

맵매칭 기술의 분류 및 맵매칭 알고리즘의 개발
Classification of Map-matching Techniques and A Development

정 연 식 *

윤 항 목 **

최 기 주 ***

Youn-Shik Chung

Yoon, Hang Mook

Keechoo Choi

ABSTRACT

Map matching technique is an essential part of the car navigation and other related positioning fields such as dead reckoning and GPS data logging upon the GIS database. This paper is to break down map matching techniques, to categorize them, and to propose a simple technique for GPS based map matching technique. For categorization of techniques, two approaches have been adopted. One is to only use geometric information, and the other is to use both geometric and topological information. Some pros and cons of each method have been described. In addition, a simple map matching technique, set forth in this paper, has been introduced for properly utilizing the advantage of GPS points after the absence of the chronic problem of selective availability, which had been prevailed recently. Some research opportunities and problems of the technique have also been discussed.

要 旨

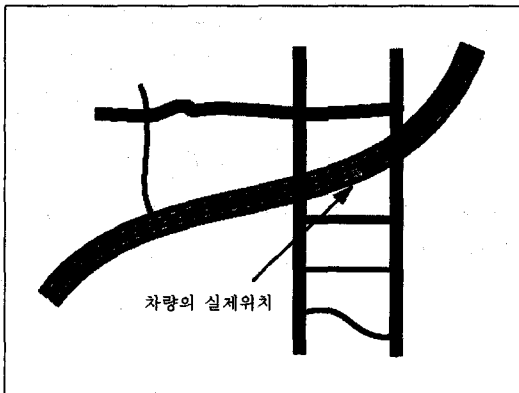
차량용 자동 항법장치가 실생활에 적용됨에 정확한 자동차의 위치를 표현하기 위해서 맵매칭 기술에 대한 중요성은 더욱 부각되고 있다. 맵매칭 기술은 GPS의 오차, 수치지도의 오차 등으로 발생하는 오류를 극복하기 위한 과정으로서 자동 항법시스템은 물론 기타 관련분야인 GPS영역등에도 반드시 필요한 기술중의 하나이다. 그러나, 지금까지 맵매칭에 대한 기술을 정확하게 구분하여 기술하거나 가장 적합한 알고리즘을 제시하지 못하고 있는 바, 본 논문에서는 그 동안 개발된 많은 맵매칭 기술에 대하여 크게 지형정보와 위상구조를 토대로 2가지 유형으로 구분하여 각 기술별 문제점과 개선 방안을 소개하였다. 더 나아가, 최근 미국 정부의 주도로 SA(Selective Availability) 개념이 사라짐에 따라, 쉽게 적용할 수 있는 간단한 형태의 맵매칭 알고리즘을 제시하였고, 향후의 응용방안에 대하여 간단히 언급하였다.

* 아주대학교 교통연구센터 연구원 ** 동의대학교 도시공학과 교수 *** 아주대학교 교통공학전공 교수

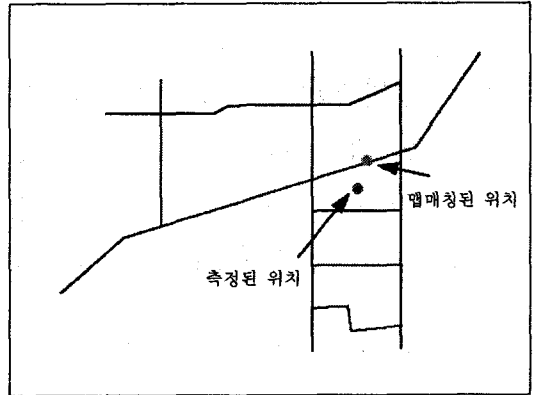
1. 서론

차량 자동항법장치(Car Navigation System)가 실생활에 적용되면서, 이 분야에 대한 연구는 더욱 활발해지고 있다. 차량 항법장치의 핵심은 차량의 현재 위치를 운전자에게 알려주는 것이다. 이를 위해 가장 많이 적용되는 방식은 GPS(Global Positioning System)이다. GPS는 일정한 시간 간격(예를 들면, 1초 간격)으로 현재 위치를 알려주기 때문에 항공과 선박의 항법 장치에 널리 적용되던 방식이다. 그러나, 그 동안 GPS를 일반 이용자가 사용할 경우 SA(Selective Availability)와 기타 GPS의 오차로 인해 차량항법에는 다소 어려움이 있었다. 즉, 차량의 위치가 나타나는 도로 네트워크는 인접 도로간의 거리가 수십미터인 경우가 많기 때문에 최대 100m이상의 오차를 지닌 GPS의 수신 정보로는 현재의 위치를 그대로 수용하기가 어려웠다. 특히, 위치를 표시하게될 수치지도(digital map) 또한 오차가 10m이상 발생하는 지역도 있기 때문에 문제는 더욱 커진다.

이러한 문제를 해결하기 위해 개발된 기술이 맵매칭(map-matching) 기술이다. 특히, 맵매칭 기술은 GPS와 지도의 오차를 극복할 뿐 아니라 고가도로, 터널과 같은 GPS 신호의 수신이 어려운 특수 지형에서도 차량의 위치를 표현하기 위해 개발된 것이다. <그림 1>은 이러한 맵매칭이 필요한 상황의 한 예를 나타낸다.



<그림 1-a> 맵매칭에 대한 문제



<그림 1-b> 맵매칭에 대한 문제

<그림 1-a>는 실제로 차량이 도로를 주행하는 모습을 구현한 것이며, <그림 1-b>는 이 차량으로부터 수집된 GPS자료를 이용하여 수치지도에 표현했을 경우 발생할 수 있는 오류를 나타낸 것이다. 만일 실제로 자료가 오른쪽 그림처럼 측정되었을 경우, 차량이 어느 도로 혹은 어느 링크를 주행중인지 알 수 없다. 그러나, 수집된 위치에 대한 정보와 네트워크의 정보를 최대한 활용한다면, 현재 주행중인 도로에 측정 자료를 이동시켜 표현할 수 있다. 이러한 과정을 맵매칭이라 한다.

차량과 도로간에는 너무나 다양한 상황이 존재하기 때문에 이러한 환경을 모두 고려하여 맵매칭 기술을 개발하는 것은 쉬운 일은 아니다. 때문에 지금까지 다양한 방식으로 맵매칭 기술이 개발되었으나, 가장 우수한 기법으로 여겨지거나 통상 가장 많이 적용되는 방법을 한가지로 정의하는 것은 어려운 문제이다. 또한, 여러 가지 문제점을 극복하기 위해 필터링(filtering)기술([8]), 퍼지(fuzzy)이론([1], [10]), DR(Dead Reckoning)과의 연계([2], [4], [7], [9]) 등 복잡한 이론을 적용하여 맵매칭 기술을 개발하였다.

그러나, 2000년 5월 1일부로 SA가 없어지면서, GPS의 오차는 100m수준에서 30m이하 수준으로 떨어졌기 때문에¹⁾ 과거의 복잡한 기

1) SA가 사라지면서 미국에서 실험한 결과 오차는 수미터 이내인 것으로 나타났으나(... a circular error of only 2.8 meters and a spherical error of 4.6 meters during the first few hours of SA-free operation. The data was

술로 개발된 맵매칭 기술을 적용하지 않고 보다 간단하고 쉬운 기법의 맵매칭 기술을 적용하여 맵매칭이 가능할 것으로 판단된다. 따라서, 본 논문에서는 그 동안에 개발된 맵매칭 기술을 유형별로 분류하며, 간단한 방식을 적용한 맵매칭 알고리즘을 개발하고자 한다.

2. 맵매칭 기법 유형

맵매칭의 유형은 크게 기하학적인 정보만을 이용하는 경우와 이 정보와 위상정보를 동시에 이용하는 경우로 나뉜다. 기하학적인 정보만을 이용하는 방식에는 다시 지점과 지점에 대한 것을 매칭시키는 "point-to-point" 매칭, 지점과 라인을 매칭시키는 "point-to-curve" 매칭, 그리고, 라인과 라인을 매칭시키는 "curve-to-curve" 매칭기법으로 나뉜다([3]). 기하학적인 정보와 위상정보를 동시에 이용하는 방식은 기하학적인 정보만을 이용할 경우 발생하는 문제점을 해결하기 위해 제시된 기법으로 point-to-curve 방식을 개선한 방식과 curve-to-curve 방식을 개선한 방식이 있다. 본 장에서는 이들 각 유형에 대하여 보다 자세히 알아보기로 한다.

2.1 기하학적 정보만을 이용하는 맵매칭 기법

2.1.1 Point-to-Point 매칭

이 방식은 가장 간단한 방식으로 시간 t 에 수집된 지점 자료 P^t 를 가장 근접한 노드에 매칭시키는 방식이다. 수집된 지점 자료와 가장 가까운 노드를 선정하기 위한 가장 선호하는 방법은 두 지점간의 유클리디안 거리(Euclidean distance)를 이용하는 것이다. 일

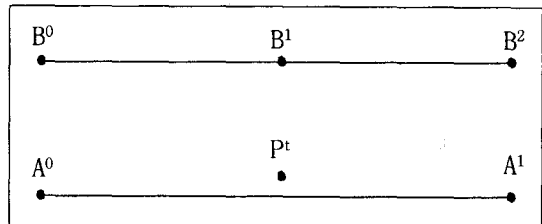
반적으로 2차원 공간(R^2)상에서 두 지점 x 와 y 사이의 유클리디안 거리는 다음과 같이 정의된다([6]).

$$\|x-y\|_2 = \sqrt{(x_1-y_1)^2 + (x_2-y_2)^2} \quad (1)$$

Point-to-Point 매칭 알고리즘에서, 가장 가까운 노드를 결정하기 위해서는 네트워크상의 각 지점과 P^t 간의 거리를 결정하는 것이 필요하다.

물론 실제로 네트워크상에서 P^t 와 모든 노드간의 거리를 구할 필요는 없다. 대신에 P^t 와 적당한 거리 내에 있는 노드를 먼저 확인하고, 그 노드와의 거리만을 계산하면 된다(예를 들면, 사용되는 GPS수신기의 오차범위 내에 포함된 노드). 이러한 인접된 지점을 찾기 위해 개발된 많은 자료구조와 알고리즘이 있다([5]).

그러나, 이 접근 방식이 쉽고 빠르다는 장점이 있지만, 실제로 구현하기에는 많은 문제점을 가지고 있다. 특히, 네트워크 상의 노드들은 모두 경로에 관련되어 있다는 점이 이 알고리즘 수행에 많은 문제점을 발생시킨다. <그림 2>는 이러한 문제점을 잘 설명하고 있다.



<그림 2> Point-to-Point 매칭 기법의 문제점

여기에서, P^t 는 노드 A^0 와 A^1 으로 이루어진 링크를 통행하고 있다는 것을 쉽게 확인 할 수 있다. 그러나, Point-to-Point 기법을 적용할 경우 P^t 는 다른 어느 노드 보다 B^1 에 더 가깝다. 때문에 링크 A^0 는 링크 B 에 매칭된다는 문제점이 발생하게 된다.

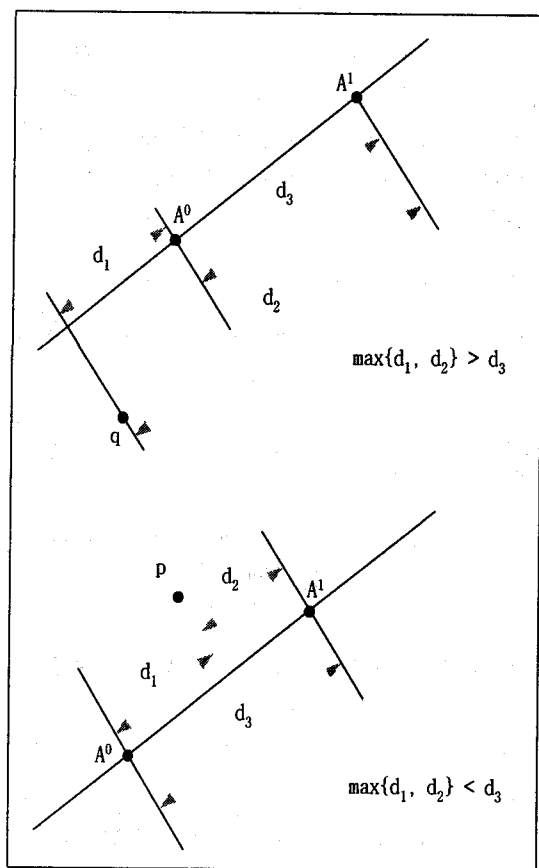
measured using a Trimble SV6 receiver([11]), 본 연구에 서는 Trimble Lassen SK-8 Starter Kit 적용.

2.1.2 Point-to-Curve 매칭

이 방식은 네트워크 상에서 P' 에 가장 가까운 노드를 탐색하는 대신 링크를 찾는 방법이다. 이 방식에서는 우선 가장 가까운 것을 정의해야 한다. 즉, 라인과 지점에 대한 거리의 비교이기 때문에 라인의 어느 지점과의 거리를 측정해야 하는 문제가 된다.

라인과 지점간의 거리를 구하는 가장 간단한 방법을 생각해 보자. a 와 b 지점을 통과하는 직선 A 를 $ax + by + c = 0$ 라 하자. 이때 이 직선 위의 한 점 $Q(x_1, y_1)$ 과 특정 지점 $P_0(x_0, y_0)$ 와의 거리는 정사투영(orthogonal projection)을 이용하여 다음과 같이 구할 수 있다([6]).

$$d(P_0, A) = \frac{|a(x_0 - x_1) + b(y_0 - y_1)|}{\sqrt{a^2 + b^2}} \quad (2)$$

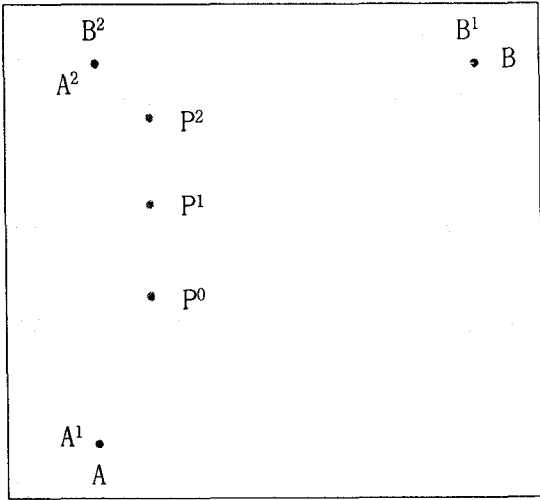


<그림 3> 점과 선분과의 거리

<그림 3>에서의 같이 점과 선분(line segment)사이의 거리를 구하는 것은 점과 직선사이의 거리를 구하는 것보다 복잡하다. 즉, d_1 과 d_2 중 큰 길이의 선분이 d_3 의 길이보다 작을 경우, 점 p 와 선분 A^0 에서 A^1 사이의 최소거리를 계산하는 것은 p 와 A^0 에서 A^1 를 지나는 직선의 최소거리를 구하는 것과 동일하기 때문에 수월하다. 그러나, d_1 과 d_2 중 큰 길이의 선분이 d_3 의 길이보다 클 경우, 점 q 와 A^0 와 A^1 을 지나는 선분 사이의 거리를 계산할 때, 수직거리를 나타내는 직선은 선분을 벗어난 선과 교차하는 것을 우리는 볼 수 있다. 때문에 우리는 q 와 A^0 , A^1 각각에 대한 거리를 계산해야 한다. 그래서, 점 P' 와 링크 A 사이의 최소거리를 계산하기 위해서는 P' 와 선분 $(\lambda A^0 + (1 - \lambda A^1), \lambda \in [0, 1])$, $(\lambda A^1 + (1 - \lambda A^2), \lambda \in [0, 1])$, ..., $(\lambda A^{n-1} + (1 - \lambda A^n), \lambda \in [0, 1])$ 사이의 거리를 구해야 한다.

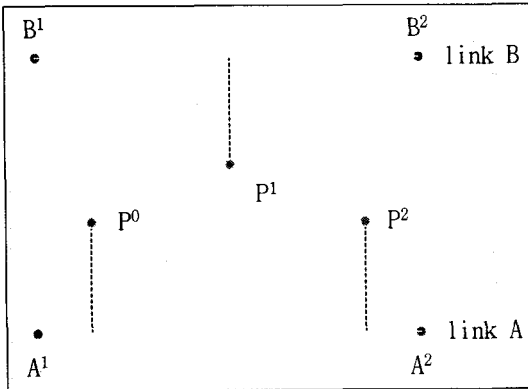
P' 를 링크에 매칭시키기 위해서는, P' 와 네트워크 상의 적당한 링크들간의 거리를 계산해야 하여 가장 가까운 링크 하나를 선정해야 한다. 이러한 접근은 간단해 보이지만 몇 가지 부적절한 항목을 가지고 있다.

첫째, Point-to-Curve 매칭은 이력정보(historical information)를 이용하지 않는다. <그림 4>는 이러한 문제를 나타내는 것이다. 측정된 위치인 P^2 는 링크 A 와 B 에 동일한 위치에 있다. 그러나, P^0 와 P^1 에 대한 정보가 주어진다면, P^2 는 링크 A 에 매칭되어야 한다는 것은 자명한 사실이다.



<그림 4> Point-to-Curve 매칭의 문제점 1

Point-to-Curve 매칭의 또 다른 문제점은 <그림 5>에 나타난 것과 같이 매우 불안정할 수 있다는 것이다. 점 \$P^0, P^1, P^2\$는 링크 \$A\$와 \$B\$로부터 각각 모두 동일한 거리에 위치해 있다. 그러나 \$P^0\$와 \$P^2\$는 \$A\$에 약간 더 가깝고 \$P^1\$은 \$B\$에 약간 더 가까운 것임이 드러난다. 그래서 이 두 개의 링크에 대한 매칭이 불안하게 된다.



<그림 5> Point-to-Curve 매칭의 문제점 2

2.1.3 Geometric Curve-to-Curve Matching

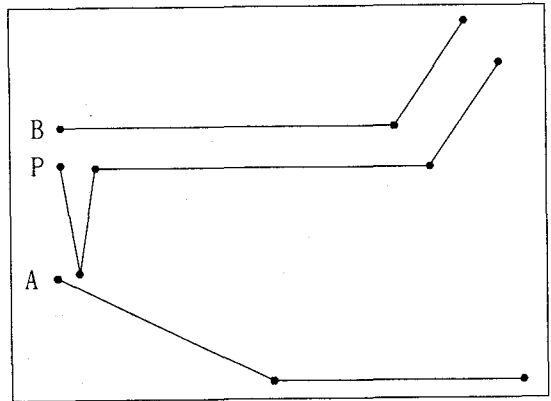
지금까지 알아본 방식보다 더 좋은 방식은 점 \$P^0, P^1, \dots, P^m\$로 정의되는 수집 자료에 대한 궤적 \$P\$

에서 가장 가까운 링크에 매칭하는 즉, 동시에 \$m\$개의 위치를 고려하는 것이다. 물론 이것은 링크들간의 거리를 측정하는 것이 필요하고, 2차원 공간상에서 2개의 링크사이의 거리를 구하는 것이 필요하다.

경로 \$A\$와 \$B\$사이의 거리를 구하기 위해 다음과 같은 식을 적용할 수 있다.

$$\|A - B\|_{\min} = \min_{a \in A, b \in B} \|a - b\| \quad (3)$$

이 정의는 몇 가지 상황에서는 적절하게 적용되지만, 과대오차(outlier)가 발생한 자료에는 매우 민감하기 때문에 문제점이 발생한다. <그림 6>은 이러한 문제점을 나타내는 것으로, 궤적 \$P\$가 경로 \$B\$에 매칭되어야 함에 불구하고 \$A\$에 매칭되는 결과를 초래하게 된다. 마찬가지로, \$\|A - B\|_{\max}\$도 동일한 유형의 문제점이 발생하게 된다.



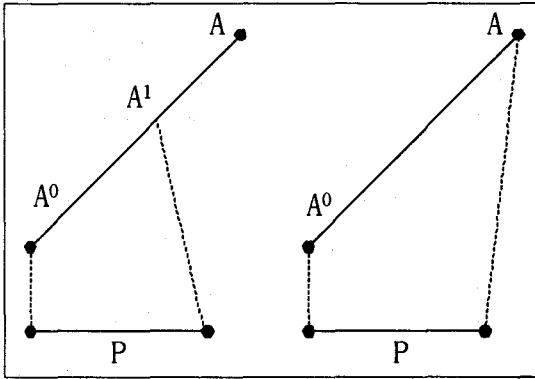
<그림 6> Curve-to-Curve 매칭의 문제점

위 방법 대신에, 링크들 사이의 평균거리를 측정하여 이용하는 것이 보다 적절하다. 이와 같은 측정은 만일 비교가 필요한 링크들을 함수화 한다면 보다 쉽게 구할 수 있다. 예를 들어, 경로 \$A\$가 함수 \$a: [0, 1] \to A\$로 가정하면, 이때 두 개의 경로 \$A\$와 \$B\$사이의 거리는 다음과 같은 식으로 구할 수 있다. 그러나, 이 방식도 각기 다른 경로의 비교에는 좋은 방법은 아니다.

$$\|A - B\|_2 = \int_0^1 \|a(t) - b(t)\| dt \quad (4)$$

만일 \$P\$의 길이를 알고 있다면, \$P\$와 현재 고려중

인 링크의 동일한 길이를 가진 부분과의 거리를 측정하는 것은 보다 쉽다. <그림 7>은 이에 대한 내용을 잘 나타내어 준다. 이 그림에서 왼쪽 그림은 P와 A 사이의 거리를 구하기 위해 A에서 P와 동일한 길이를 가지는 A⁰와 A¹만을 고려하게 된다. 반면, 오른쪽 그림은 P의 전 구간과 A의 전구간에 대한 거리를 고려하는 것이다.



<그림 7> 두 경로간의 거리 측정 방식

이와 같은 측정 방법은 두 개의 경로에 대한 끝 지점의 순서에 의존하기 때문에 링크들의 각 쌍에 대하여 모두 구하여 이중 작은 값을 선정해야 한다.

그러나, 이러한 결과 값이 제대로 나왔을 지라도, 이 같은 결과는 예기치 못한 결과를 초래할 수 있다. <그림 8>은 이러한 상황을 잘 나타낸다. 이 그림에서, P¹, ..., P⁷이 기록되어 졌고, 매칭될 2개의 링크 A, B가 후보로 등록되었다. 이 경우 3개의 경로가 다음과 같이 함수로 표시될 수 있다.

$$a(a) = \begin{pmatrix} 6t \\ 6 \end{pmatrix}, a \in [0, 1] \quad (5)$$

$$b(a) = \begin{pmatrix} 3 \\ 3t \end{pmatrix}, a \in [0, 1] \quad (6)$$

$$p(a) = \begin{pmatrix} 6t \\ 3 \end{pmatrix}, a \in [0, 1] \quad (7)$$

여기에서 P와 A사이의 거리는 다음과 같이 구해진다.

$$\int_0^1 \|p(a) - a(a)\| da \quad (8)$$

$$= \int_0^1 \sqrt{(p_1(a) - a_1(a))^2 + (p_2(a) - a_2(a))^2} da$$

$$= \int_0^1 \sqrt{(6a - 6a)^2 + (3 - 6)^2} da \quad (9)$$

$$= \int_0^1 3 da = 3 \quad (10)$$

그리고, P와 B사이의 거리는 다음과 같이 구해진다.

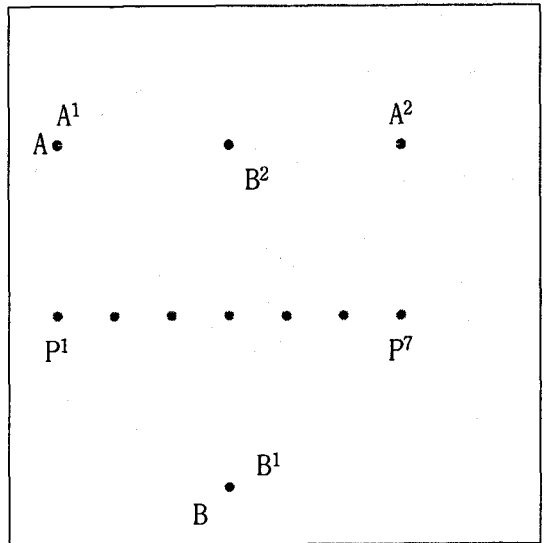
$$\int_0^1 \|P(a) - B(a)\| da \quad (11)$$

$$= \int_0^1 \sqrt{(p_1(a) - b_1(a))^2 + (p_2(a) - b_2(a))^2} da$$

$$= \int_0^1 \sqrt{(6a - 3)^2 + (3 - 6a)^2} da \quad (12)$$

$$= \int_0^1 \sqrt{2(36a^2 - 36a + 9)} da = \sqrt{4.5} \quad (13)$$

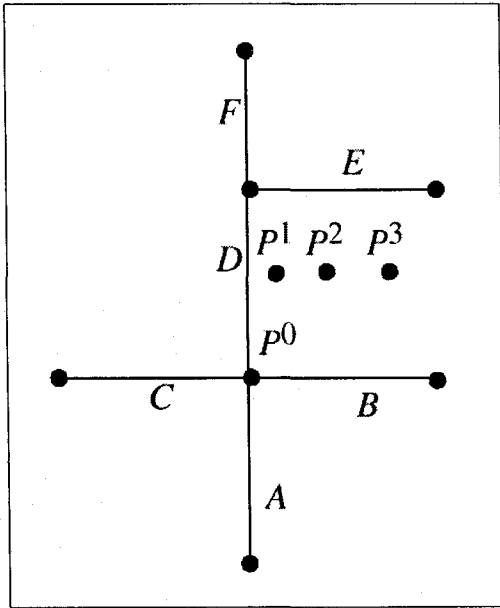
결론적으로 P는 A에 매칭되어야 함에도 불구하고 B에 매칭되는 결과를 초래하였다.



<그림 8> 결과

2.2 기하학적 정보와 위상정보를 이용한 맵매칭 기법

2.2.1 위상정보를 이용한 Point-to-Curve 매칭 개선
 위상정보의 적용방법을 설명하기 위해 전술한 Point-to-Curve 매칭에 대한 예를 다시 살펴보자. <그림 9>에서 우리는 차량이 처음에 P^0 에 있었다는 정보를 알고 있으며, D 에서 E 방향으로 진입을 할 수 없다고 가정해 보자. P^1 은 A, B, C , 혹은 D 에만 매칭될 수 있다는 것을 알 수 있다. 그러나, 단순히 기하학적인 정보만 이용한 Point-to-Curve 기법을 이용하면 P^2, P^3 는 E 에 매칭될 수 있다. 만일 우리가 네트워크에 대한 정보를 알고 있다면, P^2, P^3 가 E 에 매칭되는 것을 막을 수 있을 것이다.

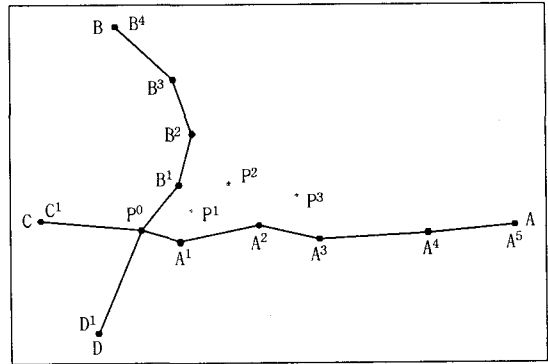


<그림 9> 위상정보의 이용

이밖에도 위상정보의 이용은 고가도로, 터널 주변에서의 맵매칭을 효과적으로 수행할 수 있도록 도와준다.

2.2.1 위상정보를 이용한 Curve-to-Curve 매칭의 개선

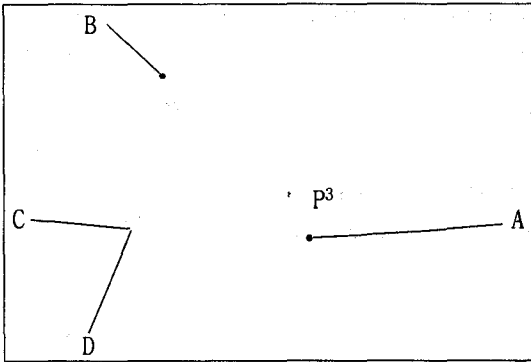
위상정보를 이용한 Curve-to-Curve 매칭기법은 놀라울 정도로 좋은 결과를 가져온다. 앞에서 살펴본 Curve-to-Curve 매칭은 궤적 P 와 P^0 로부터 매칭이 기대되는 동일한 거리를 가지는 모든 링크와의 거리를 계산했다. 그 다음에 P 에 가까운 링크에 매칭시켰다. 비록 식 (4)가 이러한 궤적을 정확하게 계산할 수 있을지라도, 여전히 문제점이 존재한다.



<그림 10> 위상정보를 이용한 Curve-to-Curve 매칭 예제

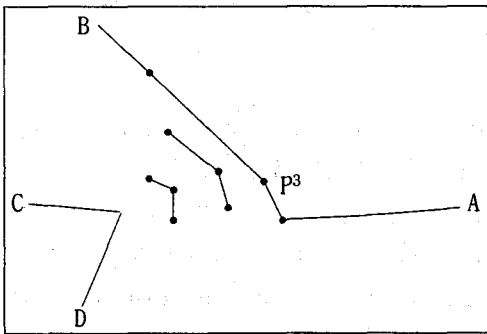
<그림 10>은 이러한 문제점을 개선하기 위해 위상정보의 활용을 설명하기 위한 예제로, 차량은 링크 A, B, C, D 가 교차($A^0 = B^0 = C^0 = D^0$)하는 P^0 에서 출발하여 P^1, P^2, P^3 가 각각 순차적으로 기록되었다고 가정하자. 이 경우 $P^0 \dots P^3$ 에 의해 형성된 궤적과 A, B, C, D 사이의 거리를 구해야 한다.

<그림 11>은 이 문제에 대한 1단계 과정을 나타낸다. 우선, P^0, P^1, P^2, P^3 에 의해 생성된 궤적을 구축한다. 그리고, 교차로에 연결된 모든 경로에 대하여, 각 지점에 의해 생성된 직선과 동일한 길이를 가지는 경로를 구축한다(여기에서는 경로 A 와 B 에 대해서만 고려해보자).



<그림 11> 알고리즘의 1단계

<그림 12>는 그 다음 단계를 나타낸다. 서로 다른 3가지의 경로는 동일한 길이를 가지는 s 개의 선분으로 나뉘어 진다(이 예제에서는 $s=3$). 마지막으로 거리를 계산하며, 이 예제에서는 경로 A가 가장 가깝기 때문에 P^3 는 A에 매칭된다.



<그림 12> 맵매칭 결과

3. 새로운 맵매칭 기법의 개발

지금까지 개발된 대부분의 맵매칭 기법들은 GPS의 심각한 오차와 지도 제작상의 오차를 극복하기 위해 매우 복잡하고 어려운 방식을 적용해 왔다. 그러나, 5월 1일 부로 SA개념이 사라졌기 때문에 100m이상으로 발생하던 GPS의 오차는 30m이하로 줄어들었다²⁾.

2) Trimble Lassen-8 Start kit을 이용하여 고정된 한 지점에

때문에 지금까지 개발된 기법들은 다소 개선되어야 할 것으로 판단되며, 이러한 과정이 필요 없을 지라도, 보다 간단한 기법을 이용한 맵매칭 기술이 다양한 분야에서 필요할 것으로 판단된다.

3.1 맵매칭을 위한 전제 및 알고리즘

맵매칭을 위해서는 벡터 형식의 수치지도가 필요하다. 또한, 벡터 형식의 수치지도는 네트워크에 대한 기하학적인 정보뿐 아니라 위상정보를 포함하기 때문에 고가도로와 같이 주변도로에 대하여 연결이 되지 않은 도로를 탐색하는데 중요한 정보로 활용된다. 그리고, 우리나라에서 사용되는 수치지도와 GPS로 수집되는 자료의 좌표체계는 다르기 때문에 동일한 좌표체계를 유지해야 한다.

한편, 맵매칭 알고리즘은 다음과 같이 이루어지며, 이 과정에서 첫 번째와 세 번째 과정이 가장 핵심이며, 이 부분에서 다양한 기술이 적용된다.

- ① 관련 링크의 선정을 위한 임계값 설정
- ② 현재 차량 위치에 인접한 모든 링크를 선정하여 맵매칭을 위한 후보 링크로 등록
- ③ 선정된 후보 링크는 차량의 궤적과 오차, 도로의 기하 및 위상정보를 이용하여 비교
- ④ 최종 매칭 링크의 선정

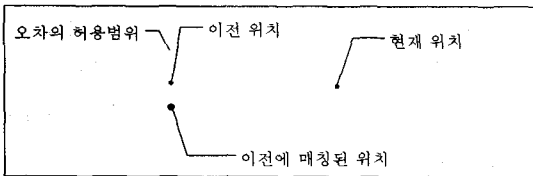
본 연구에서 사용할 맵매칭 기법의 기본 개념은 GPS point에 대한 오차와 수치지도의 오차를 감안하여 수집되는 GPS point에 대하여 버퍼를 형성하는 것이다. 즉, 버퍼에 의해 우선 후보 링크를 선정하게 되며, 선정된 링크가 1개일 경우에는 그 도로에 매칭시키는 것이다. 이러한 개념은 간단하지만 GPS의 오차가 100m이상 발생하던 시기에는 적용하기 어려운 방식이다.

3.2 버퍼를 이용한 매칭 링크 선정

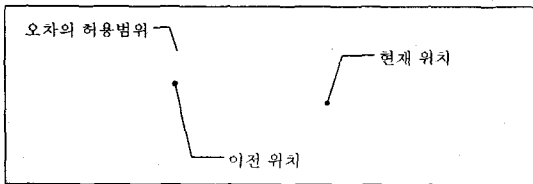
서 1시간 여 동안 수집한 결과 최대 편차는 28m를 나타내었다.

버퍼를 이용한 맵매칭에서는 크게 2가지로 구분된다. 버퍼에 의해 하나의 도로 혹은 링크 ID를 탐색한 경우와 그렇지 못한 경우이다. 그렇지 못한 경우는 다시 링크를 하나도 탐색하지 못한 경우와 2개 이상의 링크가 탐색된 경우로 나누게 된다.

링크의 ID를 탐색하는 방법은 두 가지가 있다. 하나는 이전의 위치에서 도로 ID를 탐색한 경우(<그림 13>)에 탐색 방법과 이전의 위치에서 도로 ID를 탐색하지 못했거나(<그림 14>), 처음 시작하는 경우에 탐색 방법이다.



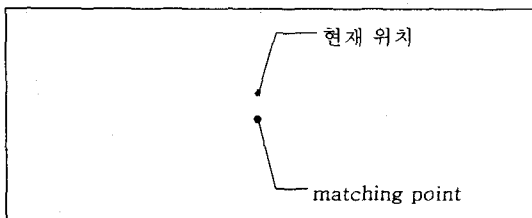
<그림 13> 이전에 링크의 ID를 탐색한 경우



<그림 14> 이전에 링크 ID를 탐색하지 못한 경우

3.2.1 이전의 위치에서 도로 ID를 찾았을 경우

<그림 13>에서처럼 이전의 위치에서 링크 ID를 찾았을 경우에는 해당되는 링크만을 검색하여 매칭시킨다.



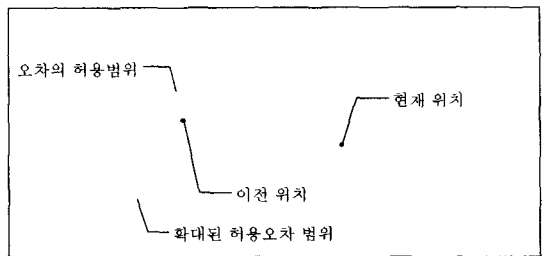
<그림 15> 한 링크 ID만 탐색하는 경우

<그림 15>는 이전의 위치에서 링크 ID를 탐색한 경우 현재의 위치에서 탐색된 링크에 대하여 가장 가까운 지점에 매칭시키는 것을 나타낸 것이다. 만일 이전의 위치에서 링크의 ID를 찾지 못했을 경우에는 다음의 방법을 적용한다.

3.2.2 이전의 위치에서 도로 ID를 찾지 못한 경우

<그림 14>의 경우처럼 오차의 허용범위를 설정하여 버퍼를 생성 시켰음에도 불구하고 과대오차(outlier)가 발생하여 이전의 위치에서 링크 ID를 찾지 못했을 경우에는 인접한 모든 링크를 비교하여 탐색해야 한다. 그러나, 인접한 링크를 어떻게 탐색하는가에 대한 문제점이 생긴다. 또한, GPS의 오차가 적어진 시점에서 인접한 모든 링크를 모두 탐색하는 것이 과연 최적인가를 고려해보아야 한다.

본 논문에서는 이전에 링크 ID를 탐색하지 못한 경우 버퍼의 사이즈를 3배³⁾로 하여 선정된 링크를 비교하는 것으로 하였다. 이 경우에 탐색된 링크는 최소 1개 이상이 선정된다. 만일 <그림 16>과 같이 선정된 링크가 1개이며, 현재 탐색된 링크가 1개라면 <그림 15>와 같이 GPS point에서 선택된 링크 혹은 도로와 가장 가까운 지점에 매칭시키게 된다.

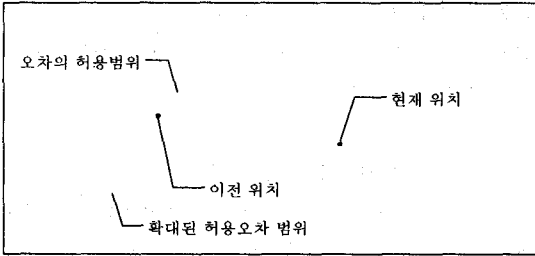


<그림 16> 허용오차의 확대로 탐색된 1개의 링크

그러나, <그림 17>과 같이 선정된 링크가 2개 이상일 경우, 그리고, 현재 탐색된 링크가 2개 이상일

3) 실험에서 이러한 결과는 나타나지 않았으나, 이러한 상황이 나타날 수도 있기 때문에 설정하였음. 3배로 설정한 근거는 없으며, 필요하다면 이 값은 언제나 조정 가능한 값임.

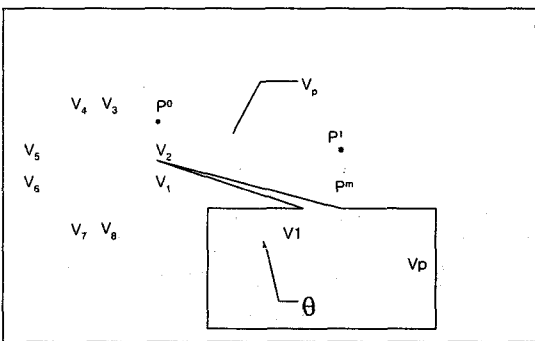
경우는 다소 복잡해진다. 이 경우는 교차로(노드)의 탐색과 관련이 있기 때문에 다음절에서 보다 자세히 알아보기로 한다.



<그림 17> 2개 이상의 링크가 탐색된 경우

3.2.3 교차로에 위치할 경우의 매칭 방법

<그림 18>의 경우와 같이 선정된 링크가 2개 이상 일 경우, 맵매칭을 위한 링크를 결정하기 위해서는 선정된 모든 링크를 비교해야 한다. 이러한 경우 선정된 링크의 위상정보를 이용하여 각 진행방향별 벡터를 구할 수 있다. 또한, 이전에 측정된 GPS point P_0 와 현재 측정된 P_1 을 연결하는 라인에 대한 벡터를 구할 수 있다. 그래서, 각 방향별 링크의 벡터와 GPS point의 주행 궤적 벡터간의 가장 작은 각도를 형성하는 링크에 P_1 을 매칭시키게 된다.



<그림 18> 벡터를 이용한 링크 탐색

<그림 18>에서는 V_1 벡터와 V_p 벡터간의 각도가 가장 작기 때문에 P_1 은 P_m 에 매칭된다. 여기에서, V_1 벡터와 V_p 벡터간의 각도 θ 는 다음의 식을 이

용하여 구할 수 있다(6).

$$\theta = \cos^{-1} \left(\frac{V_1 \cdot V_p}{\|V_1\| \|V_p\|} \right) \quad (14)$$

만일, 현재 위치가 2개 이상의 링크를 탐색했을 경우에도 동일한 방식으로 매칭지점을 구할 수 있다. 그리고, 위 방식은 이전 위치에서 여러 개의 링크를 탐색했을 때 뿐 아니라 현재 위치에서 여러 개의 링크를 탐색한 경우에도 동일하게 적용된다.

4. 자료의 적용

설정된 알고리즘에 대한 적용 가능성을 평가에 앞서 알고리즘에 적용될 허용오차 범위를 다음과 같은 자료를 토대로 설정하였다. 우선 SA가 사라지면서 발생하는 GPS의 오차를 측정하기 위해 한 지점에서 약 1시간 동안 GPS자료를 수집한 결과 형성된 GPS 자료는 최대 30m(지름)정도로 형성되었다. 또한, 실험에 사용된 지도는 자동차 부품연구원(KATECH)에서 제작한 수치지도를 사용하였으며, 이 지도는 검수시 최대 15m의 오차 내에서 수행된 바 지도의 오차는 15m로 설정하였다. 그리고, 마지막으로 차량 항법장치에 사용되는 지도는 도로를 중심선 하나로 표현하기 때문에 만일 도로가 편도 4차로 일 경우 차로의 폭은 약 15m정도가 된다. 이때 1차로와 4차로를 주행하는 차량간에는 약 10m정도의 오차가 나타날 수 있기 때문에 이러한 상황도 고려해야 한다. 그래서, 본 연구에서의 링크를 탐색하기 위해 GPS point에 설정된 버퍼의 반경은 50m로 설정하였다.

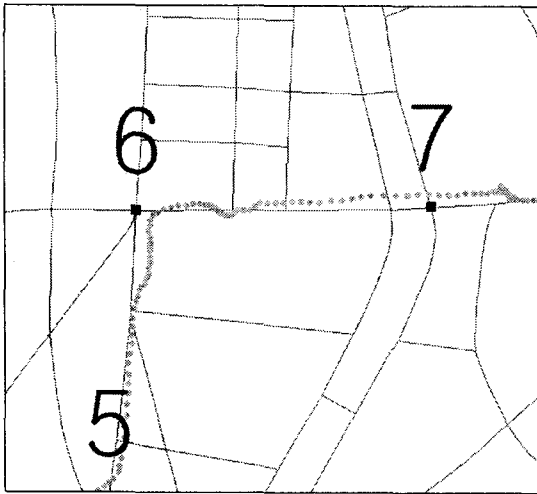
또한, 실험에 사용된 장비와 실험 일시, 실험 구간은 다음과 같다.

- 실험 장비 : Trimble Lassen SK8 Starter Kit(GPS 수신기)
- 실험 일시 : 2000년 5월 8일 17:00 ~ 22:00
- 실험 구간 : 동수원 사거리 ~ 수원역

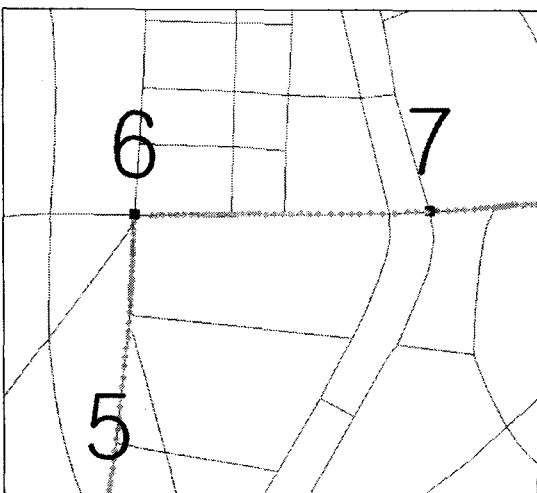
실험 시간은 지체가 심한 오후 피크시간에 맞추어 진행되었으며, 동수원 사거리~수원역 구간은 고층빌딩

이 도로변에 많이 존재하며, 비교적 차로 수가 적기 때문에 빌딩에 의한 반사 오차(outlier)가 심할 가능성이 있는 지역이기 때문에 선정하였다.

<그림 19>는 맵매칭 이전의 그림이며, <그림 20>은 맵매칭 이후의 그림을 나타낸 것이다. 맵매칭 이전에도 오차가 크지 않음을 알 수 있으며, 때문에 본 연구에서 제시한 단순한 알고리즘을 통해서 정확하게 맵매칭 되었다. 또한, 오차가 크게 발생하지 않았기 때문에 각 링크는 GPS point에 부여된 50m버퍼에 탐색되어 버퍼의 크기를 늘리는 과정은 거치지 않았다.



<그림 19> 맵매칭 전의 GPS point



<그림 20> 맵매칭 후의 GPS point

5. 결론

본 연구는 맵매칭에 대한 기법을 분류하고, GPS 오차의 주 요인이었던 SA가 사라짐에 따라 보다 간단하고 용이한 맵매칭 기술을 개발하고자 수행된 것이다. 그 동안 맵매칭 기술의 개발을 위해 필터링 기법, 퍼지 이론, DR과 같은 복잡하고 어려운 이론들을 적용하였다. 그러나, 이러한 이론은 이해하기 복잡할 뿐 아니라, 획기적인 방식으로 제시된 기술도 없었다.

본 논문에서는 우선 이러한 기술을 크게 2가지 유형으로 분류하여 분석해 보았고, 각 유형별로 장단점을 알아보았다. 또한, 본 논문에서 제시한 새로운 맵매칭 기법은 간단한 아이디어로 출발하였다. 제시된 기법의 평가를 위해 비교적 가로망이 복잡하고, 고층빌딩에 의해 발생하는 GPS 반사 오차가 크게 발생할 가능성이 많은 시내부 도로에서 수집한 자료를 적용하였다.

적용 결과 차량의 현재 위치만을 확인하는 단순한 차량 항법용 맵매칭 기법으로는 문제가 없을 것으로 판단되었다. 그러나, 이 기법을 이용하여 통행시간을 구축하거나 기타 지체 정보와 같은 다른 교통정보 수집에 적용가능성은 보다 정밀한 실험을 통해 평가해야 할 것으로 판단된다.

참 고 문 헌

- [1] Byung-chul Lee, "A study on the correction of positioning accuracy of car navigation system and map-matching algorithm", 5th ITS World Congress, 1998.
- [2] Collier, W. C., "In-Vehicle Route Guidance

- Systems Using Map Matched Dead Reckoning”, Proceedings of IEEE Position Location and Navigation Symposium, pp. 359-363, 1990.
- [3] David Bernstein, Alain Kornhauser, "An Introduction to Map Matching for Personal Navigation Assistants", 77th TRB annual meeting, 1998.
- [4] Degawa, H., "A New Navigation System with Multiple Information Sources", Proceedings of the Vehicle Navigation and Information Systems Conference, pp. 143-149, 1992.
- [5] Fuchs, H. Z. M. Kedem and B. F. Naylor. "On Visible Surface Generation by A Priori Tree Structures", Computer Graphics, Vol. 14, pp. 124-133, 1980.
- [6] Howard Anton, Chris RorresAnton, "Elementary linear algebra : applications version", 7th ed., John Wiley, pp. 131-180, 1994.
- [7] Kim, J.-S., "Node Based Map Matching Algorithm for Car Navigation System", Proceedings of the International Symposium on Automotive Technology and Automation pp. 121-126, 1996.
- [8] Krakiwsky, E. J., C. B. Harris, and R. V. C. Wong. A Kalman Filter for Integrating Dead Reckoning, Map Matching and GPS Positioning, Proceedings of IEEE Position Location and Navigation Symposium, pp. 39-46, 1988.
- [9] Mattos, P.G. Integrated GPS and Dead Reckoning for Low cost Vehicle Navigation and Tracking, Proceedings of the Vehicle Navigation and Information Systems Conference, pp. 569-574, 1994.
- [10] Sinn Kim, Jong-Hwan Kim, Ik-Hwan Hyun, "Development of a Map Matching Algorithm for Car Navigation System using Fuzzy Q-factor Algorithm", 5th ITS World Congress proceeding, 1998.
- [11] 기타 GPS 관련 참고 website: <http://www.igeb.gov/sa/diagram.shtm>.