

## 도로매칭 알고리즘 성능 향상에 관한 연구

양인철 · 전우훈\*

한국건설기술연구원 인프라안전연구본부 도로관리통합센터

## A study on performance improvement of map matching algorithm

Inchul Yang · Woo Hoon Jeon\*

Integrated Road Management Research Center, Dept. of Infrastructure Safety Research,  
Korea Institute of Civil Engineering and Building Technology, Goyang-si, Gyeonggi-do, Korea

### [요 약]

차량용 내비게이션, 자율주행과 같은 위치 기반 시스템은 실시간 GPS 좌표를 기반으로 도로망에 차차(ego-vehicle) 위치를 매칭한다. 하지만 기본적인 매칭 알고리즘의 계산량이 매우 크고 자주 반복되기 때문에 실시간으로 수행되는 작업의 속도를 향상시키기 쉽지가 않은 게 현실이다. 이에 본 연구에서는 주행하는 차량의 연속된 두 개의 위치 좌표와 도로 링크의 위상과 방향을 활용하여 더욱 빠르게 가장 가까운 링크를 찾는 효율적인 도로매칭 알고리즘을 개발하였다. 이를 위해 점-선분 간 거리를 이용하는 기본 도로 매칭 작업 전에 대상 링크의 개수를 획기적으로 감소시킬 수 있는 Buffered Boundary Method와 주행 방향 비교 방법을 제안하고 이를 포함하는 도로 매칭 알고리즘을 개발하였다. 표준ITS링크 도로망을 대상으로 성능을 테스트한 결과, 기본 매칭 알고리즘에 비해 월등히 우수한 성능을 보이는 것으로 나타났다.

### [Abstract]

The location based system such as car navigation system and automated driving system matches its location in the road network map using GPS signals on a real-time basis. The basic map matching algorithm, however, runs slow due to its enormous computational complexity and repetitive attribute. Thus the efficient map matching algorithm was proposed using the trajectory of driving vehicles and the topology and direction of road links. Two sub-modules were proposed and included in the algorithm to reduce the number of links for the basic map matching process. The first is Buffered Boundary method, and the second is Driving Direction Comparison method. The performance test results in the Standard-ITS-Link road network demonstrated that the proposed algorithm outperformed the basic map matching algorithm.

**색인어** : 도로매칭, 자율주행 시스템, Buffered Boundary, 주행 방향 비교

**Key word** : Map matching, Automated Driving System, Buffered Boundary, Driving Direction Comparison

<http://dx.doi.org/10.9728/dcs.2020.21.1.213>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Received** 15 November 2019; **Revised** 09 December 2019

**Accepted** 23 January 2020

**\*Corresponding Author; Woo Hoon Jeon**

**Tel:** +82-31-910-0170

**E-mail:** cwsoon@kict.re.kr

## 1. 서론

도로매칭(map matching) 알고리즘([1]-[3])은 차량용 내비게이션, 자율주행과 같이 차량의 위치를 이용하는 첨단 시스템이 실시간 GPS 좌표를 기반으로 도로망에서 자차(ego-vehicle) 위치를 찾는 경우 주로 이용되고, 위치 기반 정보의 공유([4]-[6]) 또는 편집([7]) 서비스에도 많이 이용되고 있다. 하지만 여러 가지 원인으로 인해 GPS를 통해 수신되는 좌표는 실제 위치와 차이를 가지며, 일반적으로 민간에서 사용되는 기기는 수 미터의 오차를 갖는다[8]. 따라서 GPS 좌표만을 이용해서 자차의 위치를 지도상에 표시할 경우 실제 위치와는 다른 곳에 차량이 그려지는 현상이 발생하고, 이를 실시간으로 표출하면 차량이 이 곳저 곳으로 이동하는, 소위 튀는 현상이 발생한다. 자율주행 시스템은 이러한 문제를 해결하기 위해 실시간으로 도로 상에 차량의 위치를 이동시키는 도로매칭이라는 작업을 수행한다. 일반적으로 도로매칭 알고리즘은 하나의 점과 헤딩값<sup>1)</sup>, 또는 두 개 이상 연속된 점으로 구성된 좌표 열과 도로의 기하학적 구조, 위상(topology) 정보를 이용하여 가장 가까운 링크(또는 도로구간)를 찾는 문제를 해결한다. 그림 1은 두 개의 연속된 차량의 위치( $V_{t-1}$ ,  $V_t$ )를 이용하여 인접한 일방통행 도로를 대상으로 도로 매칭을 수행하는 과정을 보여준다. GPS를 통해 수신된 좌표를 기반으로 주변 링크를 거리 순으로 검색하면 링크 L2와 링크 L7이 후보가 된다. 하지만 차량의 진행 방향을 고려할 때 동일한 주행 방향을 갖는 링크 L2가 실제 차량이 위치하는 도로구간일 확률이 높다. 따라서 자율주행 시스템 소프트웨어는 해당 링크에 차량이 있는 것으로 판단하게 된다. 이러한 도로 매칭 알고리즘의 가장 기본이 되는 개념은 앞서 설명한 예제에서와 같이 점과 선분 간의 거리를 이용하여 가까운 거리에 위치한 선분을 찾는 것으로, 관심 지점으로부터 주변의 모든 선분에 수선의 발을 내리고 수선의 발까지의 거리를 계산

하여 가장 가까운 선분을 선택한다. 하지만 이렇게 주변의 모든 선분에 대해 수선의 발과 그 거리를 계산하는 것은 매우 비효율적인 방법으로 시간이 많이 소요되고 사용되는 전산 자원 양도 크다. 따라서 본 연구에서는 주행하는 차량의 연속된 두 개의 위치 좌표와 도로 링크의 위상과 방향을 활용하여 더욱 빠르게 가장 가까운 링크를 찾는 효율적인 도로매칭 알고리즘을 개발한다.

II장에서는 제안된 알고리즘의 세부 내용을 설명하고, 이어지는 III장에서는 실제 도로망을 대상으로 제안된 알고리즘의 성능을 검증하고 그 결과를 분석한다. 그리고 마지막으로 IV장에서 결론을 제시한다.

## II. 효율적인 도로 매칭 알고리즘 개발

### 2-1 개요

도로 매칭 알고리즘의 성능을 향상시키기 위해서는 검색되는 링크의 개수를 감소시키는 방법이 가장 효율적이다. 자차 주변에 10개의 링크가 있을 경우, 10개 모두를 대상으로 거리를 계산하는 것보다 관련성이 높은 2-3개의 링크만 검색하여 계산할 수 있다면 속도를 증가시키고 전산 비용을 감소시킬 수 있다. 이러한 작업이 매 초마다 수행되는 자율주행 시스템 측면에서는 절약할 수 있는 비용이 엄청날 것이다. 본 연구에서는 이러한 기본 개념을 이용하여 도로 매칭 알고리즘의 성능을 향상시키기 위한 두 가지의 전처리기를 제안하였다. 첫 번째는 Buffered Boundary Method(BBM)이고, 두 번째는 주행 방향 비교 방법(Driving Direction Comparison Method, DDCM)이다. BBM은 선분이 차지하는 사각형 공간을 일시적으로 확장한 후 그 공간 내에 차량이 존재하는지 확인하여 공간적 근접성 여부

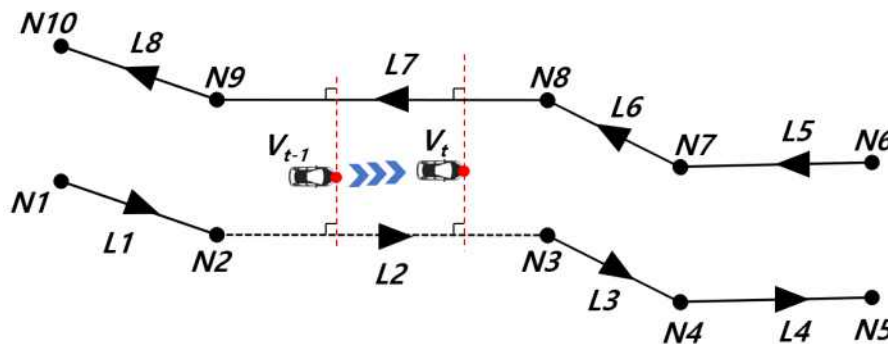


그림 1. 도로매칭 개념  
Fig. 1. Concept of map matching

1) 헤딩값(heading)은 차량의 주행 방향을 진북 기준 시계방향으로 표현한 것으로, 0~359까지의 값을 갖는다.

를 판단하는 방법이고, DDCM은 차량의 연속된 두 좌표를 이용하여 도로 링크와의 방향성을 비교하여 검색 대상 여부를 판단한다. 이러한 두 가지의 전처리기를 통해 필터링 된 소수의 도로 링크만을 대상으로 거리 계산을 수행하기 때문에 알고리즘의 효율성을 극대화할 수 있다.

그림 2는 제안된 도로 매칭 알고리즘의 절차를 보여준다. 먼저 두 개 이상의 차차 위치 좌표를 취득하고, 이를 이용하여 주변의 모든 링크를 대상으로 BBM 방법을 이용하여 공간적으로 근접성이 떨어지는 링크를 제외한다. 그 후 DDCM을 이용하여 방향성이 다른 링크를 제외하고, 남은 링크를 대상으로 점과 선분 간 최단 거리를 계산하여 가장 가까운 링크를 찾는다.

**2-2 Buffered Boundary Method**

도로 매칭 알고리즘은 주어진 좌표를 이용하여 주변 도로망의 모든 링크와의 거리를 계산하여 가장 가까운 링크를 찾는다. 따라서 주변에 링크가 많을수록 계산량이 증가하여 효율성이 감소한다. 만약 주어진 좌표와 링크 간의 거리를 계산하는 비용보다 저렴한 비용으로 대략의 거리를 가늠할 수 있다면 알고리즘을 훨씬 효율적으로 만들 수 있다.

본 연구에서 제안하는 BBM 방법은 주어진 차량의 좌표와 링크의 시작점과 끝점의 좌표, 그리고 단순 더하기, 빼기, 비교 연산만을 이용하여 관련성이 적은 링크를 배제할 수 있다. 여기서 관련성이 적은 링크는 차량이 위치할 확률이 매우 적은 링크를 의미한다.

BBM 방법의 개념은 그림 3과 같다. 모든 링크에 대해 Buffered Boundary (BB)를 설정한다. BB는 링크를 대각선으로 갖는 사

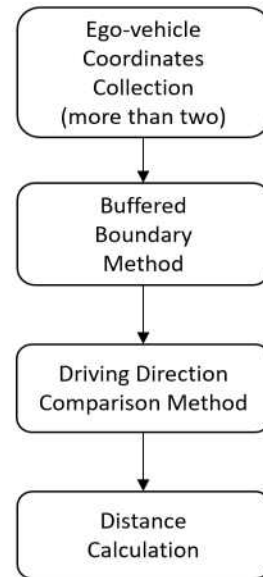


그림 2. 효율적인 도로 매칭 알고리즘 절차  
Fig. 2. Procedure of the proposed algorithm

각형의 네 꼭짓점에 buffer 값을 더하거나 빼서 확장한 사각형을 의미한다. 그림 3에서는 링크 L2와 링크 L4의 BB(점선 사각형)를 보여주고 있는데, 만약 링크가 수평이거나 수직 형태라서 사각형이 만들어지지 않더라도 시작점과 끝점을 기준으로 확장한 사각형을 설정하면 된다. 이때 buffer 값은 상하좌우(또는 동서남북)에 대해 동일하게 할 수도 있고, 상이하게 할 수도 있다. 다만, buffer 값은 적어도 차로 1개의 너비인 3~3.5미터보다는 크게 하는 것이 필요하다. 그보다 작을 경우 가까운 링크가 배제될 수 있기 때문이다.

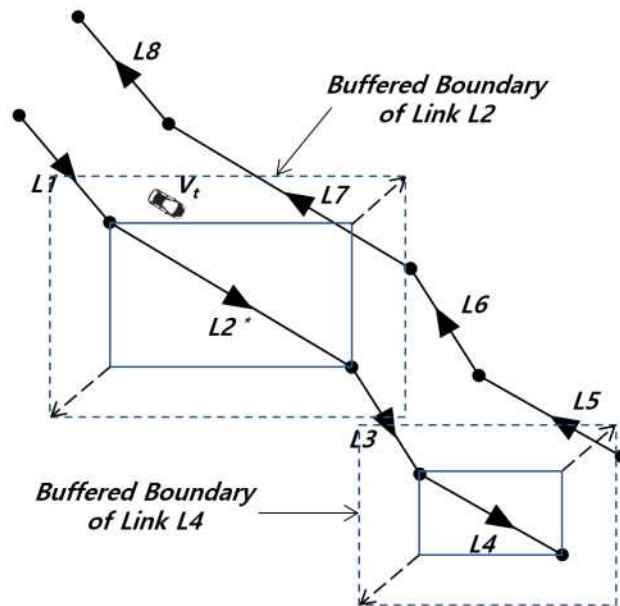


그림 3. Buffered Boundary Method 개념  
Fig. 3. Concept of Buffered Boundary Method

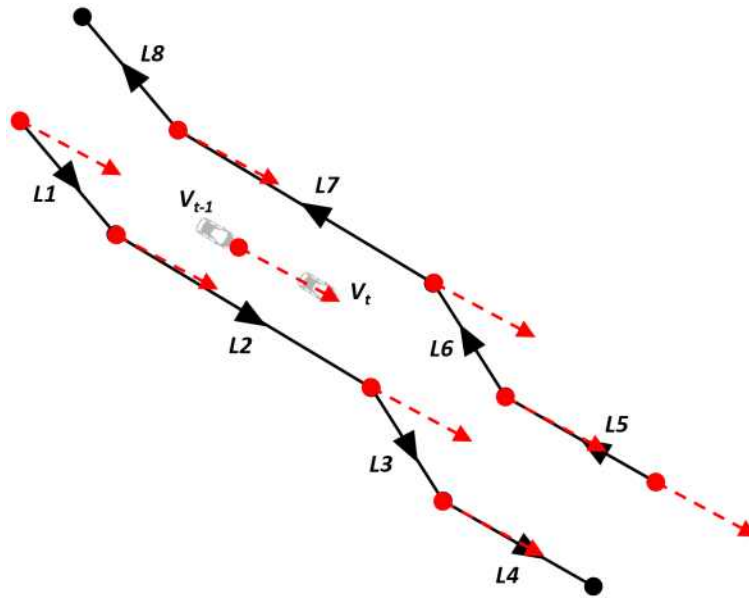


그림 4. 주행 방향 비교 방법의 개념  
 Fig. 4. Concept of finding same directional link

BB가 설정되면 다음의 네 가지 조건을 모두 만족하는 링크는 선택하고, 그렇지 않은 링크는 배제한다. 여기서  $X_t$ 와  $Y_t$ 는 주어진 차량의 좌표를,  $X_t^{LL}$ 과  $Y_t^{LL}$ 은 BB의 좌하단 점의 좌표를,  $X_t^{UR}$ 과  $Y_t^{UR}$ 은 BB의 우상단 점의 좌표를 의미한다. 조건1과 2는 차량의 X축 위치가 BB 내에 있어야 함을 의미하는 것으로, BB의 좌하단 X좌표보다 크고 우상단의 X좌표보다 작을 경우 두 조건을 모두 만족하여 BB 내 존재하는 것으로 판단한다. 마찬가지로 조건 3과 4는 차량의 Y축 위치가 BB 내에 있어야 함을 나타내는 것으로, BB의 좌하단과 우상단의 Y좌표와 크기를 비교하여 BB 내 존재 여부를 판단한다.

조건 1:  $X_t \geq X_t^{LL} - buffer$

조건 2:  $X_t \leq X_t^{UR} + buffer$

조건 3:  $Y_t \geq Y_t^{LL} - buffer$

조건 4:  $Y_t \leq Y_t^{UR} + buffer$

주어진 차량의 위치 좌표가 위의 네 가지 조건을 모두 만족하는 것은 해당 좌표가 BB 내에 존재하며, 이는 곧 해당 링크가 차량에서 가장 가까운 링크일 확률이 높은 것으로 판단할 수 있다. 하지만 네 가지 조건 중 하나라도 만족하지 않는 링크는 가장 가까운 링크일 확률이 낮기 때문에 거리 계산 대상에서 제외시킨다.

BBM과 같은 필터링 방법의 단점은 조건에 부합되지 않는 링크는 사전에 제외되기 때문에 실제 거리 계산에서는 전혀 고려되지 않을 수 있다는 것이다. 이러한 경우 제외된 링크 중 실제로

가까운 링크가 존재하면 문제가 된다. 따라서 이에 대한 보완책으로 조건을 만족하지 못하는 수준에 따라 관리하는 방법이 있을 수 있다. 조건에서 벗어나는 정도를 인덱싱 하여 그 값에 따라 순위를 관리함으로써 이후 알고리즘 절차에서 다시 고려할 기회를 가질 수 있도록 하는 것이다. 이러한 팁은 전체 알고리즘에 있어서 경미한 부분이지만 실제 서비스에서는 고객 만족도 향상에 큰 영향을 미치는 변수가 된다.

### 2-3 주행 방향 비교 방법 (DDCM)

DDCM은 BBM 이후에 수행된다. BBM 조건을 만족하는 링크 중 차량의 주행 방향과 일치하지 않는 링크를 필터링하는 방법으로, 차량의 방향 벡터와 링크의 주행 방향 벡터를 비교하여 결정한다. 그림 4는 주행 방향 비교 방법의 개념을 보여준다. 연속된 두 개의 차량 위치,  $V_{t-1}$ 과  $V_t$ 를 나타내는 벡터와, 각 링크의 시작점과 끝점을 연결하는 벡터의 내적을 계산한 후, 그 값이 음수(negative)일 경우 두 벡터 간 각도가  $90^\circ \sim 270^\circ$ 임을 의미하므로 방향이 다른 것으로 판단하고, 그 값이 양수(positive)일 경우 두 벡터 간 각도가  $0^\circ \sim 90^\circ$  또는  $270^\circ \sim 360^\circ$ 임을 의미하므로 방향이 동일한 것으로 판단한다.

DDCM을 활용함에 있어서 주의할 점은 시간적으로 연속된 차량의 위치 좌표가 방향성을 대표할 수 있어야 한다는 것이다. 예를 들어 신호 대기 중인 차량은 GPS로 수신된 위치 좌표가 이곳 저곳으로 튀기 때문에 시간적으로 연속된 두 개의 좌표로 방향성을 알아내기 쉽지 않다. 이러한 경우 이전에 매칭된 링크를 이용하거나 정지하기 전에 수집된 좌표를 벡터의 원점

으로 활용할 수 있다.

### III. 성능 검증

본 장에서는 제안된 도로 매칭 알고리즘의 성능을 평가하기 위해 정확도와 효율성 측면에서 실험 평가를 수행하였다. 이를 위해 자유로와 강변북로 일대의 62개 표준ITS링크로 구성된 도로망 데이터를 이용하였는데, 표준ITS링크는 교통정보 수집과 제공, 도로운영 등에 활용하기 위해 구축 및 관리되는 전자교통지도로, 공식 웹사이트를 통해 무료로 이용이 가능하다.

그리고 제안된 알고리즘의 성능 우수성을 비교 검증하기 위해 제안된 두 개의 모듈(BBM, DDCM), 그리고 링크와의 거리를 계산하는 모듈(Distance Measure Tool, DMT)의 사용 여부에 따라 다음과 같이 4개의 알고리즘을 준비하였다(표 1).

표 1. 알고리즘의 구분

Table 1. Algorithm Classification

| Name | BBM | DDCM | DMT |
|------|-----|------|-----|
| MM1  | ○   | ○    | ○   |
| MM2  | ○   | ✗    | ○   |
| MM3  | ✗   | ○    | ○   |
| MM4  | ✗   | ✗    | ○   |

첫 번째(MM1)는 본 연구에서 제안된 알고리즘으로, BBM과 DDCM, DMT를 순서대로 모두 사용하는 것이고, 두 번째(MM2)는 BBM과 DMT를, 세 번째(MM3)는 DDCM와 DMT를, 그리고 마지막(MM4)은 DMT만을 이용하는 알고리즘이다. BBM과 DDCM은 상호 종속적이지 않고 독립적으로 운용이 가능하지만 두 개 모두 DMT와 함께 사용해야 한다. 따라서 MM2와 MM3와 같이 BBM+DMT, 또는 DDCM+DMT 형태의 사용을 통해 도로매칭을 수행한다. MM4는 DMT만으로 도로매칭을 수행하는 것으로 제안된 알고리즘의 효율성을 검증하기 위

표 2. 성능 테스트 결과 (정확도)

Table 2. Performance test result (Accuracy)

| No | Coordinate #1 |           | Coordinate #2 |           | ITS Link ID | MM1 | MM2 | MM3 | MM4 |
|----|---------------|-----------|---------------|-----------|-------------|-----|-----|-----|-----|
|    | X             | Y         | X             | Y         |             |     |     |     |     |
| 1  | 126.725960    | 37.655063 | 126.726571    | 37.654802 | 2410000700  | ○   | ○   | ○   | ○   |
| 2  | 126.834410    | 37.593560 | 126.834576    | 37.593386 | 2180000303  | ○   | ○   | ○   | ○   |
| 3  | 126.827260    | 37.602970 | 126.827074    | 37.603097 | 2180002601  | ○   | ○   | ○   | ○   |
| 4  | 126.741698    | 37.647232 | 126.741821    | 37.647188 | 2190099800  | ○   | ○   | ○   | ○   |
| 5  | 126.743277    | 37.646793 | 126.743124    | 37.646869 | 2190099700  | ○   | ○   | ○   | ○   |
| 6  | 126.738057    | 37.649123 | 126.737845    | 37.649217 | 2190000600  | ○   | ○   | ○   | ○   |
| 7  | 126.731401    | 37.652239 | 126.731740    | 37.652129 | 2410000300  | ○   | ○   | ○   | ○   |
| 8  | 126.728378    | 37.653878 | 126.728633    | 37.653751 | 2410000500  | ○   | ○   | ○   | ○   |
| 9  | 126.825883    | 37.603309 | 126.826401    | 37.603181 | 2180002501  | ○   | ○   | ○   | ○   |
| 10 | 126.830604    | 37.599089 | 126.830366    | 37.599377 | 2180001404  | ○   | ○   | ○   | ○   |

한 기본 알고리즘으로 사용된다.

C/C++ 언어를 이용하여 표준ITS노드링크 데이터를 읽고 도로 매칭 알고리즘을 수행하는 프로그램을 만들었고, BB 모듈 수행을 위한 buffer의 크기는 5m로 하였다. 또한 사용한 서버는 Intel Core i9 (3.6GHz) CPU, 32GB 메모리로 구성되고, 운영체제는 윈도우10이다.

먼저 각 알고리즘의 정확도를 평가하기 위해 10개의 임의의 좌표 쌍에 대해 매칭을 수행한 결과, 표 2와 같이 모두 정확하게 매칭 하였다.

다음으로 효율성을 검증하기 위해 한 쌍의 좌표에 대해 알고리즘을 각각 1,000,000회씩 수행하였고, 그에 따른 소요 시간은 표 3과 같다. 제안된 알고리즘인 MM1이 188 msec로 가장 우수한 성능을 보였고, 이는 거리 계산만으로 도로 매칭을 수행하는 MM4에 비해 약 50배 빠른 것이다. 이러한 결과는 계산 비용이 낮은 방법으로 사전에 관련성이 적은 링크 들을 필터링하는 BBM과 DDCM의 접근 방법이 우수함을 나타낸다.

표 3. 성능 테스트 결과 (효율성)

Table 3. Performance test result (Efficiency)

| Name | Elapsed Time (msec) |
|------|---------------------|
| MM1  | 188                 |
| MM2  | 344                 |
| MM3  | 5641                |
| MM4  | 9141                |

MM1 다음으로 BBM만을 활용한 MM2가 우수한 성능을 보였다. DDCM만을 활용한 MM3는 MM1에 비해 30배 이상, MM2에 비해 16배 이상의 시간이 소요되는 것으로 나타났다. 이는 DDCM으로 인한 성능 향상 효과는 BBM에 비해 미미함을 의미하는 것으로, 비록 공간적으로 많이 떨어져 있더라도 주행 방향이 유사한 링크는 상당수가 존재하기 때문인 것으로 해석될 수 있다. 역으로 이러한 결과는 BBM으로 인한 성능 향상 효과

가 DDCM에 비해 월등히 크다는 것으로, 링크의 방향성 보다는 차량과의 공간적 근접성을 이용하여 관련성이 낮은 링크를 필터링하는 것이 훨씬 효과적임을 의미한다. 하지만 이러한 해석이 DDCM의 성능 향상 효과를 무시하는 것으로 오해할 필요는 없다. DDCM에 따른 성능 향상 효과는 MM1이 MM2에 비해 2배 정도 빠른 속도를 보이는 것으로 확인할 수 있기 때문이다.

#### IV. 결 론

본 연구에서는 효율적인 도로 매칭을 위해 Buffered Boundary Method (BBM)와 주행 방향 비교 방법 (Driving Direction Comparison Method, DDCM)을 제안하고 알고리즘을 개발하였다. 차량의 공간적 위치 좌표를 이용하여 전체 도로망을 구성하는 링크 중 관련성이 적은 링크를 사전에 제거함으로써 큰 계산량을 요구하는 점-선분 간 거리 계산 횟수를 획기적으로 감소시켰다. 제안된 방법의 성능 검증을 위해 표준ITS링크 도로망을 대상으로 테스트를 수행한 결과, 거리 계산만으로 도로 매칭을 하는 방법에 비해 월등히 우수한 것으로 나타났다. 그리고 BBM과 DDCM 모두 성능 향상에 기여를 하지만, 그 중에서도 BBM의 기여도가 상대적으로 매우 큰 것으로 나타났다. 이는 링크의 방향성 보다는 BBM 유형과 같이 공간적 근접성을 비교하는 방식이 우수하다는 것을 의미한다.

도심도로의 경우 연속류 도로에 비해 복잡도가 높기 때문에 제안된 알고리즘을 수행할 경우 효율성이 다르게 나타날 수 있으므로 이에 대한 추가 성능 검증이 필요하다. 또한 BBM 모듈의 경우 버퍼 크기의 조정을 통해 도심도로를 위한 알고리즘의 최적화가 가능하기 때문에 이러한 내용을 포함하는 도심도로 대상 도로매칭 알고리즘에 대한 추가 연구가 필요할 것으로 판단된다. 또한, 실제 차량용 내비게이션과 자율주행용 전자지도와 같이 대규모 데이터베이스에서는 우리가 쉽게 예측하지 못하는 다양한 상황이 존재하기 때문에 이러한 데이터를 대상으로 제안된 알고리즘의 성능을 검증함으로써 실용화 가능성을 점검할 필요가 있다.

#### 참고문헌

[1] Quddus, Mohammed A., Washington Y. Ochieng, and Robert B. Noland. "Current map-matching algorithms for transport applications: State-of-the art and future research directions." *Transportation research part c: Emerging*

*technologies* 15.5 (2007): 312-328.  
[2] Ochieng, Washington Y., Mohammed A. Quddus, and Robert B. Noland. "Map-matching in complex urban road networks." 2003  
[3] Brakatsoulas, Sotiris, et al. "On map-matching vehicle tracking data." *Proceedings of the 31st international conference on Very large data bases. VLDB Endowment*, 2005.  
[4] Yang, I., Jeon, W-H., Lee, H-M., "A Study on Dynamic Map Data Provision System for Automated Vehicle", *Journal of Korea Institute of Intelligent Transport Systems*, Vol. 16, No. 6, pp. 208~218, 2017.  
[5] Yang, I. & Jeon, W-H., "Development of lane-level location data exchange framework based on high-precision digital map", *Journal of Digital Contents Society*, Vol. 19, No. 8, pp. 1617-1623, 2018.  
[6] Yang, I. & Jeon, W-H., "Development of Lane-level Dynamic Location Referencing Method", *Journal of Korea Institute of Intelligent Transport Systems*, Vo. 17, No. 5, pp. 188~199, 2018  
[7] Yang, I. & Jeon, W-H., "Digital road event management system using high-precision map", *Journal of Digital Contents Society*, Vol. 19, No. 11, pp. 2219-2226, 2018.  
[8] Akim, E. L., and D. A. Tuchin. "GPS errors statistical analysis for ground receiver measurements." *The Proceedings of the 17th International Symposium on Space Flight Dynamics*. 2003.

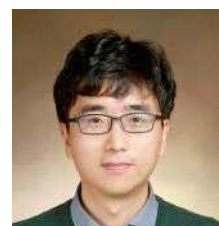


양인철(Inchul Yang)

2011년: Ph.D. in Civil Engineering at Univ. of California, Irvine  
2000년: 연세대학교 도시공학석사  
1998년: 연세대학교 도시공학 학사

2011년~현재: 한국건설기술연구원 연구위원

\*관심분야: 첨단교통, 자율주행, C-ITS, 도로안전, 도로시설



전우훈(Woo Hoon Jeon)

2016년: 서울대학교 도시계획학 박사  
2001년: 한양대학교 교통공학 석사  
1999년: 한양대학교 교통공학 학사

2001년~현재: 한국건설기술연구원 수석연구원

\*관심분야: 도로안전, 무동력 교통수단, 모바일 앱, 도로시설