

# 모바일 지도 서비스에서 벡터와 래스터 기법의 비교 평가

최진오

부산외국어대학교 컴퓨터공학부

## Comparison and Evaluation of Vector and Raster Methods in Mobile Map Services

Jin-oh Choi\*

Faculty of Computer Engineering, Pusan University of Foreign Studies

E-mail : jochoi@taejo.pufs.ac.kr

### 요 약

모바일 GIS를 구현하는 방식은 휴대 단말기로 전송되는 지도 데이터의 형식에 따라 벡터(Vector) 기법과 래스터(Raster) 기법으로 나뉜다. 각 기법에 따라 모바일 지도 서비스를 위한 서버 및 클라이언트 구현 방법을 달리 요구한다. 또한, 각각이 서로 다른 측면에서 장단점을 지니고 있다. 본 논문은 이 두 접근 방법을 구현하고 실험을 통해 그 장단점을 비교한다. 전송 성능, 지도 품질 등 다양한 요소를 비교하고 평가한다.

### ABSTRACT

There are two approaches to construct mobile GIS, Vector and Raster methods, according to the map data transformation format from server to mobile client. Each method requires a different implementation architecture of server and client modules for mobile map services. And each have advantages and disadvantages at the different aspects. This thesis implements these two approaches, thus, compares the each merits, by experiments. They include the transmission performance, map quality, and so on.

### 키워드

mobile vector GIS, mobile raster GIS, generalization, data simplification, mobile phone

### 1. 서 론

현재, 모바일 전자 지도 서비스의 구현은 WAP (Wireless Application Protocol) 기반 방식과 JAVA 기반 방식으로 나눌 수 있다. 또한, 서비스 유형은 기존의 유선 인터넷 환경과 같이 래스터(raster) 방식과 벡터(vector) 방식으로 지원될 수 있다. 래스터 방식의 서비스는, 서버에서 검색한 전자지도를 이미지로 변환하고 이것을 WML(Wireless Markup Language) 문서에 포함시켜 클라이언트로 전송하거나 소켓을 통해 직접 전송하는 방식을 취한다. 전자의 경우 클라이언트에 WAP 브라우저가 필요하고, 후자의 경우 클라이언트에 이미지를 출력하는 프로그램이 필요하다. 그러나 공통점은, 서버에서 클라이언트로 이미지를 전송한다는 것이다.

벡터 방식의 서비스는, 서버에서 검색한 전자지도 데이터를 그대로 소켓을 통해 클라이언트로 전송하고, 클라이언트에서 수신한 공간 데이터를 J2ME(Java 2 Micro Edition) 등의 프로그램으로 지도를 직접 그리는 방식을 취한다.

그림 1은 각각의 지도 출력 예를 보인다. 그림 1(가)는 래스터 지도를, (나)는 벡터 지도를 클라이언트에서 출력한 구현 화면이다.

그림 1에서 보이는 단순한 클라이언트에서의 출력 화면 차이점 외에, 두 기법의 개략적인 비교는 다음과 같다. 첫째, 서버에서의 처리 방법 차이이다. 래스터 방식은 서버에서 모든 처리가 이루어져야 하며 클라이언트에서 출력될 최종 지도 화면을 이미지로 만들어야 한다. 반면, 벡터 방식에서는, 서버는 지도 데이터를 벡터 형식으로 클라이언트

로 전송만 하면 된다. 따라서 서버의 부하는 래스터 방식에서 더 많이 걸리게 된다.



(가) 래스터지도 (나) 벡터지도  
그림 1. 두 기법의 지도 출력 예

둘째, 클라이언트의 부하에서 두 기법에 차이가 있다. 래스터 방식은 서버에서 전송되는 데이터를 그대로 이미지 출력만 하면 되지만, 벡터 방식은 벡터 데이터를 일일이 클라이언트 화면에 그리는 작업이 필요하다. 심지어, 래스터 방식에서 WAP을 이용한다면 모바일 지도 서비스를 위한 별도의 클라이언트 모듈이 필요없다.

셋째, 서비스되는 지도의 출력 용통성에 차이가 있다. 래스터 방식은 클라이언트 디바이스(device)의 특성을 고려한 지도 출력이 어렵고, 표준화된 형식과 크기의 이미지를 생성해야 하는 제약이 따른다. 벡터 방식은 클라이언트에서 수신받은 벡터 데이터를 디바이스 특성에 맞게 변환 출력할 수 있는 유연성이 있다.

넷째, 서버에서 클라이언트로 전송되는 데이터의 크기에 차이가 있다. 래스터 방식은 이미지를 전송하기 때문에 항상 일정한 데이터 볼륨을 전송하게 된다. 반면, 벡터 방식은 서비스 되는 지역에 따라 상이한 벡터 데이터 볼륨을 제공한다. 따라서, 서버의 응답시간이 래스터 방식은 일정한 반면 벡터 방식은 가변적일 수 있다.

다섯째, 클라이언트에서의 지도 활용 가능성 정도가 서로 다르다. 래스터 방식은 이미지를 전송받기 때문에 공간분석(spatial analysis), 캐싱(caching) 등의 목적으로 지도 데이터를 응용할 가능성이 적다. 반면, 벡터 방식은 서버로부터 전송받은 벡터 데이터를 저장해 두고 네트워크 분석이나 필요에 따른 지도 변형, 캐싱 등의 목적으로 활용 및 재사용이 가능하다.

본 논문에서는 이러한 두 기법의 상이한 특성을 정확히 파악하기 위해 각각의 모바일 지도 서비스 시스템을 설계 구현하고 다양한 방법으로 비교 평가하기 위한 실험을 전개한다.

본 논문의 구성은 관련 연구를 2장에서 설명하고, 3장에서 두 접근 방법의 시스템 구현 설계 개요를 소개하며, 4장에서 이들 두 기법의 장단점을 실험하고 평가한다. 결론은 5장에서 맺는다.

## II. 관련 연구

모바일 전자 지도 서비스 구현을 위한 플랫폼은 WAP기반 방식과 JAVA기반 방식이 있다. 래스터 지도 서비스 구현은 두 기법으로 모두 가능하지만 벡터 지도 서비스는 JAVA기반 방식만이 지원 가능하다. WAP은 WWW 기반 응용 서비스의 대표로, 무선 단말기에 인터넷 콘텐츠를 제공하며 통용되는 애플리케이션이 제작 가능하도록 표준화 기술로서 개발된 것으로, 이를 기반으로 한 무선 인터넷의 발전에 크게 기여하였다[1][2]. JAVA기반 방식은 J2ME의 표준 발표로 가능성이 열렸으며, 확장성, 호환성, 융통성, 비연결성 등의 많은 장점으로 [3][4] 최근 각광받고 있다.

무선 인터넷을 통한 국내 상용 전자지도 서비스 동향은 대부분이 성능 문제 때문에 래스터 지도를 선택하고 있다[5]. 일부 업체에서 벡터 방식의 무선 지도 서비스를 시행하고 있으나 대부분 모바일 서비스 전용의 간소화된 GIS 데이터베이스에 의존하고 있는 실정이다. 국외의 경우도 벡터 방식의 모바일 지도 서비스는 일반화되지 못하고 있다[6][7].

본 논문에서는 두 기법의 장단점을 실험을 통해 비교 분석하고, 이를 바탕으로 벡터 방식의 지도 서비스에서 극복되어야 할 문제점을 고찰하고자 한다.

## III. 시스템 구현 설계

그림 2는 모바일 지도 생성기 구조이다[8]. 이 그림에서 'Raster Data Provider'와 'Vector Data Provider'가 각각 래스터 지도와 벡터 지도를 생성하는 모듈이다.

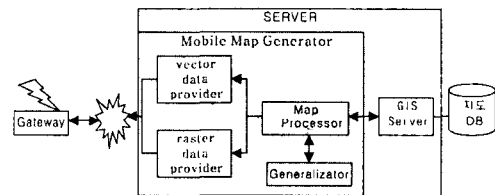


그림 2. Mobile Map Generator

그림 3은 'Raster Data Provider'의 내부 구조를 보인다. 그림 2의 'Map Processor'는 GIS 서버로부터 검색한 지도를 'Generalizator'를 통해 모바일 환경에서 출력 가능한 지도로 간소화한 후 'Raster Data Provider'로 제공한다. 그림 3에서 'Raster Data Provider'는 제공받은 공간 데이터로부터 '이미지 처리기'를 통해 이미지 파일을 생성한다. '페이지 생성기'는 이 이미지와 속성정보를 포함한 WML 문서를 생성한다.

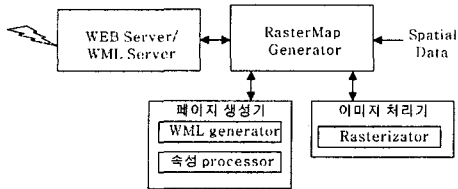


그림 3. Raster Data Provider 구조

그림 4는 'Vector Data Provider'의 내부 구조를 보이고 있다. 이는 'Raster Data Provider'에 비하여 간단하다. 넘겨받은 공간 데이터를 약속된 형식으로 소켓을 통하여 전송하기만 하면 되기 때문이다. '서버객체관리기'는 이후 서버 캐싱을 위해 들 수 있으며, 이는 벡터 방식의 장점 중 하나이다.

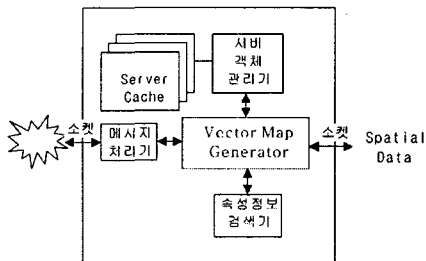


그림 4. Vector Data Provider 구조

그림 3과 4의 각각의 클라이언트 구조는 다음과 같다. 먼저, 래스터 방식에서 클라이언트에는 WAP 브라우저만이 필요하다. 그리고, 벡터 방식에서는 전송 받은 벡터데이터를 휴대 단말기에 직접 그리는 J2ME 모듈이 필요하다.

#### IV. 구현 실험

본 논문에서 설계 구현하고 실험한 환경은 다음과 같다.

- 서버 : Compaq Alphaserber DS10
- 지도데이터베이스 : Cybermap Server Ver 2.0
- Raster Data Provider : Servlet Web Server
- Vector Data Provider : JAVA 2 SE
- Raster Client : NETPLE WAP Browser Phone Emulator Ver 1.1
- Vector Client : SK-VM Phone Emulator Ver 1.1, J2ME
- 실제 휴대폰 테스트 : 삼성전자 SCH-X570

실험은 실제 지도를 사용하였으며, 도출된 수치는, 500m 영역 내에 포인트 개수가 500개 단위로 500개에서 2000개까지 각각 이내인 100개 표본 영역을 선택하여 평균값을 구한 것이다.

그림 5는 'Raster Data Provider'와 'Vector Data Provider'의 서버에서의 처리 소요 시간을 측정 비

교한 결과이다. 데이터 크기가 비교적 적은 지역은 큰 차이를 보이지 않지만 포인트 수가 증가할수록 래스터 방식에서 서버 처리 시간이 길어짐을 보인다. 이것은 데이터의 크기가 커 질수록 복잡한 이미지를 그려야 하는 오버헤드 때문이다.

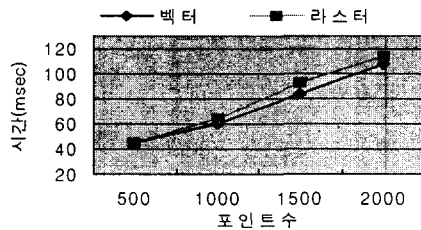


그림 5. 서버 처리시간 비교

그림 6은 서버에서 클라이언트로 데이터를 전송 시작하여 완료하는데 소요되는 시간을 측정하였다. 여기서, 포인트 수가 증가할수록 처리시간이 일정하게 증가하였다. 이는 전송되는 데이터의 크기와 연관이 있다. 래스터 방식은 WML 문서를 이미지와 함께 전송하며, 벡터 방식은 포인트 좌표를 정해진 형식으로 바로 전송한다. 이 실험에서 벡터 데이터의 크기가 상대적으로 적음을 알 수 있다.

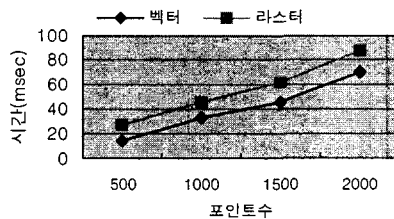


그림 6. 서버 전송시간 비교

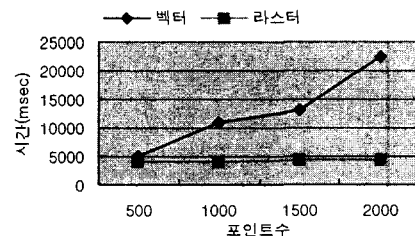


그림 7. 클라이언트 처리시간 비교

그림 7은 클라이언트가 서버에서 데이터를 전송 받기 시작하여 출력 완료하는데 소요되는 시간을 측정하고 비교한 결과이다. 여기서, 래스터 방식은 일정한 크기의 이미지를 전송받기 때문에 포인트 수에 관계없이 일정한 응답 속도를 보인다. 벡터 방식은 포인트 수가 증가함에 따라 클라이언트에

서의 처리 시간이 증가함을 알 수 있다. 이는 단말기 출력 화면에 직접 지도를 그리는 작업 때문이다.

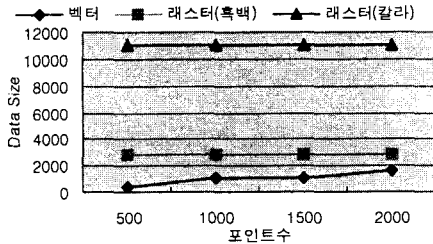


그림 8. 데이터 크기 비교

그림 8은 래스터와 벡터 방식이 각각 전송하는 데이터 크기를 측정 비교한 결과이다. 벡터 방식이 전송되는 데이터의 크기는 가장 적음을 알 수 있다. 래스터 방식은 흑백 이미지나 컬러 이미지나 포인트수에 관계없이 일정한 데이터 크기를 유지한다.

### V. 결론

본 논문은 모바일 환경에서 전자 지도 서비스를 지원하는 두 가지 기법, 래스터 기반, 벡터 기반 기법을 각각 구현 설계하고, 실험을 통하여 비교 분석한 결과를 보였다. 여기서 설계하여 보인 내용은, 래스터 지도를 생성하고 클라이언트로 전송하는 'Raster Data Provider'와 벡터 지도를 생성하고 클라이언트로 전송하는 'Vector Data Provider'이다.

실험 결과는 래스터 방식과 벡터 방식이 각각 특정 요소에 장점이 있음을 보였다. 래스터 방식은 일정한 응답 속도를 보장하며, 리소스가 부족한 클라이언트 휴대 단말기에 적합한 방법임을 알 수 있었다. 반면 벡터 방식은 클라이언트로 전송되는 데이터의 크기가 작기 때문에 전송 시간이 짧으며, 서버의 부하를 상대적으로 작게 준다.

본 논문에서 실험해 보이지 못한 부분은 벡터 방식이 캐싱, 공간 분석 등의 데이터 재사용 이득을 얻을 수 있다는 점이다. 이는 향후 연구에서 다룰 예정이다.

이 논문은 2002년도 부산외국어대학교 학술연구조성비에 의해 연구되었음

### 참고 문헌

- [1] 남기범, 이건명, 무선 웹 기술과 전망, 정보과학회지, 제 18권 6호, pp.32~38, 2000
- [2] WAP Forum Specifications, WAP June 2000 Conformance Release
- [3] Giguere, Java 2 Micro Edition, Wiley press, 2000
- [4] <http://java.sun.com/j2me>
- [5] <http://www.cybermap.co.kr>
- [6] <http://www.nttdocomo.com>
- [7] <http://www.viamichelin.com>
- [8] 김종민, 최진오, 모바일 GIS를 위한 공간 데이터 간소화 기법, 한국해양정보통신학회 2003년도 춘계종합학술대회, 제 7권 1호, pp.481~484, 2003