
모바일 GIS 환경에서 등고선 정보의 효율적인 전송 기법

최진오*

An Efficient Method for Transmission of Contour Information in Mobile GIS Environments

Jin-Oh Choi*

이 논문은 2007학년도 부산외국어대학교 학술연구 조성비에 의하여 연구되었음

요 약

모바일 GIS 환경에서 등고선을 표현하기 위해서는 지형도와는 별도로 서버로부터 해당 수치 데이터를 전송받아야 한다. 이때 전체 데이터를 전송받는 것은 모바일 환경의 특성상 받아들일 수 없다. 그리고 대표 지점들의 값을 전송받아 보간법으로 중간 데이터를 생성하는 기법은 계산 오버헤드와 등치선의 정확도 문제가 있다. 이 논문에서는 고도정보를 이용하여 생성하는 등고선 정보를 모바일 클라이언트로 효율적으로 전송하기 위한 자료구조와 알고리즘을 소개하고 구현 결과를 보인다.

ABSTRACT

For expression of contour lines in mobile GIS environments, the client needs to receive the numerical data with the topographical map from a server. At this point, a client can't get the entire raw data because of the mobile characteristics. The approach to get representative points and to make isogram by interpolation methods, has some problems. The approach requires huge computing overhead at the client and doesn't guarantee the correctness of the isogram. In this paper, a data structure, algorithm and implementation results for efficient transmission of contour information to a client which is constructed from elevation information at a server, are proposed.

키워드

Elevation Model, Isogram, Mobile GIS, Mobile Bandwidth

I. 서 론

이동 단말기에 등고선과 같은 등치선 정보를 출력하기 위해서는 자체 저장된 데이터베이스로부터 직접 지도를 생성하든지 실시간으로 서버로부터 관심 지역의 데이터를 전송받아야 한다. 고도 정보의 경우, 자체 저장된 데이터를 이용하는 접근 방법을 위해서는 클라이언트가 TIN(Triangulated Irregular Networks) 또는 DEM(Digital Elevation Models) 형태의 고도정보를 대축척 형태로 포함하고 있어야 한다. 이 방법은 이동 단말기의 저장 용량 문제로 받아들일 수 없다. 또한 등치선 생성에 많은 계산 량을 필요로 하기 때문에 모바일 단말기에서 실시간으로 처리하기 어려운 단점도 존재하다.

서버에서 실시간으로 클라이언트 관심 지역의 지도를 생성하여 전송하는 방법은 래스터(raster) 방식과 벡터(vector) 방식으로 처리할 수 있다. 래스터 방식의 경우 클라이언트에서 전송 받은 지도로부터 공간 분석 등을 위한 추가 정보 추출이 어렵다는 문제점이 있다.

벡터 방식으로 처리할 경우에도 서버에서 TIN이나 DEM 데이터로부터 고도 정보를 모바일 클라이언트로 직접 전송하여 클라이언트에서 등고선을 생성하게 하는 접근 방법은 받아들이기 어렵다. 모바일 단말기에서 등고선을 생성하는 오버헤드가 아주 큰 단점이 있다. 즉, 컴퓨팅 자원이 부족한 모바일 클라이언트에서 많은 시간이 소요되는 등고선 생성 알고리즘을 실행하여야 하기 때문이다.

벡터 방식에서 서버가 등고선을 미리 생성해서 벡터 데이터 형식으로 클라이언트에 전송하는 방법도 문제점이 있다. 전송할 데이터양이 아주 크기 때문에 전송 지연을 유발할 뿐더러 보다 정밀한 등고선을 전송하고자 한다면 데이터양이 급격히 증가한다.

따라서, 이 논문에서는 이동 장치에 등고선 출력[1]을 위하여, 벡터 방식을 기반으로 서버에서 모바일 클라이언트로 등고선 정보를 효율적으로 전달하는 방법에 초점을 맞춘다. 이 논문에서 설계하고 구현해 보이는 내용은 클라이언트로 서버가 등고선 정보를 최소 비용으로 전송하는 기법이다.

이 논문의 구성은 다음과 같다. 2장에서 관련연구에 대하여 설명하고 3장에서 고도 정보 전송을 위한 설계

내용을 보이며 4장에서 구현 결과를 제시한다. 결론은 5장에서 맺는다.

II. 관련 연구

고도 정보는 벡터 형식의 TIN 모델로 표현하거나 일정 그리드에 동일한 고도 값을 표시하는 래스터 형식의 DEM 방식으로 표현한다[2]. TIN 모델은 좌표와 함께 고도 정보를 저장하고 있어서 DEM 모델에 비해 유연성 측면에서 장점을 지닌다. 반면 데이터의 볼륨이 상대적으로 크고 지형이 복잡한 곳은 고도 정보가 중복된다. DEM 모델은 TIN에 비해 처리가 단순하고 전체적 지형 파악이 용이하다. 반면 평지에서 동일 데이터의 중복 문제가 있다. 이러한 고도 정보는 지형 파악을 위한 등고선 생성뿐만 아니라 지형 정보 또는 다양한 주제도와 함께 고도에 따른 상관관계(correlation) 분석에 주로 이용된다. 그리고 드래핑(draping) 기법에 의해 3차원 지도 생성에도 사용된다.

최근까지 고도 정보를 다루는 연구는 크게 3 분야로 분류된다. 첫째, 효율적인 DEM 또는 TIN 모델의 구축 기법과 DEM과 TIN의 효율적인 상호 변환기법[3], 둘째, 고도 정보와 기존 주제도와와의 상관관계 파악 응용[4], 셋째, 새로운 보간법(interpolation) 알고리즘 개발[5] 분야이다.

이 논문에서는 그동안 다루지 않았던 모바일 단말기로 고도정보를 전송하는 기법에 대하여 고찰하고자 한다. 클라이언트에서 등고선을 출력하기 위해서는 대용량 고도정보를 그대로 전송하는 것은 많은 대역폭이 소요되기 때문에 전송 지연이 발생한다. 또한 클라이언트에서 고도정보로부터 등고선을 생성하는 방법도 모바일 단말기의 리소스 제약으로 처리 지연이 발생하므로 받아들일 수 없다. 따라서 서버에서 고도 정보를 모바일 단말기로 효율적으로 전송하는 기법이 필요하다.

III. 고도 정보 전송을 위한 시스템 설계

이 논문에서 제안하는 등고선 데이터 구조는 그림 1과 같다. 이 구조를 FCDS(Formatted Contour Data

Structure)라 부르고 이 구조로 생성된 등고선 지도를 FCM(Formatted Contour Map)이라 부른다.

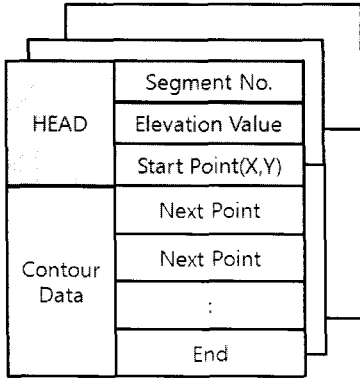


그림 1. FCDS 프레임 형식
Fig. 1 FCDS Frame Format

그림 1에서 하나의 데이터 구조는 하나의 등고선 세그먼트(segment)를 표시한다. 이러한 세그먼트들이 모여 FCM을 구성한다. 헤드는 세그먼트들을 구분하는 고유 번호, 고도값, 그리고 시작 포인트 좌표로 구성된다. 여기서 시작 포인트 좌표는 모바일 클라이언트 단말기의 물리 좌표값이다. 물리 좌표값을 사용하는 이유는 데이터 크기를 줄이기 위함이다.

표 1. FCDS 필드 크기
Table. 1 FCDS Field Size

Segment No.	Elevation Value	Start Point	Next Point	End
4byte	4byte	2byte	4bit	4bit

등고선 데이터의 'Next Point' 자료 구조는 4bit로 구성된다. 초기 시작 포인트(Start Point)로부터 인접한 8방향의 픽셀(pixel) 중 하나를 표시한다. 마지막 4bit는 리스트의 종료 표시로 사용한다. 이러한 방식의 등고선 데이터 구성 예를 그림 2에서 보았다.

서버에서 FCM을 생성하는 과정은 그림 3과 같다. 그림 3에서 서버 처리기는 고도 정보로부터 클라이언트가 요구하는 질의 영역의 고도 정보를 추출한다. 그 후 FCDS 프레임들을 구축한다. FCDS 프레임은 등고선 세그먼트마다 하나씩 생성 된다. 이 프레임들이 모여서 FCM을 구성하게 된다.

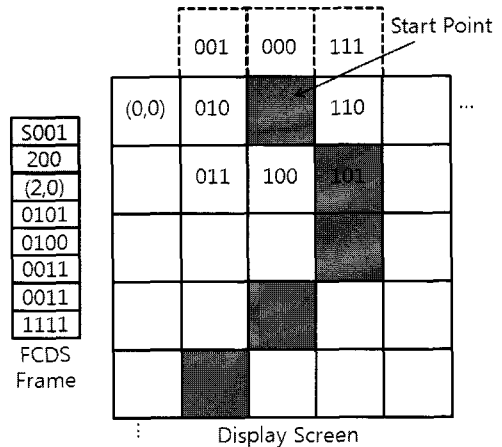


그림 2. FCDS와 Client 출력 Example
Fig. 2 Example of FCDS and Client Display

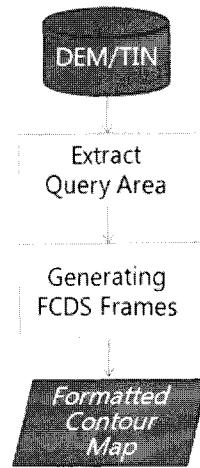


그림 3. FCM 생성 단계
Fig. 3 FCM Generation Steps

FCDS의 데이터 필드 크기는 표 1과 같다. Segment 번호는 코드화하지 않고 문자열로 표시한다. 값의 범위는 '0000'에서 '9999'이다. 이는 모바일 단말기 출력 화면의 크기를 고려하면 충분한 수치이다. Elevation값도 '9999'까지 표시할 수 있다. Start Point는 모바일 단말기의 화면 좌표이기 때문에 x, y 좌표 각각 1byte로 표현할 수 있으므로 총 2byte로 구성된다.

FCDS 프레임을 구축하는 과정은 그림 4와 같다. 생성된 벡터 등고선 리스트는 세그먼트별로 FCDS 프레임으

로 변환된다. 이 과정에서 등고선을 요청한 클라이언트의 화면 해상도 값을 이용한다.

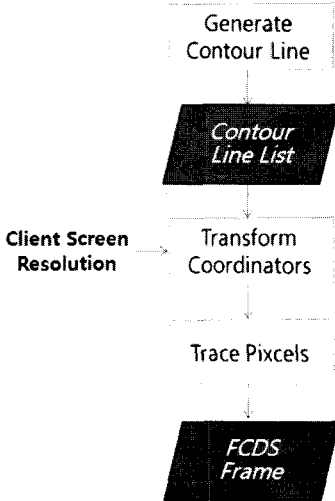


그림 4. FCDS 생성 단계
Fig. 4 FCDS Generation Steps

좌표 변환된 포인트 리스트는 서버에서 가상으로 라인을 그리고 걸치는 픽셀(pixel)을 찾는 픽셀 추적(tracing) 과정을 거쳐 FCDS 프레임을 구축하게 된다. 이 알고리즘은 그림 5에서 보였다.

```

FCM TracePixel(TransformedSegment[] s,
                ClientScreenResolution r){
    //make virtual canvas
    makeCanvas(r);

    //draw s segment
    for(s.length){
        startPoint = drawSegment(s[i]);
        //make FCDS Frame
        FFrame = makeFCDSFrame(startPoint);
        FCM.addFCDSFrame(FFrame);
    }
    return FCM;
}
  
```

그림 5. TracePixel 함수 알고리즘
Fig. 5 TracePixel Algorithm

이 알고리즘은 주어진 등고선 세그먼트 리스트와 등고선을 요청한 클라이언트의 화면 크기를 입력받

아 FCM을 만들어 반환한다. FCM을 만드는 과정은 간략히 다음과 같다. 먼저 makeCanvas()에서 가상의 클라이언트 화면을 만든다. 이때 입력 r로부터 클라이언트의 화면 해상도와 동일한 가상 캔버스를 만든다. 이는 클라이언트 상대 좌표 형태로 FCDS 프레임을 생성하기 위해서이다. 이어서 각 세그먼트에 대하여 반복하여 첫째, 가상 캔버스에 각 세그먼트를 그린다. 이때 절대 좌표로부터 가상 캔버스의 좌표로 좌표 변환에 수행한다. 둘째, makeFCDSFrame()의 호출에 의해 캔버스에 그려진 세그먼트의 startPoint로부터 픽셀 추적에 의한 'next point'값들을 구하고 해당 FCDS 프레임을 생성한다. 셋째, 생성된 FCDS 프레임을 FCM에 추가한다. 이는 FCM.addFCDSFrame() 호출에 의해 처리된다.

IV. 구현 실험

본 논문에서 설계 구현하고 실험한 환경은 다음과 같다. 서버는 Compaq Alphaserver DS10 기종의 True Unix 64이며, 서버 프로그램은 JAVA 2 SE로 구현하였다. 클라이언트는 Aroma WIPI-Emulator Ver 1.1.1.80에서 J2ME로 구현하였다.

설계 내용을 실험하기위해 서버와 클라이언트를 별도로 구현하였다. 서버는 고도 정보를 저장하고 있으며 클라이언트의 요청에 의하여 FCM을 완성하여 클라이언트로 전송한다. 서버의 구현은 Java 2 SE로 하였고 Compaq DS-10 기종(1Gbyte Main Memory)을 사용하였다. 클라이언트는 서버로 고도 정보를 요청하고 전송받은 FCM을 화면에 출력한다. 클라이언트는 Aroma WIPI Emulator V1.1.1.8을 이용하였다.

그림 6은 클라이언트에서 전송받은 FCM을 출력한 결과 화면이다. 그림6을 구성하는 데이터는 15개의 세그먼트로 이루어져 있고 562의 포인트로 구성된다. 따라서 서버에서 FCM을 구성하기 이전의 데이터 크기는 8byte/1point X 562point = 4496byte이다.

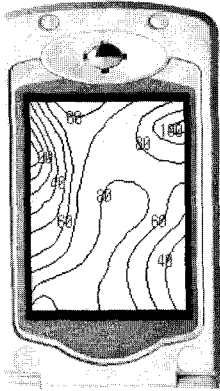


그림 6. 클라이언트 검색 결과 화면
Fig. 6 Client Output of Search Result

실험 결과 서버에서 FCM을 생성했을 때 각 FCDS 프레임의 크기는 표 2와 같았다. 표 2에서 각 프레임은 헤드 10byte(Segment No 4 byte, Elevation Value 4 byte, Start Point 2byte)와 END 4bit 크기를 포함하고 있다.

표 2. 생성된 FCDS 프레임의 크기
Table. 2 Generated FCDS Frame Size

S001	S002	S003	S004	S005	S006	S007	S008
21	33	47	59	68	80	23	124
S009	S010	S011	S012	S013	S014	S015	S016
103	107	99	89	107	82	63	44

따라서 서버에서 클라이언트로 전송해야 할 데이터의 크기는 총 1149byte로서 FCDS 프레임 처리 이전의 4496byte에 크기에 비해 약 74%의 데이터 전송량을 줄일 수 있음을 알 수 있다.

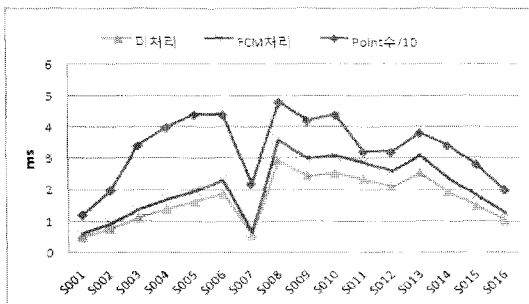


그림 7. Server에서의 처리 시간 비교
Fig. 7 Comparison of Processing Time at Server

그림 7은 각 세그먼트별 서버에서의 처리시간을 비교한 결과이다. FCM을 생성하는 오버헤드는 50point로 구성된 세그먼트에 대해서도 1ms를 넘지 않았다. 그리고 FCM 처리시간은 각 세그먼트를 구성하는 포인트의 수에 대체적으로 비례함을 알 수 있다.

실험 환경에서 서버에서 클라이언트로의 전송 시간은 미처리의 경우 세그먼트 당 1126.4ms가, FCM 처리의 경우 330.3ms가 각각 소요되었다. 약 73% 이상의 처리 효율 향상을 보였다.

V. 결 론

이 논문에서는 클라이언트의 등고선 정보 참조를 지원하기 위해 서버에서 고도정보를 효율적으로 전송하는 기법에 대하여 고찰하고 구현 및 실험 결과를 보였다. 이 논문에서 제안하고 구현 실험해 보인 기법은 서버에서 클라이언트에 출력될 등고선을 미리 완성하되, 한 등고선 세그먼트마다 클라이언트의 스크린 픽셀단위로 상대 좌표 표시방식으로 벡터화시키는 것이 주된 아이디어이다. 제안한 기법은 클라이언트로 전송하는 등고선 정보의 데이터 크기를 획기적으로 줄일 수 있으며, 클라이언트에서 등고선 출력을 위한 오버헤드를 최소화시키는 장점을 지닌다. 실험결과 서버에서의 처리 오버헤드는 수 ms로 미미한 반면 전송 데이터의 양을 줄임으로서 얻을 수 있는 전송 오버헤드는 73% 이상임을 알 수 있었다.

향후, 이 논문에서 고찰하고 구현한 결과를 바탕으로 클라이언트에서 FCDS 구조의 고도 정보로부터 다양한 부가 정보를 추출할 수 있는 전용 공간분석 알고리즘 개발이 뒤따라야 하겠다.

참고문헌

- [1] M. Price, "Creating Cool Contours Modeling - Glacial Terrain with ArcGIS", ArcUser, 2006.04
- [2] M. Kumler, "An Intensive Comparison of Triangulated Irregular Networks and Digital Elevation Models", Cartographica, Vol 31, 1994.
- [3] J. Garbrecht, "Digital Elevation Model Issues in Water

Resources Modeling”, ESRI User Conference, 1999.

- [4] 황재홍, 김상호, 류근호, “GIS 및 RS 데이터를 이용한 자연환경/생태계 정보시스템 설계 및 구현”, 한국지리정보학회지 4권3호, 2001
- [5] Z. Changqing, S. Wenzhong, W. Guangxia, C. Cheung, E. Dai, Y. Shea, “Estimation of Average DEM Accuracy Under Linear Interpolation Considering Random Error at the Nodes of TIN Model”, Int. Journal of Remote Sensing, Vol 26, No. 24, 2005
- [6] N. Jones, M. Kennard, A. Zundel, “Fast Algorithm for Generating Sorted Contour Strings”, int. Journal of Computers and Geosciences, Vol. 26, No. 7, 2000

저자소개



최진오(Jin-Oh Choi)

1991년 부산대학교 컴퓨터공학과
공학사

1995년 부산대학교 대학원
컴퓨터공학과 공학석사

2000년 부산대학교 대학원 컴퓨터공학과 공학박사
1998.3 ~ 2000.2 경동대학교 정보통신공학부 전임강사
2000.3 ~ 현재 부산외국어대학교 임베디드IT학과 부교수
※관심분야: 지리정보시스템, 모바일 GIS, 센서 네트워크