
규칙-기반 분류화 기법을 이용한 도로 네트워크 상에서의 주행 시간 예측 알고리즘

Travel Time Prediction Algorithm using Rule-based Classification on Road Networks

이현조, 니하드 카림 초우더리, 장재우
전북대학교 컴퓨터공학과

Hyun-jo Lee(hjlee@dblab.chonbuk.ac.kr), Nihad Karim Chowdhury(nihad@dblab.chonbuk.ac.kr),
Jae-Woo Chang(jwachang@chonbuk.ac.kr)

요약

동적 경로 안내 시스템과 같은 첨단 여행 정보 시스템(ATIS)의 발전에 따라 도로 네트워크 상에서 보다 정확한 주행 시간 예측 기법에 대한 연구가 활발히 진행되고 있다. 그러나 기존 대부분의 연구들은 주어진 경로 상의 평균 주행 속도만을 기반으로 주행 시간을 예측한다. 이는 러시아워 시간대의 혼잡한 도로, 주말에 교외로 나가는 대규모의 차량 등과 같은 일별 혹은 주별 도로 교통 상황을 반영하지 못하기 때문에, 주행 시간 예측의 정확도가 저하된다. 이를 해결하기 위해 본 연구에서는 규칙-기반 분류화 기법을 이용한 주행 시간 예측 알고리즘을 제안한다. 제안된 알고리즘은 데이터마이닝 기법인 규칙-기반 분류화 기법을 사용하여, 과거 차량의 궤적 데이터로부터 하루의 시간대별 교통량과 주별 차량의 운행 양식 등 도로 교통 상황을 추출하고, 이를 통해 차량의 주행 시간을 보다 정확하게 예측한다. 제안된 알고리즘을 기존의 링크-기반 예측(link-based prediction) 알고리즘, Micro T* 알고리즘[3], 그리고 스위칭(switching) 알고리즘[10]과 예측 정확도 측면에서 성능 비교를 수행한다. 예측 정확도 성능 비교 결과, 제안된 기법이 타 예측 기법에 비해 MARE (mean absolute relative error) 가 크게 감소하여 성능이 향상됨을 보인다. 그 밖에 다른 기법들과 장단점을 비교하여, 제안된 기법의 유용성을 나타낸다.

■ 중심어 : | 주행 시간 예측 | 규칙-기반 분류화 기법 | 첨단 여행 정보 시스템(ATIS) |

Abstract

Prediction of travel time on road network is one of crucial research issue in dynamic route guidance system. A new approach based on Rule-Based classification is proposed for predicting travel time. This approach departs from many existing prediction models in that it explicitly consider traffic patterns during day time as well as week day. We can predict travel time accurately by considering both traffic condition of time range in a day and traffic patterns of vehicles in a week. We compare the proposed method with the existing prediction models like Link-based, Micro-T* and Switching model. It is also revealed that proposed method can reduce MARE (mean absolute relative error) significantly, compared with the existing predictors.

■ keyword : | Travel Time Prediction | Rule-based Classification | Advanced Travelers Information System |

* 본 논문은 2008년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2008-000-11002-0)
접수번호 : #080707-001
심사완료일 : 2008년 10월 02일
접수일자 : 2008년 07월 07일
교신저자 : 장재우, e-mail : jwachang@chonbuk.ac.kr

I. 서론

첨단 여행 정보 시스템(Advanced Travelers Information Systems: ATIS) [1]과 지능형 교통 시스템(intelligent transportation system: ITS)에서는 여행 계획 수립, 동적 경로 안내 등과 같은 다양한 서비스를 제공하고 있다. 이러한 서비스에서는 주행 시간 예측 알고리즘을 사용하여 차량의 주행 시간을 예측하고, 사용자들은 이를 바탕으로 보다 효과적으로 일정 및 시간을 관리한다[2]. 따라서 출발지에서 목적지까지 주행 시간을 보다 정확하게 예측함으로써, 사용자들의 만족도를 향상시킬 수 있다.

기존에 연구된 대부분의 주행 시간 예측 알고리즘은 전체 경로에서의 평균 차량 운행 시간을 측정하고, 이를 바탕으로 주행 시간을 예측한다. 그러나 평균 주행 시간을 이용하기 때문에, 시간대에 따라 변하는 교통 상황을 전부 반영하지는 못한다. 예를 들어, 러시아워에는 교통량이 급격히 증가하여 차량의 주행 속도가 저하될 것이고, 새벽녘에는 교통량이 감소하여 차량의 주행 속도가 증가할 것이다. 이렇듯 시간대에 따라 발생하는 평균 주행 속도와 실제 주행 속도 사이에 오차때문에, 주행 시간 예측의 정확도가 저하된다.

따라서 본 논문에서는 주행 시간 예측의 정확도를 향상시키기 위해서, 도로 구간별로 발생하는 주기적인 교통 상황을 고려하고자 한다. 즉, 하루의 시간대에 따른 교통량 변화를 의미하는 일별 교통 상황 및 각 요일에 따른 교통 양식의 변화를 의미하는 주별 교통 상황을 고려한다. 예를 들어, 일별 교통 상황은 아침, 저녁, 그리고 러시아워 등 시간대에 따른 교통량의 변화를 의미하며, 주별 교통 상황은 크게 주말과 주말에 따른 교통 양식의 변화를 의미한다. 이러한 도로 구간에서의 조건별 교통 상황 정보는 단순히 제공되지 않기 때문에, 데이터 마이닝 기법인 규칙-기반 분류화 기법을 이용하여 도로 구간에서의 조건별 교통 상황 정보를 추출한다. 즉, 제안하는 알고리즘은 첫째, 차량의 궤적 데이터로부터 도로 구간별 교통 상황 정보를 추출하고, 둘째, 추출된 교통 상황 정보를 바탕으로 사용자가 입력한 임의의 경로에 대해 주행 시간을 예측하는, 크게 두가지

부분으로 나뉘어 진다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서는 제안된 알고리즘에 대하여 설명한다. 4장에서는 제안된 알고리즘의 성능 평가를 수행하며, 마지막으로, 5장에서는 결론 및 향후 연구를 제시한다.

II. 관련 연구

도로 네트워크 상에서 보다 정확한 주행 시간 예측을 위해 다양한 연구가 활발히 진행되고 있으며, 이러한 연구들은 크게 경로-기반(path-based) 기법과 링크-기반(link-based) 기법의 두 가지로 분류할 수 있다. 먼저 경로-기반 기법이란, 도시와 도시간의 이동과 같은 장거리 주행 상황, 특히 고속도로를 이용하는 경우의 주행 시간을 예측하는데 적합한 기법이다. 고속도로의 경우 도시와 도시를 이동하기 위해 선택가능한 경로의 수가 적기 때문에, 사용자의 선호도가 높거나 혹은 최단 거리를 지나는 특정 경로들을 지정하기 쉽다. 따라서 경로-기반 기법에서는 미리 지정된 특정 경로들에 대한 주행 시간을 측정하고, 이를 바탕으로 주행 시간을 예측하게 된다. 반면 링크-기반 기법이란, 도로 네트워크가 복잡하게 구성된 도시에서, 사용자가 입력하는 다양한 경로에 대해 주행 시간을 예측하는데 적합한 기법이다. 경로-기반 기법에서 미리 정의된 소수의 특정 전체 경로에서의 주행 시간을 측정하는 것과는 달리, 링크-기반 기법에서는 도시의 도로 네트워크를 구성하는 모든 도로 구간(road segment)에 대해 각각의 주행 시간을 측정한다. 이후 사용자가 원하는 이동 경로를 입력하면, 이를 구성하는 도로 구간들을 선택하여, 각 구간별 주행 시간을 합산하여 전체 주행 시간을 예측하게 된다.

대부분의 기존 연구들은 주행 시간 예측을 위해 경로-기반 기법을 사용하였다[4-10][13]. 첫째, Park[4][5] 연구는 인공 신경망(Artificial Neural Network)을 이용한 고속 도로에서의 주행 시간 예측 알고리즘을 제안하였다. 제안된 알고리즘은 시스템에 저장된 특정 경로의

교통 상황 패턴을 각각 SOFM (Self Organizing Feature Map) 기법 및 fuzzy c-means 클러스터링 (clustering) 기법을 이용하여 분류하였으며, 분류된 교통 상황 패턴을 통해 주행 시간을 예측한다. 이는 우수한 성능을 나타내지만, 알고리즘이 복잡하고, 많은 계산 시간을 요구하기 때문에 실제 응용에는 적합하지 않다. 둘째, Kwon[6] 연구는 출발 시간 범위를 고려한 선형 회귀(linear regression) 주행 시간 예측 알고리즘을 제안하였다. 제안된 알고리즘은 현재 시간과 출발하고자 하는 시간과의 시차에 따라, 시차가 20분 이내일 경우 현재 관측되고 있는 차량 주행 시간을, 시차가 20분 이상일 경우 과거 차량 주행 시간 기록을 사용하여 주행 시간을 예측한다. 셋째, Zhang[7] 연구는 출발 시간에 따라 주행 시간 측정 함수의 계수가 가변적으로 변하는 주행 시간 예측 알고리즘을 제안하였다. 즉, 출발 시간에 따라 평균 주행 시간이 변하기 때문에, 이를 주행 시간 예측을 위한 함수에 반영하여 보다 정확하게 주행 시간을 예측한다. 넷째, Rice[9] 연구는 유사 주행 시간 패턴을 고려한 최근접(nearest neighbors) 주행 시간 예측 알고리즘을 제안하였다. 즉, 현재 관측되는 데이터를 토대로 과거 주행 시간 기록 중 현재 날짜와 가장 유사한 일자의 데이터를 탐색하여, 주행 시간을 예측한다. 다섯째, Erick[10] 연구는 두 가지의 예측 기법을 이용한 스위칭(switching) 주행 시간 예측 알고리즘을 제안하였다. 제안된 알고리즘은 선형 예측 기법과 차량의 평균 주행 시간 기반 예측 기법 중 현재 도로의 교통 상황에 보다 적합한 기법을 선택하여 주행 시간을 예측한다. 우선 과거 차량의 주행 기록을 바탕으로, 두 기법 중 하나의 예측 정확도가 다른 기법에 비해 높아지는 시간 기준점을 찾는다. 이후 질의 시간과 기준점을 비교해서 두 기법 중 하나를 선택하는 것이다. 마지막으로, 김주현[13]의 연구는 차량 번호판 인식 시스템 (Automatic Vehicle Identification: AVI)을 이용하여 특정 경로에서의 차량의 주행 시간 예측 알고리즘을 제안하였다. 제안된 알고리즘은 매일의 AVI 데이터를 축적하고, 예측 시간으로부터 30분 이내에 도착한 차량의 여행 시간 패턴을 기준으로 가장 유사한 예측 시간 패턴을 검색한다. 이후 검색된 예측 시간 패턴을 통해 현

재 시간에 출발하는 차량의 도착 시간을 예측하는 것이다.

경로-기반 기법들은 전체 경로에 대한 주행 시간을 측정하고, 측정된 주행 시간의 오차를 줄이는데 초점을 맞추고 있다. 따라서 고속도로와 같이 선택 가능한 전체 경로의 수가 비교적 적으며, 일별 혹은 주별 교통 상황의 변동이 적은 경우 주행 시간 예측의 정확도가 높다. 반면 도심에서와 같이 선택 가능한 전체 경로의 수가 매우 많은 경우 사용자로부터 주어지는 다양한 질의에 대해 응답하기 어려우며, 뿐만 아니라 일별 혹은 주별 교통 상황의 변동이 심하기 때문에 예측 정확도가 저하된다. 이러한 경로-기반 기법의 문제점을 보완하기 위해, 기존 경로-기반 기법에 링크-기반에서 사용하는 도로 구간별 주행 시간 측정을 결합한 제한된 링크-기반 기법이 연구되었다[1][14]. 첫째, Chen et al [1] 연구에서는 도로 네트워크 상에서 사용자들이 선호하는 주요 경로를 미리 정의하고, 주요 경로들을 구성하는 모든 도로 구간에서 각각의 주행 시간을 측정하였다. 이를 통해 일별 혹은 주별 교통 상황의 변동에 따른 영향을 감소시켰다. 둘째, 이청원[14]의 연구는 주행 시간 예측 정확도 향상을 위해 칼만 필터를 이용한 차량의 주행 시간 예측 알고리즘을 제안하였다. 칼만(Kalman) 필터란, 과거에 측정된 차량의 주행 시간과, 주기적으로 업데이트(update)되는 차량의 주행 시간을 비교하여 둘 사이의 오차를 최소화하는 기법이다. 제안된 알고리즘은 남산 터널을 중심으로 한 선호 경로를 미리 정의하고, 이를 구성하는 도로 구간에 대한 차량의 평균 주행 속도를 칼만 필터를 적용하여 측정함으로써, 주행 시간 예측의 정확도를 향상하였다.

한편, 링크-기반 기법으로는 다음과 같은 연구가 존재한다[3][15]. 첫째, Kwon[3] 연구는 사용자 지정 경로에 대해 주행 시간을 예측하는 Micro T* 알고리즘을 제안하였다. 사용자 지정 경로란 사용자로부터 출발지점, 도착지점 및 두 지점 사이의 전체 경로가 주어지는 것이다. Micro T* 알고리즘은 먼저 전체 도로 네트워크 상의 모든 구간에 대해 구간별 평균 주행 시간을 측정하며, 만약 질의 경로가 주어지면 이를 구성하는 모든 구간의 평균 주행 시간을 합산하여 전체 주행 시간을 예측한다. 그러나 Micro T* 알고리즘은 도로의 정

체와 같은 외부적인 요인에 의해 시간대별로 그 예측 정확도가 저하되는 문제점이 존재한다. 둘째, 이영우 [15]의 연구는 퍼지 추론을 이용한 주행 시간 예측 알고리즘을 제안하였다. 차량의 통행 시간 분포는 교통 신호기가 설치된 교차로를 중심으로 신호에 의한 지체가 있는 경우와 신호 지체가 없는 경우로 분류된다. 제안된 알고리즘은 두 가지 분류에 대한 영향 요인을 선정하는 퍼지 기법을 통해 차량의 주행 시간 예측을 지원한다. 그러나 제안된 알고리즘은 퍼지 기법을 사용하기 때문에 알고리즘이 복잡하고, 많은 계산 시간을 요구한다.

III. 규칙-기반 분류화 기법을 이용한 주행 시간 예측 알고리즘

최근 네비게이션의 보급과 더불어 도시 내에서의 차량 주행에 관련된 다양한 서비스가 각광받고 있다. 차량 정체시 우회 경로를 탐색하고, 이에 대한 예측 주행 시간을 측정하는 서비스는 대표적이라 할 수 있다. 이러한 추세를 고려하여, 본 논문에서는 도시의 도로 네트워크에 적합한 링크-기반 주행 시간 예측 기법에 초점을 맞추고자 한다.

기존에 제안된 대표적인 링크-기반 기법인 Micro T* 알고리즘에서는 경로(route)를 여러 개의 도로 구간(load segments)으로 분할하여, 각 도로 구간마다 TVC(Time-Varying Coefficient) 기법을 기반으로 주행 시간을 측정한다. 최종적으로 측정된 구간별 주행 시간을 더하여 주행 시간을 예측한다. 그러나 Micro T* 알고리즘은 도로 구간의 평균 주행 시간만을 고려하며, 러시아워나 새벽 시간과 같이 서로 다른 시간대의 주행 속도를 고려하지 못하여 정확성이 저하된다. 이러한 문제점을 해결하기 위해, 본 논문에서는 규칙-기반(rule-based) 분류화 기법을 이용한 새로운 주행 시간 예측 알고리즘을 제안한다. 제안된 알고리즘은 사용자가 지정한 임의의 경로에 대해 주행 시간 예측이 가능할 뿐만 아니라, 경로를 구성하는 각 도로 구간에 대해 일별 및 주별 교통 상황에 따른 평균 구간 속도를 고려함으로써 보다 정확한 주행 시간 예측이 가능하다. 일

별 교통상황을 고려하기 위해 하루를 다수개의 시간 그룹으로 나눈다. 이를 통해 교통량이 많은 아침 러시아워의 시간대와 반대로 교통량이 적은 늦은 저녁 등, 하루의 시간대에 따라 서로 다른 도로의 교통 상황을 고려한다. 마찬가지로 주별 교통 상황을 고려하기 위해 일주일을 주일과 주말의 두 가지 그룹으로 나눈다. 이러한 일별, 주별 그룹의 조합에 따른 차량의 평균 주행 속도를 구하기 위해 도로 구간별 속도 등급(velocity class)을 고려한다. 도로 구간별 속도 등급이란 각 도로 구간마다 그룹별 교통량의 측정값을 의미하며, 이를 측정하기 위해 본 논문에서는 규칙-기반 분류화 기법을 사용한다. 도로 구간별 속도 등급을 측정하면, 이를 통해 쉽게 각 도로 구간별 주행 시간을 계산할 수 있으며, 질의 경로를 구성하는 도로 구간별 주행 시간을 합산하여 총 주행 시간을 예측할 수 있다.

1. 속도 등급, 출발 시간 그룹 및 주별 요일 그룹

도로 네트워크에서 속도 등급(velocity class)은 도로의 교통 상황에 따라 달라질 수 있다. 이를 반영하기 위해, 각 도로 구간별 속도를 세 등급으로 분류한다. 속도 등급의 각 속성들은 {VB, B, F}로써, 각각 Very Busy, Busy 그리고 Free 를 의미한다. 분류 기준은 일반 도로에서의 차량의 운행 속도이다. 즉, 일반적인 경우 운행 속도 범위인 35~60 km/hr를 기준으로, 이보다 느린 경우는 교통 체증이 발생한 경우로, 이보다 빠른 경우는 교통량이 매우 적은 경우로 분류하였다. 만약 차량의 운행 속도가 0 ~ 35 km/hour의 범위 내에 존재한다면, 속도 등급은 VB (Very Busy)이다. 도로 구간에서의 속도 등급은 [표 1]과 같이 분류한다.

표 1. 속도 등급(Velocity Class)

속도범위(km/hour)	속도등급
0~35	VB
36~60	B
>60	F

출발 시간 그룹은 대도시 지역에서의 교통량 데이터를 기준으로 분류하였다. 트래픽 데이터 튜플(tuple)에

서 g_1, g_2, \dots, g_n (where $1 \leq i \leq n$)를 출발 시간 그룹이라 하면, 하나의 튜플은 임의의 출발 시간 그룹으로 나타낼 수 있다. 예를 들어, 차량이 임의의 도로 구간에서 오전 10:00~11:00 사이에 출발하였다면, 출발 시간 그룹은 3이 된다. [표 2]는 9개의 출발 시간 그룹을 나타낸다.

표 2. 출발 시간 그룹(Velocity Class)

출발 시간 범위	출발 시간 그룹	상세 설명
07:01 ~ 09:00	1	오전 러시아워
09:01 ~ 10:00	2	오전
10:01 ~ 12:00	3	이른 점심
12:01 ~ 14:00	4	점심
14:01 ~ 15:00	5	오후
15:01 ~ 17:00	6	저녁
17:01 ~ 20:00	7	저녁 러시아워
20:01 ~ 00:00	8	심야
00:01 ~ 07:00	9	새벽

한편 차량의 운행 양식은 요일에 따라 달라질 수 있다. 특히, 주 5일제를 기반으로 통학 및 출퇴근 등 교통량이 발생하는 요인이 크게 다르다. 즉, 통학 및 출퇴근 등이 빈번한 주일(월요일~금요일)의 차량의 운행 속도 및 운행 구간이 통학 및 출퇴근이 적은 주말(토요일~일요일)과 다를 수 있다. 이를 위해 일주일을 두 가지 그룹으로 나누며, 이는 [표 3]과 같다.

표 3. 주별 요일 그룹

주별 요일 범위	주별 요일 그룹	상세 설명
월요일~금요일	W	주일
토요일~일요일	H	주말

2. 규칙-기반 분류화 기법을 이용한 속도 등급 측정

도로 구간별 속도 등급을 측정하기 위해, 규칙-기반 분류화 기법을 이용한 속도 등급 측정 알고리즘을 제안한다. 제안된 알고리즘은 도로 구간, 하루의 시간대, 요일 등의 속성을 조합하여, 도로의 속도를 분류하기 위한 조건을 생성한다. 이후 분류된 도로 조건별 차량 운행 속도의 평균을 계산하여, 특정 도로 조건 하에서의 속도 등급을 측정한다. 특정 도로 조건과 속도 등급의

조합을 결정 규칙(decision rule)이라 하며, 이는 다음과 같다.

IF (R(E) = 도로 구간 ID)
 AND (R(T_G) = 출발 시간 그룹)
 AND (R(D_G) = 출발 요일 그룹)
 THEN 속도 등급 = (VB, B, F)

IF-THEN의 형태로 표현된 규칙에서 R (E)는 출발 지점에서 도착지점 사이의 경로를 구성하는 임의의 도로 구간을 의미한다. 또한 R (T_G) 및 R (D_G) 는 각각 해당 구간을 지날 때의 출발 시간 그룹과 주별 요일 그룹을 의미한다. 규칙의 IF-부분은 전제 조건 (antecedent)이라 불리며, THEN-부분은 논리 결과 (consequent)라 불린다. 전제 조건은 R(E), R(T_G), R(D_G)의 세 가지 중 하나 이상의 속성을 조합하여 이루어지며, 논리 결과는 VB, B, 혹은 F 중 하나의 속도 등급 값을 지니게 된다. 한편, 전제 조건에 해당하는 레코드 중 논리 결과와 같은 속도 등급을 지닌 튜플의 수를 규칙의 정확도라 하며, 이는 임의의 규칙에 대한 평가 기준이 된다.

3. 규칙 생성 알고리즘

```

Rule Generation (T, A, C)
Input: Let T be the total training records,
      A be the set of attribute value pairs
      and C be the ordered set of velocity class
      {c1, c2, ..., ck}.
Output: A set of IF-THEN rules {
1) Let R= { } be the initial rule list;
2) for each velocity class c ∈ C do
3) while stopping condition is not met do
4)   r = Learn_One_Rule ( T, A, c);
5) Remove training records from T that are covered by r.
6)   R= R+ r;
7) end while
8) end for
9) return R; }
    
```

그림 1. 규칙 생성 알고리즘

[그림 1]은 규칙 생성 알고리즘을 나타낸다. 먼저 결정 규칙 리스트 R을 생성하고, 이를 초기화한다. 다음 Learn_One_Rule을 통해 전체 훈련 데이터 중 주어진 전제 조건에 해당하는 데이터들을 추출한다. 추출된

데이터들은 훈련 데이터 집합에서 삭제되며, 이를 토대로 평균 속도를 구해 속도 등급을 측정한다. 위의 과정을 반복하여 규칙들을 생성한 후 규칙 정확도가 높은 순으로 규칙들을 정렬하여 규칙 목록을 만든다.

표 4. 샘플 트래픽 데이터

차량 ID	도로구 간 ID	출발 시간 그룹	출발 시간	주별 요일 그룹	속도 (km/hour)	속도 등급
1	1	3	10:02	W	26	VB
2	1	3	10:05	W	27	VB
3	1	3	10:05	W	30	VB
4	1	4	13:00	H	86	F
5	1	3	10:01	W	27	VB
6	1	3	10:01	H	67	F
7	1	4	13:05	H	75	F
8	1	4	13:06	H	55	B
9	1	9	06:01	H	50	B
10	1	3	10:30	W	46	B

[표 4]는 샘플 트래픽 데이터이다. 만약 규칙 R1이, R1: IF 도로 구간 ID =1 AND 출발 시간 그룹 =3 AND 주별 요일 그룹 = W THEN 속도 등급 = VB 이라고 하면, 이는 10개의 전체 데이터 중 5개를 만족하므로, 50%의 적용 범위(coverage)를 지닌다. 또한 전체 조건을 만족하는 5개 중 4개가 논리 결과인 속도 등급 = VB 를 만족하므로, 규칙 정확성은 80%이다. (도로 구간 ID, 출발 시간 그룹, 주별 요일 그룹)의 세 가지 속성을 이용하여 가능한 조합을 생성하고 이를 전체 조건이라 한다. 각 전체 조건에 해당하는 데이터 들을 [표 4]에서 찾아 각 조건 별로 가장 규칙 정확도가 높은 규칙을 결정 규칙이라 하며, [그림 2]는 이러한 결정 규칙들의 집합을 나타낸다.

```

R1: IF Road_ID =1 AND Time_Group =3 AND
    Weekday_Group =W THEN Velocity_Class= VB
R2: IF Time_Group =3 THEN Velocity_Class= VB
R3: IF Time_Group =4 THEN Velocity_Class= F
R4: IF Weekday_Group =H THEN Velocity_Class= F
R5: IF Road_ID =1 AND Time_Group =4
    THEN Velocity_Class= F
R6: IF Road_ID =1 AND Weekday_Group = W
    THEN Velocity_Class= VB
    
```

그림 2. 결정 규칙 집합

한편, 전체 조건 생성 시, 전체 조건에 포함된 속성에 의해 결정 규칙 내에 혼란이 발생할 수 있다. 예를 들어 주어진 질의가 도로 구간 ID =1 AND 출발 시간 그룹 =3 AND 주별 요일 그룹 = W를 만족할 경우, 가장 정확한 전체조건은 R1이지만 R2 혹은 R6 또한 부분적으로 조건을 만족하므로 결정 규칙으로 선택이 가능하다. 이러한 혼란을 막기 위해 전체 조건에 따른 규칙 우선 순위(priority)를 정의한다. 이를 통해 결정 규칙 목록을 정렬하며, 우선순위가 높은 규칙이 낮은 규칙에 비해 먼저 선택된다. [표 5]는 규칙 우선순위를 나타낸다.

표 5. 규칙 우선순위

규칙 우선 순위	규칙
1	IF (도로 구간 ID) = " AND (출발 시간 그룹) = " AND (주별 요일 그룹) = " THEN 속도 등급= {VB,B,F}
2	IF (도로 구간 ID) = " AND (출발 시간 그룹) = " THEN 속도 등급= {VB,B,F}
3	IF (도로 구간 ID) = " AND (주별 요일 그룹) = " THEN 속도 등급= {VB,B,F}
4	IF (출발 시간 그룹) = " AND (주별 요일 그룹) = " THEN 속도 등급= {VB,B,F}
5	IF (도로 구간 ID) = " THEN 속도 등급= {VB,B,F}
6	IF (출발 시간 그룹) = " THEN 속도 등급= {VB,B,F}
7	IF (주별 요일 그룹) = " THEN 속도 등급= {VB,B,F}

4. 주행 시간 예측 알고리즘

앞서 설명한 규칙 생성 알고리즘을 바탕으로, 주행 시간을 예측하는 알고리즘을 제안한다. 제안하는 알고리즘의 수행 순서는 다음과 같다. 먼저 사용자로부터 질의 경로, 출발 시간, 출발 요일을 입력받은 후, 출발 시간과 출발 요일에 해당하는 출발 시간 그룹과 주별 요일 그룹을 결정한다. 이를 통해 전체 조건을 생성한다(1번째 줄). 다음 Rule_ Extraction 프로시저를 통해 전체 조건에 적합한 결정 규칙을 선택한다(7번째 줄). 결정 규칙을 이용하여 도로 구간에서의 주행 시간을 계산하고, 이를 통해 해당 구간의 끝에 도착할 시간을 얻는다(8번째 줄). 구간 도착 시간은 다음 구간에서의 출발 시간으로 사용된다(9번째 줄). 앞의 과정을 반복하여 경로 내 전체 구간에서의 주행 시간을 측정한다(3~11 번째 줄).

Rule-Based Travel Time Prediction (Start_Time, Weekday, Road_Segment_List)
 Input: Start_Time gives start time for initial Road segment. Weekday provides day information of a week. Road_Segment_List contains all road segments from origin to destination.
 Output: Display predicted travel time for each road segment from origin to destination {
 1) Start_Time_Group=Convert Start_Time to Group;
 2) Weekday_Group= Convert Weekday to Group;
 3) for (i=0;Road_Segment_List!=NULL; i++) do
 4) X.road_segment_id =Road_Segment_List.ID;
 5) X.group = Start_Time_Group;
 6) X.weekday_group = Weekday_Group;
 7) Velocity_Class = Rule_Extraction (X);
 8) End_Time = End Time Measure(X, Velocity_Class);
 9) Start_Time_Group =Convert End_Time to Group;
 10) Answer = End_Time;
 11) end for }

그림 3. 주행 시간 예측 알고리즘

예를 들어 임의의 차량이 도로 구간 1을 월요일 오전 10시 05분에 지나가려 한다. [그림 2]에 생성된 결정 규칙에 따르면, 해당 전제 조건 하에서의 속도 등급은 VB이다. 이때 차량의 속도는 0~35 km/hour 사이이다. 따라서 월요일 10:01~11:00 시에 도로 구간 1의 평균 속도는 28 km/hour이며, 도로 구간 1의 끝 지점에 도착하는 시간은 10시 26분이 된다. 그러나 월요일 13시 05분에 도로 구간 1을 지나는 차량의 경우, 모든 속성을 만족하는 결정 규칙이 존재하지 않는다. 이 경우 [표 5]의 규칙 우선 순위에 따라 가장 적합한 결정 규칙을 탐색한다. 여러 규칙들 중 가장 적합한 것은 (IF (도로 구간 ID= 1) AND (출발 시간 그룹 = 4)) 이며, 이때의 속도 등급은 F이며, 이를 이용하여 주행 시간을 예측할 수 있다.

IV. 성능 평가

1. 예측 정확도 성능평가

제안된 주행 시간 예측 기법의 예측 정확도를 측정하기 위해, mean absolute relative error (MARE) [1]를 사용한다. MARE는 가장 간단하고, 또한 잘 알려진 여러 측정 방법이다. MARE는 원하는 시간 범위 내에서 연관된 에러의 등급을 측정하며, 식 (1)와 같이 정의된다.

$$MARE = \frac{1}{N} \sum_t \frac{x(t) - x^*(t)}{x(t)} \quad (1)$$

$x(t)$ 는 관측값, $x^*(t)$ 는 예상값, 그리고 N 은 샘플의 수를 나타낸다. 실험에 사용된 데이터는 real 데이터 집합이다. real 데이터 집합은 부산대학교의 궤적 데이터 생성기를 사용하였으며, 인터페이스는 [그림 4]와 같다. 부산대학교에서는 GPS 센서가 부착된 차량을 운행하여, 부산시의 도로 구간별 차량 주행 속도 및 시간 등 실제 교통 상황 정보를 수집하였다. 수집된 데이터는 시간대별, 요일별로 분류되었다. 분류된 데이터를 기반으로 실제 차량 운행 정보와 매우 유사한 이동 객체 궤적을 생성한다[12]. 본 논문에서는 해당 생성기를 사용하여 훈련 데이터로 쓰일 365일간의 트래픽 데이터 및 테스트 데이터로 쓰일 30일간의 트래픽 데이터를 생성하였다. 훈련 데이터란, 분류화 기법에서 사용되는 속도 등급을 구하기 위한 기록 데이터이며, 테스트 데이터는 성능평가에서 사용되는 질의 데이터를 의미한다.

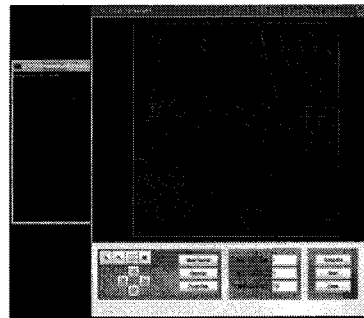


그림 4. 부산대학교 궤적 데이터 생성기의 사용자 인터페이스

또한 질의시 차량의 출발 시간은 현재 시간 $x(t)$ 로부터 30분, 60분의 두 가지 시차(1)를 적용하였으며, 이는 각각 Kwon[3]과 Rice[9]에서 사용된 것이다. 관련 분야의 연구에서는 시차의 감소에 따라 예측 정확도가 향상되는 것을 고려하여, 사용된 시차보다 작은 시차값에 대해서는 높은 예측 정확도를 나타냄을 보장하였다. 이에 따라 본 논문에서는 기존 관련 연구에서 쓰인 최대 시차값을 이용한다. 만약 $x(t) =$ 오전 8시이고 시차 1 = 30분 일 경우, 질의 경로에서 차량의 출발 시간은 각각

8:30이다. 아울러 출발 시간 그룹은 [표 2]와 같다.

한편, 예측 정확도 성능평가 대상으로는 제한된 링크-기반 기법, 링크-기반 기법, 경로-기반 기법에서 각각 대표적인 기법들을 선정하였다. 즉, Chen et al의 링크-기반(link-based) 알고리즘[1]과, Kwon et al의 Micro T*[3] 알고리즘, 그리고 Erick et al의 스위칭 알고리즘[10]을 사용하였다. 스위칭 알고리즘의 경우, 경로-기반 기법임에도 불구하고 질의 시간을 고려할 수 있으며, 높은 예측 정확도를 보이기 때문에 예측 정확도 성능평가 대상으로 선정하였다. 한편, 김주현[13]의 연구는 차량 번호판 인식 시스템(Automatic Vehicle Identification: AVI)을 기반으로 하는 연구로써, 두 가지의 문제점 때문에 성능평가를 수행하지 못하였다. 첫째, AVI의 설치가 제한적이다. 즉, AVI의 주요 설치 장소는 주요 간선 도로 및 고속도로 IC(Inter change)이며, 이러한 AVI의 제한때문에 예측의 최소 단위가 도시-도시 사이의 장거리에 적합한 기법이다. 따라서 도시 내의 모든 도로 구간에 대한 주행 시간 예측을 수행하고자 하는 본 연구와는 비교대상으로 적합하지 않다. 둘째, 김주현의 연구를 위해 수집되는 데이터의 포맷이 매우 상이하다. 즉, 김주현의 연구에서는 AVI를 활용하여 특정 간선도로 구간에서 데이터를 수집하였으며, 수집된 데이터에서 발생하는 오류를 수정하는 방법이 포함된다. 따라서 단순히 차량의 과거 궤적 데이터만으로는 실험이 불가능하므로, 예측 정확도 성능평가에서 제외하였다. 또한 제한적 링크-기반 기법 중 이청원[14]의 연구는 칼만 필터 기반으로 하는 연구로써, 다음과 같은 제한점을 지니고 있다. 첫째, 칼만 필터는 교통 패턴이 다양하고 변동이 심한 도로 네트워크에 적합한 기법으로써, 이와 같은 도로 특성을 지닌 구간을 선택하여 주행 시간을 예측하였다. 즉, 이를 적용하기 위해서는 먼저 전체 도로 구간에 대해 도로 특성을 분석하여 같은 특성을 지니는 구간을 선택할 필요가 있다. 이는 도시 내의 전체 도로 구간에 대한 주행 시간 예측을 수행하고자 하는 본 연구의 목적에 적합하지 않다. 둘째, 칼만 필터는 30분 이내의 단기간에 대한 주행 시간 예측에 적합한 기법이며, 이를 적용하기 위해서는 VMS와 같은 검지기를 사용하여 5~10분의 짧은 간격으로 현

재 교통 상황에 대한 지속적인 정보 업데이트가 필요하다. 즉, 이와 같은 도로 특성 분석의 필요성과 현재 교통 상황에 대한 빈번한 업데이트의 제약 때문에 성능평가를 수행하지 못하였다. 마지막으로 링크-기반 기법 중 이영우[15]의 연구는 단순한 차량의 궤적 데이터가 아니라, 실제 교통 신호기가 설치된 교차로를 중심으로 교통량 및 차량의 주행 시간을 측정하여야 한다. 즉, 데이터를 수집하기 위해서는 적용하고자 하는 지도 내에 존재하는 신호교차로의 기하구조 및 신호 주기에 대한 정보가 요구된다. 아울러 출발 지역의 녹색 신호 지속 시간 비율과 도착 지역의 녹색 신호 시간 비율 등 상세한 신호 조건 또한 필요하다. 이러한 신호교차로에 대한 정보를 바탕으로 개별 차량에 대한 통과시간을 측정해야 한다. 이는 본 연구에서 사용하는 궤적 데이터 생성기를 통해서는 데이터 생성이 불가능하므로, 예측 정확도 성능평가를 수행하지 못하였다.

[그림 5]와 [그림 6]은 오전 8시~ 오후 6시 사이의 시간 범위에서, 각각 시차를 30분과 60분으로 하였을 때 네 가지 예측 알고리즘들에 대한 성능 평가 결과를 나타낸다. 제안하는 알고리즘은 RBC로 표기되었다. 제안하는 알고리즘이 링크-기반 알고리즘과 Micro T* 알고리즘, 그리고 스위칭 알고리즘에 비해 전체적으로 우수한 성능을 보인다. 제안하는 알고리즘의 경우, 평균 구간 속도가 감소하면 주행 시간 예측에 대한 정확도가 향상된다. 즉, 제안하는 알고리즘은 평균 구간 속도가 가장 낮은, 러시아워의 시간대에서 가장 우수한 성능 결과를 나타낸다.

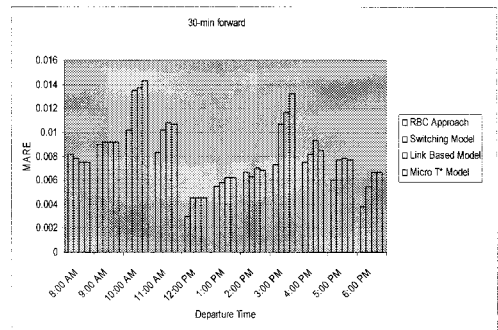


그림 5. MARE (시차 =30분, Real 데이터)

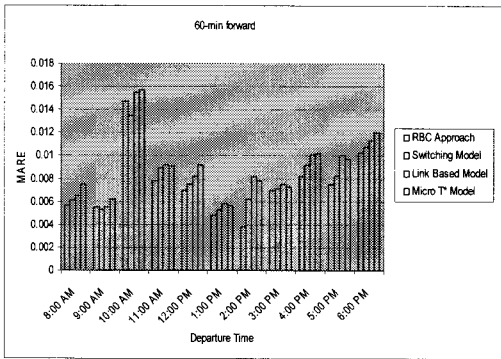


그림 6. MARE (시차 = 60분, Real 데이터)

[그림 7]은 네 가지 예측 알고리즘의 평균 MARE를 나타낸다. 시차가 30분 일 때, 제안하는 알고리즘, 스위칭 알고리즘, 링크-기반 알고리즘, 그리고 Micro T* 알고리즘의 MARE는 각각 0.00717, 0.00828, 0.00868, 그리고 0.00871이다. 따라서 제안하는 알고리즘은 다른 알고리즘들에 비해 17~18% 정도 성능이 향상되었다. 시차가 60분 일 경우, 링크-기반 알고리즘, 그리고 Micro T* 알고리즘에 비해 5%, 14%, 17% 향상된 성능을 나타낸다. 따라서 제안하는 알고리즘이 다른 두 주행 시간 예측 알고리즘에 비해 성능이 향상되었다.

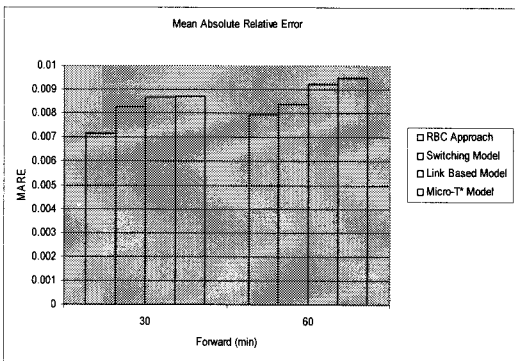


그림 7. 평균 MARE (8AM~6PM)

2. 주행 시간 예측 기법의 장단점 비교

제안된 주행 시간 예측 기법의 장단점을 기존의 다른 기법들과 비교 분석한다. 비교 대상은 Chen et al의 링크-기반 (link-based) 알고리즘[1], Kwon et al의

Micro T*[3] 알고리즘, Erick의 스위칭 알고리즘[10], 김주현[13]의 AVI 기반 알고리즘, 이청원[14]의 칼만 필터링 알고리즘, 그리고 이영우[15]의 퍼지추론 기반 알고리즘이다. [표 6]은 각 알고리즘의 장점 및 단점을 나타낸다. 제안된 주행 시간 예측 기법은 Chen et al의 연구와 비교할 때, 도시 내의 주요 도로 구간에 대해서만 주행 시간 예측뿐만 아니라, 사용자가 지정한 임의의 경로에 대하여 주행 시간 예측이 가능한 장점이 존재한다. 또한 Kwon의 연구와 비교할 때, 도로 구간의 일별 및 주별 교통 상황을 고려하여, 특히 차량의 교통량이 많은 러시아워 시간대를 중심으로 주행 시간 예측의 정확도가 향상되었다. 한편, Erick et al의 연구는 주행 시간 예측 방법의 선택 기준을 계산하기 위해 많은 차량 궤적 데이터가 필요한 반면, 제안한 주행 시간 예측 기법은 상대적으로 적은 데이터가 필요한 장점이 존재한다. 아울러 김주현의 연구와 비교할 때, 도시 내의 모든 도로 구간에 대해 주행 시간을 예측을 수행할 수 있는 장점이 존재한다. 김주현의 연구에서는 설치가 제한적인 AVI를 기반으로 하고 있으나, 본 연구에서 제안한 기법은 과거에 측정된 차량의 궤적 데이터를 이용하므로 모든 도로 구간에 대한 주행 시간 예측이 가능하다. 한편, 이청원의 연구에 비해 다음과 같은 장점을 지니고 있다. 첫째, 긴 기간에 대한 주행 시간 예측이 가능하며, 둘째, 도로망의 특성에 의한 영향을 적게 받는다. 마지막으로 지속적인 교통 정보 업데이트를 요구하지 않는다. 아울러, 이영우의 연구에 비해서는 다음과 같은 장점을 지니고 있다. 첫째, 규칙-기반 분류화 기법을 사용하여 퍼지 기법에 비해 알고리즘이 단순하고, 적은 계산량이 요구된다. 또한 교통 신호등이 설치되지 않은 도로 구간에서도 주행 시간 예측이 가능하다. 특히 제안한 기법은 신호 주기, 신호 시간 비율 등의 신호기에 대한 상세 정보를 요구하지 않기 때문에 활용이 더욱 편리하다.

제안된 주행 시간 예측 기법의 기대효과 및 의의는 다음과 같다. 첫째, 복잡한 도로 네트워크를 지나는 도시 내에서의 주행 시간 예측에 적합하다. 제안한 기법은 과거 차량의 운행 궤적을 통해 구간별 속도 등급을 측정함으로써, 모든 도로 구간에 대해 주행 시간을 예

측할 수 있다. 따라서 사용자가 지정한 임의의 경로에 대해, 해당 경로를 구성하는 모든 도로 구간에서의 주행 시간을 예측하여 이를 합산함으로써 주행 시간 예측이 가능하기 때문이다. 둘째, 일별, 주별 교통 상황 정보를 고려함으로써 높은 예측 정확도를 나타낸다. 이는 Chen et al의 링크-기반(link-based) 기법[1], Kwon et al의 Micro T*[3]기법, 그리고 Erick et al의 스위칭 기법[10]과의 예측 정확도 성능평가를 통해 예측 정확도가 향상되었음을 나타내었다. 셋째, 다른 주행 시간 예측 기법에 비해 적용이 편리하다. 즉, AVI를 기반으로 차량의 통과 시간을 측정하는 김주현의 연구와 상세한 교통 신호기 정보 등을 기반으로 하는 이영우의 연구 및 도로 구간의 특성을 분석하여 제한적으로만 적용가능한 이청원의 연구에 비해서, 제안한 기법은 과거 차량 궤적 데이터만을 사용하기 때문에 모든 도로 구간에 대하여 적용하기 쉽다. 따라서 이러한 점을 고려할 때 제안한 주행 시간 예측 기법은 국내의 지능형 교통 정보 시스템에 적용하여 활용하기에 적합하다고 할 수 있다.

표 6. 주행 시간 예측 기법의 장단점 비교

기법	장점	단점
Chen et al [1]	-주요 도로 구간에 대한 주행 시간 예측 지원 -대부분의 경로-기반 기법에 비해 주행 시간 예측 정확도 향상	-사용자들이 선호하는 주요 경로 등 도시내 도로 네트워크의 일부에 대해서만 주행 시간 예측 지원 가능
Kwon et al [3]	-사용자가 지정한 임의의 경로에 대하여 주행 시간 예측 가능	-주행 시간 예측의 정확도가 다른 기법에 비해 저하됨 -도로 구간의 교통 상황에 대하여 고려하지 못함
Erick et al [10]	-사용자의 질의 시간 및 도로 교통 상황을 고려하여, 선형 예측 방법과 평균 주행 시간 기반 예측 방법 중 선택하여 주행 시간 예측 -높은 주행 시간 예측 정확도	-경로-기반 기법이므로, 사용자가 지정한 임의의 경로에 대한 주행 시간 예측 불가능 -정확한 방법 적용의 기준점을 찾기 위해 많은 주행 시간 데이터 요구
김주현[13]	- 유사 주행 시간 패턴을 통한 장거리 주행 시 주행 시간 예측 지원	- AVI의 설치 제한 때문에, 짧은 도로 구간에 대한 주행 시간 예측이 불가능 -충분한 패턴의 축적을 위해 많은 주행 시간 데이터 요구
이청원[14]	-주기적인 교통 정보 업데이트를 통한 예측 오차 최소화 -칼만 필터링을 통한 단기 주행 시간 예측 지원	-도로망의 특성에 따라 적용이 제한적 -5~10분 이내의 단기간의 주행 시간 예측만을 지원 -빈번한 교통 상황 정보 업데이트 요구

이영우[15]	- 교통 신호에 따른 차량의 동행 시간 패턴 분석을 통해 주행 시간 예측 정확도 향상	- 알고리즘이 복잡하고, 많은 계산량이 요구됨 -교통 신호등이 설치된 교차로 중심의 주행 시간 예측만을 지원
제안한 기법	-시간대 및 요일별 주행 시간 등급을 고려 -사용자가 지정한 임의의 경로에 대하여 주행 시간 예측 가능	-각 구간별 평균 주행 시간 및 속도 등급을 측정하기 위해 많은 훈련 데이터 요구

V. 결론 및 향후 연구

본 논문에서는 도로 네트워크 상의 사용자 지정 경로를 위한 효율적이고 확장 가능한 주행 시간 예측 기법을 제안하였다. 제안하는 방법은 사용자 지정 경로에 대해 주행 시간을 예측할 수 있을 뿐만 아니라, 규칙-기반 분류화 기법을 사용하여 도로의 일별 및 주별 교통 상황을 고려하여 예측 정확도를 향상시켰다. 제안한 기법을 real 데이터 집합을 사용하여, 기존에 제안된 주행 시간 예측 알고리즘인 스위칭 알고리즘, 링크-기반(link-based) 알고리즘 및 Micro T* 알고리즘과 성능평가를 수행하였다. 성능 평가 결과, 제안하는 알고리즘이 5~17% 정도 MARE측면에서 성능이 우수함을 보였다. 향후 연구로는, 데이터 분류시 명절 혹은 기념일 등의 공휴일을 고려하여 주행 시간을 예측하는 것이다.

참고 문헌

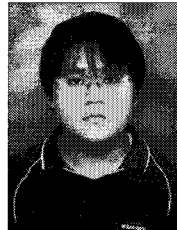
- [1] M. Chen and S. Chien. "Dynamic freeway travel time prediction using probe vehicle data: Link-based vs. Path-based," Journal of Transportation Research Record, TRB Paper No.01-2887, 2001.
- [2] W. Chun-Hsin, W. Chia-Chen, S. Da-Chun, C. Ming-Hua, and H. Jan-Ming, "Travel Time Prediction with Support Vector Regression," In IEEE Intelligent Transportation Systems Conference, 2003.
- [3] J. Kwon and K. Petty, "A travel time prediction

- algorithm scalable to freeway networks with many nodes with arbitrary travel routes," Transportation Research Board 84th Annual Meeting, 2005.
- [4] D. Park and L. Rilett, "Forecasting multiple-period freeway link travel times using modular neural networks," Journal of Transportation Research Record, Vol.1617, pp.163-170, 1998.
- [5] D. Park. L. Rilett, "Spectral basis neural networks for real-time travel time forecasting," Journal of Transport Engineering, Vol.125, No.6, pp.515-523, 1999.
- [6] J. Kwon. B. Coifman. and P. J. Bickel. "Day-to-day travel time trends and travel time prediction from loop detector data," Journal of Transportation Research Record, No.1717, TRB, National Research Council, pp.120-129, 2000.
- [7] X. Zhang and J. Rice, "Short-Term Travel Time Prediction," Transportation Research Part C, Vol.11, pp.187-210, 2003.
- [8] M. Van der Voort, M. Dougherty, and S, Watson, "Combining KOHONEN maps with ARIMA time series models to forecast traffic flow," Transportation Research Part C, Vol.4, pp.307-318, 1996.
- [9] J. Rice and E. Van Zwet, "A simple and effective method for predicting travel times on freeways," In IEEE Trans. Intelligent Transport Systems, Vol.5, No.3, pp.200-207, 2004.
- [10] J. Schmitt Erick and H. Jula, "On the Limitations of Linear Models in Predicting Travel Times," In IEEE Intelligent Transportation Systems Conference, 2007.
- [11] J. Han and M. Kamber, *Data Mining: Concepts and techniques*. Second Edition, Morgan Kaufmann publishers, 2006.
- [12] 황정래, 강혜영, 이기준, "시공간 유사성을 이용한 도로 네트워크 상의 유사한 궤적 검색", 한국정보처리학회 논문지, 제13-D권, 제3호, pp.0337-0346, 2006.
- [13] 김주현, "AVI를 활용한 실용적 여행 시간 예측 수법에 관한 연구", 수도권연구, 제2호, pp.197-211, 2005.
- [14] 이청원, 박지영, 고승영, "칼만 필터를 이용한 경로통행시간예측", 대한토목학회논문집, 제22권, 제5-D호, pp.871-880, 2002.
- [15] 이영우, "퍼지추론을 이용한 링크통행시간 분포 비율 추정모형 구축", 대한토목학회논문집, 제 26 권, 2D호, pp.233-239, 2006.

저자 소개

이 현 조(Hyun-Jo Lee)

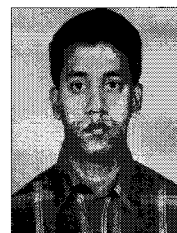
정희원



- 2006년 : 전북대학교 컴퓨터공학과(공학사)
- 2008년 : 전북대학교 대학원 컴퓨터공학과(공학석사)
- 2008년 ~ 현재 : 전북대학교 대학원 컴퓨터공학과 박사과정

<관심분야> : 데이터 마이닝, 공간 데이터베이스, 고차원 색인 구조

니하드 카림 초우더리(Nihad Karim Chowdhury) 정희원

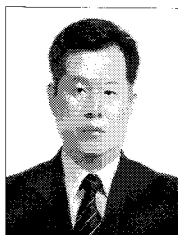


- 2004년 : Chittagong University of Engineering & Technology (공학사)
- 2006년 ~ 현재 : 전북대학교 대학원 컴퓨터공학과 석사과정

<관심분야> : 데이터 마이닝, 공간 데이터베이스, 질의 처리 알고리즘

장 재 우(Jae-Woo Chang)

정회원



- 1984년 : 서울대학교 전자계산
기공학과(공학사)
- 1986년 : 한국과학기술원 전산
학과(공학석사)
- 1991년 : 한국과학기술원 전산
학과(공학박사)

- 1996년 ~ 1997년 : Univ. of Minnesota, Visiting
Scholar
 - 2003년 ~ 2004년 : Penn State Univ., Visiting
Scholar.
 - 1991년 ~ 현재 : 전북대학교 컴퓨터공학과 교수
- <관심분야> : 공간 네트워크 데이터베이스, 상황인식,
하부저장구조