

도로 네트워크 환경을 위한 궤적 클러스터링

백지행[†] · 원정임^{**} · 김상욱^{***}

요 약

최근 궤적 정보를 이용한 많은 연구들이 진행되고 있으나, 이들 대부분의 연구는 유클리드 공간 내의 궤적들을 대상으로 하고 있다. 그러나 실제 응용에서 대부분의 이동 객체들은 도로 네트워크 공간상에 존재하므로, 유클리드 공간을 대상으로 한 연구들을 도로 네트워크 공간에 적용시키는 것은 적합하지 않다. 본 논문에서는 도로 네트워크 내 이동 객체들의 대용량 궤적 정보를 대상으로 하는 효과적인 클러스터링 기법에 대하여 논한다. 이를 위하여 우선 본 논문에서는 궤적을 각 이동 객체가 시간에 따라 지나온 도로 세그먼트들의 연속으로 정의한다. 다음, 도로 세그먼트들의 길이와 식별자 정보를 이용한 새로운 유사도 측정 함수를 제안하고, 이를 이용하여 측정된 궤적간의 유사도 정보를 기반으로 FastMap과 계층 클러스터링(hierarchical clustering) 기법을 이용하여 전체 궤적들을 클러스터링하는 방식을 제안한다. 또한, 본 논문에서는 실제 응용에서 대부분의 이동 객체는 최단 거리를 이용하여 움직인다는 특성을 반영한 새로운 궤적 생성 기법을 제안하고, 이렇게 생성된 궤적 데이터를 이용하여 제안된 클러스터링 기법에 대한 다양한 성능 평가 결과를 보인다. 실험 결과에 따르면 제안된 기법은 사람에 의하여 유사 궤적들을 클러스터링한 결과와 비교하여 95%이상의 높은 정확도를 보였다.

키워드 : 이동 객체, 궤적, 유사도 함수, 클러스터링

Trajectory Clustering in Road Network Environment

Ji-Haeng Bak[†] · Jung-Im Won^{**} · Sang-Wook Kim^{***}

ABSTRACT

Recently, there have been many research efforts proposed on trajectory information. Most of them mainly focus their attention on those objects moving in Euclidean space. Many real-world applications such as telematics, however, deal with objects that move only over road networks, which are highly restricted for movement. Thus, the existing methods targeting Euclidean space cannot be directly applied to the road network space. This paper proposes a new clustering scheme for a large volume of trajectory information of objects moving over road networks. To the end, we first define a trajectory on a road network as a sequence of road segments a moving object has passed by. Next, we propose a similarity measurement scheme that judges the degree of similarity by considering the total length of matched road segments. Based on such similarity measurement, we propose a new clustering algorithm for trajectories by modifying and adjusting the FastMap and hierarchical clustering schemes. To evaluate the performance of the proposed clustering scheme, we also develop a trajectory generator considering the observation that most objects tend to move from the starting point to the destination point along their shortest path, and perform a variety of experiments using the trajectories thus generated. The performance result shows that our scheme has the accuracy of over 95% in comparison with that judged by human beings.

Keywords : Moving Object, Trajectory, Similarity Measurement, Clustering

1. 서 론

최근, 위성 기반 시스템의 발전과 모바일 기기와 같은 휴

대용 단말기의 보편화로 인하여 텔레매틱스(telematics) 기술을 이용한 실시간 교통 정보 제공, 이동 경로 안내, 차량 사고 알림 등의 다양한 서비스가 제공되고 있다. 이들 서비스들은 GPS(global positioning system)를 이용하여 일정 시간 간격마다 측정된 이동 객체의 시간 흐름에 따른 공간적 위치 정보인 궤적 정보를 활용하고 있다.

현재, 이들 궤적 정보를 효율적으로 저장 및 관리하는 기법들에 관한 많은 연구 결과들이 보고되고 있다[1, 2, 20, 25]. 몇몇 연구에서는 주어진 이동 객체의 궤적과 유사한 궤적을 검색 또는 클러스터링하고, 이를 도로 정보 및 사용자 정보

* 이 논문은 2007년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원(KRF-2007-313-D00651, KRF-2007-314-D00221)과 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원(IITA-2009-C1090-0902-0040)으로 수행되었습니다.

† 준회원: 한양대학교 전자컴퓨터통신공학과 석사

** 정회원: 한양대학교 정보통신학부 연구교수

*** 종신회원: 한양대학교 정보통신학부 교수

논문접수: 2009년 1월 15일

수정일: 1차 2009년 2월 26일

심사완료: 2009년 3월 2일

등과 연계하여 분석하는 것을 시도하였다[2, 17, 26].

이들 궤적은 궤적이 생성되는 공간에 따라 유클리드(Euclidean) 공간상의 궤적과 도로 네트워크(road network) 공간상의 궤적으로 구분할 수 있다. 대부분의 궤적에 대한 연구는 유클리드 공간상에서 2차원의 공간 좌표(x,y)의 연속으로 표현되는 이동 객체의 궤적 정보를 대상으로 수행되어 왔다. 궤적간 유사도 측정 방식으로는 주로 유클리드 거리를 사용한다[19].

그러나 실제 텔레매틱스 응용에서 대부분의 이동 객체들은 도로 네트워크 공간상에 존재하며, 이동 객체의 공간 정보를 파악하기 위하여는 1차원의 도로 정보가 사용자에게 보다 직관적이고, 유용한 정보를 제시할 수 있다[15, 18]. 예를 들어, 이동 객체가 '한양대역'을 지나고 있다면 이를 (동경125도58분, 북위37도34분)과 같이 2차원 공간 좌표로 표현하는 것이 아니라 '왕십리길'로 표현하는 것이 더 실세계에 부합하는 표현 방식이라 할 수 있다. 최근, 유클리드 공간상에서의 연구들이 도로 네트워크 공간상의 연구로 전환되고 있으며, 이동이 제한된 도로 네트워크 공간내의 이동 객체의 궤적 정보를 효과적으로 표현, 저장, 인덱싱 하고자 하는 몇몇 연구가 시도된 바 있다[1, 2, 20, 25]. 그러나 유사 궤적 검색 및 클러스터 기법에 관한 연구는 아직 미흡한 상태이다.

본 논문에서는 도로 네트워크 공간내 이동 객체들에 대한 대용량 궤적 정보를 대상으로 하는 효과적인 클러스터링 기법에 대하여 논한다. 이를 위하여 먼저, 이동 객체의 궤적이 2차원 공간 좌표 (x,y)의 연속으로 표현되는 유클리드 공간과는 달리, 본 논문에서는 궤적을 하나의 이동 객체가 지나온 도로 세그먼트들의 연속으로 표현한다. 다음, 도로 세그먼트의 식별자 및 길이 정보를 이용한 유사도 측정 방식을 제안한다. 유클리드 거리를 두 궤적간의 유사도의 척도로 이용하는 기존의 방식을 도로 네트워크 공간상에 적용하는 것은 적합하지 않다. 예를 들어, A라는 고가도로로 움직인 이동 객체의 궤적과 해당 고가도로의 밑으로 이동하는 이동 객체의 궤적을 유클리드 거리 함수를 이용하여 유사도를 측정하는 경우 두 궤적은 서로 다른 도로를 지나도 있음에도 불구하고 유사한 것으로 판정될 수 있기 때문이다.

제안된 유사도 함수에 의하여 측정된 궤적간의 유사도 정보를 기반으로 기존의 FastMap[10]과 계층 클러스터링 기법[12]을 이용하여 궤적들을 클러스터링 한다. 구성된 클러스터와 연관된 사용자 정보, 도로 정보 등을 함께 사용자에게 제공하는 프로토타입 시스템을 제시함으로써 제안된 기법이 실제 응용에 유용하게 사용될 수 있음을 보인다. 또한, 본 논문에서 제안된 유사도 함수를 이용한 클러스터링 수행의 정확도 측정을 위한 실험을 수행하고, 제안하는 기법의 우수성을 검증한다.

본 논문의 구성은 다음과 같다. 제 2장에서는 유사 궤적을 검색하기 위한 기존의 유사도 측정 방식을 소개하고, 이들 방식을 도로 네트워크에 적용했을 때의 문제점에 대하여 설명한다. 제 3장에서는 본 논문에서 제안하는 유사도 측정 방식과 이를 이용한 클러스터 방식에 대하여 설명하고, 실제 응용에서 분석된 클러스터 정보의 활용 예를 보인다. 제

4장에서는 실제 응용에서 대부분의 이동 객체는 최단 거리를 이용하여 움직인다는 특성을 반영한 새로운 궤적 생성 기법을 제안하고, 이렇게 생성된 궤적 데이터를 이용하여 제안된 클러스터링 기법에 대한 다양한 성능 평가 결과를 보인다. 마지막으로, 제 5장에서는 본 논문을 요약하고 결론을 내린다.

2. 관련 연구

본 장에서는 기존의 유사도 측정 방식들을 살펴보고, 이들 방식을 도로 네트워크에 적용했을 때의 문제점에 대하여 설명한다.

유클리드 공간상에서 이동 객체 궤적들 간의 유사도 측정을 위하여 EU(Euclidean distance), DTW(dynamic time warping distance), ERP(edit distance with real penalty), LCSS(longest common sub-sequences), EDR(edit distance in real sequence) 등의 거리 함수를 이용한 기법들이 제안된 바 있다[4-6, 24].

EU 방식은 길이 k의 두 궤적이 주어졌을 때, 궤적을 구성하는 두 k차원 시공간 좌표들 간의 유클리드 거리를 구하는 방식이다. 이 방식은 유사도를 측정하기가 쉽고 계산이 빠르다는 장점이 있는 반면, 비교하는 두 궤적의 길이가 동일해야 한다는 제약이 있다. 실제 응용에서는 궤적들의 길이가 동일하지 않기 때문에 이 방법은 사용 범위가 제한적이다.

DTW 방식[6]은 기준이 되는 궤적의 패턴에 맞게 비교할 궤적의 패턴을 시간 좌표축을 변화시켜 두 궤적의 유사성을 측정하는 방법이다. 다시 말해, 서로 다른 길이를 갖는 궤적들 간의 유사도를 측정하기 위하여 궤적내의 특정 시공간 좌표 값을 임의의 수만큼 반복시키는 것을 허용하는 방식이다. 이 방식은 비교 대상인 궤적들의 길이가 서로 달라도 된다는 점에서 EU방식의 단점을 보완할 수 있다. 그러나, 잘못 입력된 데이터 값에 의한 잡음에 매우 민감하여 잘못된 결과를 도출 할 수 있다는 단점이 있다.

ERP 방식[4]은 에디트 거리(edit distance) 방식에 패널티 값을 이용하여 궤적간의 유사도를 측정하는 방식이다. 이 방식은 비교하려는 궤적들 간에 갭(gap)을 허용하므로 길이가 서로 다른 궤적들 간의 유사도 측정이 가능한 반면, 잡음에 매우 민감하다는 단점이 있다. EU, DTW, ERP 방식은 모두 잡음에 민감한 유사도 측정 방식이므로, 궤적의 획득 혹은 표현 과정에서 잡음이 발생할 가능성이 높은 실제의 궤적들을 대상으로 할 경우 정확도가 낮아진다는 단점을 갖는다.

LCSS 방식과 EDR 방식은 이러한 문제점을 해결한 방식이다. 이들 방식은 잡음에 의한 영향을 감소시키기 위하여 유사 궤적내의 시공간 좌표 값의 차이가 주어진 허용치보다 작으면 두 좌표가 매칭된 것으로 간주한다. 그러나 두 방식 모두 0 또는 1로 양자화 되어 유사 여부를 판단하기 때문에 허용치의 값에 따라 유사도의 값이 달라질 수 있고, 정확한 유사도의 값을 측정하기 어렵다는 단점이 있다. LCSS 방식

[24]은 잡음에 의한 영향을 줄이기 위해 임계값을 두고 유사도를 계산한다. 두 궤적을 비교하여 두 궤적의 일정 부분이 일치하게 되면 1을, 일치하지 않으면 0을 반환하고 각 궤적의 좌표들을 비교 한 후 그 값이 임계값으로 주어진 허용치에 도달하지 못하면 잡음으로 간주한다. 이 방식은 서로 다른 길이를 갖는 두 궤적간의 유사도 측정이 가능하며 잡음에 의한 영향이 적은 반면, 궤적 내에 갭을 허용하지 않는 방식으로 정확도가 낮다는 단점을 갖는다. EDR 방식[5]은 LCSS방식과 접근 방식은 비슷하지만 이 방식은 에디트 거리에 기반을 두고 있다. 두 궤적을 에디트 거리를 이용하여 비교 한 후 두 궤적이 일치하면 0을, 일치하지 않으면 1을 반환하여 주어진 허용치보다 큰 값을 갖는 궤적은 유사하지 않다고 판단하는 방식이다. 다시 말해, 유사도 측정을 위하여 에디트 거리를 사용하며, LCSS방식과 비슷하게 0과 1의 양자화 된 값을 이용하여 유사도를 판단하지만, 이 방식은 LCSS와 달리 궤적내의 갭을 허용하여 LCSS 방식보다 정확한 유사한 궤적을 찾을 수 있는 방식이다.

이들 방식들은 모두 유클리드 공간을 기반으로 하므로 본 논문에서 대상으로 하는 도로 네트워크 공간 상에서의 궤적들 간의 유사도 측정에는 적합하지 않다. 도로 네트워크 공간 상에서의 궤적은 이동 객체가 지나온 도로 세그먼트들의 연속으로 표현되며, 이 경우 연속적으로 동일 도로 세그먼트가 반복적으로 궤적 내에 출현하는 경우는 발생하지 않는다. 따라서 반복을 허용하는 DTW 방식과 ERP 방식은 적용할 수 없다. 또한, LCSS 방식과 EDR 방식은 궤적을 구성하는 도로 세그먼트의 식별자에 대하여 유사도 측정이 가능하나, 서브 궤적에 대한 유사도 측정 과정을 반복적으로 수행해야하므로 이로 인한 성능 저하 문제가 발생할 수 있다.

3. 제안하는 방법

본 장에서는 본 논문에서 제안하는 유사도 측정 함수와 이를 기반으로 하는 클러스터링 기법에 대하여 설명한다. 제 3.1절에서는 도로 네트워크를 대상으로 하는 이동 객체의 궤적을 정의하고, 제안하는 기법의 기본 개념을 설명한다. 제 3.2절에서는 두 궤적간의 유사도 측정을 위하여 도로 세그먼트들의 길이와 식별자 정보를 이용한 새로운 유사도 측정 방식을 제안한다. 제 3.3절에서는 측정된 궤적간의 유

사도 정보를 기반으로 FastMap을 이용한 클러스터링 방식을 제안한다. 마지막으로 제 3.4절에서는 제안된 클러스터링 기법의 응용 분야로 주어진 질의 궤적과 유사한 궤적을 검색하기 위한 질의 처리 기법을 보인다.

3.1 기본 개념

본 논문에서는 이동 객체의 궤적이란 세그먼트의 식별자와 길이의 리스트이며, $T_i = \{(S_1, L_1), \dots, (S_n, L_n)\}$ 으로 표현한다. 여기서, T_i 은 궤적의 식별자이고, $S_j(1 \leq j \leq n)$ 는 세그먼트의 식별자, L_j 는 세그먼트의 길이를 나타낸다. 이와 더불어 궤적 정보와 함께 사용자 정보, 도로 정보 등을 궤적에 대한 부가 정보로 함께 저장, 관리할 수 있다. (그림 1)은 궤적 표현의 예를 나타낸 것이다.

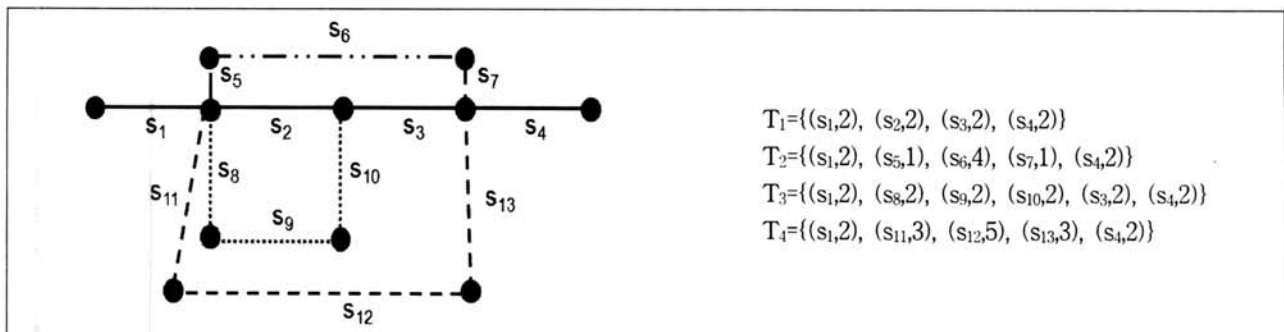
본 논문에서는 데이터베이스에 저장된 이들 대용량 궤적 정보를 대상으로 하는 효과적인 클러스터링 기법에 대하여 논한다. 이를 위하여 먼저 궤적들 간의 유사도를 측정하기 위한 유사도 측정 함수를 제안한다. 다음, 이를 이용하여 측정된 모든 궤적 쌍 간의 유사도와 FastMap 방식[10]을 이용하여 궤적들을 k차원의 점들로 매핑한 후, 이들에 대하여 클러스터링을 수행한다. 마지막으로, 주어진 질의 궤적에 대하여 해당 클러스터를 신속하게 검색하기 위한 질의 처리 기법을 제안한다.

3.2 유사도 측정 함수

제안하는 기법에서 궤적은 문자열로 표현되는 세그먼트 식별자들의 리스트로 구성된다. 주어진 두 궤적간의 유사도 측정을 위하여 문자열간의 거리 함수로 많이 사용되는 ED(edit distance) 방식을 이용할 수 있다. 그러나 비교되는 두 궤적의 세그먼트 개수가 서로 다른 경우 유사도는 세그먼트 개수가 많은 궤적에 의하여 영향을 받는다. 따라서 ED 방식에 의하여 유사 궤적을 검색하는 것은 적합하지 않다. 본 논문에서는 궤적 T_i 와 T_j 간의 유사도를 식(1)의 측정 함수 DSN(disimilarity with number)을 이용하여 계산한다.

$$DSN(T_i, T_j) = \frac{T_i \text{와 } T_j \text{의 비공통 세그먼트의 개수}}{T_i \text{의 세그먼트 개수} + T_j \text{의 세그먼트 개수}} \quad (1)$$

예를 들어, (그림 1)의 궤적들에 대하여 DSN 방식에 의



(그림 1) 궤적 표현의 예

한 유사도 값을 구하면 다음과 같다. T_1 과 T_2 의 세그먼트 개수는 각각 4와 5이고, T_1 과 T_2 의 비공통 세그먼트는 S_2, S_3, S_5, S_6, S_7 이므로 비공통 세그먼트 개수는 5이다. 따라서 $DSN(T_1, T_2)=0.56$ 이다. 같은 방식으로 모든 궤적에 대하여 DSN을 적용하면 다음과 같다.

$$\begin{aligned} DSN(T_1, T_2) &= 0.56, & DSN(T_1, T_3) &= 0.4, \\ DSN(T_1, T_4) &= 0.56, & DSN(T_2, T_3) &= 0.64, \\ DSN(T_2, T_4) &= 0.6, & DSN(T_3, T_4) &= 0.64 \end{aligned}$$

T_1 과 비교되는 T_2, T_3, T_4 에 대한 ED 방식에 의한 유사도는 모두 동일한 3이라는 값이 나온다. 반면, 제안된 DSN 방식에 의한 값은 위에 제시된 바와 같이 T_1 에 대해 T_3 이 가장 유사한 값을 보인다. 실제 T_1 과 T_3 의 공통된 세그먼트가 더 많기 때문에 ED에 의한 유사도 값보다 DSN에 의한 유사도 값이 더 의미가 있다. 이는 DSN 방식이 두 궤적의 세그먼트 개수의 합을 이용하여 두 궤적의 거리를 계산하기 때문이다. 즉, 유사도 계산 시 ED 방식에서와 같이 두 궤적 중에서 세그먼트의 개수가 많은 임의의 한 궤적에 의한 영향을 감소시킬 수 있다. 그러나 DSN 방식 역시 다음과 같은 문제점을 갖는다.

- 1) 세그먼트 식별자의 매치 여부와 개수만을 가지고 유사도를 측정하므로 매치되는 세그먼트의 길이 정보를 무시하게 된다. 예를 들어, DSN 방식에 의하여 궤적 T_1 과 T_2 의 유사도와 궤적 T_1 과 T_4 의 유사도가 같은 값을 갖는다. 이는 T_1 과 T_2 의 비공통 세그먼트 개수와 T_1 과 T_4 의 비공통 세그먼트의 개수가 동일하기 때문이다. 그러나 (그림 1)에 나타난 바와 같이 T_4 의 궤적보다 T_2 의 궤적이 T_1 에 더 근접해 있으므로, T_1 과 T_2 가 더 유사하다고 할 수 있다.
- 2) 동일한 도로라 하더라도 세그먼트를 나누는 기준에 따라 궤적에 포함된 세그먼트의 개수가 달라지므로 유사도가 달라질 수 있다. 제안된 DSN 방식은 세그먼트의 개수에 영향을 받기 때문에 하나의 세그먼트가 다수의 세그먼트로 분할되어 표현될 경우, 유사도 값이 전혀 다른 값을 갖게 된다. 예를 들어, (그림 1)에서 두 궤적 T_1 과 T_2 의 $DSN(T_1, T_2)=0.56$ 이다. 만약, 궤적 T_2 의 세그먼트 s_6 를 3개의 세그먼트로 분할했다면 $DSN(T_1, T_2)=0.64$ 가 된다.

따라서 두 궤적의 유사도를 세그먼트의 개수만을 이용하여 판단하는 방식은 적절하지 않다. 본 논문에서는 이러한 문제점을 해결하기 위하여 세그먼트의 길이를 이용한 유사도 측정 함수 DSL(dissimilarity with length)를 제안한다.

$$DSL(T_i, T_j) = \frac{T_i \text{와 } T_j \text{의 비공통 세그먼트의 길이의 합}}{T_i \text{의 세그먼트 길이의 합} + T_j \text{의 세그먼트 길이의 합}} \quad (2)$$

(그림 1)의 궤적에 대하여 DSL 방식에 의한 유사도 값을 구하면 다음과 같다. T_1 과 T_2 의 세그먼트 길이의 합은 각각

8와 10이고, T_1 과 T_2 의 비공통 세그먼트 S_2, S_3, S_5, S_6, S_7 의 길이의 합은 10이다. 따라서 $DSL(T_1, T_2)=0.56$ 이다. 같은 방식으로 모든 궤적에 대하여 DSL을 적용하면 다음과 같다.

$$\begin{aligned} DSL(T_1, T_2) &= 0.56, & DSL(T_1, T_3) &= 0.4, \\ DSL(T_1, T_4) &= 0.65, & DSL(T_2, T_3) &= 0.59, \\ DSL(T_2, T_4) &= 0.68, & DSL(T_3, T_4) &= 0.7 \end{aligned}$$

DSN 방식에 의한 $DSN(T_1, T_2)$ 과 $DSN(T_1, T_4)$ 의 유사도 값이 동일한 반면에, DSL 방식에 의한 $DSL(T_1, T_2)$ 과 $DSL(T_1, T_4)$ 의 유사도 값은 다름을 알 수 있다. 즉, DSL 방식에 의하여 궤적 T_2 가 궤적 T_4 보다 궤적 T_1 과 더 유사함을 파악할 수 있다. 제안된 DSL 방식은 두 궤적 간에 공통 세그먼트의 개수가 많고, 세그먼트의 길이의 차가 적은 궤적들을 보다 유사하다고 판단한다.

또한, 제안된 방식은 세그먼트를 나누는 기준에 따른 세그먼트 개수의 변화에 영향을 받지 않는다. 예를 들어, 궤적 T_1, T_2 의 유사도 측정 시 s_6 의 세그먼트를 3개로 분할할 경우에도 분할된 세 개의 세그먼트의 길이를 합한 전체의 길이는 4로 분할 전과 길이면에서는 변함이 없다. 따라서 분할 전과 분할 후의 $DSL(T_1, T_2)$ 의 값은 0.56으로 동일하다.

3.3 클러스터링 방법

제 3.2절에서 제안한 유사도 측정 함수를 이용하여 주어진 질의 궤적과 유사한 궤적을 가지는 이동 객체를 검색할 수 있다. 그러나 본 연구의 주목적은 검색된 유사 궤적을 클러스터링하여 보다 유용한 정보를 사용자에게 제공하는 것이다.

궤적 클러스터링이란 궤적간의 유사도를 이용하여 전체 궤적들을 그룹화하는 것을 말하며, 일반적인 클러스터링 방법에서는 클러스터의 무게 중심이라 할 수 있는 중심점의 반복적인 변경에 의해 클러스터를 구성한다. 따라서 궤적간의 상대적인 거리로 클러스터를 구성한다면 중심점이라는 기준이 모호해 진다. 이러한 문제점을 해결하기 위하여 본 연구에서는 FastMap[10]을 이용하여 각 궤적을 k차원 공간상의 한 점으로 표현한 후, 전체 궤적들과 대응되는 점들을 대상으로 클러스터링을 수행한다. 이때, 서로 다른 길이를 갖는 궤적들을 하나의 차원으로 매핑시키기 위하여 본 연구에서는 제 3.2절에서 제안된 DSL 방식에 의해 측정된 두 궤적간의 유사도 값을 이용한다. 여기서, 측정된 DSL 값은 두 궤적간의 거리를 의미한다.

FastMap은 n개의 객체들과 모든 객체쌍간 거리 함수가 주어졌을 때, 이 객체들을 k차원 상의 점으로 매핑하는 기법이다. FastMap을 이용하면, 측정된 두 궤적간의 거리를 이용하여 궤적을 쉽게 k차원 공간상의 점으로 매핑할 수 있다. FastMap을 이용하여 궤적을 k차원의 점으로 변환 방법은 다음과 같다.

FastMap을 이용하여 각 궤적을 k차원의 한 점으로 변환한 후, 변환된 k차원 점들을 대상으로 클러스터링을 수행한다.

단계 1: 궤적 T_i 와 T_j 간의 거리 $DSL(T_i, T_j)$ 의 값 중에서 가장 큰 값을 갖는 두 궤적 T_i 와 T_j 를 찾는다. 여기서, 이들 두 궤적을 T_{min^k}, T_{max^k} 라 한다.

단계 2: 각 궤적 $T_i(1 \leq i \leq n)$ 는 다음 식에 의해 T_{min^k}, T_{max^k} 를 기준으로 하는 k차원의 T_i^k 값으로 매핑된다.

$$T_i^k = \frac{DSL(T_{min^k}, T_i)^2 + DSL(T_{min^k}, T_{max^k})^2 - DSL(T_{max^k}, T_i)^2}{2 \times DSL(T_{min^k}, T_{max^k})}$$

단계 3: 단계 2에 의해 k차원에 매핑된 T_i^k 값들을 이용하여 궤적 T_i 와 T_j 간의 거리 $DSL(T_i, T_j)$ 를 아래 식에 의해 재 계산 한다.

$$DSL(T_i, T_j) = \sqrt{DSL(T_i, T_j)^2 - (T_i^k - T_j^k)^2}$$

단계 4: k를 1씩 증가시켜 차원을 확장하고, 단계 1~3을 반복하여 궤적 T_i 를 $(T_i^1, T_i^2, \dots, T_i^k)$ 로 표현되는 k차원의 한 점으로 변환한다.

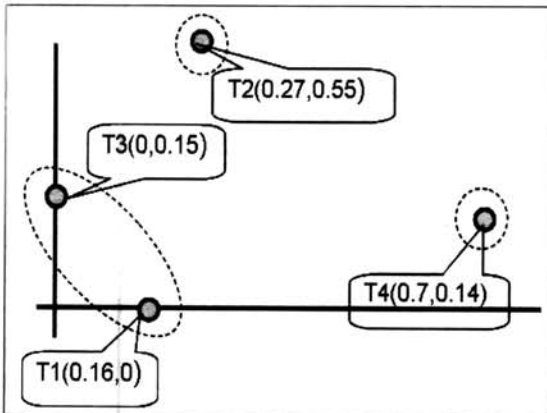
(그림 2)는 FastMap을 이용하여 (그림 1)의 궤적을 2차원의 점으로 매핑한 후, 이를 3개의 클러스터로 구성한 예를 보인다. 여기서 궤적 T_1 의 2차원 점 (0.16, 0)은 궤적 T_1 을 1차 변환하여 얻어진 점 $T_1^1 = 0.16$ 과 2차 변환하여 얻어진 점 $T_1^2 = 0$ 으로 구성된 것이다. 또한 클러스터링 결과를 보면, 유사도가 가장 높은 T_1 과 T_3 가 같은 클러스터로 그룹화되어 있는 것을 확인할 수 있다.

본 연구에서는 클러스터링 과정을 통하여 얻어진 클러스터에 대한 정보를 각 클러스터 C 에 대하여 세그먼트 요약 정보 $S = \langle s_1, w_1 \rangle, \dots, \langle s_n, w_n \rangle$ 와 사용자 요약 정보 $U = \langle u_1, \dots, u_m \rangle$ 를 구성하여 함께 저장 관리한다. 여기서, s_j 는 해당 클러스터 내에 속하는 궤적들 내에 포함된 각 세그먼트를 의미하며, w_j 는 클러스터 내에서 해당 세그먼트의 발생 빈도를 의미하며 식 (3)에 의하여 얻어진다.

발생빈도수

$$= \frac{\text{클러스터 내 해당 세그먼트 리스트가 포함된 궤적의 개수}}{\text{클러스터 내 궤적의 총 개수}} \quad (3)$$

예를 들어, (그림 1)의 각 궤적에 대하여 클러스터링 과정을 수행하여 얻어진 $C_1 = \{T_1, T_3\}$ 이고, 각 궤적 사용자가 각



(그림 2) 클러스터링 예

각 user1, user2라면, 클러스터 C_1 에 대한 세그먼트 요약 정보 $S = \langle \langle s_1, 1 \rangle, \langle s_3, 1 \rangle, \langle s_4, 1 \rangle, \langle s_2, 0.5 \rangle, \langle s_8, 0.5 \rangle, \langle s_9, 0.5 \rangle, \langle s_{10}, 0.5 \rangle \rangle$ 이고, 사용자 요약 정보 $U = \langle \text{user}_1, \text{user}_2 \rangle$ 이다.

3.4 응용

본 절에서는 제 3.3절에서 구성된 클러스터의 세그먼트 요약 정보와 사용자 요약 정보를 이용한 질의 처리 기법을 제안하여 제안된 기법이 실제 응용에 유용하게 사용될 수 있음을 보인다. 제안된 기법에서는 주어진 질의 궤적 내에 포함된 각 세그먼트를 각 클러스터의 세그먼트 요약 정보내의 해당 세그먼트의 가중치 값과 비교한다.

예를 들어, (그림 1)로부터 구성된 클러스터의 세그먼트 요약 정보에 대하여 질의 궤적 $Q = \langle s_1, s_2, s_{10}, s_9, s_8, s_5 \rangle$ 가 주어지면 Q를 구성하는 각 세그먼트에 대하여 클러스터의 세그먼트 요약 정보를 검색한다. 각 세그먼트와 비교된 각 클러스터의 세그먼트 요약 정보내의 해당 세그먼트의 가중치 값들의 합을 구하고, 그 중 가장 높은 가중치 값을 갖는 클러스터를 검색한다. 예에서 Q내의 세그먼트 $s_1, s_2, s_{10}, s_9, s_8$ 은 클러스터 C_1 에 존재하므로, 이때의 $s_1, s_2, s_{10}, s_9, s_8$ 의 가중치를 모두 합하면 $1+0.5+0.5+0.5+0.5=3$ 이 된다. 다음 Q내의 세그먼트 s_5, s_3 는 클러스터 C_2 에 존재하므로 가중치의 합은 $1+1=2$ 이고, 세그먼트 s_1 은 클러스터 C_3 에 존재하므로 가중치의 합은 1이다. 질의 궤적 Q와 매핑되는 클러스터는 가중치 값이 가장 큰 클러스터 C_1 이 된다. 따라서 주어진 질의 궤적에 대하여 클러스터를 재구성하는 추가적인 오버헤드 없이 질의 궤적과 매치되는 클러스터를 검색할 수 있다.

다음, 검색된 클러스터와 연관된 사용자 정보와 도로 정보 등을 사용자에게 함께 제시한다. 예를 들어, 클러스터 C_1 에는 서울 지역에서 유사한 궤적으로 움직인 사용자 A, B, C가 존재하고, 클러스터 C_2 에는 부산 지역에서 유사 궤적으로 움직인 사용자 A, B가 존재한다고 가정한다면, 우리는 사용자 C가 부산 지역을 방문할 경우 클러스터 C_1 에 속한 사용자 A와 B가 부산 지역에서 이동한 경로를 사용자 C에게 추천할 수 있다.

4. 성능 평가

본 장에서는 실험에 의한 평가를 통하여 제안된 클러스터링 기법에 의하여 구성된 클러스터의 정확도를 정량적으로 검증한다. 제 4.1절에서는 실험 환경을 설명하고, 제 4.2절에서는 본 논문에서 제안하는 새로운 궤적 생성 기법을 설명한다. 제 4.3절에서는 실험 결과를 분석한다.

4.1 실험 환경

본 연구에서는 실험을 위하여 개발된 도로 네트워크 기반의 궤적 데이터 생성기를 이용하여 100개와 10,000개의 궤적 데이터를 생성하여 사용하였다. 도로 네트워크 데이터로는 [9]에서 7,035개의 세그먼트로 구성된 올덴버그(Oldenburg) 도로 네트워크를 다운로드 받아 사용하였다. 궤적 데이터

생성을 위하여 출발지 노드와 목적지 노드는 임의 선택하여 총 100개를 사용하였으며, 생성된 궤적 데이터들의 평균 세그먼트의 개수는 25개이며, 평균 궤적의 길이는 300이다.

FastMap[10]을 통하여 k차원 상의 점들로 변환된 궤적들을 대상으로 계층 클러스터링(hierarchical clustering) 방식 [12]을 이용하여 클러스터링 수행한다. 계층 클러스터링 방식은 가장 유사한 객체들을 동일 클러스터로 그룹화하는 방식으로 단계적인 병합 과정을 수행하여 모든 객체들이 하나의 클러스터에 속할 때까지 클러스터링 과정을 반복하는 방식이다. 제안된 기법의 성능 평가를 위하여 유사도 함수의 오차율과 클러스터링의 정확도를 비교 분석한다. 정확도 분석을 위한 비교 대상으로는 사람에 의하여 구성된 클러스터 집합을 사용한다.

실험을 위한 플랫폼으로는 MS Windows XP Professional을 운영 체제로 사용하고, 1GB의 주기억 장치, 120GB 디스크를 갖는 Pentium IV 3GHz의 PC를 사용한다.

4.2 궤적 생성기

궤적 정보를 이용한 많은 연구들이 진행되고 있으나 대용량의 실제 궤적 데이터는 획득하기 어렵기 때문에 대부분의 연구들은 궤적 생성기에 의해 생성된 데이터를 이용하고 있다[1, 2, 14, 16, 20, 25]. 이에 따라, 보다 실제와 가까운 궤적을 생성하기 위하여 여러 가지 궤적 생성기들이 연구되어 왔다[3, 21-23]. 그러나 대부분의 궤적 생성기들은 유클리드 공간을 바탕으로 궤적을 생성하기 때문에 본 연구에서와 같이 도로 네트워크 공간을 대상으로 하는 경우에는 이들을 직접 활용할 수 없다.

따라서, 본 연구에서는 도로 네트워크 공간을 대상으로 하는 새로운 궤적 데이터 생성 기법을 제안한다. 제안된 기법에서는 실제 이동 객체의 움직임과 비슷한 궤적을 생성하기 위하여 이동 객체는 출발지에서 목적지까지 최단 경로에 근접하게 움직인다는 현실 세계의 특징을 반영한다. 이를 위하여, 먼저 출발지 노드와 목적지 노드가 주어지면 이들 간의 최단 경로를 검색하고, 검색된 최단 경로를 구성하는 각 노드들과 연결되어 있는 근접 세그먼트들을 이용하여 목적지 노드까지 도달할 수 있는 근접 최단 경로를 검색하여 이를 궤적 데이터로 생성한다. 이때, 실제 최단 경로와 검색된 근접 최단 경로간의 거리 차가 얼마나 발생하는지를 확률값으로 계산하여 궤적 생성 과정에서의 이동 노드를 선택하기 위한 값으로 사용한다. 본 연구에서는 최단 경로 검색을 위하여는 다익스트라 알고리즘(Dijkstra algorithm)을 사용한다[7]. <표 1>에 제안된 궤적 생성 알고리즘을 보인다.

확률 값을 부여하기 위하여 다음의 세 가지 사항을 고려한다. 첫째, 임의의 한 노드에 연결된 모든 세그먼트들의 확률값의 합은 1이다. 이는 확률의 기본 개념인 모든 확률값의 합은 1이 되어야 한다는 조건을 만족시키기 위함이다. 둘째, 최단 경로와 유사한 경로의 세그먼트일수록 높은 확률 값을 가져야 한다. 또한 최단 경로에 포함되는 세그먼트의 확률 값은 다른 세그먼트들에 비해 상대적으로 높은 값

<표 1> 궤적 생성 알고리즘.

단계 1:	임의의 노드에 연결된 각 세그먼트들을 포함하는 최단 경로 검색 및 길이를 측정한다.
단계 2:	최단 경로 길이를 오름차순으로 정렬 후 순위를 결정한다.
단계 3:	각 최단 경로에 대한 확률 값을 계산한다.
단계 4:	확률 값에 의해 다음에 이동할 노드를 선택한다.
단계 5:	선택 되지 않고 계산된 세그먼트는 그래프에서 삭제한다.
단계 6:	다음 옮겨간 노드가 도착지 노드일 때까지 위의 단계 1~5를 반복한다.

을 가져야 한다. 이는 최단 경로와 근접한 경로로 움직이려는 이동 객체의 특성을 반영하기 위함이다. 마지막으로, 목적지 노드에 가까워질수록 최단 경로를 선택할 확률이 높아야 한다. 목적지 노드에 가까워질수록 이동 가능한 경로의 수는 적어지고, 제한적이게 되므로 최단 경로를 구성하는 경로와 동일하게 움직여야 하기 때문이다. 이러한 확률 값 부여 요건을 만족하는 확률식을 다음과 같이 정의한다.

i번째 세그먼트의 확률값

$$= \frac{\left(\frac{1}{i\text{번째 세그먼트를 포함하는 최단 경로의 길이}^{rank(i)}} \right)}{\left(\sum_{j=1}^n \frac{1}{j\text{번째 세그먼트를 포함하는 최단 경로의 길이}^{rank(j)}} \right)}$$

확률식은 근접 최단 경로를 검색하기 위하여 노드에 연결된 임의의 세그먼트를 이용하여 목적지 노드에 도달 가능한 경우에 해당 세그먼트가 경로로 선택되어질 확률을 나타낸다. 이때 i번째 세그먼트를 포함하는 최단 경로의 길이란 노드에 연결된 i번째 세그먼트를 이용하여 목적지 노드에 도달 가능한 최단 경로의 길이를 말한다. rank(i)란 i번째 세그먼트를 이용하는 최단 경로들의 길이를 오름차순으로 정렬한 후의 순위를 나타낸다. 여기서, rank(i)에 대하여 지수승을 하는 이유는 최단 경로에 근접할수록 더 높은 확률 값을 갖게 하기 위함이다. 또한, 역수를 취하는 것은 확률 값들의 합이 1이 되어야 한다는 조건과 목적지 노드에 가까워질수록 최단 경로에 포함되어 있는 세그먼트의 확률이 커져야 한다는 조건을 만족시키기 위함이다. 따라서, 제안된 확률식은 앞에서 언급한 확률 값 부여 시 고려해야 할 세 가지 조건을 모두 만족시킨다. 본 실험에서는 제안된 궤적 생성기를 이용하여 100개와 10,000개의 궤적을 생성하여 사용한다.

4.3 실험 결과

실험 1과 실험 2에서는 클러스터링을 수행하기 전에 FastMap을 위한 최적의 차원수를 결정하는 실험을 수행한다. 먼저 실험 1에서는 제안된 유사도 함수 DSL을 이용하여 측정된 실제 궤적간의 거리 차이와 FastMap을 이용하여 궤적을 k차원 공간상의 점으로 변환했을 때 궤적들 간의 거리 차이를 이용한 오차율을 다음과 같이 정의하여 사용한다.

$$\text{오차율} = \frac{\sum_{i=1}^n \sum_{j=i+1}^n (DSL(T_i, T_j) - EU(T'_i, T'_j))^2}{n(n-1)}$$

위의 식에서 n 은 전체 계적의 개수를 의미하며, $n(n-1)$ 는 이들 계적을 조합하여 생성 가능한 전체 계적 쌍의 개수를 의미한다. $DSL(T_i, T_j)$ 는 제안된 DSL 함수를 이용하여 계적 T_i 와 T_j 간의 유사도를 측정된 값이며, T'_i 와 T'_j 는 계적 T_i 와 T_j 를 FastMap을 이용하여 변환된 k 차원 공간상의 점을 말하며 $EU(T'_i, T'_j)$ 는 변환된 두 점간의 유클리드 거리를 의미한다.

(그림 3)은 FastMap의 차원수 변화에 따른 오차율을 측정된 결과를 보인다. 실험 데이터로는 100개와 10,000개의 계적 데이터를 사용하였으며, X축은 FastMap의 변환 차원수, Y축은 오차율을 나타낸다. 실험 결과에 따르면, 차원이 커질수록 오차율이 작아지는 것을 볼 수 있다. 100개의 계적 데이터인 경우에는 10차원 이상에서 0.1 이하의 오차율을 보이며, 10,000개의 계적 데이터의 경우에는 50차원 이상에서 0.1 이하의 오차율을 보였다.

실험 2에서는 실제 계적들 간의 유사도 순위가 FastMap으로 변환 후에 변화가 있는지를 실험한다. 이를 위하여 참고 문헌 [8, 13]에서 정의된 KSim을 사용한다.

$$KSim(\tau_1, \tau_2) = \frac{|(u, v) : \tau_1, \tau_2 \text{ agree on order of } (u, v), u \neq v|}{n(n-1)}$$

여기서, τ_1 는 제안된 유사도 함수 $DSL(T_i, T_j)$ 을 이용하여 도로 네트워크 공간 상의 실제 계적 T_i 와 계적 T_j 간의 유사도 측정된 후, 계산된 유사도를 기준으로 순위화한 순위 집합이며, τ_2 는 이들 두 계적을 FastMap에 의하여 k 차원 공간상의 한 점으로 변환한 계적 T'_i 와 계적 T'_j 간의 유사도를 유클리디언 함수 $EU(T'_i, T'_j)$ 를 이용하여 측정된 후, 계산

된 유사도를 기준으로 순위화한 순위 집합이다. KSim은 이들 두 집합 τ_1 와 τ_2 에서의 임의의 유사도 순위 (u, v) 의 상대적 순위가 일치하는 확률을 의미한다.

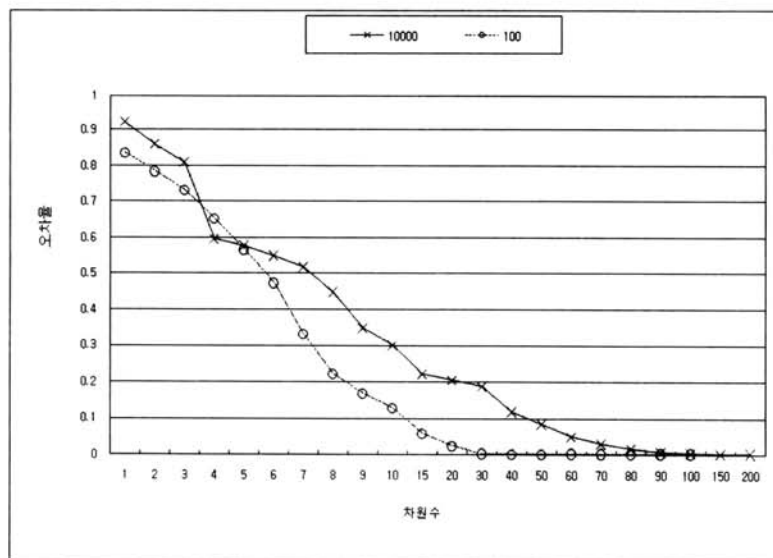
(그림 4)는 FastMap의 차원수 변화에 따른 계적간의 유사도 순위 변화를 측정된 결과를 보인다. X축은 FastMap의 차원수를 나타내며, Y축은 유사도 순위가 변하는 계적의 비율을 나타낸다. 실험 결과에 따르면, 변환 차원이 증가됨에 따라 유사도 순위가 변하는 계적의 수가 확연히 감소하는 것을 알 수 있다.

실험 1과 실험 2의 결과에 따라 본 실험에서는 FastMap의 변환 차원을 50차원으로 고정하고 이후 클러스터링 실험을 수행한다.

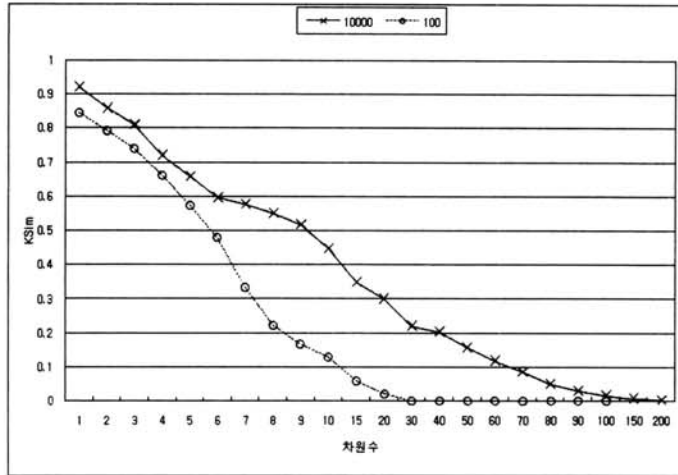
실험 3에서는 정답 집합과 제안하는 기법을 이용하여 얻어진 클러스터링 결과 집합을 비교 분석한다. 정답 집합은 일반인 5명을 대상으로 본 실험에서 사용하고 있는 10,000개의 계적 데이터를 5개부터 50개까지의 클러스터로 구성하게 하여 얻어진 집합이다. 제안된 기법의 클러스터링 결과에 대한 정확도 측정을 위한 기본 척도로 참고 문헌 [11]에서 제안된 정답 집합과 구성된 클러스터들간의 중심 값들을 비교하는 $clusterdistance$ 를 사용한다.

$$clusterdistance(A, B) = \sum_{i=1}^n \sum_{j=i}^n \min(dist(\bar{a}_i, \bar{b}_j))$$

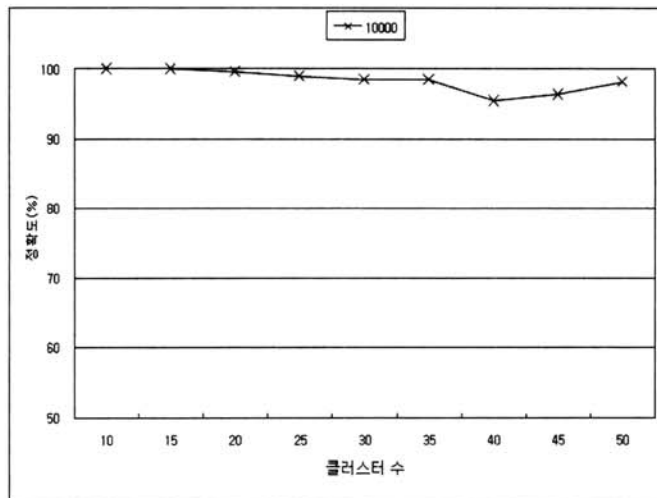
수식은 본 논문에서 제안된 클러스터링 기법을 사용하여 구성된 클러스터 집합 $A = (\bar{a}_1, \bar{a}_2, \dots, \bar{a}_k)$ 와 사람에 의하여 구성된 클러스터 집합 $B = (\bar{b}_1, \bar{b}_2, \dots, \bar{b}_k)$ 간의 거리를 구하는 것으로, 여기서 \bar{a}_i 와 \bar{b}_j 는 클러스터 집합 A와 B를 구성하는 각 클러스터들의 중심값이다. $dist(\bar{a}_i, \bar{b}_j)$ 는 두 클러스터 중심값 \bar{a}_i 와 \bar{b}_j 간의 유클리드 거리를 의미한다. 즉, cluster-



(그림 3) FastMap의 차원수 변화에 따른 오차율



(그림 4) FastMap의 차원수 변화에 따른 궤적간의 유사도 순위 변화



(그림 5) 클러스터의 개수 변화에 따른 정확도의 변화

distance는 두 클러스터 집합간의 유사 정도를 나타낸다.

(그림 5)는 10,000개의 궤적 데이터에 대하여 정확도를 측정 한 결과를 보인다. 이때 정확도는 구성된 클러스터내의 궤적들을 대상으로 동일 클러스터로 구성된 궤적의 비율을 나타낸다. 실험 결과에 따르면 클러스터의 개수가 작을수록 높은 정확도를 보였으며, 클러스터의 개수가 많아지면 정확도가 다소 낮아짐을 알 수 있다. 그러나 제안하는 기법은 95%이상의 높은 정확도를 나타냄으로써, 비교적 정확한 클러스터링의 결과를 얻을 수 있음을 알 수 있다.

5. 결 론

본 논문에서는 도로 네트워크 공간상에서의 이동 객체들을 대상으로 하는 효과적인 유사 궤적 검색을 위한 유사도 측정 함수 및 클러스터링 기법에 대하여 논하였다. 이를 위하여 먼저 궤적을 각 이동 객체가 시간에 따라 지나간 도로 세그먼트들의 리스트로 정의하고, 궤적을 이루는 도로 세그먼트들의 길이와 식별자 정보를 이용한 새로운 유사도 측정 함수를 제안하였다. 클러스터링을 위하여는 측정된 궤적 쌍

간의 유사도를 기반으로 궤적을 FastMap을 이용하여 k차원 공간상의 점들로 사상한 후, 이들을 클러스터링하는 기법을 제안하였다. 또한, 구성된 클러스터와 연관된 사용자 정보, 도로 정보 등을 함께 사용자에게 제공하는 질의 처리 기법을 제시하여 제안된 기법이 실제 응용에 유용하게 사용될 수 있음을 보였다.

제안된 기법의 성능 평가를 위하여 본 논문에서는 도로 네트워크 공간상에서 이동 객체는 출발지에서 목적지까지의 최단 경로에 근접하여 움직인다는 현실 세계의 특징을 반영한 궤적 생성기를 제안하였다. 이를 이용하여 생성된 궤적 데이터들을 대상으로 클러스터링을 수행한 실험 결과에 따르면 제안된 기법은 사람에 의하여 유사 궤적들을 클러스터링한 결과와 비교하여 95%이상의 높은 정확도를 보였다.

참 고 문 헌

[1] V. Almeida and R. Güting, "Indexing the Trajectories of Moving Objects in Networks," *Geoinformatica*, Vol.9, No.1, pp.33-60, 2005.

- [2] S. Brakatsoulas, D. Pfoser, and N. Tryfona, "Modeling, Storing, and Mining Moving Object Databases," In *Proc. Int'l. Symp. on Database Engineering and Applications*, pp.68-77, 2004.
- [3] T. Brinkhoff, "A Framework for Generating Network-Based Moving Objects," *GeoInformatica*, Vol.6, No.2, pp.153-180, 2002.
- [4] L. Chen and R. Ng, "On the Marriage of Lp-norms and Edit Distance," In *Proc. Int'l. Conf. on Very Large Data Bases, VLDB*, pp.1040-1049, 2004.
- [5] L. Chen, M. Ozsu, and V. Oria, "Robust and Fast Similarity Search for Moving Object Trajectories," In *Proc. Int'l. Conf. on Management of Data, ACM SIGMOD*, pp.491-502, 2005.
- [6] S. Chu et al., "Iterative Deepening Dynamic Time Warping for Time Series," In *Proc. SIAM Int'l. Conf. on Data Mining*, 2002.
- [7] E. Dijkstra, "A Note on Two Problems in Connection with Graphs," *Numeriche Mathematik*, Vol.1, pp.269-271, 1959.
- [8] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing top k lists," In *Proc. Int'l. Symp. on Discrete algorithms*, ACM SIAM, pp.28-36, 2003.
- [9] FH Oldenburg/Ostfriesland/Wilhelmshaven, Network-based Generator of Moving Objects, <http://www.fh-oow.de/institute/iapg/personen/brinkhoff/generator/>, 2005.
- [10] C. Faloutsos and K. Lin, "Fastmap: A Fast Algorithm for Indexing, Data-Mining, and Visualization of Traditional and Multimedia Datasets," In *Proc. Int'l. Conf. on Management of Data, ACM SIGMOD*, pp.163-174, 1995.
- [11] D. Goldin, R. Mardales, and G. Nagy, "In Search of Meaning for Time Series Subsequence Clustering: Matching Algorithms Based on a New Distance Measure," In *Proc. Int'l. Conf. on Information and Knowledge Management*, pp.347-356, 2006.
- [12] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Academic Press, 2001.
- [13] T. Haveliwala, "Topic-Sensitive PageRank," In *Proc. Int'l. Conf. on World Wide Web, WWW*, pp.517-526, 2002.
- [14] X. Huang et al., "The Islands Approach to Nearest Neighbor Querying in Spatial Networks," In *Proc. Int'l. Symp. on Spatial and Temporal Databases, SSTD*, pp.73-90, 2005.
- [15] K.-S. Kim et al., "Fast Indexing and Updating Method for Moving Objects on Road Networks," In *Proc. IEEE Int'l. Conf. on Web Information Systems Engineering*, pp.34-42, 2003.
- [16] M. Kolahdouzan and C. Shahabi, "Voronoi-Based K-Nearest Neighbor Search for Spatial Network Databases," In *Proc. Int'l. Conf. on Very Large Data Bases, VLDB*, pp.840-851, 2004.
- [17] J. Lee, J. Han, and K. Whang, "Trajectory Clustering: A Partition-and-Group Framework," In *Proc. Int'l. Conf. on Management of Data archive, ACM SIGMOD*, pp.593-604, 2007.
- [18] D. Papadias et al., "Query Processing in Spatial Network Databases," In *Proc. Int'l. Conf. on Very Large Data Bases, VLDB*, pp.802-813, 2003.
- [19] D. Pfoser, C. Jensen, and Y. Theodoridis, "Novel Approaches to the Indexing of Moving Object Trajectories," In *Proc. Int'l. Conf. on Very Large Databases*, pp.395-406, 2000.
- [20] D. Pfoser and C. Jensen, "Indexing of Network Constrained Moving Objects," In *Proc. Int'l. Symp. on Advances in Geographic Information Systems, ACMGIS*, pp.25-32, 2003.
- [21] Y. Theodoridis and M. Nascimento, "Generating Spatiotemporal Datasets on the WWW," *ACM SIGMOD Record*, Vol.29, No.3, pp.39-43, 2000.
- [22] Y. Theodoridis, J. Silva, and M. Nascimento, "On the Generation of Spatiotemporal Datasets," *LNCS*, pp.147-164, 1999.
- [23] T. Tzouramanis, M. Vassilakopoulos, and Y. Manolopoulos, "On the Generation of Time-Evolving Regional Data," *GeoInformatica*, Vol.3, No.3, pp.207-231, 2002.
- [24] M. Vlachos, G. Kollios, and D. Gunopulos, "Discovering Similar Multidimensional Trajectories," In *Proc. Int'l. Conf. on Data Engineering, IEEE ICDE*, pp.673-684, 2002.
- [25] O. Wolfson et al., "Moving Object Databases: Issues and Solutions," In *Proc. Int'l. Conf. on Scientific and Statistical Database Management, SSDBM*, pp.111-122, 1998.
- [26] Y. Yanagisawa, J. Akahani, and T. Satoh, "Shape-Based Similarity Query for Trajectory of Mobile Objects," In *Proc. Int'l. Conf. on Mobile Data Management*, pp.63-77, 2003.



백지행

e-mail : oracle@zion.hanyang.ac.kr
 2005년 8월 단국대학교 전자컴퓨터학부(학사)
 2008년 8월 한양대학교 전자컴퓨터통신공학과(석사)
 관심분야 : 데이터베이스 시스템, 데이터 마이닝, 공간 데이터베이스/GIS, 이동 객체 데이터베이스/텔레매틱스



원정임

e-mail : jiwon@hanyang.ac.kr
 1992년 2월 한림대학교 전자계산학과(학사)
 1997년 8월 한림대학교 전자계산학과(석사)
 2004년 2월 한림대학교 전자계산학과(박사)
 2000년 3월~2004년 2월 한림대학교 교양교육부 강의전담교수
 2004년 3월~2006년 2월 연세대학교 컴퓨터과학과 연구교수
 2006년 3월~6월 서울대학교 유전자이식연구소 선임연구원
 2006년 7월~현 재 한양대학교 정보통신학부 연구교수
 관심분야 : 데이터베이스 시스템, 데이터 마이닝, XML 응용, 바이오 정보공학, 데이터베이스 보안, 이동객체 데이터베이스/ 텔레매틱스



김 상 욱

e-mail : wook@hanyang.ac.kr

1989년 2월 서울대학교 컴퓨터공학과(학사)

1991년 2월 한국과학기술원 전산학과(석사)

1994년 2월 한국과학기술원 전산학과(박사)

1991년 7월~8월 미국 Stanford University,

Computer Science Department 방문

연구원

1994년 2월~1995년 2월 KAIST 정보전자연구소 전문연구원

1999년 8월~2000년 8월 미국 IBM T.J. Watson Research Center

Post-Doc.

1995년 3월~2000년 8월 강원대학교 컴퓨터정보통신공학부 부

교수

2003년 3월~현 재 한양대학교 정보통신학부 교수

관심분야: 데이터베이스 시스템, 저장 시스템, 트랜잭션 관리, 데

이터 마이닝, 멀티미디어 정보 검색, 공간 데이터베이

스/GIS, 주기억장치 데이터베이스, 이동 객체 데이터베

이스/텔레매틱스, 사회 연결망 분석, 웹 데이터 분석