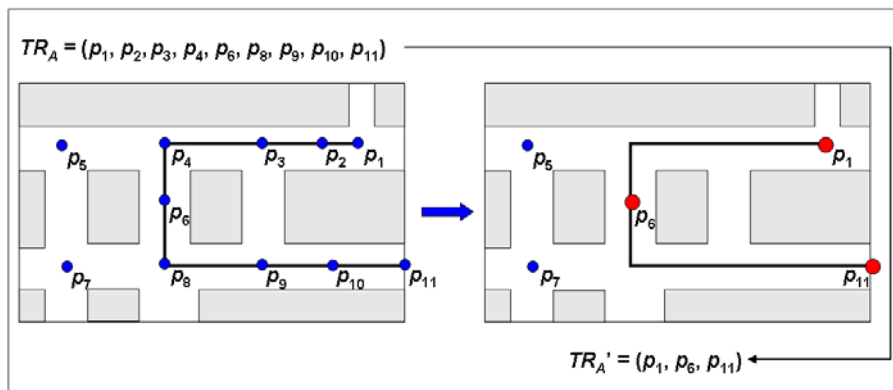
end if

$i++;$

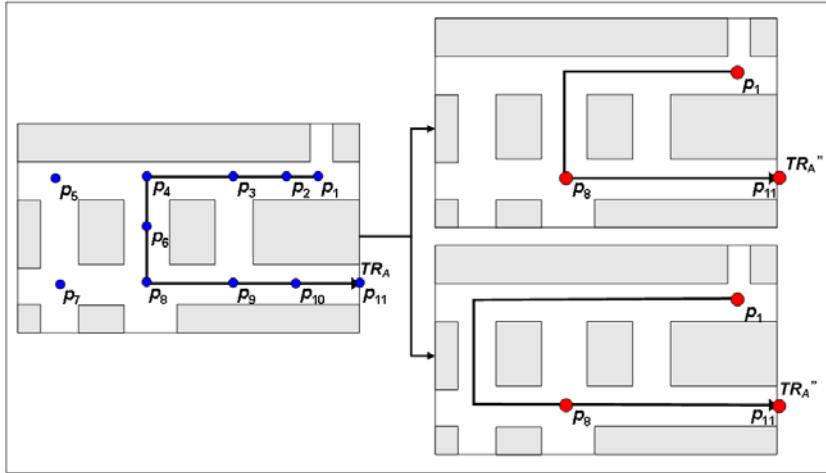
end while

return $ReducedTR;$

end



<그림 8> 간략화 된 궤적의 결과 모습



<그림 9> 궤적의 잘못된 간략화의 예

이 방법의 단점은 궤적의 간략화에 대하여 규칙 3과 4를 적용하는 데 있어 도로 네트워크 상의 POI들 간의 경로를 조사하여야 하는 부담이 발생한다는 것이다. 그림에도 불구하고, 이 방법의 장점은 궤적을 간략화 하는데 있어 도로 네트워크 상에 존재하는 POI의 종류나 가중치에 따른 영향을 전혀 받지 않는 반면에, 다른 나머지 방법들은 POI의 성향에 따라 궤적의 간략화가 달라지게 된다는 것이다. 따라서, 본 논문에서는 본 절에서 제시된 방법을 궤적을 간략화 하는 방법으로 선택하고, 이 방법을 이용하여 궤적 데이터를 간략화 하고, 본 논문의 실험에 적용한다.

5. 실험 평가

본 논문에서 제시하는 방법의 효율성을 입증하기 위하여, 본 장에서는 다양한 실험을 수행하였다. 본 장에서는 실험 환경을 소개하고, 실험에 따른 결과들을 보인다.

5.1 실험 환경

본 논문에서는 다음과 같이 [4]에서 제시된 도로 네트워크 상에서 유사한 궤적들을 검색하는 세 가

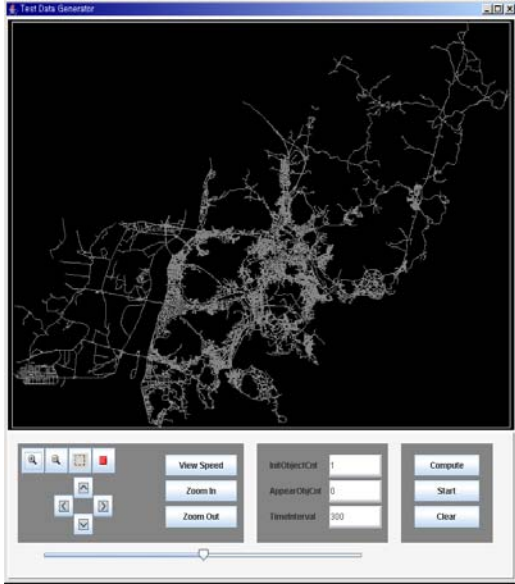
지 방법들을 실험의 대상으로 정하였다.

- 방법 1: 공간적 필터링과 시간적 거리 기반의 유사한 궤적 검색
- 방법 2: 시간적 필터링과 공간적 거리 기반의 유사한 궤적 검색
- 방법 3: 시공간 필터링과 시공간 거리 기반의 유사한 궤적 검색

본 논문에서는 본래 궤적 데이터와 본 논문의 4.3 절에서 제시한 방법에 의하여 간략화 된 궤적 데이터를 [4]에서 제시된 유사한 궤적들을 검색하는 방법들에 적용함으로써 성능을 비교 실험하였다. 본 논문에서는 다음과 같은 실험 환경을 가진다.

- 실제 부산 도로 네트워크 기반의 이동 객체 데이터 생성기를 사용하였다.
- 부산 도로 네트워크 상에 10,000 개의 POI들을 생성하였다.
- 도로 네트워크 상의 이동 객체 궤적들에 대하여 24 개의 TOI들을 정의하였다.
- 이동 객체 데이터 생성기를 이용하여 100,000 개의 이동 객체 궤적을 생성하였다.
- 이동 객체 궤적들로부터 5,000 개의 질의 궤적들을 생성하였다.

<그림 10>은 본 논문에서 사용된 실제 부산 도로 네트워크 공간을 보여주고 있다.



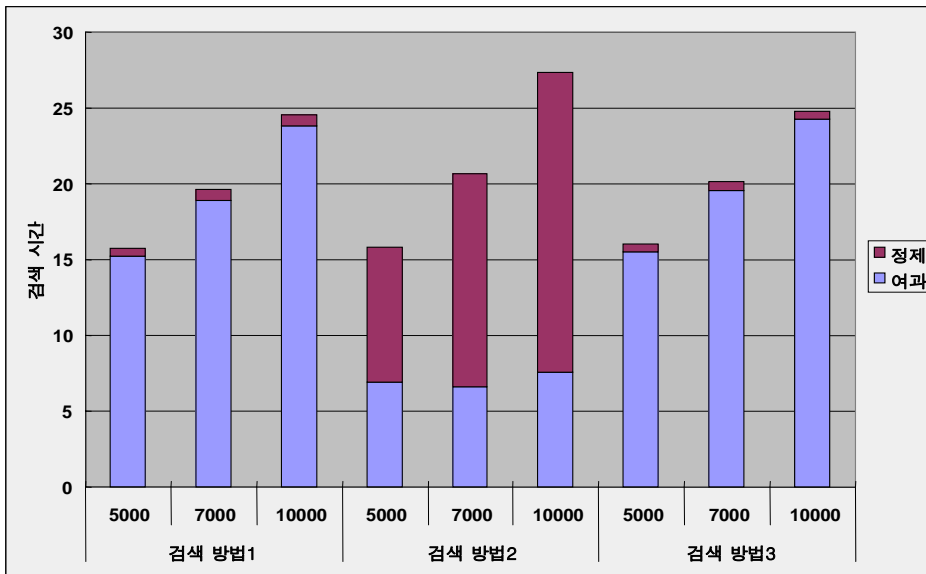
<그림 10> 부산 도로 네트워크 공간 기반의 이동 객체 궤적 생성기

5.2 실험 결과

본 절에서는 본래 궤적 데이터와 그것을 간략화 시킨 궤적 데이터 간의 검색 시간과 검색 결과에 대한 일치율에 대하여 살펴본다.

- POI 수의 변화에 따른 검색 시간 비교
도로 네트워크 상에는 수많은 POI들이 존재한다. 이러한 POI들을 이용하여, 본 논문에서는 도로 네트워크 상의 이동 객체 궤적을 표현하였고, 그 궤적을 간략화 하는 방법을 제시하였다. 따라서, 이 실험은 도로 네트워크 상의 POI의 수가 [4]에서 제시된 유사한 궤적들을 검색하는 방법들의 성능에 어떠한 영향을 미치는지를 실험하였다. <그림 11>은 POI 수의 변화에 따른 질의 처리 시의 평균 검색 시간을 보여주고 있다. 이 실험에서 사용된 POI의 수는 5,000개, 7,000개 그리고 10,000개이다.

<그림 11>에서 알 수 있듯이, POI의 수는 유사한 궤적들을 검색하는 시간에 영향을 미친다. 다시



<그림 11> POI 수의 변화에 따른 검색 시간 비교

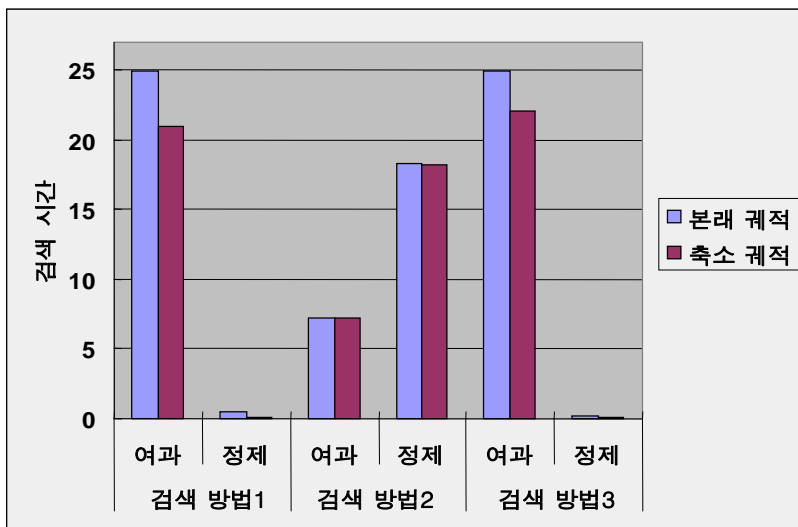
말해서, POI의 수가 많을수록 각 방법들의 검색 시간은 많이 소요된다.

- 본래 궤적과 간략화 된 궤적 간의 검색 시간 비교 이 실험은 동일한 질의들에 대하여 본래 궤적 데이터와 그것의 간략화 시킨 궤적 데이터 간의 평균 검색 시간을 비교한 실험이다. [4]에서 제시된 유사한 궤적들을 검색하는 데 있어, POI 수가 많을수록 각 방법에 대한 검색 시간이 많이 요구된다는 것은 이미 앞의 실험에서 살펴보았다. 이 실험은 각 방법에 대하여 본래 궤적을 이용한 검색 시간과 그것의 간략화 시킨 궤적을 이용한 검색 시간을 비교 실험하였다. 이 실험에서는 10,000 개의 POI가 정의된 도로 네트워크를 사용하였으며, 가장 유사한 5 개의 궤적을 검색하는 질의 1,000 개를 수행하였다.

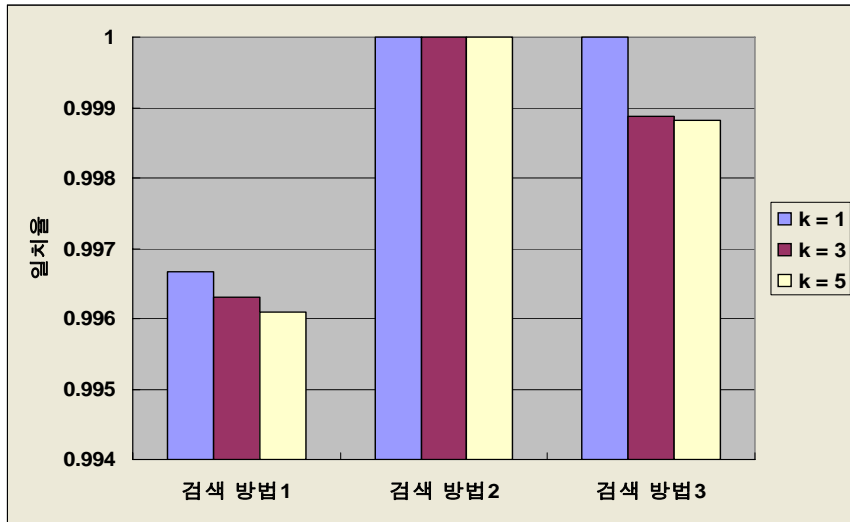
<그림 12>는 본래 궤적과 그것의 간략화 된 궤적 간의 유사한 궤적을 검색하는 데 수행되는 시간을 보여주고 있다. 그림에서 알 수 있듯이, 간략화 된 궤적을 이용한 검색에서 유사한 궤적을 검색하기 위하여 제시된 방법 2의 검색 시간은 거의 변함

이 없다. 반면에, 방법 1과 방법 3의 여과 시간이 단축되었다. 그 이유는 [4]에서 제시된 검색 방법에서 공간적 여과와 시공간 여과 단계가 POI 기반의 공간적 유사성에 의하여 수행되기 때문이다. 다시 말해서, 궤적의 간략화로 인하여 궤적이 가지는 POI 수가 줄어들었기 때문이다.

- 본래 궤적과 간략화 된 궤적 간의 검색 결과에 대한 일치율 비교 이 실험 역시 동일한 질의들에 대하여 본래 궤적 데이터와 그것의 간략화 시킨 궤적 데이터 간의 검색 결과에 대한 일치율을 비교한 실험이다. 본 논문에서는 도로 네트워크 상의 이동 객체 궤적을 간략화 하기 위한 방법들을 제시하였으며, 그중에서 4.3절에서 제시된 간략화 방법을 선택하여 실험에 적용하였다. <그림 13>은 각 방법에 대하여 본래 궤적에 대한 검색 결과와 간략화 된 궤적에 대한 검색 결과 간의 일치율을 보여주고 있다. 이 실험에서는 10,000 개의 POI가 정의된 도로 네트워크를 사용하였으며, 1,000 개의 질의를 수행하였다. 유사한 궤적의 검색 수는 1 개, 3 개 그리고 5 개로 하였다.



<그림 12> 본래 궤적과 간략화 된 궤적 간의 검색 시간 비교



<그림 13> 본래 궤적과 그것의 간략화 된 궤적 간의 검색 결과에 대한 일치율 비교

<그림 13>에서 보는 바와 같이, 본래 궤적을 이용한 검색 결과와 간략화 된 궤적에 대한 검색 결과의 비교에서 방법 2는 100% 일치하였으며, 방법 1과 방법 3 역시 100%에 가깝게 일치하였다.

이러한 이유는 유사한 궤적들을 검색하기 위한 검색 방법 1과 3은 POI를 기반으로 한 공간적 유사성을 적용하고, 검색 방법 2는 POI를 고려하지 않은 시간적 유사성을 적용하기 때문에 검색 결과에 대한 일치율에 영향을 주지 않는다.

6. 결론 및 향후 연구

도로 네트워크 상에는 수많은 이동 객체들이 존재한다. 그것들의 움직임을 나타낸 대용량 궤적 데이터를 분석하는 것은 많은 시간이 요구된다. 실질적으로, 도로 네트워크를 기반으로 하는 응용 분야들이 유사한 궤적들을 검색하는 방법 등을 적용하고 있다. 이러한 초점에 맞추어, 본 논문에서는 도로 네트워크 상의 이동 객체 궤적을 POI와 그것의 가중치 기반으로 한 실질적인 방법으로 표현하고, 이동 객체 궤적을 간략화 하는 방법을 제시하였다. 본 논문에서 제시된 이동 객체 궤적을 간략화 하는 방법은 간략화 된 후에도 본래 궤적의 형태를 그대로

유지할 뿐만 아니라 이동 객체 궤적 데이터를 분석하는 시간을 단축시킨다는 장점을 가진다. 본 논문의 실험을 통하여 본래 궤적 데이터와 그것의 간략화 된 궤적 데이터를 가지고 본 논문에서 제시한 방법의 성능이 우수하다는 것을 보여주었다. 본 논문에서 제시한 이동 객체 궤적을 간략화 하는 방법은 도로 네트워크 상의 이동 객체 궤적을 분석하는 실질적인 응용 분야에 매우 효율적으로 사용될 것으로 예상된다.

참고 문헌

1. Nirvana, Meratnia and Rolf A. de By, "Representation in Location-based Services: Problems & Solution", Proc. 3rd International Workshop on Web and Wireless Geographical Information Systems, 2003.
2. Christian S. Jensen, Jan Kolarvr, Torben Bach Pedersen, and Igor Timko, "Nearest Neighbor Queries in Road Networks", Proc. 11th ACM International Symposium on Advances in Geographic

Information Systems, 2003, pp. 1-8.

3. Michalis Vazirgiannis and Ouri Wolfson. "A Spatiotemporal Model and Language for Moving Objects on Road Networks", Proc. 7th International Symposium on Spatial and Temporal Databases, 2001, pp. 20-35.
4. Jung-Rae Hwang and Hye-Young Kang and Ki-Joune Li, "Searching for Similar Trajectories on Road Networks using Spatio-Temporal Similarity", Proc. 10th East European Conference Advances in Databases and Information Systems, 2006, pp. 282-295.
5. Jung-Rae Hwang and Hye-Young Kang and Ki-Joune Li, "Spatio-temporal Similarity Analysis between Trajectories on Road Networks", ER Workshop on CoMoGIS, 2005, pp. 280-289.
6. Eleftherios Tiakas and Apostolos N. Papadopoulos and Alexandros Nanopoulos and Yaninis Manolopoulos and Dragan Stojanovic and Slobodanka Djordjevic-Kajan, "Trajectory Similarity Search in Spatial Networks", Proc. 10th International Database Engineering and Applications Symposium, 2006.
7. Alexander Kolesnikov and Pasi Frönti, "Data reduction of large vector graphics", Pattern Recognition, Vol.38 No.3, 2005, pp. 381-394.

8. Karen S. K. Cheung and Douglas R. Vogel, "Complexity Reduction in Lattice-Based Information Retrieval", Information Retrieval, Vol.8 No.2, 2005, pp. 285-299.
9. Wasim Sadiq and Maria E. Orlowska, "Analyzing Process Models Using Graph Reduction Techniques", Information System, Vol.25 No.2, 2000, pp. 117-134.
10. 정훈조, 서성보, 최경주, 박정석, 류근호, "다변량 스트림 데이터 축소 기법 평가", 한국정보처리학회 논문지 D, 제13-D권 제7호, 2006, pp. 889-900.

황정래

1999년 경성대학교 컴퓨터공학과(공학사)
 2001년 부산대학교 대학원 지형정보공학과(공학석사)
 2007년 부산대학교 대학원 지형정보공학과(공학박사)
 2007년~현재 부산대학교 컴퓨터 및 정보통신연구소
 기금교수(조교수)
 관심분야 : 지리정보시스템, 시공간 데이터베이스,
 시공간데이터마이닝, 공간 데이터 모델링

강혜영

2000년 부산대학교 컴퓨터공학과(공학사)
 2002년 부산대학교 대학원 컴퓨터공학과(공학석사)
 2003~현재 부산대학교 대학원 컴퓨터공학과(박사과정)
 관심분야 : 유비쿼터스 컴퓨팅, 지리정보시스템, 시
 공간 데이터베이스

이기준

1984년 서울대학교 계산통계학과(이학사)
 1986년 서울대학교 대학원 계산통계학과(이학석사)
 1992년 프랑스 국립응용과학원 전자계산학과(박사)
 1990년~1991년 프랑스 Logism사 선임엔지니어
 1993년~현재 부산대학교 정보컴퓨터공학부 교수
 관심분야 : 시공간 데이터베이스, 지리정보시스템,
 유비쿼터스 컴퓨팅 등