

# 알맵 지도 인터페이스를 이용한 궤적 데이터의 시간적 표현 방법<sup>1)</sup>

이정훈, 홍영신  
제주대학교 전산통계학과  
e-mail:{jhlee, yshong}@cheju.ac.kr

## A sequential presentation method for trajectory data using Almap interface

Junghoon Lee, Youngshin Hong  
Dept of Computer Science and Statistics, Cheju National University

### 요 약

본 논문에서는 막대한 양의 위치 정보가 축적되는 차량 텔레매틱스 시스템을 대상으로 이동 이력 데이터에 대한 효과적인 분석을 위하여 이동 객체들의 궤적과 위치 변화를 시간적인 흐름에 따라 디지털 맵에 표현하는 인터페이스를 설계하고 구현하였다. 분석기 모듈은 쓰레드로 구현되어 윈도우즈 운영체제의 쓰레드 제어함수에 의해 분석 모듈도 같이 수행이 제어될 수 있으며 상용 디지털 맵인 알맵에 기반하여 이의 API에 따라 지도 인터페이스를 구현하였다. 또한 도로상에서의 분석을 위해 도로 네트워크 상에서의 표현도 구현하였다. 본 논문에서 구현된 분석 인터페이스의 구조는 쓰레드, 디지털 맵 등에 대한 요소들을 적절히 결합하여 새로운 Add-in 분석 기능을 추가할 수 있도록 한다.

### 1. 서론

텔레매틱스 기술은 무선 통신기술과 정보기술이 결합하여 다양한 위치기반 서비스를 가능하게 한다. 더욱이 텔레매틱스 장치들이 GPS (Global Positioning System) 수신기를 장착하게 됨에 따라 각 이동객체들이 자신의 위치를 정확하게 파악할 수 있을 뿐 아니라 무선 통신 인터페이스를 통하여 자신의 현재 위치를 중앙의 서버에게 보고할 수도 있다. 이 데이터는 객체에 대한 실시간 트래킹, 근거리 목적 객체의 검색 등 다양한 목적으로 사용될 수 있으며 이 데이터들이 축적된 이동 이력 데이터는 객체의 이동 패턴 분석이나 궤적 추적 등과 같은 사후 분석 및 이를 기반으로 한 미래 예측 등에 기반한 응용에서 사용될 수 있다[1].

현재 텔레매틱스는 차량 텔레매틱스, 헬tm 텔레매틱스 등 응용분야를 다양하게 확대해가고 있으며 이동객체들이 자신의 위치를 주기적으로 중앙서버에게 보고한다. 이동객체들은 환자, 노인, 차량 등 다양한 객체들이 포함되며 일반적으로 이들의 보고 주기는 객체의 속도들에 의해 결정된다. 예를 들어 사람이 이동하는 경우의 보고 주기는 5분에서 10분 정도로도 충분한 반면 차량의 경우에는 최소한 1분의 보고 주기를 가져야 한다. 결국 차량의 경우 짧은 보

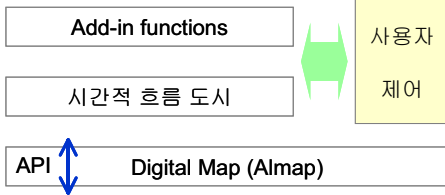
고 주기 때문에 축적되는 데이터의 양이 막대해지게 되며 이를 효율적으로 분석하기 위해서는 그래픽에 기반한 인터페이스가 필수적이다. 더욱이 차량 객체의 경우에는 도로를 통해서만 이동할 수 있으므로 디지털 맵에 기반한 사용자 인터페이스를 제공하여야 한다. 또한, 모듈화된 구조에 의해 새로운 분석기 기능을 추가할 수 있어야 한다. 본 논문에서는 차량 텔레매틱스의 이동이력 데이터를 이용하여 객체들의 이동 궤적과 이동 패턴을 분석하기 위한 그래픽 인터페이스를 설계하고 구현한다[2]. 이동이력 데이터는 현재 운영 중인 제주 택시 텔레매틱스 시스템에서 축적된 데이터를 기반으로 하는데 이 시스템은 1분마다 각 택시로부터 위치를 받아 고객으로부터의 호출이 왔을 때 호출 지점에서 가장 가까운 택시를 배차한다[3].

### 2. 시스템의 설계

이동이력 데이터의 분석 시스템은 <그림 1>에서 보는 바와 같이 우선적으로 위치 이력 데이터를 디지털 맵에 표시할 수 있어야 하며 이 과정에서 시간적인 흐름이 표현되어야 한다. 사용자 혹은 분석기가 설정한 일정 간격마다 각 객체의 위치 변화가 나타나야 하며 각 객체의 현재 상태가 표현될 수 있어야 한다. 예를 들어 각 객체가 택시인 경우는 택시에 승객이 탑승했는지, 공차인지, 배차를 받은 상태인지, 휴식 상태인지 등에 대한 정보가 표현될 수 있다. 또 시간적은 흐름은 사용자 입력 메뉴에 의해 제어될

1) 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-C1090-0801-0040)

수 있어야 하는데 이는 사용자가 제적 이동을 잠시 중지 시키거나 추후 재개할 수 있어야 한다. 이를 위해서는 쓰레드로 추적을 구현하여 쓰레드에 대한 제어 기능을 이용하는 것이 유용하다.



위치 이력 데이터베이스

<그림 1> 위치이력 분석 시스템의 구조

디지털 맵은 분석의 목적에 따라 요구사항이 달라질 수 있는데 예를 들면 도로 상에서의 패턴이나 맵 매칭에 의한 링크의 분석 등의 기능이 필요하다면 노드와 링크로 이루어진 도로 네트워크가 필요하다. 여기서 노드는 교차로를 나타내며 링크는 두 교차로간의 길에 대한 정보를 나타낸다. 반면 지형이나 지명 혹은 관심 지역에 대한 패턴이 필요하다면 상용 디지털 맵을 채택하는 것이 바람직하며 구글 맵, 알맵 등과 같은 GIS 서비스 제공자들은 지도에 대한 API (Application Program Interface)를 제공하여 지도 정보, 지도에 대한 처리, 지도를 기반으로 한 새로운 응용의 개발을 지원하고 있다[4]. 이 경우 줌인, 줌아웃, 팬, 거리 재기 등 기본적인 지도 함수를 이력 분석에 그대로 사용할 수 있는 장점이 있어서 보다 고수준의 분석을 수행할 수 있다.

### 3. 분석기 인터페이스의 구현

본 장에서는 먼저 상용 맵 중 하나인 알맵 지도와 이 인터페이스를 이용하여 로그 파일에 포함된 각 객체의 이동 이력을 지도상에 표시하고 이의 상태를 표현하며 지도 상에서 관심 지역에 따르는 객체의 이동 패턴과 다른 객체와의 관련성을 분석할 수 있도록 한다. 이를 위해 먼저 각 객체의 속성을 정의하면 <표 1>과 같으며 각 객체들은 그 개수가 자주 변경이 되므로 연결 리스트로 구성한다. 객체의 속성에는 현재 위치의 경위도 좌표, 상태, 식별자 등이 기본적으로 포함되며 색을 구별하기 위한 필드와 역할 등이 예약되어 사용된다.

<표 1> 이동 객체를 위한 속성 자료구조

```

struct t_movement {
    int      tid;
    long    color;
    int     state;
    double  x, y;
    int     role;
    struct t_movement *next;
} *Move=NULL;
    
```

알맵은 기본적으로 플러그인 형태로 사용자정의 메뉴의 확장을 지원하며 이는 DLL (Dynamic Link Library) 형태의 응용을 개발하고 이 라이브러리 함수들을 구현하도록 한다. 응용의 개발을 지원하기 위해 개발 템플릿을 배포하고 있으며 사전에 정의된 함수에 개발자가 자신의 프로그램 코드를 개발할 수 있다. 먼저 almap\_auth 라는 스켈리톤 함수에는 <표 2>에서 보는 바와 같이 응용에서 새로이 추가할 메뉴들의 이름이 순서적으로 정의된다. 본 시스템에서는 Initialize, Start, Suspend, Resume, End 등 5 개의 새로운 플러그-인 함수들이 추가된 것을 보이고 있다.

<표 2> 새로운 메뉴 추가를 위한 스켈리톤 함수 정의

```

char* almap_auth () {
    return "dll,Plug-in Name, ID, ***, 5, Initialize,
        Start, Suspend,Resume, End";
}
    
```

새로운 메뉴가 추가되면 이에 대한 처리 함수가 정의 되어야 하는데 이는 ShowSDKFORM 스켈리톤 함수에 의해 정의되고 <표 3>은 그 구현을 보인다. 추가된 메뉴들은 순서대로 nMenuIdx 인자로 매핑되며 각 값에 따라 처리 코드를 구현하면 된다. 예를 들어 Initialize를 선택한 경우 nMenuIdx는 1이 되고 결국 InitDisplay 서브루틴을 호출한다. <표 3>에서 Start 메뉴를 선택하면 새로운 쓰레드를 생성하고 제적에 대한 추적 기능이 자동적으로 시작되도록 한다. 반면 Suspend, Resume, Stop의 경우에는 Windows API의 쓰레드에 대한 정지, 재개, 종료 함수를 호출하여 쓰레드의 수행을 제어한다.

<표 3> 메뉴의 처리

```

void ShowSDKForm (... long nMenuIdx, ...) {
    if (nMenuIdx == 1) {
        InitDisplay();
    }
    else if (nMenuIdx == 2) {
        dThread = CreateThread(0, 500,
            (LPTHREAD_START_ROUTINE) DisplayProc,
            NULL, 0, &dwld);
    }
    else if (nMenuIdx == 3) {
        SuspendThread(dThread);
    }
    else if (nMenuIdx == 4) {
        ResumeThread(dThread);
    }
    else if (nMenuIdx == 5) {
        fclose(tracefp);
        Move = NULL;
        TerminateThread(dThread, 0x11);
    }
}
    
```

생성된 쓰레드는 메인 프로그램의 쓰레드와 독립적으로 제어의 흐름을 갖게 되며 계속하여 이동이력 데이터를 읽어들이며 각 객체의 위치를 갱신하는데 이는 <표 1>에서 보인 속성에서 x, y 값을 변경함으로써 수행된다. 이 과정에서 알맵 화면에 각 객체의 위치 변화를 반영해주기 위해서는 일정시간 혹은 일정 좌표이동마다 Windows API의 InvalidateRect와 같은 화면 재도시 함수를 호출하여야 하는데 이는 플러그인에서 호출할 수 없으므로 알맵에서는 자신의 API를 제공한다. <표 4>에서 보는 바와 같이 Almap(3, NULL, NULL)에서 3은 화면을 재도시하라는 명령에 해당한다.

<표 4> 쓰레드 루틴과 화면 재도시

```
void DisplayProc(DWORD * ThreadArg) {
    while (1) {
        res = GetTaxiMovement(tracefp, s1, s2, &tid,
            &x, &y, &dir, &speed, &status,
            &link, &pos);
        Taxi = GetMyTaxi(tid);
        Taxi->x = x; Taxi->y = y; Taxi->dir = dir;
        Taxi->status = status;
        if (i == 50) {
            AlMapExec(3, NULL, NULL);
            Sleep(1000); i =0;
        }
        i++;
    }
}
```

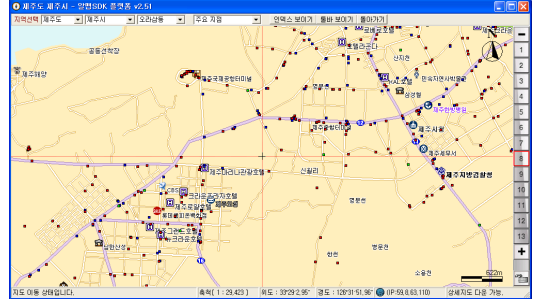
화면을 재도시하는 함수는 이에 해당하는 메시지를 응용의 이벤트 큐에 넣으며 이에 의해 WM\_PAINT 사건이 발생한다. 알맵은 자신의 기능을 수행한 후 플러그-인 부분의 AfterDrawMap 함수 스켈리톤을 통해 응용이 자신만의 화면 표시 함수를 구현할 수 있도록 한다. 이 부분은 각 객체에 대한 연결 리스트를 순회하면서 각 객체의 위치와 상태를 표시하는데 알맵 함수 16번은 알맵에서 사용하는 WGS 좌표와 화면좌표계의 변환을 수행한다.

<표 5> 화면의 재도시 함수

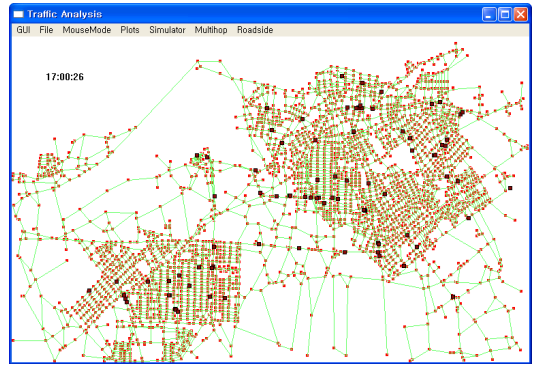
```
void AfterDrawMap (...) {
    SDKDATA_TRANSVIEWPOINT_IN GpsPoint;
    SDKDATA_TRANSVIEWPOINT_OUT MapPoint;

    t = Move;
    while (t != NULL) {
        GpsPoint.dLon = t->x;
        GpsPoint.dLat = t->y;
        AlMapExec(16, &GpsPoint, &MapPoint);
        Rectangle(HDC MapDC, MapPoint.nX,
            MapPoint.nY, MapPoint.nX+5, MapPoint.nY+5);
        t = t->next;
    }
}
```

결국, 이를 수행하면 <그림 2>와 같이 알맵 지도 상에서 각 객체의 위치들이 추적되어 분석될 수 있으며 도로 네트워크 상에 표현하면 <그림 3>과 같이 되어 다양한 객체들의 이동 패턴 분석과 속도 분석이 가능하다.



<그림 2> 알맵 상에서의 객체 추적 기능



<그림 3> 도로 네트워크 상에서의 객체 추적 기능

4. 결론

본 논문에서는 이동 이력 데이터에 대한 효과적인 분석을 위하여 이동 객체들의 궤적과 위치 변화를 시간적인 흐름에 따라 알맵과 도로 네트워크에 표현하는 분석 시스템을 설계하고 구현하였다. 알맵의 API에 기반을 둔 구현과 쓰레드 기반 추적에 의하여 시간적 재생이 제어가 가능하게 되었으며 새로운 Add-in 분석 기능을 추가할 수 있는 기반을 마련하였다.

참고문헌

[1] S. Kim, et al. "Path prediction of moving objects on road networks through analyzing past trajectories," LNCS4693, Sep. 2007, pp. 379-389.  
 [2] J. Lee, G., Park, H., Kim, Y., Yang, P., Kim, P., and S. Kim, "A telematics service system based on the Linux cluster," LNCS, Vol. 4490, 2007, pp. 660-667.  
 [3] Z. Liao, "Real-time taxi dispatching using global positioning systems," CACM, Vol. 46, 2003, pp. 81-83.  
 [4] Eastsoft, <http://www.almap.co.kr>