

도로네트워크 기반의 3차원 객체를 위한 연속원근질의처리

김준석* · 이기준** · 장병태*** · 유재준***

Continuous Perspective Query Processing for 3D Objects on Road Networks

Joon-Seok Kim* · Ki-Joune Li** · Byung-Tae Jang*** · Jae-Joon You***

요 약

최근 도로네트워크 환경에서 위치기반 서비스를 제공하는데 있어서 2차원 공간 데이터를 기반으로 하고 있다. 3차원 데이터는 사용자에게 2차원 보다 친근감을 주는 중요한 매체가 되지만, 데이터의 크기가 커지기 때문에 이동단말기의 메모리 공간의 한계로 모든 데이터를 저장할 수 없게 되는 문제점이 발생한다. 본 논문에서는 도로네트워크에서 이동객체에게 3차원 데이터를 제공할 수 있는 연속원근질의를 정의하고 이를 처리하는 방법론을 제안하였다.

주요어 : 연속원근질의, 도로네트워크, 세밀도, 3차원, 이동객체

ABSTRACT : Recently people have been offered location based services on road networks. The navigation system, one of applications, serves to find the nearest gas station or guide drivers to the shortest path based 2D map. However 3D map is more important media than 2D map to make sense friendly for the real. Although 3D map's data size is huge, portable devices' storage space is small. In this paper, we define continuous perspective queries to support 3D map to mobile user on road networks and propose this queries processing method.

Keywords : continuous perspective query, road network, lod, 3D, moving object

*부산대학교 컴퓨터공학과 박사과정
**부산대학교 컴퓨터공학과 교수
***한국전자통신연구원 텔레매틱스 USN 연구단

1. 서 론

대부분의 LBS(location-based service)는 이동 사용자에게 건물이나 도로와 같은 공간객체를 2차원 또는 2.5차원으로 제공한다. 하지만 이들 공간객체를 3차원으로 표현하면 사실적으로 묘사할 수 있게 되어 사용자에게 현실감을 줄 수 있다. 예를 들면, 건물을 2차원으로 표현할 경우는 단순 다각형으로 나타내지만 3차원으로 표현할 경우 기하형태와 질감 정보를 가진 실제 모습을 묘사할 수 있다. 이러한 3차원 데이터는 현실적인 표현을 통해 사용자의 요구를 만족시켜 줄 수 있는 중요한 데이터이다.

이동 사용자에게 3차원으로 공간객체를 제공할 때는 기존의 2차원으로 제공하는 것과 다른 문제점이 발생하게 된다. 2차원으로 표현할 때 보다 데이터의 크기가 커지게 되는데 반면 모바일 단말기기는 데이터를 저장할 메모리의 크기 제약 때문에 모든 3차원 데이터를 저장하는 것이 불가능하다. 한 가지 방법으로 부분적으로 3차원 데이터를 제공하는 것을 생각할 수 있다. 본 논문에서는 이동 객체의 시야에 따라 필요로 하는 데이터를 선택적으로 처리하는 원근질의를 통해서 이 문제에 접근한다. 그리고 질의가 이동하는 점을 고려하여 새로운 타입의 질의인 연속원근질의를 정의하고, 이를 스트림 사용하여 처리하는 방법을 제시한다.

후반부의 논문 구성은 다음과 같다. 2장에서 기존의 관련연구를 살펴보고 연구 방향을 제시한다. 3장에서 원근질의 및

연속원근질의를 정의한다. 4장에서는 실제 도로 상에서 적용할 수 있는 방법과 구조를 제안한다. 5장에서 제안된 질의 처리 방법론 실험을 통해 검증한다. 마지막 장에서 결론을 정리하고 향후 연구에 대해 논의한다.

2. 관련 연구

원근질의는 사용자의 요구에 따른 다른 중요도를 가지는 공간정보를 제공해주는 것이다. 이는 3차원 객체를 묘사하는 과정에서 사람이 인식하지 못하는 범위에서 정보의 양을 최대한으로 줄여서 묘사 성능을 최적화하는 것에 응용된다(Tomas Moller, 2002). 원근질을 처리하는 방법을 제시한 Michael Kofler(2000)와 Jun Li(2001)의 연구들은 모두 범위(range) 기반 세밀도(LOD; Level of Detail) 선택 방법으로 원근질을 처리한다. 세밀도는 정보의 세밀한 정도를 구분한 것으로 여기서는 구간을 크게 세 구간으로 나누어 가까운 거리에 있는 범위는 높은 세밀도(high LOD)로 묘사하고, 그 다음 가까운 범위는 중간 세밀도(medium LOD)로 그리고 먼 거리 범위는 낮은 세밀도(low LOD)로 묘사한다. Michael Kofler(2000)의 연구에서는 기존의 R-tree에 LOD를 접목시켜 데이터 색인을 구성하고 이 색인으로 원근질을 처리한다. Jun Li(2001)의 연구에서도 이와 유사하게 R-tree의 확장인 V-Reactive tree를 이용하여 중요도를 고려하여 데이터 색인을 구성하고, 일반화(generalization)를 통해 다축척(multi-scale)을 지원하는 원근

질의를 처리한다.

하지만 실제 응용에서는 시점이 연속적으로 변한다. 그러나 원근질의에 관련된 연구에서는 이러한 점을 고려하지 않았다. 이와 비슷한 문제점을 해결하기 위해서 연속질의(continuous query) 연구에서 많이 연구되어 왔다.

연속질의는 질의를 한번만 처리하는 속사(snapshot) 질의와는 다르게 시간이 변함에 따라 특정 조건을 계속적으로 만족하는 결과를 계속 갱신하는 질의를 말한다(Douglas B. Terry,1992). 특히, 최근에는 이동객체를 다루기 위한 이동객체 데이터베이스 시스템에 대한 연구가 활발해지면서 위치 기반의 연속질의 처리에 대한 방법론들이 제안되고 있다. <표 1>은 위치 기반 연속질의에 대한 연구를 이동패턴과 질의형태로 분류한 것이다.

Douglas B. Terry(1992)의 연구에서는 고정객체나 이동객체를 대상으로 속사질의의 순서열로 동적 질의를 정의하고 점진적인 질의 실행을 통하여 연속 영역질을 처리하는 방법을 제시하였다. 그러나 Ying Cai(2002)의 연구에서는 고정된 질의 영역에 대하여 질의 영역을 여러 개로 분

할시키고 각 영역에 존재하는 이동 객체를 연속적으로 모니터링을 하는 방법을 제안하였다. 또한, Dragan Stojanovic(2006)의 연구에서 도로라는 국한된 공간에서 객체나 질의가 움직이거나 고정인 모든 경우를 고려하여 연속 영역질을 처리하는 방법을 제안하였다.

이 외에도 Sunil Prabhakar(2002)와 Mohamed F. Mokbel(2004)의 연구에서 여러 개의 연속 영역질을 처리하기 위한 기법을 보여준다. Sunil Prabhakar(2002)의 연구에서 소개된 Q-Index는 객체 대신에 여러 개의 질의영역을 이용하여 색인 구조를 생성한 후 일정 시간 단위로 각 질의영역에 포함되는 이동객체를 찾는다. 그러나 Q-Index 방법은 고정된 영역질을 대상으로 하고 있어 움직이는 영역질을 처리하는 경우에는 상당히 성능이 저하된다. 그래서 Mohamed F. Mokbel(2004)의 연구에서 격자를 이용한 해시 기반의 알고리즘을 이용하여 객체와 질의의 조인(join)연산을 통하여 동시에 요청되는 다수의 연속 영역질을 처리하는 방법을 제안하였다.

또 다른 질의형태인 최근접 질의에 관한 연구들도 연속질을 처리하는 방법들

<표 1> 연속질의의 이동패턴과 유형 분류

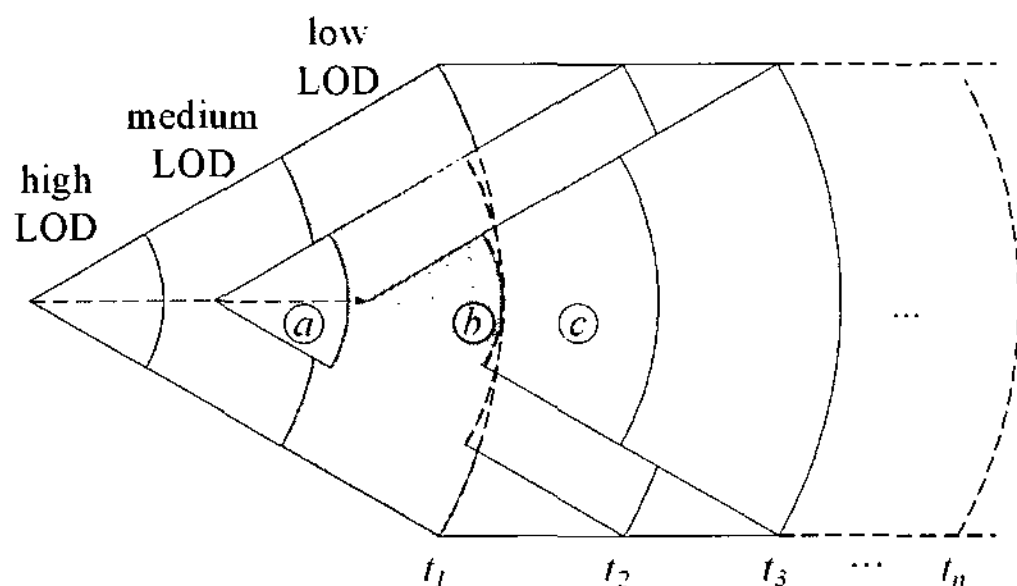
패턴 (질의-객체)	질의유형	범위	최근접
고정-이동		Ying Cai(2002), Dragan Stojanovic(2006) Sunil Prabhakar(2002), Mohamed F. Mokbel(2004)	Xiaopeng Xiong(2005), Kyriakos Mouratidis(2006)
이동-고정		Iosif Lazaridis(2002), Dragan Stojanovic(2006), Mohamed F. Mokbel(2004)	Yufei Tao(2002), Xiaopeng Xiong(2005), Mohammad R. Kolahdouzan(2004), Hyung-Ju Cho(2005)
이동-이동		Iosif Lazaridis(2002), Dragan Stojanovic(2006), Mohamed F. Mokbel(2004)	Xiaopeng Xiong(2005)

을 제안하는데, 질의의 형태가 본 연구의 질의영역과는 다소 차이를 보이기 때문에 자세히 언급하지 않는다(Xiaopeng Xiong, 2005; Kyriakos Mouratidis, 2006; Yufei Tao, 2002; Mohammad R. Kolahdouzan, 2004; Hyung-Ju Cho, 2005).

공간 객체를 다루는 연속질의 처리방법들은 그 질의의 종류가 영역질의 또는 최근접 질의 형태로 국한되어 있다. 원근질의는 단일 세밀도로(특히 high LOD) 처리하는 영역 질의와는 다르게 다중 세밀도로 처리하는 것이 필요하다. 그리고 원근질의 결과가 많기 때문에 연속 질의의 결과가 빈번하게 바뀌는 점을 고려해야 한다.

3. 연속원근질의의 개념

본 논문에서는 연속원근질을 원근질의와 연속질의의 조합으로 정의한다. [그림 1]에서는 시간이 $t_1, t_2, t_3, \dots, t_n$ 으로 변함에 따라 원근질의의 위치가 바뀌어 결과가 바뀌는 것을 보여준다. b 는 낮은 세밀도에서 중간 세밀도, 높은 세밀도로 바뀌고, c 는 시야에서 보이지 않다가 t_2 시간부터 낮은 세밀도로 보이기 시작한



[그림 1] 연속원근질의의 예

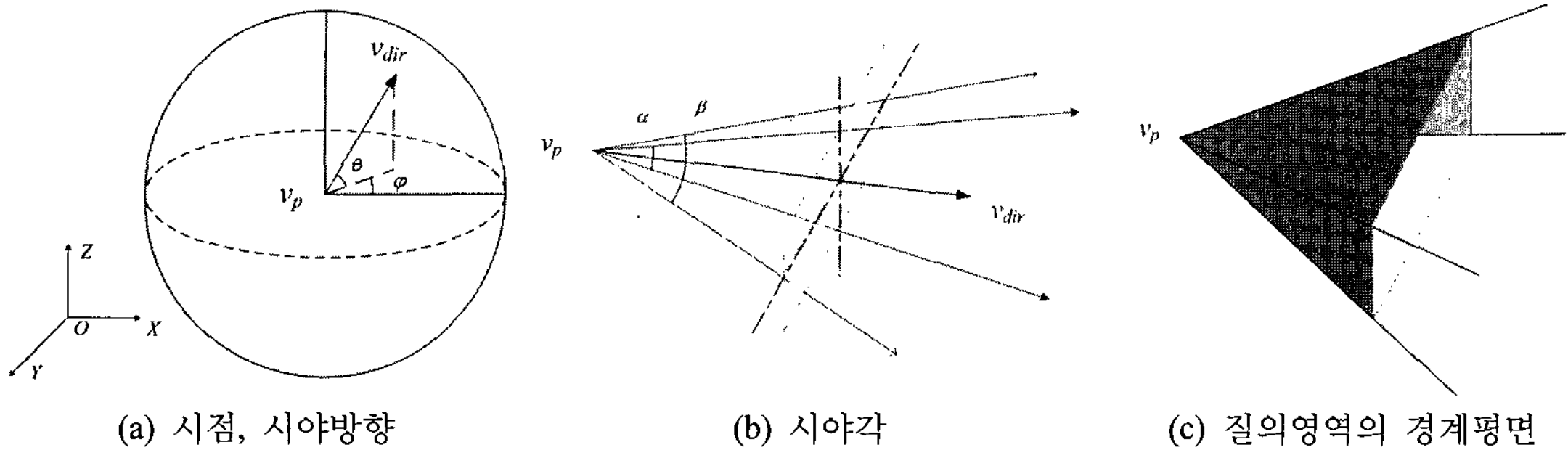
다. a 는 t_2 시간부터 시야에서 벗어난다. 원근질의에 대한 정의를 바탕으로 시간의 연속성을 추가하여 연속원근질을 확장하여 정의한다.

3.1 원근질의

원근질의는 시각화 영역 내에 있는 공간 객체들을 찾아내고 각 공간 객체의 세밀도를 결정하는 질의를 말한다. 이 때 시각화 영역은 공간영역질의로 정의되고, 세밀도는 세밀도 함수에 의해 결정된다. 우선 공간영역질이는 세 가지의 매개변수를 이용하여 생기는 영역을 질의영역으로 가지게 되는데, 세 가지 매개변수는 다음과 같다.

- 시점: 3차원 공간에서 한 점으로 표현된다. [그림 2.a]에서 $v_p = (x, y, z)$ 로 나타낸다.
- 시야방향: 시점을 중심으로 3차원 공간에서 하나의 방향을 말하는데, [그림 2.a]에서 $v_{dir} = (\theta, \varphi)$ 의 두 각으로 나타낸다. θ 는 (x, y) 평면과 이루는 각이다. φ 는 z 축의 양의 방향 중심축으로 (x, y) 평면에서 회전각이다.
- 시야각: 시점과 시야방향에서 상하(α), 좌우(β) 두 각을 나타내는 것으로 [그림 2.b]에서 $v_{ang} = (\alpha, \beta)$ 으로 표현한다.

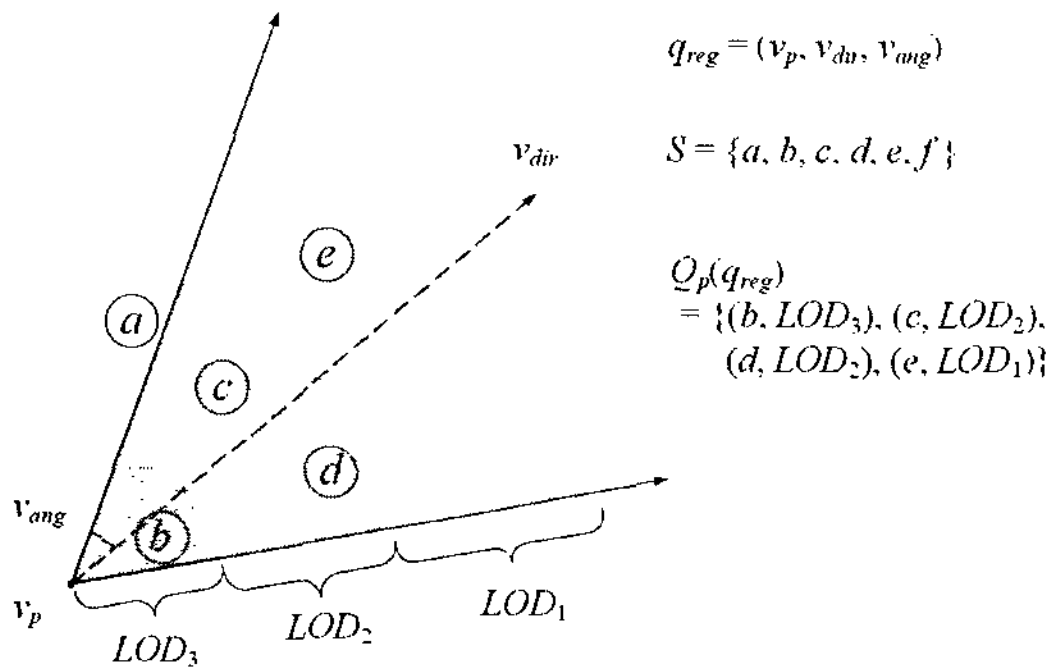
세 가지 매개변수를 가지면 [그림 2.c]와 같은 평면으로 둘러싸인 공간이 생기게 되는데 이를 질의 영역 $q_{reg} = (v_p, v_{dir}, v_{ang})$ 으로 정의한다. 이 때 원근질의의 정의는 다음과 같다.



[그림 2] 원근질의 매개변수와 질의영역

정의 1. 원근질의(Perspective Query)

$$Q_p(q_{reg}) = \{(s, l) | s \in S, l \in L\}$$



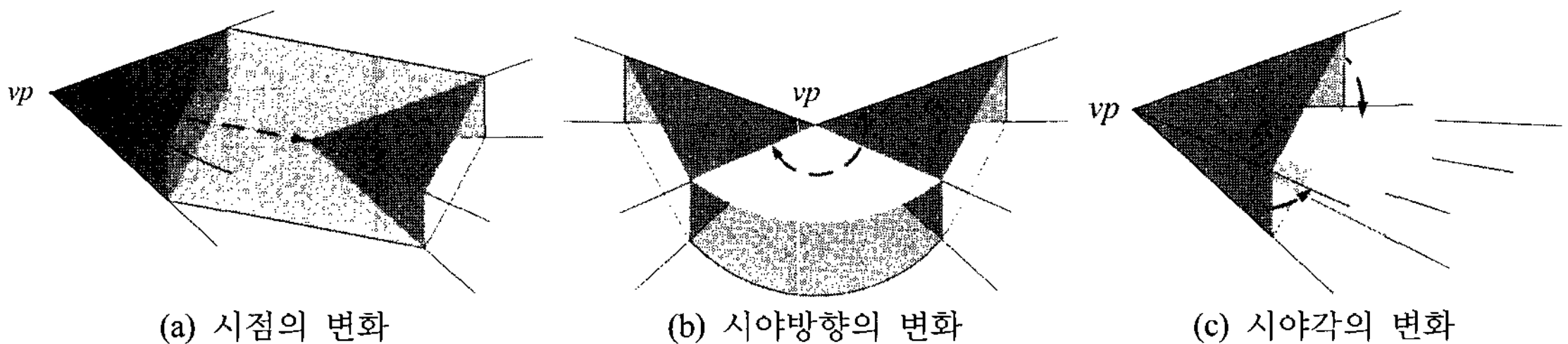
[그림 3] 원근질의의 예

[그림 3]은 원근질의와 질의 결과를 보여주는 예제이다. a, b, c, d, e 다섯 개의 공간 객체가 존재하고, 질의영역의 매개변수 $q_{reg} = (v_p, v_{dir}, v_{ang})$ 가 주어질 때,

원근질의의 결과는 $\{(b, LOD_3), (c, LOD_2), (d, LOD_2), (e, LOD_3)\}$ 의 집합으로 나타난다. a는 질의영역 밖에 있으므로 결과에서 제외된다. b, c, d, e는 각각 $LOD_3, LOD_2, LOD_2, LOD_1$ 의 범위에 존재하므로 질의를 만족한다.

3.2 연속원근질의

시간이 변함에 따라 원근질의 또한 계속 바뀌는데, 이를 갱신하는 질의를 연속원근질의라 한다. 영역질의나 최근접 질의가 질의점의 이동에 따라 연속질의가 되는 것처럼 원근질의도 매개변수 $q_{reg} = (v_p, v_{dir}, v_{ang})$ 가 시간에 따라 변하게 되면 연속원근질의가 된다. [그림 4]는 매개변수가 변함에 따라 시야의 영역이 변하는



[그림 4] 원근질의의 매개변수의 변화

것을 나타낸 것이다.

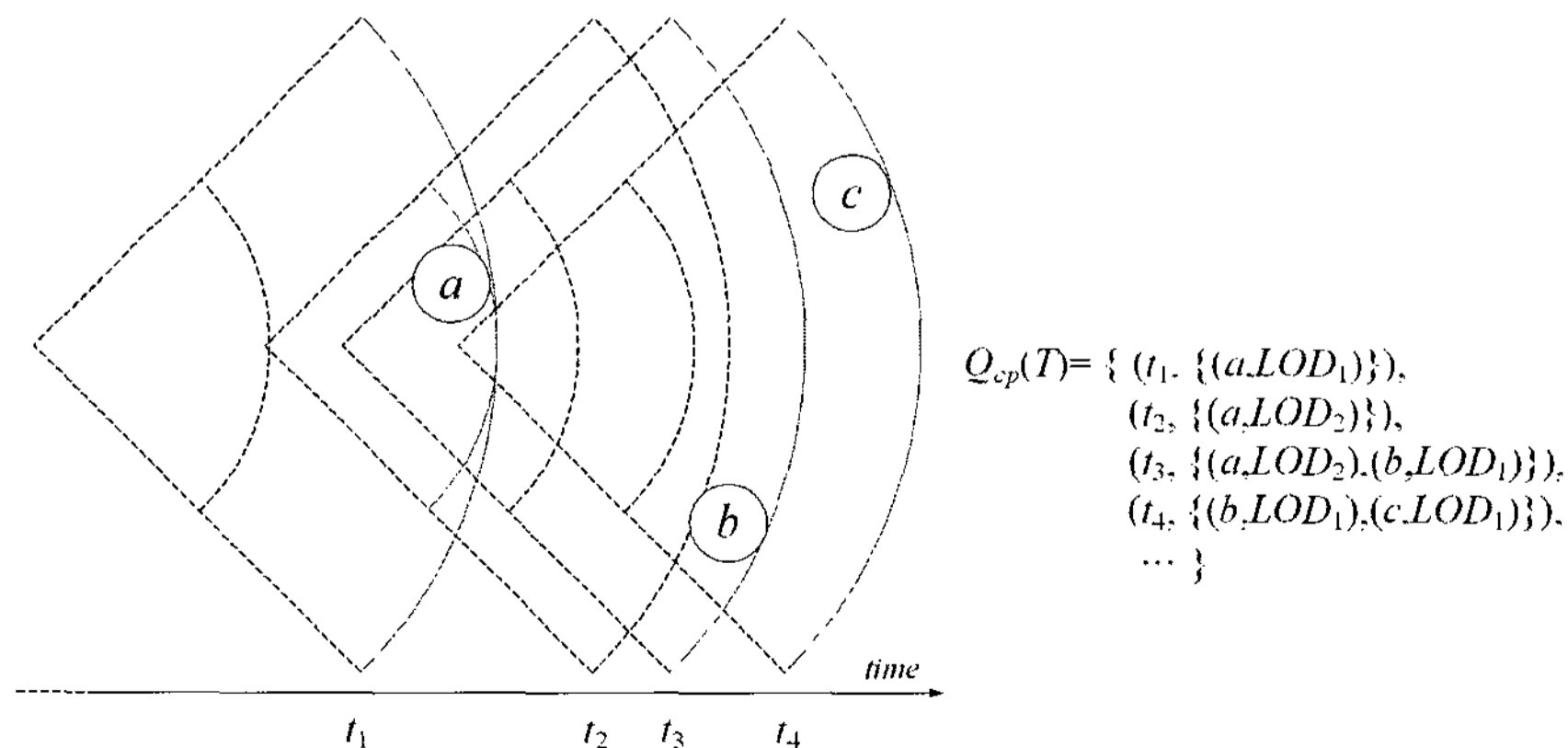
결과적으로 연속원근질의는 매개변수가 시간에 따라 변하는 함수로 정의할 수 있다. 어떤 t 시간에 시점, 시야 방향, 시야 각은 각각 $v_p(t)$, $v_{dir}(t)$, $v_{ang}(t)$ 이 되고, 질의영역은 $q_{reg}(t)$ 으로 정의한다. t 시간에 원근질의는 $Q_p(q_{reg}(t))$ 가 된다.

원근질의의 결과가 변하여 갱신이 필요한 시간을 갱신 시간(update time)이라 한다. 이 시간들을 특정 시간 구간(T)에서 만족하는 연속 원근질의를 다음과 같이 정의한다.

정의 2. 연속원근질의(Continuous Perspective Query)

$$Q_{cp}(T) = \{(t, Q_p(q_{reg}(t))) | t \in T\}$$

[그림 5]는 갱신 시간과 연속원근질의를 설명해주는 그림이다. 시간이 지나면서 원근질을 수행하다가 이전 원근질의 결과가 바뀌는 시간이 t_1 , t_2 , t_3 그리고 t_4 이다.



[그림 5] 갱신 시간과 연속원근질의

4. 질의처리

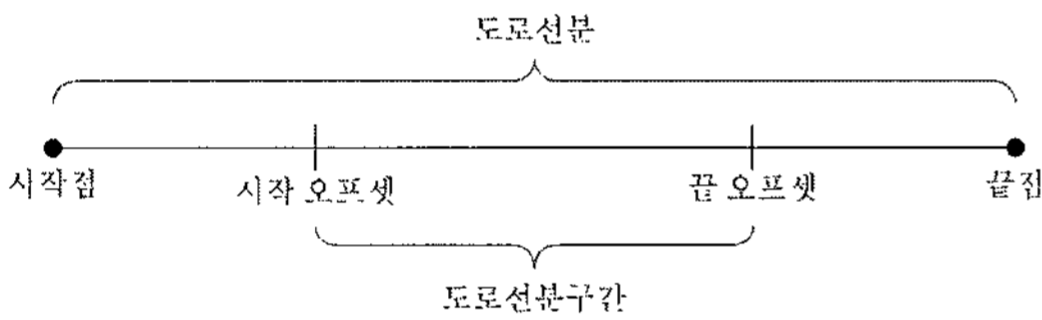
4.1 도로네트워크 기반의 연속원근질의

이 절에서는 앞서 정의한 연속원근질의를 바탕으로 제약 조건을 적용한 연속원근질을 정의한다. 앞에서 언급한 바와 같이 도로 상에서 3차원 지도를 이용하면 사용자에게 높은 친밀도를 제공하기 때문에 중요한 매체로 작용한다. 따라서 연속원근질을 도로의 영역(domain)에 맞춰서 적용할 필요가 있다.

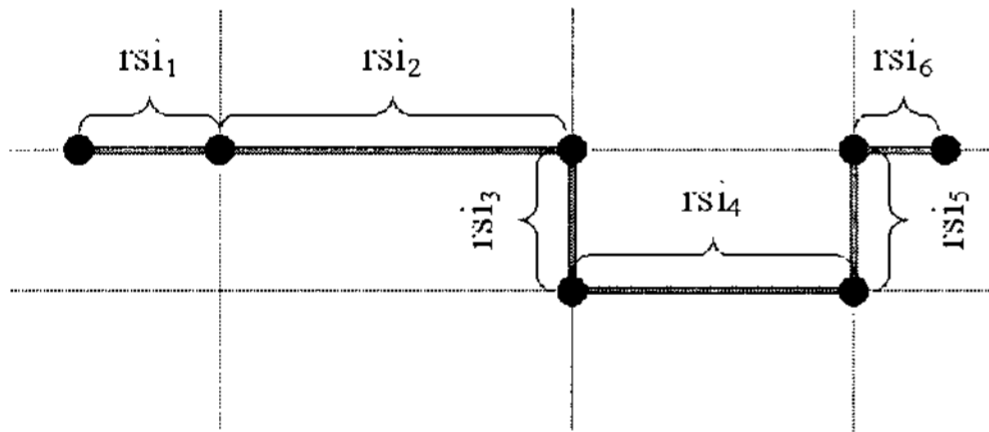
차량과 시점은 동일한 위치이기 때문에 시점은 도로 위에 고정되고, 시야 방향은 진행 방향으로 고정된다. 도로 위에서의 차량은 특정 시간에 특정 위치에 있게 된다. 공간 객체의 위치는 고정되어 있기 때문에 시점의 위치와 방향에 대한 값으로 질의가 결정된다. 도로에서의 위치는 $p = (segment, offset)$ 으로 나타내고, 질의 영역 매개변수는 $q_{reg} = (p, direction)$ 으로 정의된다.

시간적인 연속 개념을 도로에서는 공간적인 구간으로 변환하여 문제를 해결한다. 연속원근질의에서 시간의 구간(T)에 대응되는 구간이 도로네트워크 공간에서도 필요하다. 도로는 도로선분으로 구성되므로, 도로의 구간은 도로선분 구간들의 순서열(sequence)로 나타낼 수 있다. 다음 [그림 6]과 [그림 7]은 도로선분구간(road segment interval)과 도로선분구간 순서열을 나타낸 것이다.

연속원근질의의 갱신 시간(update times)



[그림 6] 도로선분구간(rsi)의 예



[그림 7] 도로선분구간(rsi)의 순서열

은 도로에서는 갱신 위치(change points)로 대응된다. 본 논문에서는 도로네트워크 기반의 연속원근질을 정의하기 위해 질의를 도로 선분 단위로 나누어 해결한 후에 결과를 합치는 방법을 택한다. 먼저 도로 선분에서의 연속원근질을 정의하고 이를 도로네트워크에서 합치는 방법으로 다음과 같이 정의한다.

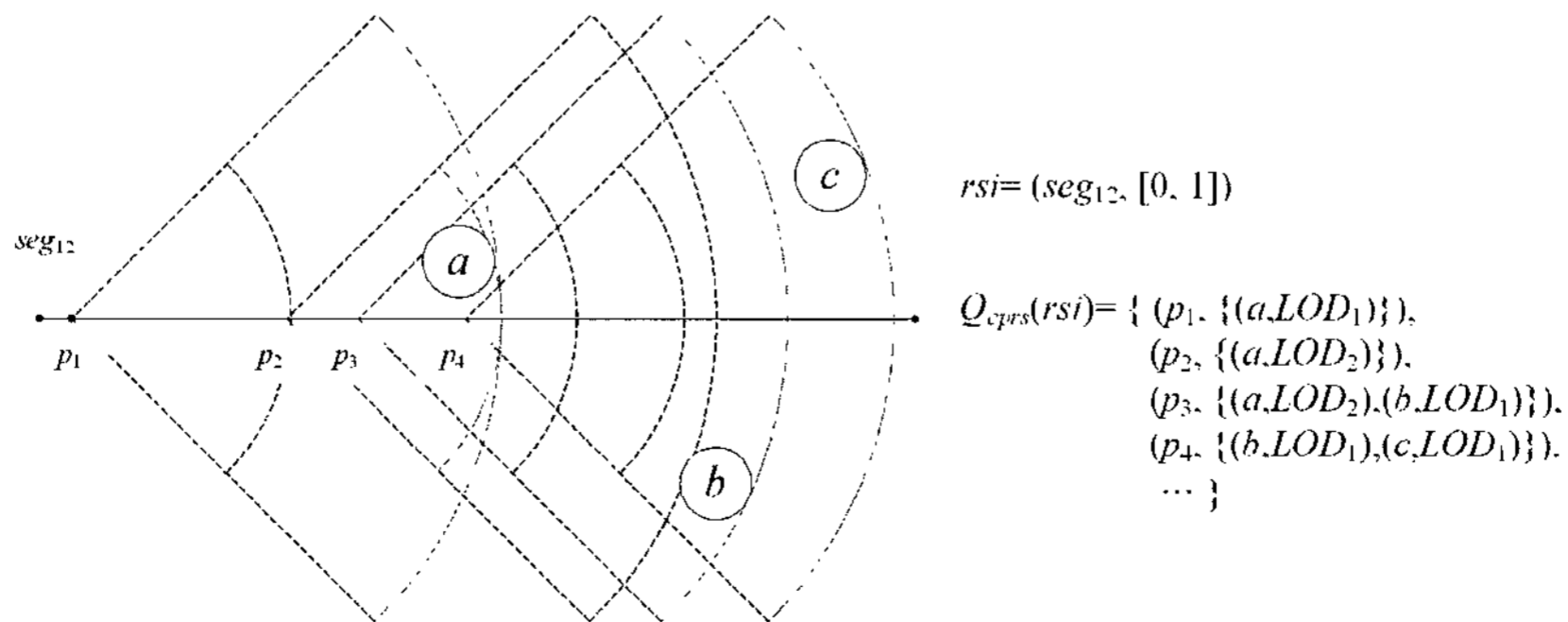
정의 3. 도로선분 기반 연속원근질의 (Continuous Perspective Query on Road Segment)

$$Q_{cprs}(rsi) = \{(p, Q_p(p) | p \in CP(rs_i))\}$$

정의 4. 도로네트워크 기반 연속원근질의 (Continuous Perspective Query on Road Networks)

$$Q_{cprn}(SEQ_{rsi}) = \bigcup_{rsi \in SEQ_{rsi}}^n Q_{cprs}(rsi_i)$$

[그림 8]은 갱신 위치와 도로선분위에서 연속원근질을 설명해주는 그림이다. [그림 5]와 거의 동일한데, 시간 구간 T 가 도로선분 구간 rsi 로 바뀌고, 갱신 시간



[그림 8] 갱신 위치와 도로선분에서 연속원근질의

이 갱신위치로 대체된다. 시간이 지나면서 원근질의 수행하다가 이전 원근질의 결과가 바뀌는 위치가 p_1, p_2, p_3 그리고 p_4 이다.

정의 5. 갱신(Update)

$$U = \{(p, s, LOD, op)\}$$

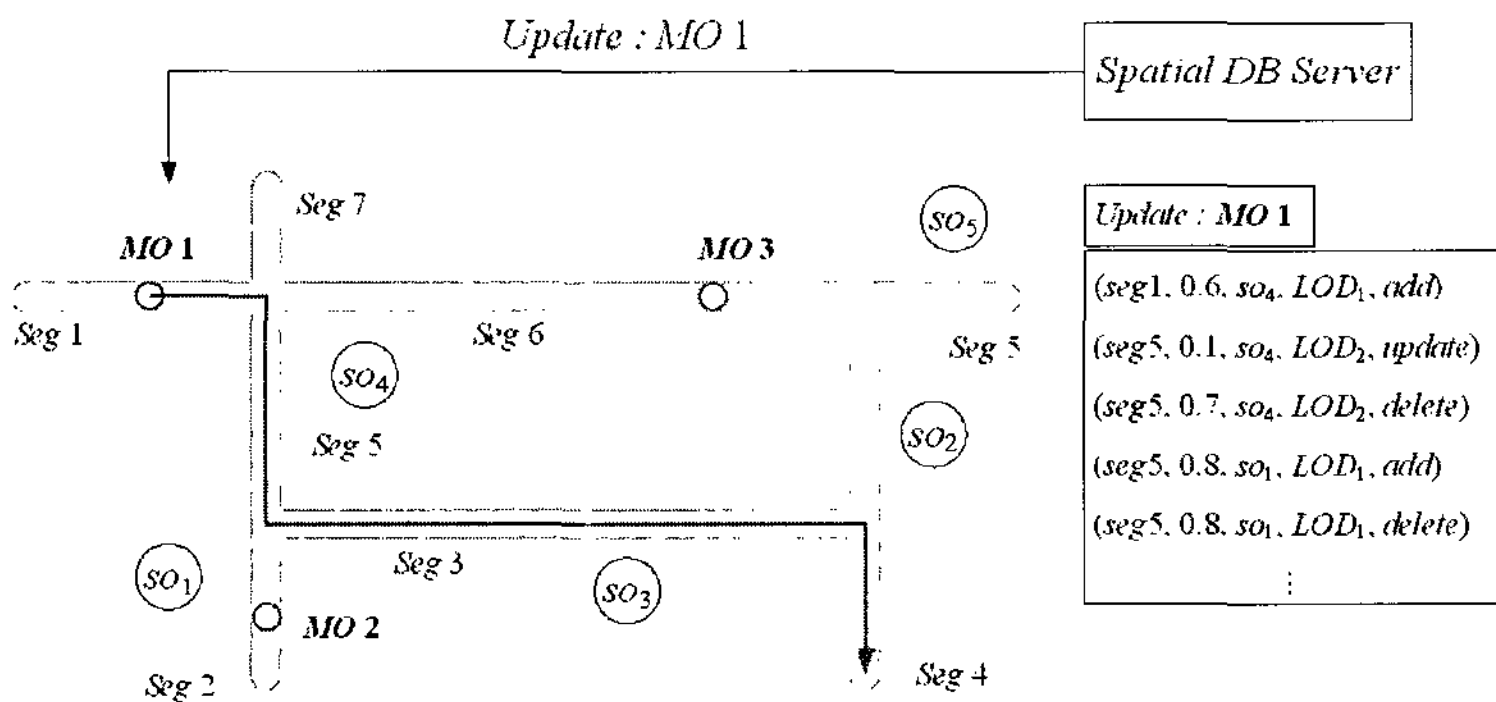
갱신정보는 도로에서의 위치 p 와 공간 객체, 그것의 그리고 갱신 op 로 구성된다. op (operation)은 추가(add), 삭제(remove) 그리고 갱신(update) 중에 한 가지 연산을 사용한다. 이해를 돕기 위하여 다음과 같은 전체적인 구조를 제시한다. [그림 9]와 같이 3차원 공간 DB 서버는 도로의 각 차량에게 질의를 받고, 처리된 결과를 스트림의 형태로 각 차량에게 무선통신으로 전달된다. 이 때 이동객체 $MO1$ 이 $Seg1, Seg5, Seg3, Seg4$ 의 순서로 도로선분을 이동하고 하고 있다. $MO1$ 는 $Seg1$ 위치부터 초기 지도와 이동경로에 따라 갱신 정보를 얻게 된다.

4.2 도로선분 단위의 동적인 질의처리

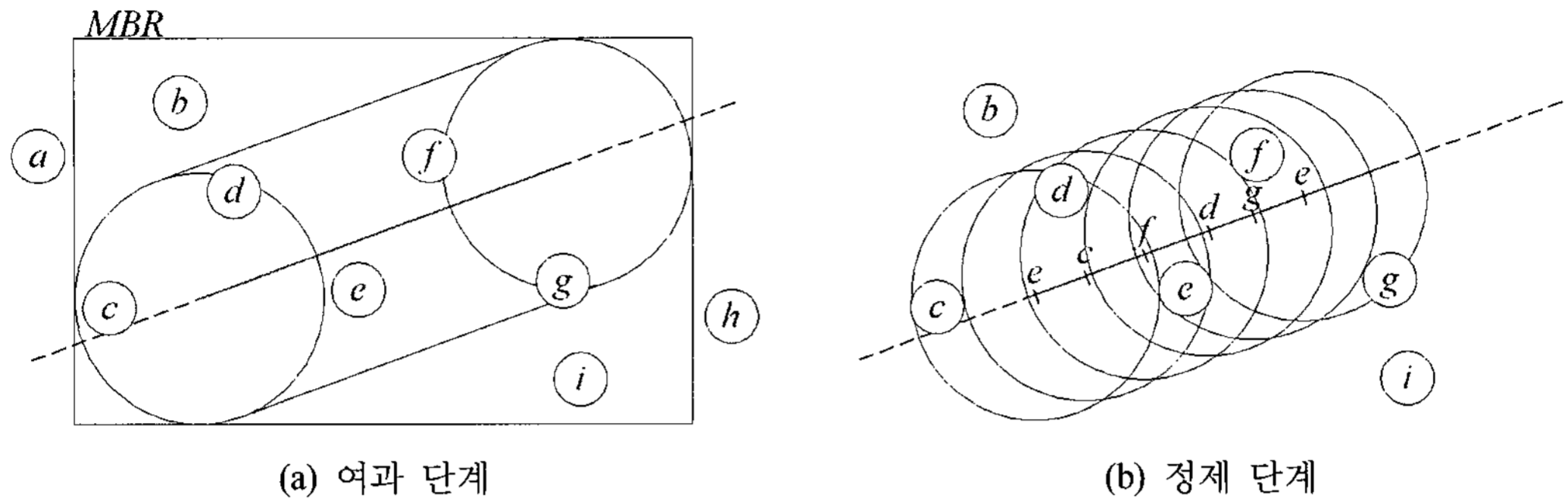
도로 선분단위 질의처리는 다음의 단계들로 구성된다.

- 단계 1: 클라이언트가 도로선분구간에 진입하면 이를 서버에게 알린다.
- 단계 2: 서버는 도로선분구간으로부터 질의 영역을 계산한다.
- 단계 3: 서버는 질의 영역에서 갱신 위치를 찾는다.
- 단계 4: 서버는 갱신 위치마다 갱신(u)을 계산한다.
- 단계 5: 서버에서 결과를 정렬시켜 클라이언트로 전송한다.
- 단계 6: 단계 1에서 5를 계속 반복한다.

[그림 10]은 문제를 단순화시켜 하나의 세밀도에서 질의처리를 설명하는 그림이다. [그림 10.a]는 단계 2와 3에서 설명한 바와 같이 질의영역을 계산하고 공간객체를 찾는 것을 보여준다. a, h 는 MBR 을 벗어나기 때문에 후보객체에서 제외된다. [그림 10.b]는 단계 4와 5를 나타내는 것



[그림 9] 3차원 데이터베이스 서버의 시스템의 환경



[그림 10] 도로선분 단위 질의처리 과정

으로, b 와 i 는 질의영역에서 벗어나서 결과에서 제외된다. 그리고 결과를 만족하는 공간객체 (c, d, e, f, g) 들의 갱신 위치를 계산하고, 이를 정렬된 목록으로 구성하여 클라이언트로 전송하게 된다. 클라이언트는 결과로 받은 이 정렬된 목록을 통해 도로선분구간을 이동하면서 현재 위치에 맞도록 갱신을 트리거(trigger)시킨다.

4.3 정렬된 오프셋 목록(Sorted Offset List)

도로와 공간객체의 위치가 고정되어 있다는 점을 감안하면 갱신의 위치가 변하지 않는 것을 알 수 있다. 따라서 반복적으로 갱신 위치를 찾는 작업을 수행할 필

요가 없이 각 도로마다 갱신 작업을 미리 수행하고, 이를 특정 자료구조로 저장하면 반복된 처리과정에 따른 부하를 막을 수 있다.

본 논문에서는 전처리과정을 통해 효과적으로 연속원근질의를 처리할 수 있는 자료구조를 제안한다. [그림 11]은 정렬된 오프셋 목록으로 저장한 구조를 보여준다. 첫 번째 *Mapping Table*에서 도로선분의 식별자와 정렬된 목록을 저장할 디스크 페이지의 주소를 기록한다. 두 번째 각 도로에 해당하는 디스크 페이지에서 정렬된 목록을 계산하여 [그림 11]의 오른쪽과 같이 갱신 정보를 기록한다. 클라이언트로 전송할 데이터의 크기에 맞게

<i>MappingTable</i>		231				
<i>RSI ID</i>	<i>Page No.</i>	<i>RSI ID</i>	1440			
...	...	<i>displacement</i>	<i>SOID</i>	<i>LOD</i>	<i>operation</i>	<i>size(byte)</i>
1440	231
1441	242	0.11	8412	1	add	132
...	...	0.15	3231	2	delete	1523
		0.21	63263	3	delete	17123
		0.3	15234	1	add	104
		0.31	44141	3	update	23412
	

[그림 11] 정렬된 오프셋 목록(SOL)의 구조

전송할 수 있도록 각 갱신마다 갱신의 크기를 저장한다.

5. 실험 평가

이번 장에서는 도로네트워크 기반의 연속원근질의 처리하는 시스템을 구현하고 이를 실험을 통해 평가한다. 본 연구의 내용을 실제 응용 분야에 적용할 수 있게 하기 위해서 시스템의 환경을 설정하였다. 대부분의 네비게이션 시스템에서 데이터로 건물과 도로를 사용하는 것을 감안하여 본 실험에서는 3차원 건물과 도로네트워크로 데이터를 구성하였다. <표 2>는 질감정보를 포함하는 대전시 3차원 건물 데이터를 분석하여 표로 나타낸 것이다. 건물 생성기는 실제 건물의 분포와 유사하게 생성하기 위해서 가로와 세로의 비율은 1:3을 넘지 않는 범위에서 각 타입의 비율에 맞게 생성한다. 세밀도에 따른 건물 데이터의 크기는 세밀도3, 세밀도2, 세밀도1에서 평균 100kb, 1kb, 200bytes로 구성되었다. 세밀도3은 건물의 질감정보를 사용하였고, 세밀도2, 세밀도1은 질감정보를 사용하지 않았다. 세밀도1은 복잡한 건물을 단순한 블록형태로 일반화시켜 데이터 크기를 줄였다. <표 3>은 실험

에서 사용하는 데이터 집합의 특징을 나타낸 것이다.

실험 변수로는 세밀도의 영역이 성능에 미치는 영향을 알아보기 위해 세밀도를 사용한다. 그리고 질의처리의 단위는 클라이언트의 메모리 여유 공간에 따라 변화하기 때문에 메모리 공간 크기를 실험 변수로 적용한다. 메모리 공간 크기는 10MB, 20MB, 30MB으로 세 가지를 사용한다. 세밀도의 단계는 모두 세 단계로 하고, 다섯 가지 그룹을 사용한다. 200mm × 100mm 인 모바일 기기에서 차량에서 120도 시야각으로 디스플레이 되는 것을 가정하고, 세밀도1 거리는 화면에 가로 5mm를 실제 가로 10m로 맵핑시킨 거리를 계산하여 117m로 계산하였다. 나머지 세밀도2, 세밀도3 거리는 10mm, 50mm를 각각 10m로 맵핑시켰을 때, 58m, 11m로

<표 3> 데이터셋의 특징

특징	값
도로선분의 총 개수	5247
도로선분의 평균길이	80m
도로선분의 총 길이 합	457km
도로선분의 평균 교차점 개수	1.74
건물의 평균 영역	260m ²
건물의 밀도	72.8%
3차원 데이터의 크기	3GB

<표 2> 건물 데이터

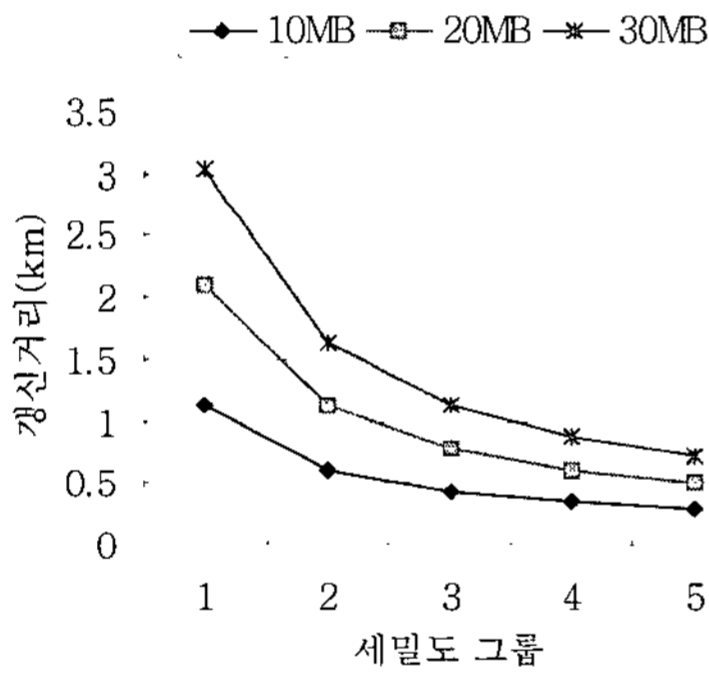
	평균 영역(m ²)	평균 높이(m)	개수	비율(%)
A 유형	217.7	3.3	433	6.2
B 유형	2279.8	32.5	105	1.5
C 유형	612	11.8	245	3.5
D 유형	193.3	3.9	6157	88.3
E 유형	4998.3	25.6	28	0.4

계산된다. 세밀도 그룹-1(117m, 58m, 11m)를 하나의 세밀도 집합으로 하고, 각 거리에 2배, 3배, 4배, 5배하여 세밀도 그룹-2, 세밀도 그룹-3, 세밀도 그룹-4, 세밀도 그룹-5 다섯 가지 세밀도 집합을 사용한다.

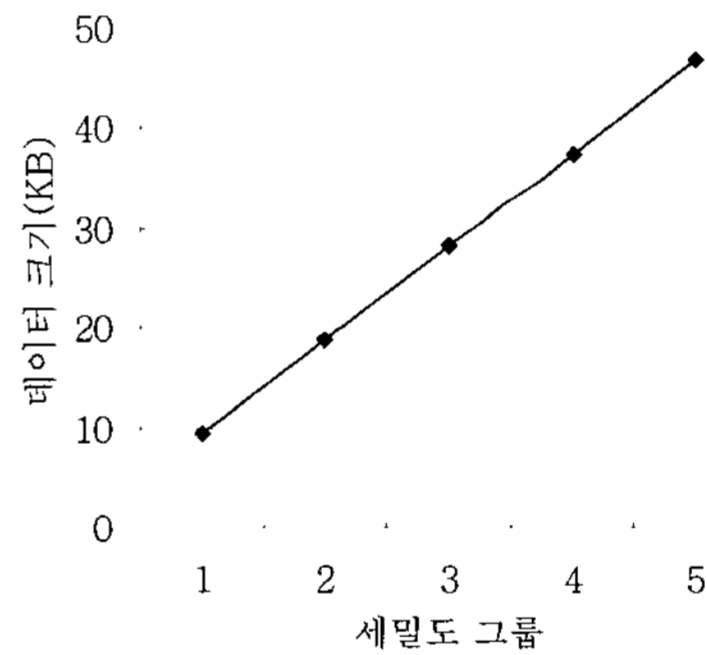
[그림 12]은 세밀도 그룹에 따른 갱신 거리와 갱신되는 데이터의 크기를 나타낸 것이다. 갱신 거리는 클라이언트의 메모리 크기만큼 갱신되는 데이터의 주기를 거리로 계산한 것으로, [그림 12.a]의 결과로 나타났다. 10MB 메모리크기를 가질 때 세밀도 그룹-1의 갱신 거리는 1km 정

도로 나타났다. 갱신 거리는 메모리 크기에는 비례하고, 세밀도의 거리에는 반비례한다. [그림 12.b]는 1m 움직일 때의 평균 갱신되는 데이터 크기를 나타낸 것이다. 즉, 세밀도 그룹-1에서는 1m 움직일 때 마다 11KB의 데이터를 갱신하게 된다. 시간으로 바꾸어 계산하면 60km로 주행할 때 1초에 178KB 크기 정도의 데이터 갱신이 일어난다.

[그림 13]은 동적인 질의처리의 성능을 보여준다. [그림 13.a]는 10MB 데이터 질의처리 시에 500번 정도의 노드 접근을

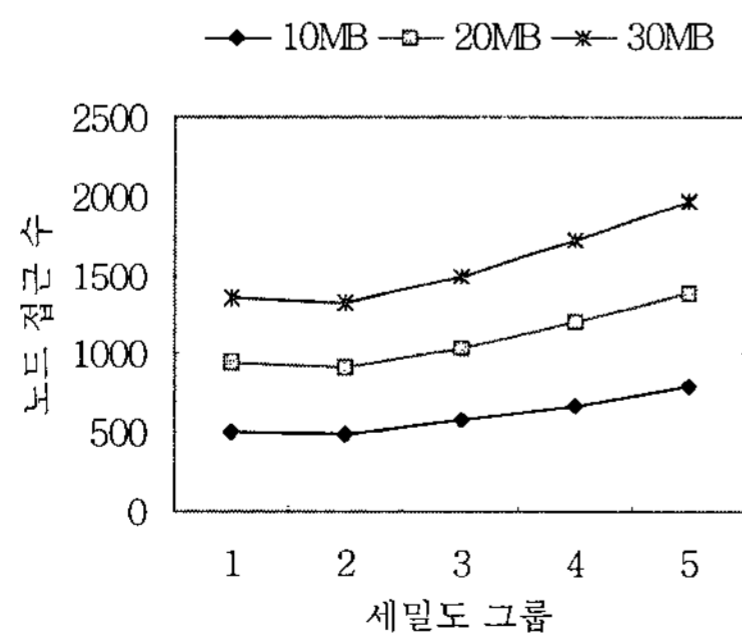


(a) 갱신 거리

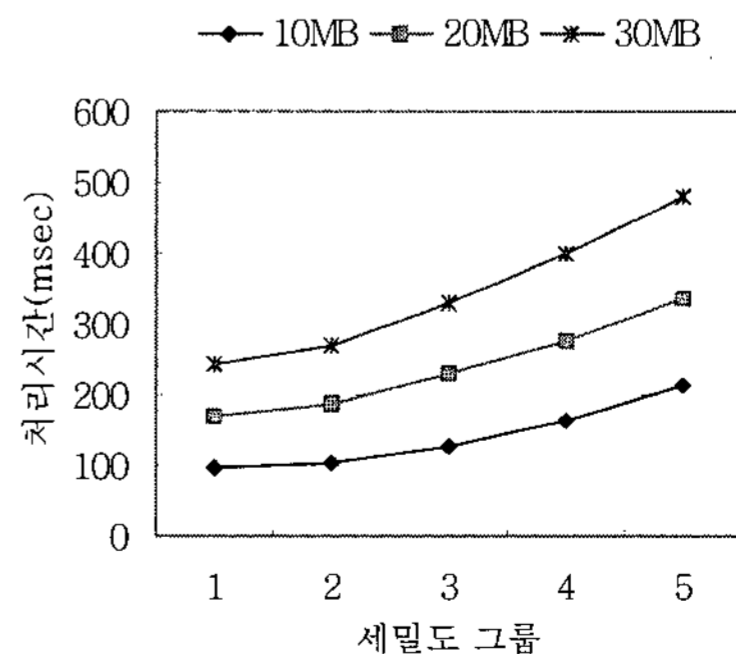


(b) 1m당 갱신되는 데이터 크기

[그림 12] 갱신 거리와 갱신 데이터 크기



(a) 질의단위 노드 접근수



(b) 질의단위 처리시간

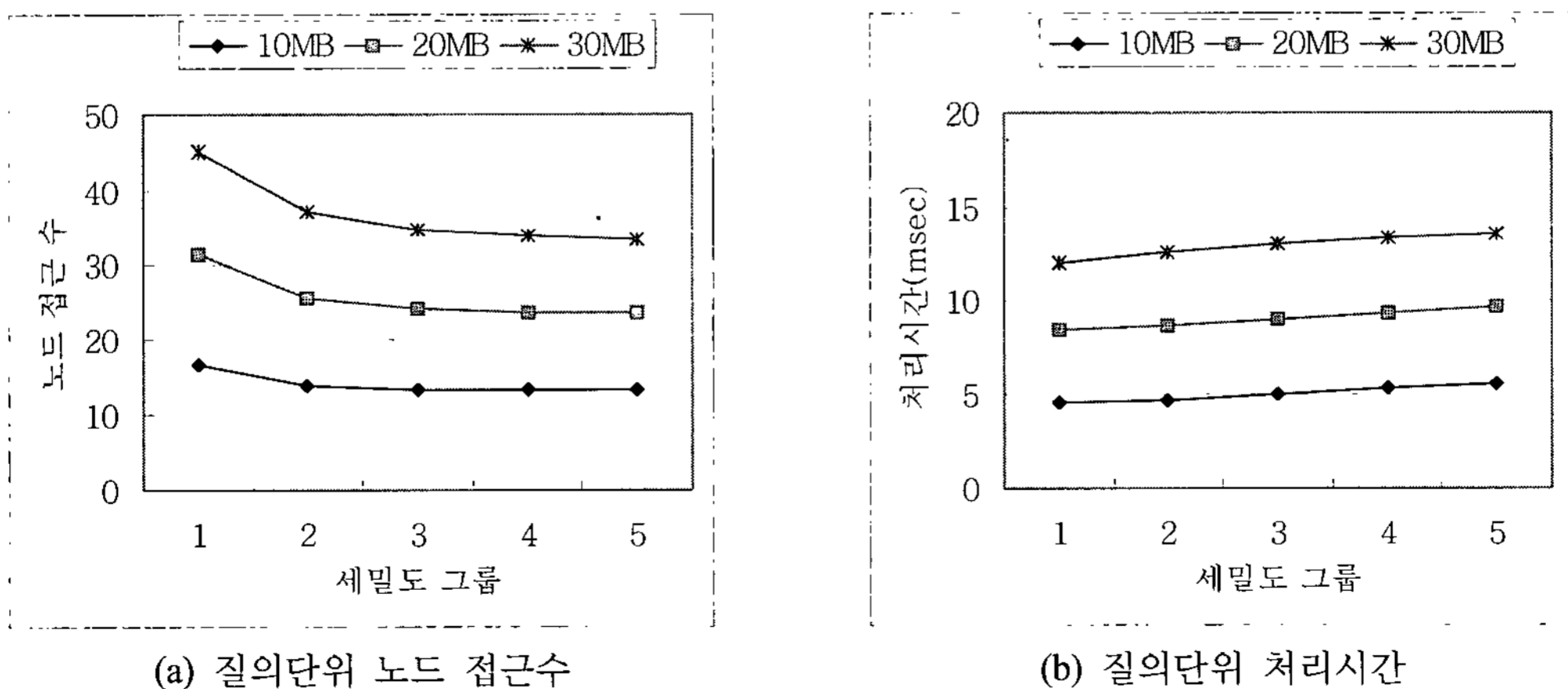
[그림 13] 동적 질의처리 성능 분석

수행하고, 세밀도의 거리가 증가함에 따라 노드 접근 수가 증가한다. 분석결과 도로 선분의 크기 단위로 질의를 처리하기 때문에 버퍼 크기를 넘어선 구간에 대해서도 질의를 수행하고 버퍼 크기만큼 클라이언트에게 전달한다. 따라서 동적 질의수행 시에 불필요한 부분에 대한 검색이 이루어진다. [그림 13.b]는 질의처리 시간을 보여준다. 앞서 살펴본 결과에 영향을 받아서 세밀도 범위가 커짐에 따라 질의처리 시간도 비례하여 증가한다.

[그림 14]는 정렬된 오프셋 목록(이하 SOL)의 질의처리 성능을 나타낸 것이다. 10MB 데이터의 질의를 처리하는데 17번 이하의 노드접근을 수행한다. 세밀도 범위가 증가해도 노드 접근수가 증가하지 않는데, 그 이유는 질의를 만족하는 데이터의 크기를 누적하면서 원하는 메모리 크기에서 질의를 중단하기 때문에 추가적인 노드 접근이 필요 없다. 세밀도 범위가 증가해도 질의처리 시간 또한 거의 증가하지 않은 것으로 나타났다.

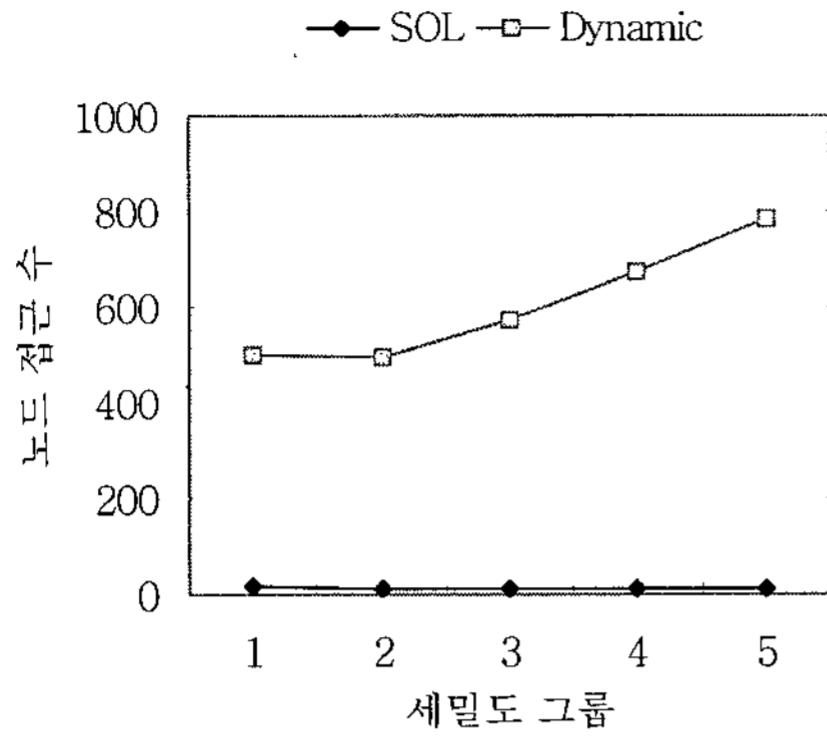
[그림 15]는 동적인 질의처리와 SOL을 이용한 질의처리 성능을 비교한 것이다. 네 실험 모두 클라이언트의 메모리 크기는 10MB이다. [그림 15.a]는 10MB 크기만큼의 질의를 처리할 때 디스크 노드 접근 수를 나타낸 것이다. 동적인 질의처리가 SOL 방법에 비해 30배 이상의 노드 접근 수가 많은 것으로 나타났다. [그림 15.b]는 100m를 주행할 때 질의처리에 수행하는 노드 수를 나타낸 것이다. SOL 방법이 앞서 말한 이유대로 세밀도 범위에 영향을 거의 받지 않는다. [그림 15.c]와 [그림 15.d]는 질의처리 시간을 비교한 것으로 20배 정도의 성능 차이를 나타낸다.

SOL 질의처리 방법이 동적인 방법에 비해 뛰어난 것으로 나타났지만 이는 전처리과정이 있기 때문에 이를 고려해야한다. [그림 16]은 SOL의 전처리 비용을 분석한 결과이다. [그림 16.a]는 SOL의 생성 크기를 세밀도 그룹 별로 나타낸 것이다. 전체 3차원 데이터 3GB 중의 0.01% 정도의 데이터를 필요로 하며 이는 건물의 밀

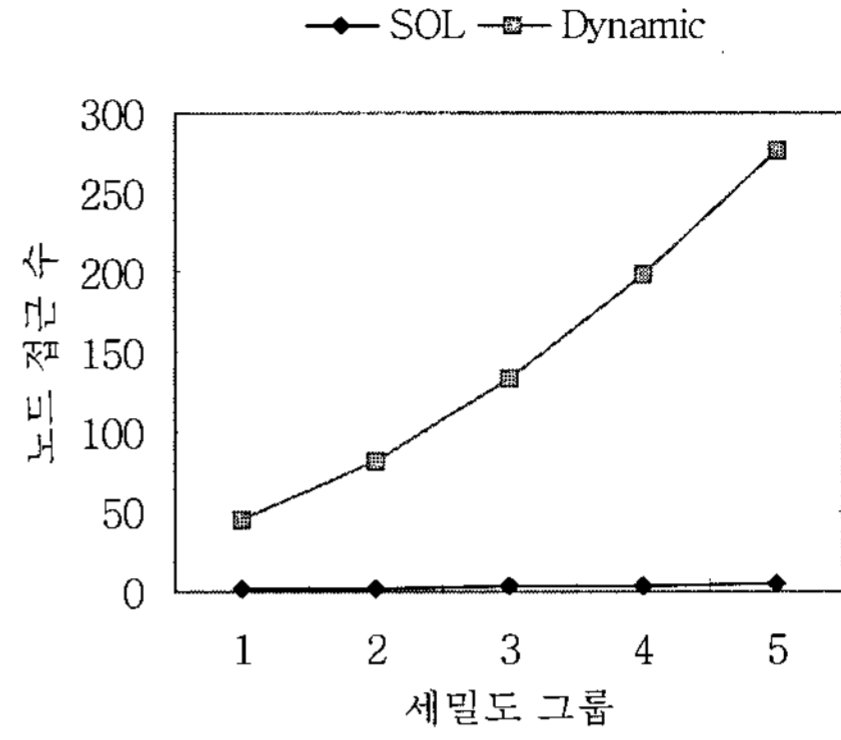


[그림 14] SOL 질의처리 성능 분석

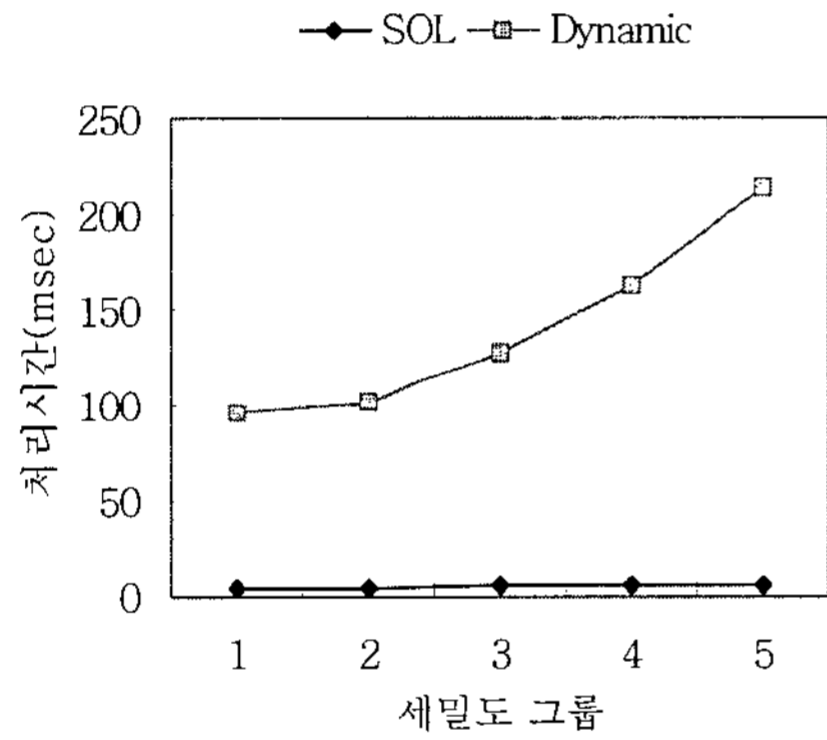
도로네트워크 기반의 3차원 객체를 위한 연속원근질의처리



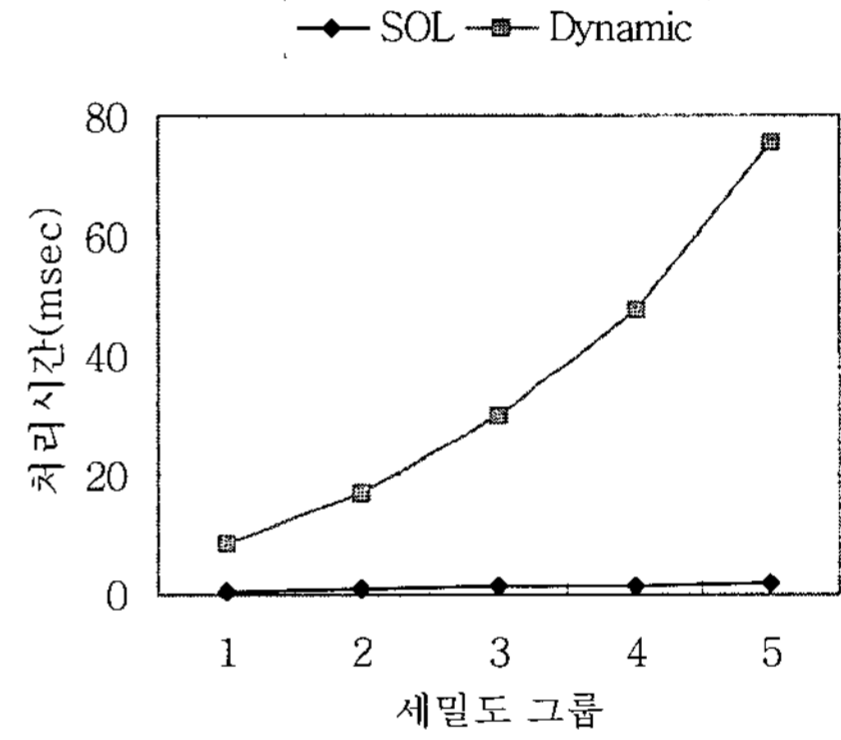
(a) 질의단위 노드 접근수



(b) 100m단위 노드 접근수

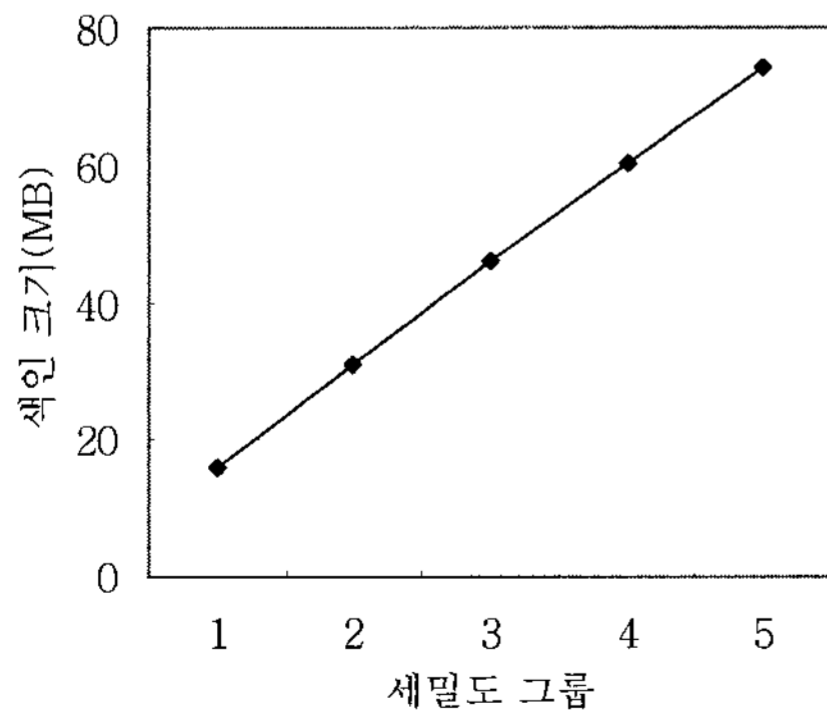


(c) 질의단위 처리시간

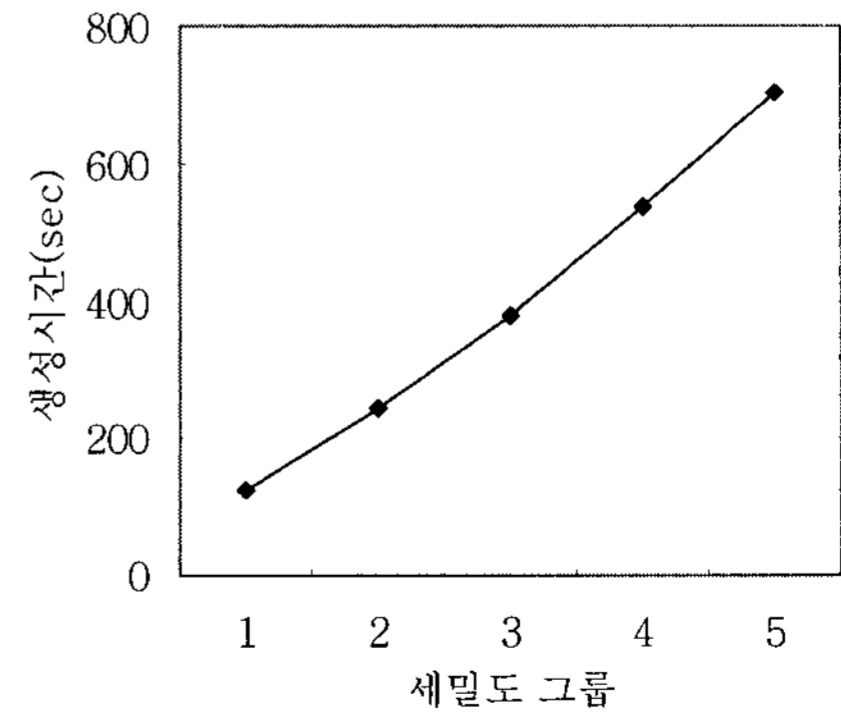


(d) 100m단위 처리시간

[그림 15] 질의처리 성능 비교



(a) 생성된 데이터 크기



(b) 생성 시간

[그림 16] 전처리 비용 분석

도와 도로네트워크의 밀도에 영향을 받는다. [그림 16.b]는 SOL의 생성 시간을 나타낸 것이다. 이는 도로 네트워크의 전체 길이에 비례하는데 모든 도로선분의 길이 합인 457km만큼을 지나는 이동객체의 질의처리 시간과 동일하게 된다.

전처리 수행시간은 도로선분 전체에 대해서 연속원근질의를 수행하는 것과 같은 것이므로 전체 도로선분의 총합에 비례하게 된다. 그리고 전처리 수행시간은 도로선분하나에서 질의를 처리하는 것이 비해 오래 걸리게 된다. 다수의 이동 객체에게 3차원 데이터를 제공하는 것은 SOL을 이용한 질의처리가 유리하다. 이는 각 질의처리의 선택 기준이 될 수 있다.

6. 결 론

본 논문에서는 이동객체의 위치를 중심으로 3차원 데이터를 효율적으로 제공하기 위한 방법으로, 데이터의 중요도에 따라 필요로 하는 데이터를 구분하여 제공할 수 있는 원근질의를 통해 접근하였다. 기존의 원근질의는 이동객체와 같이 질의점이 시간에 따라 변하는 것을 고려하지는 못했다. 따라서 본 논문에서 시간의 변화에 따라 연속적으로 원근질의를 갱신하는 연속원근질의를 정의하였다.

이를 바탕으로 실제로 응용할 수 있는 도로네트워크 공간에서 연속원근질의를 처리하는 방법론을 제시하였다. 질의처리는 도로선분구간을 단위로 처리하는데 동적인 질의처리방법과 전처리를 거친 질의처리방법을 제안하였다. 동적인 질의 처

리는 전처리 과정에 따른 시간은 없지만 하나의 도로에서 움직이는 이동객체들이 같은 질의를 요청하기 때문에 중복된 질의처리 비용이 드는 단점을 가진다. 반면 정렬된 오프셋 목록을 이용한 질의처리는 전처리 과정의 비용이 들지만 질의처리의 중복을 줄이는 효율적인 방법이다. 실험 결과 두 질의처리방법 중에 전처리를 만들어진 정렬된 오프셋 목록을 이용하는 것의 성능이 20-30배 이상 좋은 것으로 나타났다.

향후에는 동적질의 처리에서 남은 메모리크기를 감안하여 질의처리를 하도록 개선해야한다. 또한 정렬된 오프셋 목록이 여러 도로에 3차원 데이터가 중복되는 점을 감안하여 클러스터링을 하는 연구가 필요하다. 그리고 본 논문에서는 사용하는 거리 기반의 세밀도 방법 이외에도 각 공간객체 마다 다른 세밀도를 적용하는 것이 향후 연구 과제이다.

참고문헌

- Douglas B. Terry, David Goldberg, David Nichols, and Brian M. Oki, 1992, Continuous Queries over Append-Only Databases, In Proceedings of ACM SIGMOD, p.321-330.
- Dragan Stojanovic, Slobodanka Djordjevic-Kajan, Apostolos N. Papadopoulos, and Alexandros Nanopoulos, 2006, Continuous Range Query Processing for Network Constrained Mobile Objects, In Proceedings of the 8th ICEIS, pp. 63-70.
- Michael Kofler, Michael Gervautz, and Michael Gruber, 2000, R-trees for Organizing and

- Visualizing 3D GIS Databases, *Journal of Visualization and Computer Animation*, vol. 11, No. 3, pp. 129-143.
- Hyung-Ju Cho, and Chin-Wan Chung, 2005, An Efficient and Scalable Approach to CNN Queries in a Road Network, In *VLDB*, pp. 865-876.
- Iosif Lazaridis, Kriengkrai Porkaew, and Sharad Mehrotra, 2002, Dynamic Queries over Mobile Objects, In *Proceedings of the 8th International Conference on Extending Database Technology*, pp. 269-286.
- Jun Li, Ning Jing, and Maoyin Sun, 2001, Spatial Database Techniques Oriented to Visualization in 3D GIS, In *Proceedings of the 2nd International Symposium on Digital Earth*.
- Kyriakos Mouratidis, Man Lung Yiu, Dimitris Papadias, and Nikos Mamoulis, 2006, Continuous Nearest Neighbor Monitoring in Road Networks, In *VLDB*.
- Mohamed F. Mokbel, Xiaopeng Xiong, and Walid G. Aref, 2004, SINA: Scalable Incremental Processing of Continuous Queries in Spatio-temporal Databases, *SIGMOD Conference*, pp. 623-634.
- Mohammad R. Kolahdouzan, and Cyrus Shahabi, 2004, Continuous K-nearest neighbor queries in spatial network databases, In *STDBM*.
- Sunil Prabhakar, Yuni Xia, Dimitri Kalashnikov, Walid Aref, and Susanne E. Hambrusch, 2002, Query Indexing and Velocity Constrained Indexing: Scalable Techniques for Continuous Queries on Moving Objects, *IEEE Transactions on Computers*, October, Vol. 51, No. 10.
- Tomas Moller, Eric Haines, and Tomas Akenine-Moller, 2002, *Real-Time Rendering (2nd Edition)*, A. K. Peters Ltd, 880 pages.
- Xiaopeng Xiong, Mohamed F. Mokbel, and Walid G. Aref, 2005, SEA-CNN: Scalable Processing of Continuous K-Nearest Neighbor Queries in Spatio-temporal Databases, In *ICDE*, pp. 643-654.
- Ying Cai, Kien A. Hua, and Guohong Cao, 2004, Processing Range-Monitoring Queries on Heterogeneous Mobile Objects, In *Mobile Data Management*, pp. 27-38.
- Yufei Tao, Dimitris Papadias, and Qiongmao Shen, 2002, Continuous Nearest Neighbor Search, In *Proceeding of VLDB*, pp. 287-298.