

# 도로 네트워크 환경에서 사용자의 이동 성향을 고려한 경로 생성 기법

## A Path Planning Scheme Using the Moving Tendencies of Users in Road Network Environments

황동교, 박혁, 김동주, 리하, 박준호, 박용훈, 복경수, 유재수  
충북대학교 정보통신공학과

Donggyo Hwang(corea1985@gmail.com), Hyuk Park(agodsun@naver.com),  
Dongjoo Kim(dongjoo.k@gmail.com), He Li(lihelol@gmail.com),  
Junho Park(junhopark@chungbuk.ac.kr), Yonghoon Park(yhpark1119@chungbuk.ac.kr),  
Kyoungsoo Bok(ksbok@chungbuk.ac.kr), Jaesoo Yoo(yjs@chungbuk.ac.kr)

### 요약

사용자들은 시간, 거리, 도로 혼잡도와 같은 속성들에 의해 선호하는 경로가 있기 때문에 사용자의 이동 성향에 맞는 경로를 생성하는 기법들이 필요하다. 기존의 기법들은 이동 성향을 고려하여 경로를 생성하기 위해서는 사용자의 이동 성향 정보를 추가적으로 입력하여야 한다. 그러나 네비게이션 및 모바일 장치의 불편한 인터페이스 특성상 이러한 정보 입력은 거의 하지 않고 출발지와 목적지만을 입력하여 경로를 추천 받는 경향이 있다. 본 논문에서는 추가적인 이동 성향에 대한 정보 입력 없이 이동 성향에 맞는 경로를 생성하는 기법을 제안한다. 성능평가를 통해 최소 시간 경로나 최단 거리 경로와 비교하여 제안하는 기법이 사용자의 이동 성향을 고려한 경로가 생성됨을 입증한다.

■ 중심어 : | 경로 생성 | 이동 객체 | 이동 성향 | 암시적 추출 |

### Abstract

A path generation method considering the tendencies of users is required because users has the preferred routes through the properties such as time, distance, and congest. The existing method must input the additional information about user's tendency to generate path considering moving tendencies. However, mobile device have the inconvenient interface. Therefore users are reluctant to input additional information for path generation. In this paper, we propose the path generation method considering the tendencies of users without additional input of users tendencies. To show the superiority of the proposed method, we compared it with shortest distance or time path through performance evaluation.

■ keyword : | Path Generation | Moving Object | Move Tendency | Implicit Extraction |

## I. 서론

최근 GPS가 장착된 네비게이션이나 모바일 단말기

가 널리 보급됨에 따라 이러한 장치를 통해 사용자의  
궤적 정보를 기록 할 수 있게 되었다. 이러한 사용자의  
과거 궤적 정보를 이용하여 사용자의 이동 성향이나 사

\* 본 연구는 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 산학협력 선도대학(LINC)육성사업과 KISTI의 지원(K-12-L06-C02-S03)으로 수행되었음

접수번호 : #120716-004

접수일자 : 2012년 07월 16일

심사완료일 : 2012년 08월 16일

교신저자 : 유재수, e-mail : yjs@chungbuk.ac.kr

용자의 개인화된 특성에 맞는 다양한 서비스 제공하기 위한 연구들이 진행되고 있다[1]. 특히, 사용자의 과거 궤적 정보를 분석하여 현대 생활에 필수적인 응용 분야인 네비게이션 서비스에 적용시키기 위한 연구가 활발히 진행 중이다[2-4]. 이러한 네비게이션 시스템에서 필수적인 기반 기술로 경로 생성 기법이 있다. 경로 생성 기법은 최단 거리 경로와 최소 시간 경로로 분류할 수 있다. 최단 거리 경로는 출발점과 목적지까지 가장 짧은 거리를 갖는 경로이고, 최소 시간 경로는 최단 거리보다는 빠른 시간 안에 도달 할 수 있는 경로이다. 일반적인 네비게이션 시스템은 사용자의 이동 성향과 상관없이 앞에서 언급한 일반적인 최소 시간 경로나 최단 거리경로를 안내해준다.

각각의 사용자들은 최단 거리 경로나 최소 시간 경로 외에도 사용자의 이동 성향에 따라 여러 가지 경로를 제공 받기를 원한다. 예를 들어, 사용자들은 도로의 혼잡도가 일정 수준 이상이면 우회하여 돌아가거나 주로 자주 다니는 도로를 이용하는 등 다양한 이동 성향을 가지고 있다. 최근 사용자의 이동 성향에 따른 경로 생성이 중요해지면서 사용자의 이동 성향에 맞는 경로를 생성하는 연구가 활발히 진행되고 있다.

[3]에서는 경로를 생성하기 위해 경로의 거리, 시간, 신로도와 도로의 유형, 신호등, 주변시설, 길이, 폭, 품질 등 각 도로 세그먼트의 속성을 고려한다. 다양한 속성 정보에 사용자가 입력한 이동 성향 속성의 선호도를 적용하여 사용자의 이동 성향에 맞는 경로를 생성한다. 하지만 네비게이션이나 모바일 단말기는 불편한 인터페이스로 인해 추가적인 정보 입력이 용이하지 않아 사용상에 문제점이 발생한다. 따라서 대부분 사용자들은 추가적인 정보 입력을 수행하지 않고 사용자의 성향에 적합한 경로를 추천받기를 원한다. 즉, 사용자의 정보 입력이 없는 상황에서 사용자의 이동 성향을 추출하고 이동 성향에 맞는 경로를 생성하는 기법들이 필요하다. 특정 사용자는 평일 출근을 위해 빠른 경로를 선호하지만 주말에는 혼잡하지 않는 경로를 선호하는 것처럼 시간에 따라 사용자의 선호하는 경로가 서로 다르게 나타날 수 있다[4]. 그러나 기존 연구들은 이러한 시간과 같은 상황 정보를 고려하지 않는다.

본 논문에서는 시간과 요일 속성마다의 이동 성향을 추출한 후 이동 성향에 맞는 최적의 경로를 생성하는 기법을 제안한다. 경로를 생성하기 위해 이동 객체들의 궤적 데이터를 수집하여 도로 세그먼트에 대한 시간과 혼잡도와 같은 속성을 추출하고 각각의 이동 객체들에 대해 이동 성향을 추출한다.

본 논문의 나머지 구성은 다음과 같다. 2장에서는 기존에 제안된 경로 생성 기법의 특징과 문제점에 대해 기술한다. 3장에서는 본 논문에서는 제안하는 사용자의 이동 성향을 생성하는 방법과 사용자의 이동 성향을 고려한 경로 생성 기법에 대해 기술한다. 4장에서는 기존에 제안된 경로 생성 기법과 성능 비교를 통해 제안하는 기법의 우수성을 입증한다. 마지막 5장에서 결론 및 향후 연구에 대하여 기술한다.

## II. 관련연구

경로 생성 기법은 크게 두 가지로 과거 궤적 데이터를 기반으로 경로를 생성하는 기법과 사용자들이 입력한 선호도를 기반으로 사용자의 선호도에 맞는 경로를 생성하는 기법으로 나눌 수 있다. 과거 궤적 데이터를 분석하여 최적의 경로를 생성하는 기법으로 사용자들의 과거 궤적을 통해 이동 경로 생성하는 기법을 제안하였다[5]. 과거 궤적으로부터 사용자의 이동성을 탐색해 Coherence Expanding 알고리즘을 사용하여 방향성을 고려하였다. 또한, Absorbing Markov Chain을 사용하여 이동 확률을 추론하고 이를 토대로 인기 경로(popular route)를 발견한다. 그러나 이 기법은 사용자의 선호도를 고려하지 않고 다수 사용자들 정보를 기반으로 확률적인 모델을 통해 경로를 생성한다.

[6]에서는 속도 패턴과 운전 패턴을 고려하여 경로를 생성하는 기법을 제안하였다. 속도 패턴은 날씨, 시간, 차의 종류 등과 같은 요소를 적용하여 도로 세그먼트에 대한 속도를 수집하는 것이다. 운전 패턴은 “R1이란 도로세그먼트가 여름에는 좋지만, 겨울에는 위험하다”와 “R2는 속도가 빠르지만 범죄 발생이 높은 곳에 위치한다.”와 같은 규칙을 정의하여 정보를 수집하는 것이다.

[6]에서는 이러한 요소들을 고려하여 최적의 경로를 생성한다.

과거 궤적 데이터를 기반으로 경로를 생성하는 연구들은 별도의 추가적인 정보를 입력하는 것이 아니라 과거 궤적 데이터를 분석하여 암시적으로 경로 생성에 대한 정보를 추출하는 기법을 사용한다. 하지만 사용자마다의 선호도와 같은 개인화된 특성에 대해 고려하지 않는다는 문제점이 있다.

사용자의 선호도에 따라 적합한 경로를 생성하는 기법으로 도로의 복잡도에 따라 다른 경로 생성을 수행하는 기법이 제안되었다[7]. [7]에서는 도로의 복잡도를 계산하기 위해 특정 도로 세그먼트와 연결된 도로 세그먼트의 수와 연결된 도로 세그먼트의 각도와 평균 도로 세그먼트의 길이를 고려하여 CWC 값을 계산한다. CWC 값에 사용자가 입력한 성향 값을 토대로 가중치를 적용하여 도로들을 군집화하고 복잡도에 따른 경로 생성을 수행한다.

[3]에서는 경로의 거리, 시간, 신뢰도와 도로 세그먼트의 다양한 속성을 고려하여 경로를 생성하는 기법을 제안하였다. 다양한 속성 값을 적용시키기 위해 사용자들은 각 속성에 대한 선호도를 입력한다. 입력된 선호도를 적용하기 위해 각 속성에 대해 fuzzy quantifier-guided OWA라는 기법을 사용하여 가중치가 계산하고 계산된 가중치를 통해 경로를 생성한다. [8]은 도로의 수용력과 차량의 속도, 도로의 길이를 토대로 D-S evidence theory[9]를 통해 도로 세그먼트의 가중치를 계산한다. 계산된 가중치를 토대로 PCNN(Pulse Coupled Neural Network)을 통해 경로를 생성한다.

사용자들이 입력한 선호도를 기반으로 사용자의 선호도에 맞는 경로 생성 기법들은 사용자의 선호도를 고려하기 위해 경로 생성에 필요한 다양한 속성 정보를 사용자가 입력해야 한다는 문제점이 있다. 또한, 사용자의 이동 성향을 고려하지 않으며 경로를 생성하기 위해 많은 양의 데이터를 입력해야 한다. 하지만 사용자들은 경로를 생성시킬 경우 받은 경우 네비게이션 및 모바일 장치의 불편한 인터페이스 때문에 출발지 및 목적지를 입력할 뿐 부가적인 정보는 입력하지 않는다. 따라서

사용자의 추가적인 정보 입력이 없는 환경에서 사용자의 이동 성향에 적합한 경로를 제공하기 위한 기법이 필요하다.

### III. 제안하는 이동 객체 추적 기법

#### 3.1 개요

본 논문에서는 사용자의 추가적인 정보의 입력 없이 사용자의 과거 궤적 데이터를 통해 이동 성향을 암시적으로 추출하고 사용자의 이동 성향을 고려한 최적의 경로를 생성하는 기법을 제안한다. 사용자의 이동 성향을 고려한 경로 생성 기법의 전체적인 구조는 [그림 1]과 같다. 사용자의 과거 궤적 데이터와 도로 세그먼트 정보를 이용하여 도로 세그먼트의 특성 정보를 생성하고 사용자가 경유한 과거 경로와 추천 받은 경로를 토대로 사용자 이동 성향정보를 생성한다. 이렇게 생성된 도로 세그먼트의 특성 정보와 사용자 이동 성향정보를 이용하여 사용자의 이동 성향을 맞는 경로를 생성하여 생성한다.

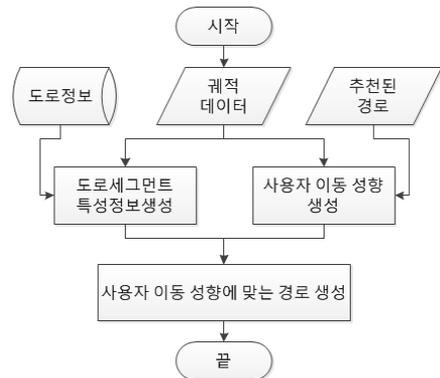


그림 1. 사용자의 이동 성향을 고려한 경로 생성 시스템

제안하는 기법에서는 경로 생성을 위해 사용자의 과거 궤적 정보, 도로 세그먼트의 정보, 사용자가 기존에 안내 받은 경로가 필요하다. 사용자의 과거 궤적 정보에 대한 정의는 다음과 같다. 사용자  $i$ 의 과거 궤적 정보는  $tr_i : (p_1 \rightarrow p_2 \rightarrow \dots \rightarrow p_n)$ 과 같이 정의한다. 각각의 점

$p_i$ 는  $(lat_i, lng_i, t_i)$ 로 구성되며  $lat_i$ 는 위도,  $lng_i$ 는 경도,  $t_i$ 는 시간을 의미한다. 사용자에게 제공된 경로 정보는 과거 궤적 데이터와 유사하게 표현된다. 도로 세그먼트의 기본적인 정보는  $r_i = (len_i, spe_i)$ 와 같이 정의한다. 이때,  $len_i$ 은 도로 세그먼트의 길이를 의미하며  $spe_i$ 는 제한 속도를 의미한다.

### 3.2 도로세그먼트 특성 정보 생성

기본적인 도로 세그먼트의 정보인  $r_i$ 와 과거 궤적 데이터를 분석하여 도로 세그먼트의 특성 정보를 생성한다. 도로 세그먼트의 특성 정보는 거리, 속도, 혼잡도로 구성되며  $rpr_i = (len_i, at_i, ac_i)$ 와 같이 정의된다. 이때,  $len_i$ 은 도로 세그먼트의 길이,  $ac_i$ 는 도로 세그먼트를 통과하는데 소요되는 평균 시간,  $at_i$ 는 도로 세그먼트를 통과하는 시간 (식 1)을 이용하여 계산한다. 이때,  $r_i$ 는 도로 세그먼트를 의미하며,  $N$ 은  $r_i$ 을 지나간 객체들의 총 수,  $p_i.s$ 와  $p_i.e$ 는 차량이 도로 세그먼트  $r_i$ 을 들어온 시간과 나간 시간을 의미한다.

$$at_i = \frac{\sum_{i=1}^N (p_i.e - p_i.s)}{N} \quad (1)$$

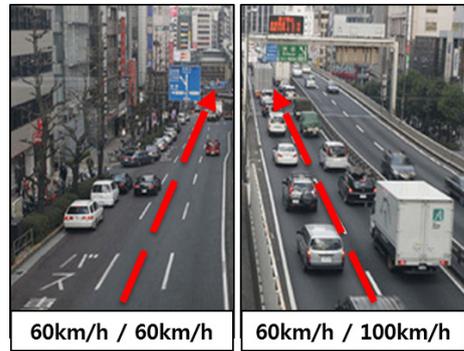
$ac_i$ 는 도로 세그먼트의 혼잡도를 의미하며 본 논문에서는 도로 세그먼트에서 차량의 밀집도 및 속도로 정의하지 않고 각 도로 세그먼트마다의 제한 속도를 기준으로 한 속도의 비율로 정의한다. [그림 2](a)와 같이 왼쪽 도로는 제한 속도가 60km/h인 일반도로이고 [그림 2](b)와 같이 오른쪽 도로는 제한 속도가 100km/h인 고속도로이다. 일반도로에서 특정 차량이 60km/h로 진행하는 경우와 고속도로를 60km/h로 진행하는 경우의 사용자가 느끼는 혼잡도는 다를 것이다.

따라서 도로의 제한 속도에 따라 사용자가 느끼는 상대적인 속도차로 혼잡도로 사용한다. 즉, 혼잡도란 제한 속도와 해당 도로 세그먼트를 통과한 차량의 평균 속도를 나누어 제한 속도를 기준으로 더 빠르게 통과하였는지 아닌지를 계산하여 나온 값을 의미한다. 혼잡도는 (식 3)에 의해 계산된다. (식 3)에서  $r_i.spe_i$ 는 도로의 제

한 속도이고,  $s_i$ 는 속도를 의미하며 (식 2)를 통해 계산할 수 있다.

$$s_i = \frac{r_i.len_i}{at_i} \quad (2)$$

$$ac_i = \frac{r_i.spe_i}{s_i} \quad (3)$$



(a) 일반 도로 (b) 고속 도로

그림 2. 도로의 상대적인 혼잡도

### 3.3 사용자 이동 성향 생성

사용자마다 거리, 시간, 혼잡도와 같은 이동 성향이 다를 수 있다. 예를 들어 두 경로가 비슷할 경우 시간을 우선시 하여 시간이 약간 적게 걸리는 경로를 선택할 것인가 아니면 거리가 약간 짧은 경로를 선택할 것인가는 사용자마다 다를 것이다. 또한, 도로가 혼잡하다고 느끼는 기준도 사용자마다 다르다. 따라서 사용자마다 이동 성향을 추출하여 사용자에게 맞는 경로를 생성할 때 사용해야 한다.

제안하는 기법에서는 사용자의 이동 성향을 경로의 거리, 시간, 혼잡도로 분류한다. 이렇게 각각의 경로에 대한 속성 값을 이용하여 사용자의 이동 성향을 추출하기 위해 실제 이동 경로와 안내받은 경로를 비교한다. 사용자는 네비게이션 시스템에 의해 최단 거리 경로나 최소 시간 경로를 안내 받게 된다. 만약 사용자가 최단 거리 경로를 통해 이동한다면 거리 속성에 대한 선호도가 증가하고 최소 시간 경로를 통해 이동한다면 시간 속성에 대한 선호도가 증가한다. 하지만 기존에 안내받은 경로를 통해 이동하지 않을 경우 사용자가 실제 이

동한 경로와 안내받은 경로를 비교하여 사용자의 이동 성향을 추출한다.

이러한 이동 성향 추출 과정에서 세 가지 경우가 발생한다. 예를 들어, [그림 3]과 같이 안내 받은 경로와 실제 이동한 경로가 있다고 가정하자. 유형 1은 추천 경로를 이용해서 목적지까지 이동한 경우이며 유형 2는 추천받은 경로를 이용하여 이동을 하는 상황에서 특정 구간에서 다른 경로로 이동 경로를 변경한 경우이다. 유형 3은 처음부터 추천받은 경로를 이용하지 않고 다른 경로를 이동한 경우이다. 유형 1인 경우 안내 받은 경로가 최단 거리 경로인지 최소 시간 경로인지에 따라 해당 속성에 가중치를 부여한다. 유형 2인 경우 추천 경로와 사용자가 이동한 경로를 비교하여 두 경로가 겹치지 않는 부분을 비교하여 속성 가중치를 부여한다.

유형 2인 경우 안내 받은 경로가 아닌 다른 경로를 이용한다는 것을 안내 받은 경로에 속하는 도로 세그먼트가 혼잡하기 때문에 다른 도로 세그먼트를 이용하여 진행한 것으로 판단한다. 따라서 사용자는 특정 혼잡도를 가진 도로 세그먼트를 만나면 우회하는 경향이 있다고 판단하여 혼잡도 한계치로 저장하게 된다. 그 후 안내 받은 경로와 사용자가 진행한 경로를 비교하여 효율적인 속성에 가중치를 부여한다. 만약 추천 받은 경로보다 사용자가 간 경로가 모든 속성에서 효율적이지 않다면 사용자의 잘못된 판단으로 이동을 한 상태로 판단하여 사용자 이동 성향 추출 에서 제외시킨다. 유형 3인 경우 추천받은 전체 경로와 사용자가 경유한 경로를 비교하여 특정 속성 값에 대해 효율적인 경로라면 효율성에 따라 속성에 가중치를 부여한다.

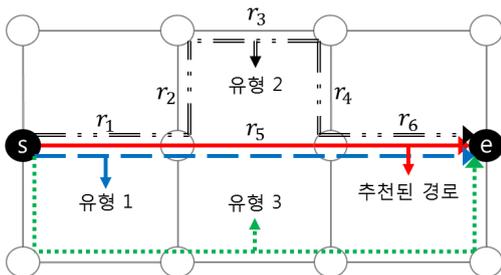


그림 3. 추천된 경로와 사용자 과거 궤적의 비교

[그림 4]는 선호도 가중치 계산하는 알고리즘을 나타낸 것이다. 우선 사용자가 안내받은 경로 GR과 사용자가 진행한 경로 UR을 비교하여 겹치는 부분인 overlap을 `overlapDiscovery(UR,GR)`를 이용하여 추출하게 된다. 만약 overlap이 존재하지 않는다면 두 경로를 전혀 다른 경로이므로 `compareThree(UR,GR,overlap)`을 이용하여 유형 3의 추출 과정을 적용시킨다. 만약 두 경로의 크기가 같고 overlap의 크기도 같다면 두 경로가 동일한 경로이므로 `compareOne(UR,GR)`을 이용하여 유형 1의 추출 과정을 진행한다. 두 경우가 아니라면 경로 중 일부만 유사한 유형 2이므로 `compareTwo(UR,GR,overlap)`을 이용하여 유형 2의 추출 과정을 진행한다. 이렇게 각각의 유형에 대한 추출과정이 끝나면 사용자의 이동 성향을 반환하게 된다.

알고리즘1 사용자 선호도 추출

```

입력:
UR = 사용자 경로
GR = 안내된 경로
출력:
timeWeight = 시간 가중치
distanceWeight = 거리 가중치
congestThreshold = 혼잡 한계치
1: overlap := overlapDiscovery(UR,GR);
2: if overlap.size() == (UR.size()+GR.size())/2 then
//유형1일 경우 사용자의 이동성향 추출하여 3
3: 가지 속성 값 반환
compareOne(UR,GR);
4: else
5: if overlap.size() > 0 then
//유형2일 경우 사용자의 이동성향 추출하여
6: 3가지 속성 값 반환
compareTwo(UR,GR,overlap);
7: else
//유형3일 경우 사용자의 이동성향 추출하여
8: 3가지 속성 값 반환
compareThree(UR,GR,overlap);
9: end if
10: end if
    
```

그림 4. 사용자 선호도 추출 알고리즘

예를 들어, 사용자에게 최단 경로 ( $r_1 \rightarrow r_5 \rightarrow r_6$ )가 추천된 상황에서 사용자가 실제 이동한 경로는 ( $r_1 \rightarrow r_2 \rightarrow r_3 \rightarrow r_4 \rightarrow r_6$ )라고 하자. 두 경로가 겹치지 않는  $r_5$ 와  $r_2 \rightarrow r_3 \rightarrow r_4$ 을 비교하면 [표 1]과 같다.  $r_5$ 의 거리는 4km, 소요 시간은 20분, 혼잡도는 2이다. 이에 반해  $r_2 \rightarrow r_3 \rightarrow r_4$ 의 거리는 10km, 소요 시간 15분, 혼잡도는 3이다.  $r_2$ 와  $r_5$ 의 혼잡도는  $r_5$ 의 혼잡도가 높다는 것을 확인할 수 있고 이를 혼잡도 한계치로 저장한다. 또한,  $r_5$ 와  $r_2 \rightarrow r_3 \rightarrow r_4$ 의 거리를 비교할 경우  $r_2 \rightarrow r_3 \rightarrow r_4$ 가 거리는 길지만 시간이 짧은 것을 확인할 수 있다. 결론적으로 사용자의 혼잡도 한계치는 2이고 거리는 좀 더 길지만 혼잡하지 않으면서 빠른 경로를 선택한다고 판단할 수 있다.

표 1. 도로세그먼트 속성 정보

도로 세그먼트	거리(km)	시간(m)	혼잡도
$r_1$	4	10	1
$r_2$	4	5	1
$r_3$	3	5	1
$r_4$	3	5	1
$r_5$	4	20	2
$r_6$	4	16	2

따라서 시간 속성에 가중치를 부여한다. 이렇게 각각의 궤적들에 대해 위의 알고리즘을 수행한다. 예를 들어 한 사용자의 25개의 궤적 데이터를 이용하여 추출했을 경우 거리가 5회, 시간은 20회가 나왔다면 (식 4)을 통해 이동 성향을 계산한다. 이때,  $W_{time}$ 은 시간 속성 값이고,  $W_{dist}$ 는 거리 속성 값을 의미한다. 사용자의 거리와 시간에 대한 이동성향을 계산하게 되면 사용자의 거리 속성 성향은 0.2, 시간 속성성향은 0.8이 나오게 된다. 이렇게 추출된 사용자의 이동 성향을 이용하여 각각 도로 세그먼트 특성 정보에 적용시켜 경로를 생성하기 위한 가중치로 사용된다. 또한, 알고리즘을 통해 추출된 사용자의 혼잡도 한계치는 경로를 생성할 경우 혼잡도 한계치보다 높은 값을 갖은 도로 세그먼트를 제외하여 경로를 생성하게 된다.

$$\frac{W_{dist,or\ time_i}}{W_{dist} + W_{time}} \quad (4)$$

### 3.4 사용자 이동 성향을 고려한 경로 생성

추출한 사용자의 이동 성향을 바탕으로 경로를 생성해야 한다. 경로를 생성하기 위해 각각의 속성 값을 이용하여 도로 세그먼트의 가중치 값을 계산한다. 도로 세그먼트에 대한 가중치는 (식 5)을 이용하여 부여된다. 이때,  $P_{dist}$ 는 거리 속성값,  $P_{dist} \cdot MAX$ 는 거리 속성의 최댓값,  $W_{dist}$ 는 거리 속성의 가중치,  $P_{time}$ 은 시간 속성값,  $P_{time} \cdot MAX$ 는 시간 속성의 최댓값,  $W_{time}$ 은 시간 속성의 가중치를 의미한다. 거리와 시간의 값의 크기가 다르기 때문에 거리와 시간의 크기를 동일하게 하기 위해서 데이터에 존재하는 거리와 시간의 최댓값으로 각각 나누어주고 사용자의 이동 성향을 고려한 도로 세그먼트들의 속성 값이 계산된다.

$$\frac{P_{dist}}{P_{dist} \cdot MAX} \times W_{dist} + \frac{P_{time}}{P_{time} \cdot MAX} \times W_{time} \quad (5)$$

각각의 도로 세그먼트에 대해 사용자의 이동 성향이 적용된 속성 값이 계산되면 A\*알고리즘[10]을 통하여 비용이 최소가 되는 경로를 생성한다. 생성된 경로 중 혼잡도 한계치 이상의 경로가 존재한다면 혼잡도 한계치 이상의 경로를 제외하고 주변의 경로 중 비용이 적은 경로로 대체한다. 이를 통해 사용자의 이동 성향에 맞는 경로를 생성한다.

[그림 5]는 경로 생성 알고리즘을 나타낸 것이다. 우선 경로 생성 알고리즘에서 열린 목록과 닫힌 목록을 사용하는데 열린 목록은 검사를 해야 할 도로 세그먼트들의 집합이고, 닫힌 목록은 검사를 완료했거나 갈 수 없는 도로 세그먼트들의 집합이다. 사용자는 경로 생성을 위한 출발지점 SP와 도착지점 EP를 입력한다. 출발지와 목적지에 대해 추출된 사용자 이동 성향 정보  $P_{dist}$ ,  $P_{time}$ ,  $P_{con}$ 와 도로 세그먼트의 속성 정보를 이용하여 이동 성향이 적용된 도로 세그먼트의 비용을  $getWeight(distance, time, P_{dist}, P_{time})$ 을 통해 계산한다. 계산된 도로 세그먼트의 비용을 갖고 최소 비용을 갖는 경로를 생성하게 된다. 경로를 생성하는 과정에서 사용자의 이동성향 중 하나인 혼잡도 한계치보다 큰 도로 세그먼트들은 필터링하여 경로 생성 과정에 포함하지 않는다. 필터링 과정을 통하여 경로 생성 비용을 줄여

보다 빠른 경로 생성을 할 수 있다. 이렇게 생성된 경로는 트리 구조를 가지고 있으므로 경로의 역순으로 저장되어 있다. 따라서 reconstruct\_path(ep)를 통해 출발지부터 목적지까지의 경로를 재 생성하여 최종적인 경로를 반환한다.

**알고리즘2 경로 생성**

```

입력:
SP = 출발지, EP = 목적지, Pdist = 거리 선호도,
Ptime = 시간 선호도, Pcon = 혼잡도 한계치
출력:
path = 경로
1: while openset is not empty do
2:   current := 열린 목록 중 가장 작은 값을 갖고 있는
      도로 세그먼트;
3:   if current == EP then
4:     return reconstruct_path(ep);
5:   end if
6:   열린 목록에서 현재 도로 세그먼트를 제거
7:   닫힌 목록에 현재 도로 세그먼트를 추가
8:   for each neighbor in neighbor_nodes(current) do
9:     if Pcon < current.congest then
10:      닫힌 목록에 현재 도로 세그먼트 추가
11:     end if
12:     if neighbor in closedset then
13:       continue;
14:     end if
15:     distance := 현재 도로 세그먼트부터 이웃 도로
      세그먼트까지의 거리 계산
16:     time := 현재 도로 세그먼트부터 이웃 도로
      세그먼트까지의 시간 계산
17:     weight :=
      getWeight(distance, time, Pdist, Ptime);
18:     nextStepCost = current.cost + weight;
19:     if (nextStepCost < neighbor.cost) then
20:       if (inOpenList(neighbor)) then
21:         열린 목록에서 neighbor 제거
22:       end if
23:       if (inClosedList(neighbor)) then
24:         닫힌 목록에서 neighbor 제거
25:       end if
26:     end if
27:     if (!inOpenList(neighbor)
      && !(inClosedList(neighbor))) then
28:       neighbor의 비용을 nextStepCost로 저장
29:       neighbor의 부모로 current 추가
30:       열린 목록에 neighbor 추가
31:     end if
32:   end for
33: end while
    
```

그림 5. 경로 생성 알고리즘

**IV. 성능평가**

본 논문에서는 제안하는 기법의 우수성을 보이기 위해 실험을 수행하였다. 실험을 위해 [그림 6]와 같은 Oldenburg지역의 환경에 [표 2]와 같이 도로 네트워크 환경을 구성하였으며 Brinkhoff generator[11]을 이용하여 도로 및 객체 데이터를 생성한 후 Oracle Database 10g와 JAVA로 구현하였다.

사용자의 궤적은 실 데이터를 구할 수 없어서 이동 객체 생성기를 이용하여 랜덤하게 궤적 데이터를 생성하였다. 이렇게 생성된 궤적 데이터를 이용하여 각각의 객체마다의 이동 성향을 추출하여 성능평가를 진행하였다. 실제 데이터가 아닌 생성된 데이터지만 궤적 데이터들의 특징을 추출하는 과정이 동일하고 이렇게 추출된 객체의 이동 성향이 객체의 성향과 일치함을 볼 수 있다. 이렇게 추출된 이동 성향을 가지고 경로를 생성한 결과도 이동 성향이 잘 반영된 경로가 생성됨을 확인 할 수 있었다.

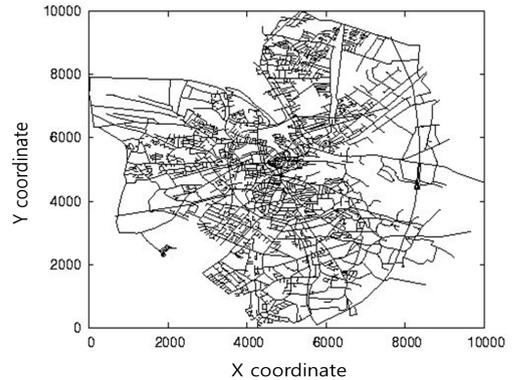


그림 6. Oldenburg 지역

표 2. 성능 평가 파라미터

구분	내용
도로 세그먼트의 수(개)	7,035
도로 세그먼트 교차점의 수(개)	6,105
객체의 수(개)	50,000
객체 속도(km/h)	0 - 120

사용자의 과거 경로 정보와 추천된 경로 50개를 분석

하여 사용자의 이동 성향을 추출한 결과는 [표 3]과 같다. 사용자는 다수의 이동 성향을 갖고 있었는데 그 중 시간 선호 사용자와 거리 선호 사용자의 속성을 추출해보았다. 추출된 사용자의 이동 성향을 가지고 경로를 생성한 결과 [그림 7][그림 8]과 같다.

[그림 7]은 시간 선호 사용자의 경로에 대한 거리, 시간에 대한 속성 값의 비율이다. 시간 선호 사용자의 이동 성향을 바탕으로 생성된 경로와 기존의 최소 시간 경로와 최단 거리 경로를 비교해보았다. 시간 선호 사용자의 이동 성향을 보면 거리속성은 0.17, 시간속성은 0.83만큼 선호하는 것을 볼 수 있다. 실제 그래프와 비교해본 결과 최소 시간 경로의 속성 값과 유사함을 볼 수 있다.

[그림 8]은 거리 선호 사용자의 경로에 대한 거리, 시간에 대한 속성 값의 비율이다. 거리 선호 사용자의 이동 성향을 바탕으로 생성된 경로와 기존의 최소 시간 경로와 최단 거리 경로를 비교해보았다. 거리 선호 사용자의 이동 성향을 보면 거리속성은 0.87, 시간속성은 0.13만큼 선호하는 것을 볼 수 있다. 실제 그래프와 비교해본 결과 최단 거리 경로의 속성 값과 유사함을 볼 수 있다. 이처럼 사용자의 이동 성향에 맞게 경로가 생성됨을 확인 할 수 있다.

표 3. 추출된 사용자 이동 성향 정보

사용자	거리	시간	혼잡도 한계치
시간 선호 사용자	0.17	0.83	4.0
거리 선호 사용자	0.87	0.13	3.5

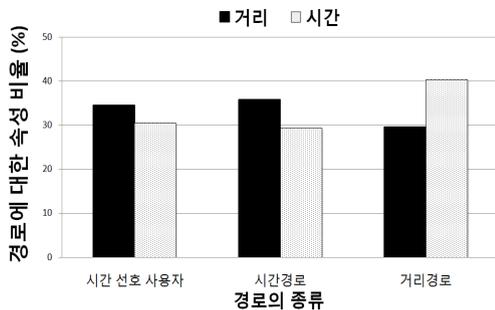


그림 7. 시간 선호 사용자와 다른 경로와의 속성 비교

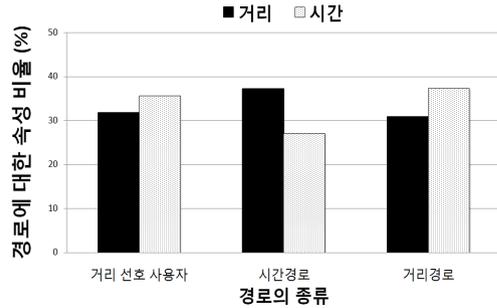


그림 8. 거리 선호 사용자와 다른 경로와의 속성 비교

[그림 9]와 [그림 10]은 사용자의 이동 성향에 맞게 생성된 경로를 나타낸다. 그림에서 굵은 검은색은 사용자의 혼잡도 한계치보다 높은 도로 세그먼트를 의미하며 이어진 실선은 생성된 경로를 의미한다. [그림 9](a)와 [그림 9](b)는 기존의 일반적인 경로이며 [그림 9](c)는 시간 선호 사용자를 위한 경로를 나타낸다. 시간 선호 사용자의 경로가 최단 거리 경로보다는 최소 시간 경로와 유사함을 볼 수 있다. 이는 사용자가 시간 속성을 더 많이 선호하기 때문이다.

기존의 최소 시간 경로나 최단 거리 경로는 혼잡도 한계치 이상의 도로 세그먼트를 이용하여 경로를 생성하였다. 하지만 제안하는 기법을 이용한 경로는 사용자가 경유하기 싫어하는 혼잡도 한계치 이상의 도로 세그먼트들을 제외한 것을 볼 수 있다. 이처럼 사용자의 혼잡도 한계치가 적용된 경로가 생성되었으며, 시간과 거리의 이동 성향도 반영된 경로가 생성됨을 확인 할 수 있다.

[그림 10]은 거리 선호 사용자를 위해 생성된 경로를 나타낸 것이다. 다른 경로와 비교한 결과 거리 속성을 선호하기 때문에 최단 거리 경로와 유사한 경로가 생성됨을 볼 수 있다. 또한, 혼잡도 한계치 이상의 도로 세그먼트들을 제외하고 경로가 생성됨을 볼 수 있다. 이처럼 제안하는 기법을 통해 사용자의 이동성향이 고려된 경로가 생성됨을 확인 할 수 있었다.

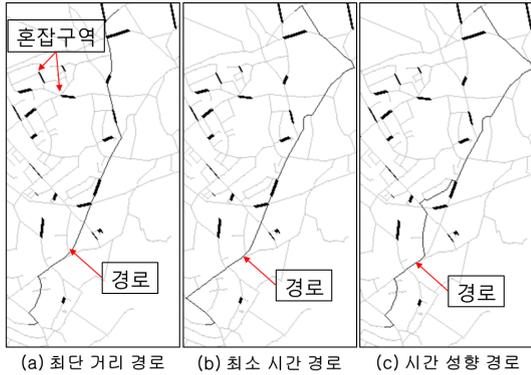


그림 9. 시간 선호 사용자와 다른 경로의 비교

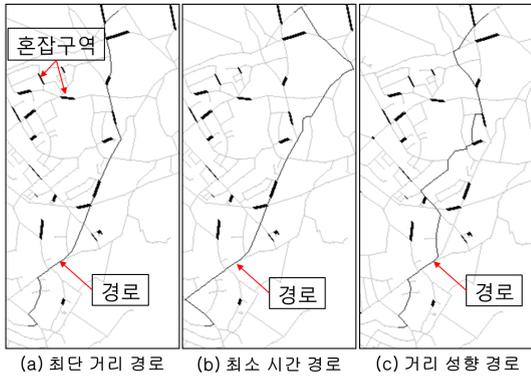


그림 10. 거리 선호 사용자와 다른 경로의 비교

## V. 결론

본 논문에서는 추가적인 정보를 입력을 수행하지 않고 사용자의 과거 경로 정보 분석을 통해 이동 성향을 추출하고 이동 성향에 맞는 최적의 경로를 생성하는 기법을 제안하였다. 제안하는 기법의 우수성을 입증하기 위해 사용자 이동 성향을 적용시킨 경로를 생성한 결과 사용자의 이동 성향에 일치하는 경로가 생성됨을 확인할 수 있었다. 향후 연구로는 경로 생성 비용을 감소시키기 위해 도로 세그먼트들을 군집화하는 기법을 연구할 예정이며 사용자와 다른 사용자들의 이동 성향을 이용하여 유사도를 측정하고 유사한 이동 성향을 갖는 사용자에게 경로를 추천 받는 기법을 제안할 예정이다.

## 참고 문헌

- [1] 장수민, 황동교, 강수, 김은주, 박준호, 장기훈, 유재수, “시각 장애인을 위한 네비게이션 시스템 설계 및 구현”, 한국콘텐츠학회논문지, 제12권, 제1호, pp.38-47, 2012.
- [2] B. Ziebart, A. Maas, A. Dey, and J. Bagnell, “Navigate like a Cabbie: Probabilistic Reasoning from Observed Context-aware Behavior,” Proc. International Conference on Ubiquitous Computing, pp.322-331, 2008.
- [3] S. Nadi and M. R. Delavar, “Multi-criteria, Personalized Route Planning using Quantifier-guided Ordered Weighted Averaging Operators,” International Journal of Applied Earth Observation and Geoinformation, Vol.13, No.3, pp.322-335, 2011.
- [4] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang, “T-drive: Driving Directions based on Taxi Trajectories,” Proc. ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, pp.99-108, 2010.
- [5] Z. Chen, H. T. Shen, and X. Zhou, “Discovering Popular Routes from Trajectories,” Proc. International Conference on Data Engineering, pp.900-911, 2011.
- [6] H. Gonzalez, J. Han, X. Li, M. Myslinska, and J. P. Sondag, “Adaptive Fastest Path Computation on a Road Network : a Traffic Mining Approach,” Proc. International Conference on Very Large Databases, pp.794-805, 2007.
- [7] K. F. Richter, “Adaptable Path Planning in Regionalized Environments,” Proc. International Conference on Spatial Information Theory, pp.453-470, 2009.
- [8] Y. Xu and J. Wang, “Optimal Path Solution of

Urban Traffic Road,” Proc. International Conference on Natural Computation, pp.799-802, 2011.

[9] S. Wang and Y. Xu, “Improvement of Priority Computation in Exemplar-Based Image Inpainting Base on D-S Evidence Theory and TV Model,” Proc. International Congress on Image and Signal processing, pp.1-3, 2009.

[10] P. E. Hart, N. J. Nilsson, and B. Raphael, “A Formal Basis for the Heuristic Determination of Minimum Cost Paths,” IEEE Transactions on Systems Science and Cybernetics, Vol.4, No.2, pp.100-107, 1968.

[11] T. Brinkhoff, “A Framework for Generating Network-Based Moving Objects,” GeoInformatica, Vol.6, No.2, pp.153-180, 2002

저 자 소 개

황 동 교(Donggyo Hwang) 준회원



- 2011년 2월 : 충북대학교 정보통신공학과(공학사)
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 석사과정

<관심분야> : 무선 센서 네트워크, 데이터베이스 시스템, 클라우드 컴퓨팅, 위치 기반 서비스 등

박 혁(Hyuk Park) 준회원



- 2011년 2월 : 충북대학교 정보통신공학과(공학사)
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 석사과정

<관심분야> : 무선 센서 네트워크, 위치 기반 서비스, 데이터베이스 시스템, 위치 기반 서비스 등

김 동 주(Dongjoo Kim) 준회원



- 2011년 2월 : 충북대학교 정보통신공학과(공학사)
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 석사과정

<관심분야> : 데이터베이스 시스템, 무선 센서 네트워크, Cloud, 분산 시스템 등

이 하(He Li) 정회원



- 2006년 7월 : 운남대학교 컴퓨터공학과(공학사)
- 2010년 8월 : 충북대학교 정보통신공학과(공학석사)
- 2010년 8월 ~ 현재 : 충북대학교 정보통신공학과 박사과정

<관심분야> : 데이터베이스 시스템, 모바일 소셜 네트워크 등

박 준 호(Junho Park) 정회원



- 2008년 2월 : 충북대학교 정보통신공학과(공학사)
- 2010년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2010년 3월 ~ 현재 : 충북대학교 정보통신공학과 박사과정

<관심분야> : 무선 센서 네트워크, 데이터베이스 시스템, RFID, 차세대 웹, LMS/LCMS, 바이오인포매틱스 등

박 용 훈(Yong-Hoon Park) 정회원



- 2005년 : 호원대학교 정보통신공학과 및 건축공학과(공학사)
- 2007년 : 충북대학교 정보통신공학과(공학석사)
- 2011년 : 충북대학교 정보통신공학과(공학박사)

▪ 2011년 3월 ~ 현재 : 충북대학교 Postdoc

<관심분야> : 데이터베이스 시스템, 정보검색, 시공간  
데이터베이스, 센서 네트워크 및 RFID

북 경 수(Kyoung-Soo Bok)

정회원



- 1998년 2월 : 충북대학교 수학과 (이학사)
- 2000년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2005년 2월 : 충북대학교 정보통신공학과(공학박사)

- 2005년 3월 ~ 2008년 2월 : 한국과학기술원 전산학과 Postdoc
- 2008년 3월 ~ 2011년 2월 : (주)가인정보기술 연구소
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 초빙부교수

<관심분야> : 데이터베이스 시스템, 자료저장 시스템, 위치기반서비스, 모바일 P2P 네트워크, 센서네트워크 및 RFID 등

유 재 수(Jaesoo Yoo)

종신회원



- 1989년 2월 : 전북대학교컴퓨터공학과(공학사)
- 1991년 2월 : 한국과학기술원 전산학과(공학석사)
- 1995년 2월 : 한국과학기술원 전산학과(공학박사)

- 1995년 3월 ~ 1996년 8월 : 목포대학교 전산통계학과 (전임강사)
- 1996년 8월 ~ 현재 : 충북대학교 정보통신공학부 및 충북BIT연구중심대학교육성사업단 교수

<관심분야> : 데이터베이스시스템, 정보검색, 센서네트워크 및 RFID, 멀티미디어데이터베이스, 분산객체컴퓨팅, 바이오 인포매틱스 등