
실시간 교통 정보를 이용한 교통 혼잡 예측 시스템

유영중* · 조미경**

The System for Predicting the Traffic Flow with the Real-time Traffic Information

Young-Jung Yu* · Mi-Gyung Cho**

요 약

텔레매틱스 서비스 중에서 가장 많이 이용하고 있는 것 중 하나는 출발지와 목적지에 이르는 최단 경로를 찾아 주는 서비스이다. 현재 보편적으로 사용되고 있는 최단 경로 찾기 서비스는 실시간 교통 정보를 고려하지 않는 정적인 최단 경로 알고리즘을 사용하고 있다. 본 연구에서는 실시간 교통 정보를 반영하여 현재 시간으로부터 일정 시간 경과한 후의 교통 정보를 예측하기 위한 방법을 제안하고 예측 결과의 정확성을 평가하기 위해 실시간 데이터를 이용하여 실험하였다. 제안된 방법은 도로 위를 달리는 차량의 평균 속도를 5분 단위로 입력 받아 누적된 데이터를 동일한 시간과 요일별로 분석하여 구한 누적 속도패턴과 칼만 필터 방법을 통합한 것이다. 제안한 방법은 현재 사용되고 있는 누적 속도 패턴만을 이용한 예측보다 더 정확한 예측 결과를 보여 주었다. 예측된 결과는 동적인 최단 경로를 구하기 위해 사용될 뿐만 아니라 교통 혼잡이 예측되는 지역을 피하여 여행하도록 정보를 제공할 수 있다

ABSTRACT

One of the common services of telematics is the car navigation that finds the shortest path from source to target. Until now, some routing algorithms of the car navigation do not consider the real-time traffic information and use the static shortest path algorithm. In this paper, we proposed the method to predict the traffic flow in the future. This prediction combines two methods. The former is an accumulated speed pattern, which means the analysis results for all past speeds of each road by classifying the same day and the same time interval. The latter is the Kalman filter. We predicted the traffic flows of each segment by combining the two methods. By experiment, we showed our algorithm gave better precise prediction than only using accumulated speed pattern that is used commonly. The result can be applied to the car navigation to support a dynamic shortest path. In addition, it can give users the travel information to avoid the traffic congestion areas.

키워드

텔레매틱스, 속도 예측, 교통 혼잡, 실시간 교통 정보

I. 서 론

달리는 차량 내에서 사무실에서와 같은 정보통신 서비스를 받을 수 있는 것을 목표로 하는 텔레매틱스는 차량

내에서 지원 받을 수 있는 모든 정보통신 서비스를 지칭하는 용어로 지리정보시스템, GPS 계측, 무선 인터넷 등 다양한분야가 통합된 학문이다[4]. 가장 많이 사용되고 있는 서비스는 출발지-목적지 최단 경로 찾기 서비스와

* 부산외국어대학교 컴퓨터공학부
** 동명대학교 멀티미디어공학과

현재 위치에서 특정 장소를 찾는 문제 - 예를 들면, "현 위치에서 가장 가까운 주유소를 찾아라" - 등이다. 현재까지 사용되고 있는 최단 경로 서비스는 도로의 거리와 허용된 차량 속도 등 정적인 데이터를 이용하여 출발지에서 목적지에 이르는 최단 경로를 찾아주는 알고리즘을 이용한다. 이 경우 실시간 교통 정보를 반영하지 않기 때문에 출발지로부터 출발하여 일정한 시간 경과 후의 차량이 경유하는 지역에서 교통 혼잡이 생길 경우 선택된 경로는 최단 경로가 될 수 없다. 만약 특정 도로의 교통 혼잡을 미리 예측할 수 있다면 그 도로를 피하여 다른 길을 우회하여 가도록 경로 배정을 하므로 질적으로 향상된 최단 경로 서비스를 제공할 수 있을 것이다.

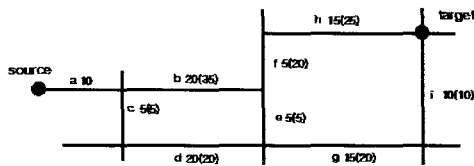


그림 1. 교통 혼잡을 고려할 때와 고려하지 않을 때의 세그먼트 통과 시간
 Fig. 1. Traveling time of each segment depending on the existence of traffic congestion

도로 네트워크는 그림 1에서 보는 것처럼 세그먼트 단위로 구성되어 있다고 가정한다. 세그먼트는 교차로와 교차로 사이의 도로를 의미하며 그림에서 a, b, c 등으로 표현하였다. 실시간 교통 정보를 반영하지 않고 단순히 세그먼트의 길이를 가지고 계산한 각 세그먼트를 통과하기 위해 걸리는 시간이 a의 경우 10, b의 경우 20, c의 경우 5이다. 괄호 안의 시간은 차량이 출발지를 출발하여 해당 세그먼트에 도착하였을 때 각 세그먼트를 통과하는데 걸리는 시간을 의미한다. 세그먼트 b의 경우 정상시 평균 통과시간은 20이지만 현재 교통 정체로 인해 35로 증가하였음을 알 수 있다. 따라서 정적인 교통 정보만을 이용하여 출발지로부터 목적지에 이르는 최단 경로를 구할 경우 최단 경로는 a, b, f, h 이지만 동적인 교통 정보를 실시간으로 추가할 경우 a-c-d-e-f-h가 최단 경로가 된다.

실시간 교통 정보를 사용할 경우 사용자에게 더 정확한 교통 정보를 제공할 수 있기 때문에 질적인 텔레매틱스 서비스 향상을 위해 실시간 교통 정보를 반영하기 위한 노력을 많이 하고 있다[4]. 본 논문에서는 실시간 교통 정보를 반영하여 일정 시간이 경과한 이후의 교통 정보

를 예측하기 위한 방법을 연구하였다. 즉, 출발지에서 출발한 차량이 세그먼트 a를 통과하는 시간은 출발지에서 10분이 경과한 시간으로 그때 세그먼트 b나 c에서 교통 정체가 발생했는지 아니면 일상시의 교통 속도를 유지하는지에 대한 교통 정보를 예측하는 것이다.

개발된 시스템은 모든 세그먼트들에 대한 실시간 교통 정보는 세그먼트 위를 달리는 차량들의 평균 속도(km/시속)로 5분 단위로 입력 받도록 하였고 현재 시각을 중심으로 5분 후부터 3시간 후까지의 차량 속도 및 교통 혼잡 유무를 예측하도록 하였다. 예측된 결과는 "50분 후 세그먼트 f위를 달리는 차량들의 평균 속도는 무엇인가?" 혹은 "50분 후 세그먼트 f에서 교통 정체가 예측되는가?"라는 질문에 답할 수 있도록 하였다.

실시간 교통 정보를 반영한 동적인 최단 경로 배정 알고리즘이나 여행 시간 지연에 대한 예측 등에 대한 연구는 1998년 이후로 연구 결과가 발표되었다[1,2,3]. 또한 대도시를 중심으로 실시간 교통 정보를 이용한 각종 ITS(Intelligent Transportation Systems) 서비스가 현재 실시되고 있다. Chon[2]의 연구 결과에 따르면, 도로 혼잡을 고려한 동적인 경로 배정이 정적인 경로 배정에 비해 전체 이동 객체의 평균 이동 시간을 감소하는 결과로 이어진다고 보고하였다. 그가 제안한 방법으로 특정 세그먼트에 대한 교통 혼잡을 측정하기 위해서는 모든 이동 객체가 경로 정보를 서버에 제공해야 하는데 이것은 현실적으로 불가능하다.

Handley[2,3] 등은 고속도로에 설치한 115개의 센서로부터 주기적으로 차량의 평균 속도를 수집하여 출발지에서 목적지까지 차량이 여행하는데 걸리는 시간을 예측하였다. Handley의 연구는 센서를 통해 실시간으로 주어지는 현재 데이터를 반영하여 소요되는 여행 시간을 측정하였다. 즉, 여행 시간을 측정할 때 차량이 통과될 미래 시간에 대한 고려 없이 모든 세그먼트에 대해 현재 시각을 기준으로 차량의 평균 속도를 사용하였다. 본 논문에서는 각 세그먼트들에 대한 교통 혼잡 예측을 위해 모든 이동 객체들의 경로 정보를 가지고 있어야 한다는 전체 조건을 필요로 하지 않는다. 또한 현재 실시간 교통 정보를 반영하여 일정한 시간이 경과된 이후의 미래 시간에 대한 교통 혼잡을 예측하는 것으로 Handley의 연구와도 차별성이 있다.

본 논문의 구성은 다음과 같다. 2장에서는 연구에 사용된 실시간 교통 정보 데이터에 대한 설명과 본 논문에서

개발한 예측 시스템의 대략적인 구조를 설명할 것이다. 3장에서는 누적된 속도 데이터로부터 분석한 속도 패턴과 칼만 필터 법을 적용한 속도 예측 알고리즘에 대해 설명할 것이다. 4장에서 실시간 데이터를 이용하여 제안한 예측 시스템의 성능을 실험한 결과를 살펴보고 5장에서 결론을 맺고자 한다.

II. 예측 시스템 구조와 실시간 데이터

실시간 교통 데이터는 세그먼트위를 달리는 차량들의 평균 속도로 모든 세그먼트에 대해 5분 단위로 웹을 통해 입력 받았다. 실시간 속도 데이터는 ETRI LBS 팀을 통해 전달 받았다. 예측 결과는 현재 시간으로부터 5분 경과한 후부터 3시간까지 모든 세그먼트들에 대해 예측되는 속도와 예측 정확도 및 교통 혼잡 유무로 표시된다. 본 논문에서 개발한 교통 예측 시스템의 대략적인 구조는 그림 2와 같다.

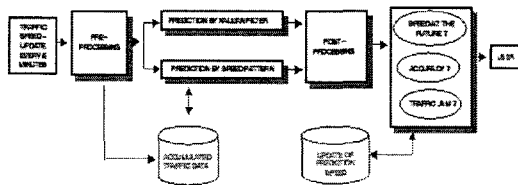


그림 2. 교통 예측 시스템의 구조
Fig. 2. The architecture of the prediction system

전처리 과정에서는 입력된 실시간 데이터들을 검사하여 평균 속도가 누락된 세그먼트들에 대한 처리를 수행한다. 원칙적으로 실시간 속도 데이터는 모든 세그먼트에 대해 주어져야 하지만 실제 실시간 데이터들을 검사했을 때 평균 속도가 누락된 세그먼트들이 발견되었으며 이들의 값은 -1로 표시되어 있었다. 따라서 이와 같은 세그먼트들에 대해서 먼저 평균 속도를 계산해 주는 과정이 필요하다. 이를 결측 구간에 대한 속도 측정이라고 부른다. 외국의 경우 결측 구간에 대한 속도 측정을 하기 위한 상용 소프트웨어가 개발되어 사용되고 있다.

본 논문에서는 도로 네트워크에서 결측 세그먼트 주위의 그래프 위상 구조를 이용하여 결측 구간에 대한 속도를 계산하였다. 도로 네트워크를 그래프 구조로 표현할 때 교차로는 정점으로 표시되고 교차로와 교차로 사이의

세그먼트는 간선으로 표현된다. 각 세그먼트는 방향을 가지기 때문에 도로 네트워크는 방향 그래프로 표현된다. 결측 세그먼트에 대한 속도를 계산하기 위해 먼저 해당되는 링크의 from_node와 to_node를 검색하였다. 해당 링크의 from_node에서 들어오는 링크들(in-going links)의 속도와 to_node에서 나가는 링크들(out-going links)의 속도의 평균을 계산하여 결측 구간에 대한 속도로 사용하였다. 결측 세그먼트들에 대한 전처리가 끝난 실시간 데이터는 누적 교통 정보 데이터베이스에 저장함과 동시에 속도 예측을 위하여 사용된다.

본 논문에서는 속도 예측을 위해 누적 속도 패턴 방법과 칼만 필터 법을 사용하였다. 이 방법들에 대해서는 3장에서 자세히 설명할 것이다. 두 방법으로 예측된 속도는 후처리 과정을 거쳐서 최종 예측 속도와 예측의 정확성 그리고 교통 혼잡 유무를 결정하는데 사용된다. 예측 결과들은 데이터베이스에 저장되고 웹을 통해 예측 정보를 요청한 사용자들에게 전송된다.

그림 3은 본 논문에서 사용한 서울시의 도로 네트워크를 보여 준다. 왼쪽 그림(a)는 실제 데이터를 축소한 도로이며 오른쪽 그림 (b)는 왼쪽의 사각형 내의 도로들을 확대한 것이다. 그림에서 보는 것처럼 서울시는 매우 복잡한 도로 네트워크를 가지고 있으며 정점의 개수는 6,735개이고 세그먼트를 나타내는 간선의 개수는 20,271개이다. 실시간 속도 데이터는 모든 세그먼트에 대해 5분 단위로 들어오며(time_sequence, link_id, speed) 형태로 입력된다. 첫 번째 파라메타인 time_sequence 값이 200410231205 이면 데이터가 입력되는 현재 시간이 2004년10월 23일 12시 05분을 의미한다. 모든 세그먼트는 link_id로 구분되며 speed는 세그먼트 위를 달리는 차량들의 평균 속도로 시속 km로 표현된다.

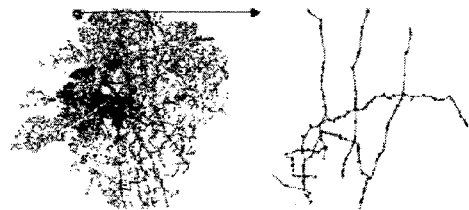


그림 3. 서울시의 도로네트워크의 일부
Fig. 3 The part of the road network of Seoul city

III. 속도 예측 방법

속도 예측을 위해 두 가지 방법을 사용하였다. 첫째는 각 세그먼트들에 대한 누적된 속도 데이터로부터 평상시 차량의 흐름 정보를 분석한 결과인 속도 패턴을 이용한 방법이다. 두 번째는 칼만 필터를 이용한 예측 방법을 사용하였다. 두 가지 방법으로 예측된 결과는 최종 예측 속도를 계산하는데 사용되어진다.

3.1. 누적 속도 패턴 분석 및 예측

누적 데이터베이스는 요일과 시간대로 분석된 누적 속도 패턴에 대한 정보를 유지한다. 이것은 실시간 데이터를 동일한 요일과 시간대로 분류하여 각 세그먼트들에 대하여 속도 평균과 표준편차 값을 구한 것이다. 요일은 일곱 개로, 시간 구간은 30분 단위로 분리하여 하루를 모두 48개의 구간으로 나누어 분석하였다. 따라서 각 세그먼트들에 대한 누적 데이터는 336개의 속도 평균과 표준편차로 구성된다. 이러한 데이터는 각 세그먼트들에 대한 일상적인 교통 흐름을 파악할 수 있는 자료가 된다. 평균 속도와 표준 편차를 분석한 결과 각 세그먼트들에 대해 동일한 시간대와 동일한 요일들에 대해서는 일정한 속도 패턴을 유지하고 있음을 알 수 있었다.

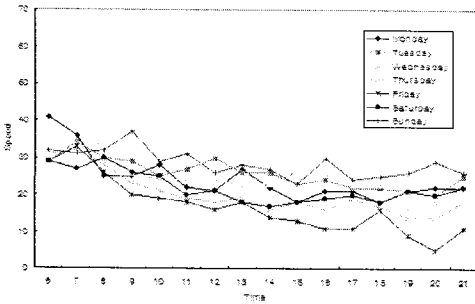


그림 4. 도심에 위치한 한 세그먼트의 요일과 시간대 별 평균 속도 패턴

Fig. 4 An accumulated speed pattern of a segment located in downtown

그림 4는 도심에 위치한 특정 세그먼트에 대한 요일과 시간대로 분석한 누적 속도 패턴을 보여 준다. 그래프에서 보여주는 특징들을 살펴보면 금요일 저녁 6시부터 9시에 해당되는 구간과 토요일의 경우 오후 1시부터 밤 8시 구간 사이에 교통 정체 현상이 심하게 나타나고 있는데

이는 사람들이 도심으로 모이는 현상 때문임을 추측할 수 있다. 일요일의 경우 다른 요일에 비해 모든 시간대에 걸쳐 평균 속도가 높은 것을 알 수 있다. 이것은 도심에 속한 세그먼트에 대한 결과이며 공원등에 근접한 세그먼트의 경우 이와는 다른 속도 패턴을 보여 주었다.

각 세그먼트들에 대한 실시간 교통 흐름을 분석하기 위해 현재 시간을 기준으로 한 시간 이전부터 현재 시간까지의 실시간 속도 데이터를 서버에 유지하였다. 현재의 속도 패턴과 누적된 데이터로부터 추출한 속도 패턴을 비교하여 현재의 교통 상태가 특별한 교통 정체 없이 일상적인 교통 흐름을 유지하고 있는 경우 누적 속도 패턴에 의해 미래 시간에 대한 속도를 예측하였다. 미래 시간은 현재 시간을 기준으로 5분 후부터 3시간 이후까지의 시간을 의미한다. 현재의 교통 상태가 일상적인 교통 흐름인지 아닌지에 대한 결정은 누적 데이터베이스에 있는 속도 평균과 표준편차 값을 이용하였다. 3.3절에서 이에 대하여 자세히 설명할 것이다.

누적 속도 패턴을 이용하여 미래 시간에 대한 속도를 예측하기 위한 방법은 알고리즘3-1과 같다.

알고리즘 3-1: 누적속도 패턴을 이용한 속도 예측

1. 현재 속도 데이터 $CV_{k-j}(L_i), (0 \leq j < 12)$ 를 가져온다.
2. 실시간 데이터의 현재 요일과 시간에 대응되는 누적 속도 데이터 $AV_{k-j}(L_i), (0 \leq j < 12)$ 를 가져온다.
3. 실시간 데이터와 대응되는 누적 데이터의 속도 차이 함수 $Speed(CV_{k-j}(L_i))$ 를 계산한다. 함수는 아래와 같이 정의되며 $1 \leq j \leq 12$ 의 값을 가진다.

$$\delta = AV_{k-j}(L_i) - \sigma(AV_{k-j}(L_i))$$

$$Speed(CV_{k-j}(L_i)) = -1, \text{if } \delta \geq CV_{k-j}(L_i)$$

$$Speed(CV_{k-j}(L_i)) = 1, \text{if } \delta \leq CV_{k-j}(L_i)$$

$$Speed(CV_{k-j}(L_i)) = 0, \text{otherwise}$$
4. 예측속도 $PV_{k+p}(L_i), (1 \leq p \leq 68)$ 를 계산한다.

$$PV_{k+p}(L_i) = AV_{k+p}(L_i) + \sum_{j=1}^{12} (Speed(CV_{k-j}(L_i)) * f(L_i)),$$

$$f(L_i) = \alpha^* \sqrt{\frac{\sum_{j=i}^{i-12} (CV_{k-j}(L_i) - AV_{k-j}(L_i))^2}{12}}$$

알고리즘 3-1은 하나의 세그먼트 L_i 의 미래시간 속도를 예측한다. 따라서 모든 세그먼트 $L_i (1 \leq i \leq n)$ 에 대한 속도 예측은 알고리즘의 반복을 통해 구할 수 있다. 알고리즘 3-1에서 $CV_k(L_i)$ 는 현재 시간 k 에 대한 특정 링크 L_i 의 실시간 속도를 의미하고 $CV_{k-1}(L_i)$ 는 현재 시간을 기준으로 5분 전의 실시간, $CV_{k-2}(L_i)$ 는 10분 전의 실시간 속도를 각각 의미한다. 현재 시간 k 에 대응되는 특정 링크 L_i 의 누적 속도는 $AV_k(L_i)$ 로 표시하고, 5분 전 누적 속도는 $AV_{k-1}(L_i)$ 로 각각 표현된다. 또한 $\alpha(AV_k(L_i))$ 는 현재 시간 k 에 대응되는 특정 링크 L_i 의 누적 속도들에서 구한 표준 편차를 의미한다. 예측 속도 $PV_{k+1}(L_i)$ 는 현재시간 k 로부터 5분 경과한 후의 예측 속도를, $PV_{k+68}(L_i)$ 는 3시간 경과한 후의 예측 속도를 각각 의미한다.

3.2. 칼만 필터에 의한 속도 예측

실시간 속도 패턴이 누적 속도 패턴과 다른 경우 칼만 필터에 의해 속도를 예측한다. Rudolf E. Kalman이 발명한 칼만 필터는 최소 자승법(least-square method)을 사용해서 실시간으로 잡음 운동 방정식을 가진 시간 의존 상태 벡터(time-dependent state vector)를 추적하는 효율적인 반복 계산 방법이다[5,6]. 칼만 필터는 하나의 시스템이 시간에 따른 변화를 적절하게 예측할 수 있도록 잡음으로부터 신호를 찾아내기 위해 주로 사용된다.

이산시간에서의 칼만 필터는 $x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}$ 로 표현된다. 식에서 x_k 는 시간 k 에서 상태 벡터 x 의 값을 나타내고, u_{k-1} 은 시간 $k-1$ 에서 입력으로 주어지는 제어 값을 의미한다. 그리고 w_{k-1} 은 발생 가능한 잡음을 나타내며, A 와 B 는 각각의 값들을 대응시키는 상수 행렬이다. 칼만 필터를 사용한 추정은 결국 이전의 상태 값 x_{k-1} 과 제어 값 u_{k-1} 을 이용해 다음 시간에서의 상태 값 x_k 를 추정하는 것이라고 말할 수 있다.

칼만 필터를 각 세그먼트들에 대한 속도 예측에 적용하기 위해 상태벡터 x_k 를 세그먼트수 N 과 같은 크기를 가지는 벡터로 가정하면, A 는 각 링크사이의 공간상의 관계를 나타내는 N^*N 행렬로 나타낼 수 있다. 본 논문에서 사용한 칼만 필터 계산과정은 그림 5와 같다. 초기화

과정에서 사용되는 속도가 현재 입력된 실시간 속도와 5분전의 실시간 속도이기 때문에 각 행렬은 2×2 또는 2×1 로 초기화 된다. 또한 예측 속도와 측정 속도에 대한 잡음의 분산은 10으로 초기화하고 같은 값을 계속 사용하였다.

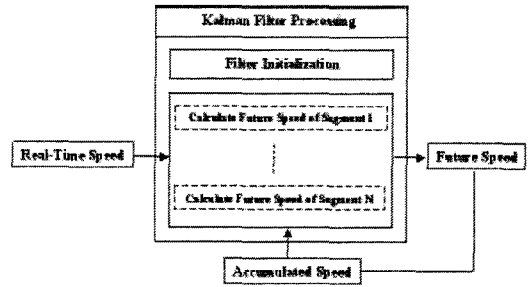


그림 5. 칼만 필터를 이용한 예측 속도 개선 과정
Fig. 5. The computational process for predicting the speed by the Kalman filter

본 논문에서 사용한 데이터는 서울시 도로 네트워크로 세그먼트의 개수가 대략 20,000 정도이다. 실시간 데이터가 입력되는 간격은 5분이며 현재시간에서 3시간까지의 미래 속도를 5분 단위로 예측해야 한다. 이것을 위해서는 다음 시간대에서의 실시간 속도가 입력되기 전까지 예정된 속도를 계산하는 것이 필수적이다. 빠른 시간 안에 칼만 필터를 이용한 속도예측계산을 완료하기 위해 각 링크 사이의 공간적인 관계에 대한 고려를 배제하고, 각 링크를 다른 링크와 독립적으로 처리하였다. 제안한 칼만 필터를 이용하여 3시간 후까지의 속도를 5분 간격으로 계산하는데 소요되는 시간은 대략 10초 정도였다.

3.3. 속도 패턴과 칼만 필터의 통합

두 가지 방법으로 예측한 결과 중 하나를 실제 예측 결과로 선택하기 위해 누적 데이터베이스에 저장된 모든 세그먼트들에 대한 속도 표준 편차를 사용한다. 실시간 속도 $CV_k(L_i)$ 와 동일한 요일과 시간대에 대응되는 누적 속도 패턴에 나타난 평균 속도 $AV_k(L_i)$ 를 비교하여 유의한 범위 내에 있는지를 조사한다. 서버에 유지하고 있는 한 시간 분량의 실시간 데이터에 대해 현재 속도가 누적 속도의 유의 범위 안에 있는지를 검사하기 위해 아래 식을 사용한다.

$$\alpha = AV_{k-j}(L_i)$$

$$\beta = \delta^* \sigma(AV_{k-j}(L_i))$$

$$\alpha - \beta \leq CV_{k-j}(L_i) \leq \alpha + \beta$$

위 식에서 δ 는 상수 값으로 95% 신뢰 수준 안에 있는지 검사를 경우 1.96의 값을 가진다. 한 시간 분량의 현재 속도 $CV_{k,j}(L_i), (0 \leq j < 12)$ 가 모두 식의 범위 안에 있으면 현재의 차량 속도가 일상적인 차량 흐름을 유지하고 있음을 의미하므로 누적 속도 패턴에 의한 예측 결과를 선택한다. 현재 속도 $CV_{k,j}(L_i), (0 \leq j < 12)$ 가 유의 구간 안에 얼마나 포함 되는지에 따라 최종적인 예측 결과를 얻는 방법은 아래식과 같다.

$$PV_{k+j}(L_i) = a * sbp_{k+j}(L_i) + b * sbk_{k+j}(L_i)$$

여기서, $b = 1 - a$ 이다.

위식에서 a 는 $CV_{k,j}(L_i), (0 \leq j < 12)$ 가 식의 유의 구간 안에 포함되는 비율이며 b 는 포함되지 않은 비율이다. $sbp_{k+j}(L_i)$ 는 누적 속도 패턴에 의한 예측 결과를 $sbk_{k+j}(L_i)$ 는 칼만 필터에 의한 예측 결과를 의미한다.

IV. 실험 결과

제안한 예측 시스템의 정확성을 검증하기 위해 2004년 9월 한달 동안의 서울시 모든 세그먼트들에 대한 실시간 교통 정보를 사용하여 제안한 예측 알고리즘을 실험하였다. 실험 방법은 다음과 같다. 먼저 앞선 3주의 데이터를 이용하여 누적 속도 데이터베이스를 생성하고 나머지 4주 데이터를 실시간 데이터로 입력하면서 제안한 예측 알고리즘을 이용하여 미래 속도를 예측하였다. 예측된 속도는 실제 속도와 비교함으로써 예측 결과의 정확성을 분석하였다. 미래 시간에 대한 속도 예측은 본 논문에서 제안한 알고리즘과 누적 속도 패턴 데이터만을 이용한 방법을 사용하였다. 누적 속도 패턴 데이터만을 사용하는 예측 방법은 데이터베이스에 저장된 속도 데이터를 예측된 결과로 사용하는 것으로 과거 실시간 교통 정보가 없는 경우 이 방법을 이용하여 속도 예측을 하였다.

예측 정확성을 측정하는 측도로 예측 속도와 실제 속도사이의 오차를 나타내는 예측 속도 오차 $e_k(L_i)$ 를 정의

하였다. 9월 넷째 주에 속한 특정일, 예를 들면 9월 24일 0시 0분부터 실시간 데이터를 5분 단위로 입력하여, 현재 시간을 기준으로 5분 후부터 3시간까지의 속도를 예측하여 예측 속도를 구한 다음, 예측 속도와 9월 24일 0시 5분부터 3시 5분까지의 실제 속도의 차이를 아래식과 같이 계산하였다.

$$e_{k+j}(L_i) = |PV_{k+j}(L_i) - RV_{k+j}(L_i)|$$

여기서 $PV_{k+1}(L_i)$ 는 5분 후의 세그먼트 L_i 에 대한 예측 속도이고 $PV_{k+60}(L_i)$ 는 3시간후의 세그먼트 L_i 에 대한 예측 속도를 각각 나타낸다. $RV_{k+j}(L_i)$ 는 현재 시간으로부터 $j * 5$ 분이 경과하여 알 수 있는 세그먼트 L_i 의 실제 속도를 나타낸다.

그림 6은 서울 전체 도로 네트워크 중 강남구에 속한 390여 개의 세그먼트들에 대하여 3시간 분량의 미래 속도를 본 논문에서 제안한 알고리즘의 예측 결과와 누적 속도 데이터만을 이용하여 예측한 결과를 비교하여 보여 준다. 제안한 알고리즘의 경우 예측 속도 오차가 없는 세그먼트의 개수는 약 2400여 개이지만 누적 데이터만을 사용할 경우 약 1500여 개였다. 예측 속도 오차가 시속 6km 이하일 경우 제안한 알고리즘의 세그먼트의 개수가 누적 속도 패턴만을 이용한 세그먼트의 개수보다 많고 7km 이상인 경우는 반대의 결과를 보여준다. 이것은 제안한 알고리즘의 예측 결과가 누적 속도 데이터만을 이용한 속도 예측보다 예측 오차가 적으므로 더 정확하게 예측하였음을 보여준다.

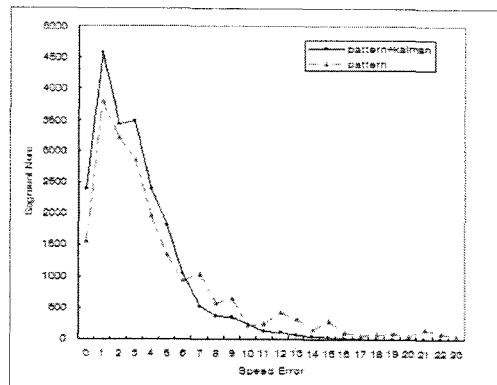


그림 6. 누적 속도 패턴 및 제안된 알고리즘을 이용한 예측 결과의 비교

Fig. 6. The comparison of the results of two methods

전체적으로 살펴보면 390여 개의 세그먼트들에 대한 3시간 분량의 예측 결과에 대해 제안된 알고리즘은 전체의 91%가 예측 속도 오차가 7km이하였고, 98%가 예측 속도 오차가 10km이하로 예측하였다. 반면 누적 속도 패턴만을 사용할 경우 7km이하의 예측 속도 오차를 가지는 것은 전체세그먼트의 77%이고 10km이하의 예측 속도 오차를 가지는 세그먼트는 전체의 89%였다. 누적 속도 패턴만을 사용할 경우 실시간 발생할 수 있는 교통 정체나 돌발 상황들을 반영하지 못하므로 정확성에 있어 실시간 데이터를 반영하는 것과 비교할 때 예측 정확도가 떨어짐을 알 수 있다.

V. 결 론

본 논문에서는 실시간 교통 정보를 반영하여 현재 시간 이후의 교통 정보를 예측하기 위한 알고리즘을 제안하였다. 예측 알고리즘은 세그먼트들에 대한 평균 속도를 5분 단위로 입력 받아 현재시간으로부터 3시간 이후까지 각 세그먼트들에 대한 속도를 예측한다. 예측은 두 가지 방법으로 이루어졌으며 첫째는 모든 세그먼트들에 대한 평균 속도를 요일과 시간별로 분석한 누적 속도 패턴을 사용하는 것이고 두 번째는 실시간 데이터들에 대하여 칼만 필터법을 적용한다. 예측 정확성을 측정하기 위해 예측 속도와 실제 속도사이의 오차를 나타내는 예측 속도 오차를 정의하여 실험한 결과 제안한 알고리즘은 예측 속도 오차가 지속 7km이하인 것이 전체 세그먼트의 91%로 누적 속도 데이터만으로 예측하는 방법보다 정확한 예측 결과를 주었다.

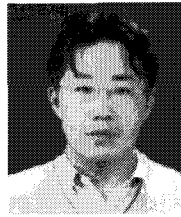
참고문헌

[1] Joonwan Kim, Trends of Services and Technology of Telematics, IITA IT Korea, 2004.
 [2] H. D. Chon, D. Agrawal and A. E. Abbadi, "FATES: Finding A Time dEpendent Shortest path," Proc. of the 4th International Conference on Mobile Data Management, pp. 165-180 2003.
 [3] S. Handley, P. Langley and F. Rauscher, "Learning to Predict the Duration of an Automobile Trip," Proc. Of the 4th International Conference on Knowledge Discovery and Data Mining, pp. 219-223, 1998, New Work.

[4] D. Moriarty, S. Handley, P. Langley, "Learning Distributed Strategies for Traffic Control," Proceedings of the Fifth International Conference of the Society for Adaptive Behavior, pp. 437-446, Zurich, Switzerland, 1998.
 [5] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," Transaction of the ASME Journal of Basic Engineering, pp. 35-45, March, 1960.
 [6] grewal, Mohinder S., and Angus P. Andrews, *Kalman Filtering : The theory and Practice*, Prentice Hall, 1993.

저자소개

유 영 중 (Young-Jung Yu)



1996년 2월 부산대학교 전자계산학과 (이학사)
 1998년 2월 부산대학교 전자계산학과 (이학석사)
 2002년 2월 부산대학교 전자계산학과 (이학박사)

2002년 3월 - 현재 부산외국어대학교 컴퓨터공학부(조교수)
 ※ 관심분야: 컴퓨터 그래픽스, 애니메이션, 시뮬레이션

조 미 경 (Mi-Gyung Cho)



1990년 2월 부산대학교 전자계산학과 (이학사)
 1992년 2월 부산대학교 전자계산학과 (이학석사)
 1999년 2월 부산대학교 전자계산학과 (이학박사)

2000년 9월-2002년 8월 부산대학교 기금교수
 2002년 9월 - 현재 동명대학교 멀티미디어공학과(조교수)
 ※ 관심분야: 컴퓨터 알고리즘, GIS