

# 공간 네트워크 데이터베이스에서 시간제약을 고려한 경로 내 최근접 질의처리 알고리즘

## (In-Route Nearest Neighbor Query Processing Algorithm with Time Constraint in Spatial Network Databases)

김 용 기 <sup>†</sup>      김 상 미 <sup>††</sup>  
(Yong-Ki Kim)    (Sang-Mi Kim)

장 재 우 <sup>†††</sup>  
(Jae-Woo Chang)

**요 약** 최근 공간 네트워크 데이터베이스를 위한 질의 처리 알고리즘에 관한 연구가 많이 진행되어 왔으나, 경로-기반 질의에 대한 연구는 매우 미흡한 실정이다. 공간 네트워크 데이터베이스에서는 이동 객체가 공간 네트워크상에서만 이동하기 때문에 LBS(Location-Based Services) 및 Telematic와 같은 응용에서는 경로-기반 질의가 매우 유용하게 사용된다. 따라서 본 논문에서는 경로-기반 질의의 대표적인 방법인 경로 내 최근접(In-Route Nearest Neighbor, IRNN) 질의처리 알고리즘을 제작하고, 시간 제약을 지닌 새로운 경로 내 최근접 질의처리 알고리즘을 제안한다. 아울러, 성능 분석을 통하여 시간 제약을 지닌 제안하는 질의 처리 알고리즘이 기존 경로 내 최근접 질의처리 알고리즘에 비하여 검색 성능이 우수함을 보인다.

**키워드 :** 공간 네트워크 데이터베이스, 질의처리 알고리즘, 경로 내 최근접 질의

**Abstract** Recently, the query processing algorithm in spatial network database (SNDB) has attracted many interests. However, there is little research on route-based query processing algorithm in SNDB. Since the moving objects moves only in spatial networks, the route-based algorithm is very useful for LBS and Telematics applications. In this paper, we analyze In-Route Nearest Neighbor (IRNN) query, which is a typical one of route-based queries, and propose a new IRNN query processing algorithm with time constraint. In addition, we show from our performance analysis that our IRNN query processing algorithm with time constraint is better on retrieval performance than the existing IRNN query processing one.

**Key words :** Spatial network database, Query processing algorithm, In-Route Nearest Neighbor query

### 1. 서 론

최근 공간 네트워크 데이터베이스를 위한 질의 처리 알고리즘에 관한 연구가 활발히 진행되어 왔다. 그러나 현재 좌표-기반 질의에 대한 연구는 활발히 진행 중이나, 경로-기반 질의에 대한 연구는 매우 미흡한 실정이다. 공간 네트워크 데이터베이스에서는 이동 객체가 공간 네트워크상에서만 이동하기 때문에 경로에 기반을 둔 질의의 유용성이 매우 증대되며, 따라서 경로-기반 질의에 대한 효율적인 질의처리 알고리즘 연구가 필수적이다. 미국 Minnesota 대학의 S.Y JIN과 S. Shekhar는 대표적인 경로-기반 질의로써 경로 내 최근접(In-Route Nearest Neighbor, IRNN) 질의를 정의하였으며[1], 이는 사용자가 운행하고자 하는 경로를 가장 적게 벗어나면서 최근접점을 찾는 데에 초점을 맞추고 있다. 그러나 경로 내 최근접 질의처리 알고리즘은 시간 제약을 고려하지 못하기 때문에 시간 제약 범위내의 POI(Point of Interest)를 검색하지 못하는 문제점이 발생한다. 이러한 문제점을 극복하기 위해서 본 논문에서는 기존 경로 내 최근접 질의처리 알고리즘을 확장하여 시간 제약을 고려한 질의처리 알고리즘을 설계한다. 그리고 이를 구현하여 기존의 경로 내 최근접 질의처리 알고리즘과 본 논문에서 제안하는 시간 제약을 고려한 경로 내 최근접 질의처리 알고리즘에 대해 성능평가를 수행한다. 본 논

- 본 논문은 2005년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2005-041-D00656).
- 이 논문은 2007 한국컴퓨터종합학술대회에서 '공간 네트워크 데이터베이스에서 시간제약을 고려한 In-Route Nearest Neighbor 질의처리 알고리즘'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 전북대학교 컴퓨터공학과  
ykkim@dblab.chonbuk.ac.kr

†† 학생회원 : 대우정보시스템 자판시스템팀 사원  
smkim81@disc.co.kr

††† 종신회원 : 전북대학교 컴퓨터공학과 교수  
jwchang@chonbuk.ac.kr  
(Corresponding author임)

논문접수 : 2007년 9월 27일  
심사완료 : 2008년 1월 4일

Copyright@2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

문에서 제안하는 질의처리 알고리즘의 목적은 기존 경로 내 최근접 질의처리 알고리즘에서 고려하지 못한 시간 제약을 이용하여 텔레메티克斯(telematics), CNS(Car Navigation System), 자동항법장치, L-commerce 등과 같은 실제 응용에서 시간 제약을 이용하여 최적 조건의 POI(Point of Interest) 및 경로를 탐색하는 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로써 경로 기반 질의처리 알고리즘에 관한 연구들을 소개하고, 3장에서는 본 논문에서 설계한 시간 제약을 고려한 경로 내 최근접 질의처리 알고리즘을 제시한다. 4장에서는 시간 제약 경로 내 최근접 질의처리 알고리즘의 성능평가를 수행한다. 마지막으로 제 5장에서는 결론 및 향후연구를 제시한다.

## 2. 관련 연구

기존의 경로-기반 질의는 연속 최근접(continuous Nearest Neighbor) 질의, 연속 k-최근접(continuous k-Nearest Neighbor) 질의, 경로 내 최근접(In-Route Nearest Neighbor) 질의 등으로 구성된다. 일반적으로 이를 경로-기반 질의들의 공통점은 이동 객체가 이동하면서 최근접점 POI를 탐색한다는 점이다. 아울러 이를 경로-기반 질의는 유클리디언 공간에서의 연속 k-최근접(연속 최근접포함) 질의와, 공간 네트워크에서의 연속 k-최근접 질의 및 경로 내 최근접 질의로 분류할 수 있다. 상대적으로 유클리디언 공간에서의 연속 k-최근접 질의에 대한 연구가 다수 진행된 데 반해, 공간 네트워크에서의 연속 k-최근접 질의 및 경로 내 최근접 질의에 대한 연구는 초보적인 단계에 있다.

첫째로, 미국 Illinois 대학에서는 유클리디언 공간에서 연속 최근접 질의에 대해 처음으로 정의하였다[2]. 아울러 미국 Maryland대학에서는 fixed upper bound algorithm을 통해 이동객체가 새로운 최근접 질의를 수행하지 않고 움직일 수 있는 최소 거리를 정의하여, 연속 최근접 질의를 처리하기 위한 알고리즘을 제시하였다[3]. 둘째로, 중국 홍콩 과학기술 대학(HongKong University of Science and Technology)에서는 time parameterized 질의의 개념에 기초하여 연속 최근접 질의를 처리하는 알고리즘을 제시하였다[4]. 아울러 홍콩 과학기술 대학에서는 유클리디언 공간에서 객체의 전체 경로에 대해 하나의 최근접점 질의로 근사화시키는 방법을 통해 연속 k-최근접 질의처리 알고리즘을 제시하였다[5]. 셋째로, 미국 Southern California 대학에서는 공간 네트워크상에서 연속 k-최근접 질의를 처리하기 위한 알고리즘을 제시하였다[6]. 이 알고리즘에서는 Voronoi network diagram을 통해 미리 찾고자 하는 POI들의 네트워크 거리를 미리 계산하고, 이동객체의 경

로가 주어지면 각각의 노드별로, 또는 경로 상에 POI가 있으면 노드와 POI로 경로를 세그먼트로 분할한다. 각각의 세그먼트별로 K개의 POI를 Voronoi diagram으로 찾은 다음, POI의 방향성에 따라 세그먼트의 분할 점을 설정하여 그 분할 점에서의 K개의 후보 셋을 만든다. 이와 같은 방법은 성능이 비효율적이므로 Maryland 대학에서 제시한 upper bound algorithm을 사용하여 분할 점의 개수를 줄이는 방법을 제시하였다. 마지막으로, Minnesota 대학의 연구에서는 경로 내 최근접 질의를 제시하였다[1]. 연속 최근접 또는 연속 k-최근접 질의가 가장 짧은 경로로 갈 수 있는 최근접점을 찾는데 초점을 맞추었다면, 경로 내 최근접 질의는 경로를 가장 적게 벗어나면서 최근접점을 찾는 데에 초점을 맞춘다. 이를 위해 4가지 알고리즘을 제안하였다. 첫째, 단순 그래프(Simple graph) 기반 방법으로, 경로상의 각 노드마다 최근접점을 찾은 후 각 노드의 최근접점을 경유하여 목적지에 도달하는 최단 거리를 계산하여 가장 최단거리가 되는 최근접점을 찾는 방법이다. 이 방법은 노드가 많아지면 비용이 증가한다는 단점이 있다. 둘째, recursive spatial range 질의 기반 방법으로, 현재 경로상의 최근접점을 최단거리 알고리즘(shortest path algorithm)을 통해 찾은 후, 다음 노드에서의 최근접점을 맨 처음에 찾은 최근접점까지의 거리로 영역 질의를 통해 찾는다. 만일 여기서 찾은 최근접점의 거리가 더 작다면 영역 범위를 이 거리로 조절한 후 다음 노드에서 영역질의를 수행한다. 이 방법은 단순 그래프 방법에 비해 비용이 감소하나 각 노드에서의 영역질의를 수행하는 비용이 많이 든다. 셋째, spatial distance 조인 기반 방법으로, 각 노드에서의 영역질의를 수행하는 비용을 줄이기 위해 제안되었다. 이 방법은 노드와 POI에 대한 두 가지의 트리를 가지고 경로 상에 존재하는 각 노드와 POI의 유클리디언 거리 조인을 통해 영역질의 범위를 구하여 최근접점을 구한다. 마지막으로, precomputed zone 기반 방법은 각 노드마다 가장 가까운 POI를 설정하여 각각의 POI에 해당하는 서비스 지역(service zone)을 설정하여 노드에서 POI까지의 네트워크 거리를 미리 계산한다. 따라서 이동객체의 경로가 주어지면 해당경로를 포함하는 서비스 지역을 검색하여 최근접점을 빠르게 찾을 수 있다. 제안된 4가지 방법 중에서 precomputed zone 기반 방법이 가장 좋은 성능을 보이고 있다.

## 3. 시간 제약을 고려한 경로 내 최근접 질의처리 알고리즘

기존 경로 내 최근접 질의는, 경로를 가장 적게 벗어나면서 최근접점을 찾는 데에 초점을 맞추고 있다. 이러한 질의의 예는 “출발 지점에서 목적지까지 표시된 경

로로 운행하고자 계획한 자동차가, 그 경로 내에서 가장 적게 경로를 벗어나면서 방문할 수 있는 주유소를 찾아라”이다. 그러나 기존 경로 내 최근접 질의는 주유소와 같은 POI(Point of Interest) 검색 시, 시간 제약을 고려하지 않기 때문에 시간 제약 범위를 벗어난 POI를 검색하는 문제점이 존재한다. 예를 들면, “주어진 경로로 운행하고자 계획한 자동차가, 출발지점(S)에서 자동차에 기름이 거의 없기 때문에 주어진 경로를 가장 적게 벗어나면서 S에서 20분 이내에 도달할 수 있는 주유소를 찾아라”의 질의는, 기존 경로 내 최근접 질의처리 알고리즘으로 처리하지 못하지만, 실제 응용에서는 매우 유용하게 사용될 수 있다. 즉, 텔레매틱스(telematics), CNS(Car Navigation System), 자동항법장치, L-commerce 등과 같은 응용을 효과적으로 처리하기 위해서는, 시간 제약을 이용하여 최적 조건의 POI 및 경로를 탐색하는 시간 제약을 고려한 경로 내 최근접 질의 처리 알고리즘에 대한 연구가 필수적이다.

이를 위해 그림 1에서, “현재 지점(S)에서 시청(T)까지 최근단 경로를 탐색하고 그 경로를 운행하고자 할 때, 설정된 경로를 가장 적게 벗어나면서 현재 지점(S)에서 20분 내에 방문할 수 있는 주유소를 검색하라”의 질의를 고려 한다. 그림 1에서 A, B, C로 표시된 것은 주유소 POI를 나타내며, 기존 경로 내 최근접 질의처리 알고리즘은 계획된 경로를 가장 적게 벗어나면서 최근접점을 찾는 데에 초점을 맞추기 때문에 주유소 C를 검색한다. 하지만, 만약 현재 자동차가 20분 내에 급유를 해야 운행이 가능하다면, 경로를 가장 적게 벗어나면서 20분이라는 시간 제약 범위 내에 존재하는 주유소 B를 검색해야 한다.

따라서 이러한 응용을 지원하기 위해 기존 경로 내 최근접 질의에 시간제약 개념을 고려한 시간 제약 경로 내 최근접 질의처리 알고리즘의 설계가 요구된다. 한편, 시간 제약 개념을 고려한 경로 내 최근접 질의처리 알고리즘의 설계 시, 시간제약 정도의 강약에 따라 POI를

검색하는 연구가 필요하다. 앞선 예의 경우, 주어진 시간 내에 급유하지 않으면 더 이상 운행이 불가능하므로 시간 제약의 정도가 매우 강하지만, 식당의 경우는 시간 제약의 정도가 상대적으로 강하지 않다. 즉, 검색대상 POI의 성격에 따라 시간 제약 정도에 가중치를 두고, 시간범위 내 이동할 수 있는 거리를 계산하여 최적 조건의 POI를 찾는 질의처리 알고리즘에 대한 연구가 필수적이다. 가중치에 따른 시간범위 내 이동할 수 있는 거리 계산을 위해 정의 1은 다음과 같다. 아울러 가중치  $\alpha$ 는  $0 \leq \alpha \leq 1$  값이다.

**정의 1.** 가중치에 따라 시간범위 내 이동할 수 있는 거리

$$\text{이동 거리} = (\text{속력} \times \text{시간 제약 범위}) / \alpha$$

현재, 사용자의 운행속도가 60km/h, 시간 제약 범위가 20분이라면 주유소와 같이 시간 제약 정도가 매우 강한 경우 가중치  $\alpha$  값을 1이라 한다면, 위와 같은 정의 1의 공식을 이용하여 시간 범위 내 이동할 수 있는 거리를 구하면 20km이다. 반면, 식당의 경우처럼 시간 제약 강도가 약한 경우에는 가중치  $\alpha$  값을 0.5라 한다면, 시간 범위 내 이동할 수 있는 거리는 40km이다. 이처럼, 검색대상 POI의 성격에 따라 가중치를 두고, 시간 범위 내 이동할 수 있는 거리를 유연하게 계산함으로써 효율적인 질의처리를 할 수 있다.

이러한 정의를 바탕으로 설계한 시간 제약 경로 내 최근접 질의처리 알고리즘은 그림 2와 같다. 첫째, 사용자에 의해 지정된 경로내의 노드를 B+-트리에서 검색하여 각 노드마다 가장 가까운 POI를 테이블에 저장한다. 둘째, 정의 1에 제시된 공식을 통해 사용자의 현재 위치에서 시간 제약 범위 내에 이동할 수 있는 거리를 구한다. 셋째, 테이블로부터 시간 제약 범위 내에서 경로에서 가장 가까운 POI를 검색한다. 마지막으로, 사용자로부터 입력받은 시간 제약에 대한 가중치  $\alpha$ 가 0인 경우에는 기존 경로 내 최근접 경로에서 가장 가까운 POI를 전달한다. 아울러  $0 < \alpha \leq 1$ 인 경우에는  $\alpha$ 를 반영한 시간범위 내 이동할 수 있는 거리 내에서 가장 가까운 POI를 전달한다.

#### 4. 성능 평가

본 논문에서 제안하는 시간 제약을 고려한 경로 내 최근접 질의처리 알고리즘(TCIRNN)과, 기존 경로 내 최근접 질의처리 알고리즘 가운데 가장 좋은 성능이 좋은 Precomputed Zone 방법(이하 PCZ로 명명)과 성능 평가를 수행한다. 성능평가에서 사용한 데이터는 Brinkhoff의 알고리즘에서 제공하는 대표적인 공간 네트워크 데이터인 샌프란시스코 만 데이터를 사용하였다[7]. 샌프란시스코 만 공간 네트워크의 에지의 수는 약 220,000

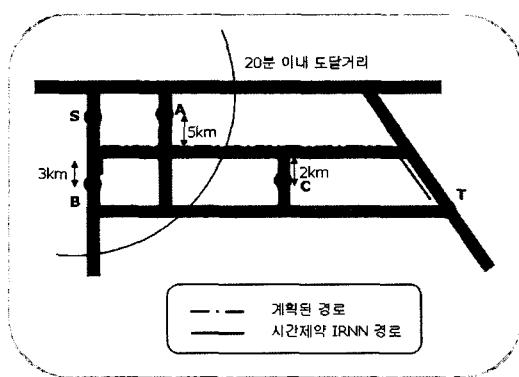


그림 1 시간 제약 경로 내 최근접 질의 개념

---

```

TCIRNN (BaseRoute, POI, Time,  $\alpha$ )
// BaseRoute: 기준 경로, POI: 검색 대상, Time:시간 제약,  $\alpha$ :시간 제약 가중치
1. Result=Load_B+Tree(BaseRoute);
2. if(  $\alpha==0$  )
   2.1 return POI=IRNN(Result);
3. else
   3.1 시간제약 범위 내 이동거리 = (속력 x 시간제약) /  $\alpha$ ;
   3.2 return TCPOI=FindTimeConstrainedPOI(시간제약 범위 내 이동거리, Result);
End TCIRNN

FindTimeConstrainedPOI(시간제약 범위 내 이동거리, Result)
// Result : 기준경로내의 각 노드마다 가장 가까운 POI를 저장한 결과 테이블
1. Limit_length = 시간 제약 범위 내 이동거리;
2. if(Result->length < Limit_length)
   2.1 Limit_length=Result->length;
   2.2 TCPOI=Result->POI;
3. return TCPOI;
End FindTimeConstrainedPOI

```

---

그림 2 시간 제약 경로 내 최근접 질의처리 알고리즘

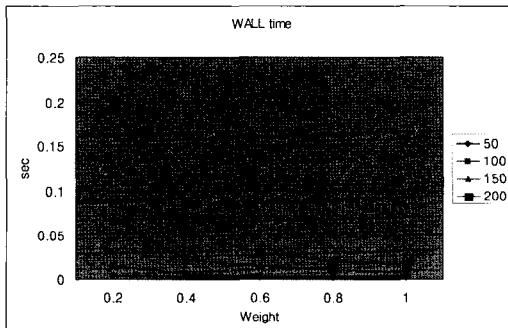


그림 3 시간제약 가중치에 따른 성능평가

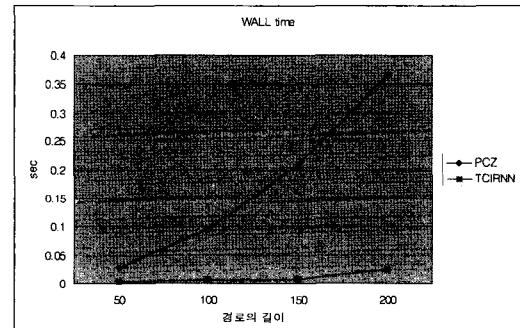


그림 4 시간제약 경로 내 최근접 알고리즘 성능비교

개이며, 노드의 수는 약 170,000개이다. 아울러 Runtime[8] 알고리즘을 사용하여 센프란시스코 만 데이터 내에서 임의의 10846개의 POI를 생성하였다. 성능평가 항목은 경로의 길이를 50, 100, 150, 200개로 증가하면서 각각 전체 응답시간(WALL\_Time)을 측정한다. 그림 3은 제안하는 알고리즘에서 시간제약 가중치( $\alpha$ )를 0.2, 0.4, 0.6, 0.8, 1.0으로 변화하면서, 각각 50, 100, 150, 200개의 경로별로 WALL\_Time을 측정한 결과를 나타낸다. 결과에서 나타나듯이, 시간제약 가중치 값이 증가 할수록 WALL\_time이 줄어드는 것을 알 수 있다. 시간 제약 가중치 값이 커질수록 시간제약 정도가 강하기 때문에, 시간 범위 내 이동거리가 짧아진다. 따라서 검색 해야 할 경로의 길이와 계산 시간이 줄어들기 때문에 성능이 좋아짐을 알 수 있다.

그림 4는 가중치  $\alpha$ 가 1일 때, 경로의 길이를 50, 100, 150, 200개로 증가하면서 제안하는 방법(TCIRNN)과 기존 경로 내 최근접(PCZ) 방법 사이의 WALL\_Time을 측정한 결과를 나타낸다. 그림에서 보듯이 경로

의 길이가 증가 할수록 TCIRNN이 PCZ에 비해 좋은 성능을 보인다. 경로의 길이가 50일 때는 비슷한 결과를 보이지만, 경로의 길이가 100일 때는 TCIRNN의 WALL\_time값이 약 10배, 경로의 길이가 200일 때는 약 13배의 성능이 더 좋다. 이는 TCIRNN은 경로내에서 주어진 시간제약 범위내의 경로만을 선택해서 처리하기 때문에, WALL\_time값이 줄어들고 따라서 성능 향상을 가져온다. 한편 경로가 길어짐에 따라, 즉, 찾어야하는 범위가 많을수록 본 논문에서 제안하는 시간 제약 알고리즘의 성능이 기존 경로 내 최근접 질의처리 알고리즘에 비해 훨씬 우수함을 알 수 있다.

그림 5는 시간 제약 범위를 전체 경로 길이의 10%로 설정한 후에, 시간 제약 범위 이내에 POI가 존재하는 경우의 비율이다. 그림에서 보듯이 시간제약 가중치 값이 0.2인 경우에는, 시간 제약 범위 이내의 POI를 검색하는 것이 거의 가능하다. 반면에 시간 제약 정도가 증가하여 가중치 값이 1에 가까울수록, 시간 범위 내 이동 거리가 줄어들기 때문에 찾는 경우가 줄어든다. 아울러

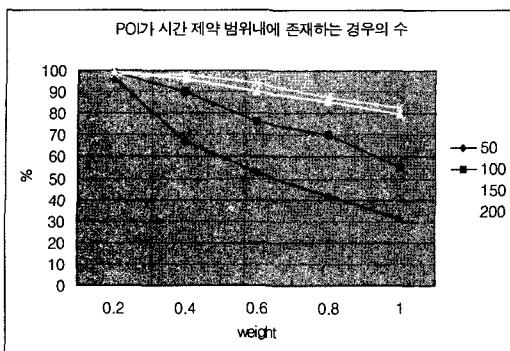


그림 5 시간 제약범위 내 POI가 존재하는 경우의 수

전체 경로의 길이가 길수록 찾아야 하는 범위가 넓어짐에 따라 훨씬 많은 POI가 찾아짐을 알 수 있다.

## 5. 결론 및 향후연구

본 논문에서는 공간 네트워크상의 시간 제약을 고려한 경로기반 질의처리 알고리즘을 제안하였다. 공간 네트워크 데이터베이스에서는 이동객체가 공간 네트워크 상에서만 이동하기 때문에 경로에 기반을 둔 질의의 유용성이 매우 증대되고 있다. 따라서 경로-기반 질의에 대한 효율적인 질의처리 알고리즘 연구가 필수적이다. 대표적인 경로-기반 질의로써 경로 내 최근접 질의가 제시되었으며[1], 이는 경로를 가장 적게 벗어나면서 최근접점을 찾는 데에 초점을 맞추고 있다. 그러나 경로 내 최근접 질의처리 알고리즘은 시간제약을 고려하지 못하기 때문에 시간제약 범위안의 POI를 검색하지 못하는 문제점이 존재한다. 이러한 문제점을 극복하기 위해서 본 논문에서는 기존 경로 내 최근접 질의처리 알고리즘에서 가장 좋은 성능을 보인 PCZ를 기반으로 시간제약을 고려한 경로 내 최근접 질의처리 알고리즘을 제안하였다. 또한, 기존 PCZ 방법과의 성능비교를 통하여, 시간제약을 고려한 제안하는 경로 내 최근접 방법이 시간제약 기준치가 증가함에 따라 탐색성능이 현저히 향상됨을 제시하였다.

향후 연구로는 본 논문에서 제안한 시간 제약을 고려한 경로 내 최근접 질의처리 알고리즘을 현재의 응용 시스템에 접목시켜 실세계에 적용하는 것이다.

## 참 고 문 헌

- [1] S. Shekhar, and J.S. Yoo, "Processing In-Route Nearest Neighbor Queries: A Comparison of Alternative Approaches," Proc. of ACM GIS, pp.9-16, 2003.
- [2] P. Sistla, O. Wolfson, S. Chamberlain, and S.Dao, "Modeling and Querying Moving Objects," Proc. of IEEE ICDE, pp. 422-432, 1997.
- [3] Z. Song, and N. Roussopoulos, "K-Nearest neighbor Search for Moving Query Point," Proc. of SSTD, pp. 79-96, 2001.
- [4] Y. Tao, and D. Papadias, "Time Parameterized Queries in Spatio-Temporal Databases," Proc. of ACM SIGMOD, pp. 334-345, 2002.
- [5] Y. Tao, D. Papadias, and Q. Shen, "Continuous Nearest Neighbor Search," Proc. of VLDB, pp. 287-298, 2002.
- [6] M.R. Kolahdouzan, and C. Shahabi, "ContinuousK Nearest Neighbor Queries in Spatial Network Databases," Proc. of STDBM, pp. 33-40, 2004.
- [7] <http://www.fh-oow.de/institute/iapg/personen/brinkhoff/generator/>
- [8] T. Brinkhoff, "A Framework for Generating Network-Based Moving Objects," GeoInformation, Vol. 6, No.2, pp. 153-180, 2002.