

모바일 지도 서비스를 위한 공간 객체 포맷의 설계 및 구현

문진용*

요약

기존의 유선 환경기반의 GIS 솔루션으로 무선 환경에서 서비스를 하기에는 제한된 통신 속도, 처리속도, 화면 사이즈 등의 한계점이 존재한다. 본 논문에서는 GIS 개발자들에 의해 개척되고 있는 최신의 기술인 모바일 지도 서비스에서 공간 데이터의 효율적 사용을 위한 공간 객체 표현, 저장 및 전송을 위한 파일 포맷인 MVF를 개발하였다. 본 논문에서는 먼저 MVF의 문법을 정의하고, API를 통한 MVF 생성을 위해 헤드섹션 생성모듈과 메인섹션 생성모듈을 구현하였다. 특히, 모바일 지도 서비스를 위해 개발된 MVF는 네트워크의 트래픽을 감소하기 위한 공간 데이터의 크기와 이용 효율을 위한 한글 처리에 대해 고려하였다.

Design and Implementation of Spatial Object Format for Providing Mobile Map Services

Jin-Yong Moon*

Abstract

Providing services in a wireless environment with existing wired-based GIS solutions have many limitations such as slow communication, processing rates, and screen size. Among the components, data is considered to be the most important factor in GIS. In this paper, I describe MVF, the most recent technology pioneered by GIS researchers. MVF is a format for efficient usage of spatial object in order to express, save and transform in mobile GIS. First, I describe the definition of MVF syntax, then the implementation of Header Section Generation Module and Main Section Generation Module through the API. Since the issues of network traffic and Hangul processing in mobile GIS are important, I especially consider development of MVF in terms of spatial data size.

Keywords : Mobile GIS, Geospatial, Vector Data

1. 서론

아이폰이나 구글폰은 터치스크린과 햅틱 인터페이스를 장착하고 있어 사용자가 좀더 편리하게 사용할 수 있다는 장점 이외에도 풀 브라우징과 같은 고성능 하드웨어를 필요로 하는 기능들을 추가함으로써 휴대단말의 범용화를 가속화하고 있다 [8, 15].

모바일 GIS(Geographic Information System)는 모바일을 기반으로 하고 있기 때문에 언제

어디서나 일반 사용자들도 손쉽게 사용할 수 있는 장점을 가지고 있으나 현재와 같이 폭발적으로 발생하는 네트워크의 트래픽 때문에 상대적으로 용량이 큰 기존의 포맷들은 모바일 지도 서비스 사용자들에게 지루함을 느끼게 한다 [12, 13].

또한, 교통량과 도로현황을 자동차 내에서 알 수 있으며 시시각각 변화하는 기상정보를 처리하거나 어느 장소에서도 그 지역의 지질정보나 지하 시설물의 정보를 알 수 있도록 전 국토의 건설 도로 및 항공 환경 등 각종 공간 데이터를 표현하기 위한 다양한 그래픽 포맷들이 등장하였다 [14, 15].

이중 MVF(Mobile Vector Format)는 레이어를 지원하는 2차원 벡터 공간 데이터 포맷이다.

※ 제일저자(First Author) : 문진용
접수일:2011년 02월 11일, 수정일:2011년 03월 25일,
완료일:2011년 03월 30일
* 극동정보대학 방송영상미디어과 부교수
jmoon@kdc.ac.kr

MVF는 Point, Line, Polyline, Rectangle, Circle, Arc, Text와 같은 공간 객체들의 타입을 포함하고 있다. 내부는 아스키 텍스트 파일로 되어 있어 일반 텍스트 에디터로도 편집이 가능하며, 선택적으로 이진파일로 저장하는 것을 지원한다. 그리고 한글 데이터를 효율적으로 처리하기 위하여 국립지리정보원에서 전국에 대한 수치지도에 사용하고 있는 KS C 5601-1987 완성형 코드를 사용한다.

본 논문의 구성은 다음과 같다. 2장에서 공간 데이터 포맷에 대한 관련 연구를 살펴보고, 3장에서 MVF의 문법을 정의하고, 4장에서는 MVF를 위한 라이브러리 형태의 API의 구현에 대하여 논한다. 마지막으로 5장에서는 결론 및 향후 연구과제에 대하여 언급한다.

2. 관련 연구

본 장에서는 현재 GIS 분야에서 널리 사용되고 있는 데이터 포맷 중에서 공간 데이터 상호 변환 표준인 SDTS, 공간 데이터를 생성하는 분야에 있는 사람들이 널리 사용하고 있는 DXF, 그리고 Adobe에서 개발한 PostScript에 대해 간단히 살펴본다.

2.1 SDTS

SDTS(Spatial Data Transfer Standard)는 모든 종류의 공간 데이터들을 서로 변환 가능하게 해주는 표준으로 상이한 하드웨어, 소프트웨어, 운영체제 간의 데이터 교환을 가능하게 한다. 미연방에 의해 추진된 SDTS는 1992년 미연방 표준(FIPS Publication 173)으로 채택되었다 [6, 9]. SDTS는 공간 데이터 전환의 조직과 구조, 공간 형상과 공간 속성의 정의, 그리고 데이터 전환의 코드화에 대한 규정을 상세히 제공한다.

SDTS는 많은 다른 형식, 모델, 구조들로 서로 연관된 속성 데이터들을 FIPS 173을 사용하여 다른 시스템 간에 교환할 수 있도록 정의되어 있으며, 이는 크게 3개의 부분인 Part 1, Part 2, Part 3으로 구성된다. 또한 Part 1, 2, 3에 의해 특정 데이터 유형에 관한 구체적인 변환 형식으로서 Part 4인 벡터 데이터를 위한 TVP(Topological Vector Profile)가 존재한다.

2.2 DXF

DXF(Drawing eXchange Format) 데이터 포맷은 아스키 텍스트 파일로 구성되어 있으며, GIS 및 CAD 시스템에서 데이터간의 교환 중에서 가장 일반적으로 사용되고 있는 데이터 포맷이다 [2].

DXF는 데이터를 관리, 사용, 변환하는 것이 용이하고, 변환 효율도 뛰어나 널리 사용된다. 최근까지 표준화된 교환수단으로서 많이 사용되고 있다가 GIS 데이터 교환표준(SDTS, DIGEST)의 개발로 점점 영역이 좁아지고 있다. DXF 데이터 구조는 크게 Header, Table, Block, Entities 등 4개의 Section으로 구성되어 있으며, 도면요소는 그룹 코드에 의해 여러 가지 형태의 데이터 요소로 정의된다.

2.3 PostScript

Adobe에서 개발된 PostScript는 레이저 프린터가 발매된 이후 전자출판의 중심점이 되어 표준 출력기의 핵심을 이루었다 [1, 11]. PostScript는 일반적인 프롤로그와 스크립터로 구성된다. 프롤로그는 스크립터에서 사용되는 응용 프로그램 특유의 정보가 저장되어 있고, 스크립터는 작성되는 각 페이지의 내용을 기술하기 위해 응용 프로그램에 의해서 자동적으로 생성된다. 스크립터란 PostScript의 기본 기능과 프롤로그에서 이루어진 정의에 대한 참조 항목과 이들 연산이 요구하는 피연산자와 데이터가 혼합된 것이다.

3. MVF 문법

3.1 고려사항

3.1.1 벡터와 라스터 모델

모바일 GIS는 벡터 모델과 라스터 모델의 서로 근본적으로 다른 형태의 모델을 사용한다. 이들 두 가지 방식에 대한 특징 및 장단점을 간단히 살펴보면 벡터 모델은 지도와 CAD와 같은 시스템에서 시스템의 자원을 보다 적게 소비하며, 데이터를 조작하거나 저장하기에 편리하고, 전송이 효율적이고, 다양한 배율로 나타내기 쉽고, 공간상의 데이터 분석이 용이하며, 그리고 객체들 간의 위상정보를 나타내기 쉽고 공간 테

이터를 표현하기 편리하다. 예를 들면 ESRI의 ArcInfo와 ArcView, MapInfo의 MapInfo, 그리고 Autodesk의 AutoCAD 등이 벡터 모델을 사용하였다.

반면에 라스터 방식은 데이터의 표현은 용이하나 많은 데이터의 용량을 필요로 하고 다양한 GIS 연산을 처리하기가 힘들다. MVF에서는 사용자가 다양한 지도를 검색해 가며 원하는 공간 객체를 쉽게 찾을 수 있고, 라스터 방식보다는 주제별 혹은 여러 레이어의 표현이 용이한 벡터 모델을 사용한다.

3.1.2 관계형과 객체지향 모델

모바일 GIS를 위한 데이터 모델은 크게 관계형 모델과 객체지향 모델로 구분된다. 공간 데이터를 관계 모델을 이용하게 되면 데이터 검색, 보안, 데이터베이스 회복, 트랜잭션 로깅 등과 같은 종래의 데이터베이스 시스템의 기능들을 사용할 수 있으며 표준 질의 언어인 SQL과 공간 데이터 검색을 위한 SQL의 확장을 사용할 수 있다는 장점이 있으나 정수나 문자열과 같은 간단한 데이터 타입을 처리하기 위해 설계된 질의 최적화나 데이터 클러스터링과 같은 것들이 공간 데이터에 적용되었을 때는 만족스러운 결과를 얻지 못하며 공간 데이터와 같은 복잡한 구조의 데이터를 모델링할 수 있는 능력이 관계형 모델에서는 빈약하다.

반면에 객체지향 데이터 모델은 비구조적이고 복잡한 데이터를 자연스럽게 표현하고 데이터의 계층구조를 이용한 연산이 용이하며 새로운 함수의 확장과 데이터의 무결성 검사가 쉽고 설계 단계의 모델과 구현 단계의 모델 사이에 발생하는 불일치 문제를 줄일 수 있어 다양한 타입의 지리 데이터를 처리하는 응용에 적합하다. 위와 같은 특성들로 인하여 GIS 모델링시 관계 데이터 모델보다 객체지향 데이터 모델로 처리하는 것이 적합하다.

3.1.3 모바일 지도 서비스를 위한 요구사항

모바일 지도 서비스는 모바일을 기반으로 하고 있기 때문에 언제 어디서나 쉽게 연결할 수 있는 장점이 있지만, GIS의 경우 애플리케이션의 다운로드와 방대한 양의 지도 데이터를 주고받아야 하므로 속도상의 문제와 사용자의 응

답 시간 지연의 문제가 발생한다. 따라서 모바일 지도 서비스 응용을 위해 정의된 MVF는 용량에 많은 고려를 하여야 한다. 일반적으로 많이 사용되고 있는 SDTS, DXF와 같은 기존의 포맷들은 상대적으로 그 크기가 너무 크며, 실제 모바일 지도 서비스 응용분야에 직접 이용하기에는 불필요한 정보와 용량 때문에 적합하지 않다.

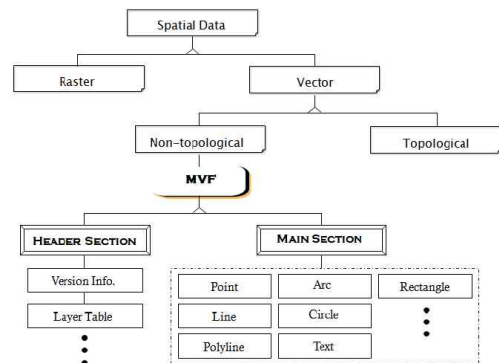
또한, 내국인을 위한 전용 응용 서비스나 농수업 모바일 서비스에서는 대부분 일반인들을 대상으로 하므로 쉽게 알아볼 수 있도록 한글로 서비스를 하는 것이 바람직하며, 국립지리정보원에서 제작한 국내 수치지도 등과 같이 기존에 구축된 공간 데이터를 자동으로 수입하여 시간과 비용이 많이 드는 입력 단계를 최소화하여 기존 데이터의 이용 효율을 높여야 한다.

3.2 MVF 문법

3.2.1 MVF 데이터 구조

GIS에서 사용하는 데이터는 공간 데이터, 속성, 위상 정보, 그리고 시간의 4가지로 이루어진다. 다시 공간 데이터는 벡터와 라스터 모델로 나뉜다. 이중 벡터 모델은 위상과 비위상의 두 개로 나눌 수 있다.

MVF는 공간 데이터 중에서 벡터 모델을 따르고, 비위상 정보를 가지는 공간 객체를 위한 데이터 포맷이다. 이것을 그림으로 도식화하면 (그림 1)과 같다.



(그림 1) MVF 데이터 구조

MVF는 헤드섹션과 메인섹션으로 이루어져 있다. 헤드섹션은 파일의 일반적인 특성이 나열되어 있고 메인섹션은 실제 출력 및 저장에 필

요한 데이터의 리스트가 저장된다. MVF는 NGIS와의 호환하기 위하여 좌표 값의 단위는 [m]로 하고 소수 2자리까지 나타내며, 표현대상을 9개의 레이어(도로, 철도, 하천, 건물, 지물, 시설물, 지형 행정 및 지역 경계, 주기)로 구분한다. MVF에서 사용되는 기본적인 공간 객체의 종류는 다음과 같다.

- ▶ Point : 공간상의 한 위치를 나타내는 (x, y)좌표로 표현된다.
- ▶ Line : 두 Point 좌표로 표현되는 선분이다.
- ▶ Polyline : n개의 점을 가지고 이들이 연결된 선분들의 집합이다.
- ▶ Circle : 중심점 좌표와 반지름 크기로 표현되는 원이다.
- ▶ Arc : 중심점과 반지름, 시작 각도, 끝 각도로 표현되는 호이다.
- ▶ Text : 시작점과 문자열이다.
- ▶ Rectangle : 2개의 (x, y)좌표로 표현되는 사각형이다.

```

<HEAD>
Comment Syntax for the MVF(Mobile Vector
Format) v0.9:
Comment number -> #:
MVF_FileName string of Filename:
MVF_Version version #:
MVF_Extents xmin + ymin + xmax + ymax:
MVF_LayerTable # of entries:
MVF_LayerElement LayerName + # of Layer:
MVF_Background color:
</HEAD>
    
```

(그림 2) MVF Header Section Syntax

```

<MAIN>
MVF_Point x + y;
MVF_MoveTo x + y;
MVF_LineTo x + y;
MVF_PolyLine # of points + { x + y };
MVF_RelMoveTo xoffset + yoffset;
MVF_RelLineTo xoffset + yoffset;
MVF_RelPolyLine # of points + {<xoffset> +
<yoffset>};
MVF_Rectangle xmin + ymin + xmax + ymax;
MVF_Circle xcenter + ycenter + radius;
MVF_Arc xcenter + ycenter + radius + startangle +
endangle;
MVF_Text xstart + ystart + text string;
</MAIN>
</MVF>
    
```

(그림 3) MVF Main Section Syntax

3.2.2 MVF 문법

(그림 2)는 MVF의 헤드섹션에서 사용하고 있는 문법의 일부분을 나타낸다. 헤드섹션에서는 Markup Language에서 이용하는 것처럼 <HEAD>와 </HEAD>라는 태그를 사용하여 시작과 끝을 알려주며, 그 내부에는 MVF에 관한 파일이름, 파일포맷 버전, 레이어 정보, 그리고 배경색깔과 같은 일반적인 정보를 담고 있다.

MVF에서의 메인섹션 문법이 (그림 3)에 나타나 있다. 메인섹션은 Point, Line, Text, 그리고 Polyline 등과 같은 각 공간 객체에 관련된 정보를 담고 있다. MVF는 대부분의 벡터 포맷이 따르는 좌측 하단을 원점(0,0)으로 한다. 절대좌표 값은 반드시 양의 정수이어야 하며, offset은 음수일 수도 있다.

```

FILE *MVFOpen(char *filename)
int MVFClose(FILE *MVFfp)
void MVFHeadStart(FILE *MVFfp)
void MVFHeadEnd(FILE *MVFfp)
void MVFVersionInfo(FILE *MVFfp)
void MVFExtents(FILE *MVFfp,int xmin,int
ymin,int xmax,int ymax)
void MVFLayerTable(FILE *MVFfp,int numentries)
void MVFLayerEntry(FILE *MVFfp, int
layernum,char *layername)
void MVFBinWrite(FILE *MVFfp,int BinWrite)
void MVFBackground(FILE *MVFfp,int color)
void MVFComment(FILE *MVFfp,char* com)
    
```

(그림 4) MVF Header Section Generation Module

4. MVF API

본 장에서는 앞에서 언급한 MVF 문법을 이용하여 API를 헤드섹션 생성모듈과 메인섹션 생성모듈로 나누어 설명한다.

4.1 구현 환경

본 논문에서는 HP xw6000 Workstation에서 운영체제로는 Fedora Core 6을 사용하였고, 컴파일러는 gcc 4.4.0을 사용하였다.

4.2 헤드섹션 생성모듈

MVF 헤드섹션 생성모듈은 (그림 4)와 같이 헤드섹션을 생성할 수 있는 라이브러리 형태의 함수로 되어 있다. 이 중 MVFBinWrite는 선택적으로 이진파일로 저장하는 것을 설정하는 함수이며, MVFLayerTable와 MVFLayerEntry는 레이어 테이블을 만들어 주는 함수들이다.

4.3 메인섹션 생성모듈

(그림 5)는 MVF 메인섹션을 생성하기 위한 함수들을 나열하고 있다. 이 중 MVFSetLayer는 현재 작업하는 레이어를 이동하는데 사용되며, Rel이 붙은 함수는 현재 위치로부터 상대적인 좌표로 작동하며, 붙지 않은 함수는 절대 좌표로

동작한다. MVFFillMode는 FillOn/Off로 현재로부터 작업하는 영역을 채우는지 여부를 설정하는 함수이다.

```

void MVFMainStart(FILE *MVFfp)
void MVFMainEnd(FILE *MVFfp)
void MVFPoint(FILE *MVFfp,int x,int y)
void MVFMoveTo(FILE *MVFfp,int x,int y)
void MVFRelMoveTo(FILE *MVFfp,int x,int y)
void MVFLineTo(FILE *MVFfp,int x,int y)
void MVFRelLineTo(FILE *MVFfp,int x,int y)
void MVFPolylineStart(FILE *MVFfp,int n)
void MVFRelPolylineStart(FILE *MVFfp,int n)
void MVFPolyPoint(FILE *MVFfp,int x,int y)
void MVFRelPolyPoint(FILE *MVFfp,int x,int y)
void MVFRectangle(FILE *MVFfp,int xmin,int
ymin,int xmax,int ymax)
void MVFCircle(FILE *MVFfp,int xcenter, int
ycenter,int radius)
void MVFArc(FILE *MVFfp,int xcenter, int
ycenter,int radius,int startangle,int endangle)
void MVFText(FILE *MVFfp,int x, int y, char *text)
void MVFSetLayer(FILE *MVFfp,int layer)
void MVFFillMode(FILE *MVFfp,int mode)
    
```

(그림 5) MVF Main Section Generation Module

5. 성능 평가 및 고찰

본 논문에서의 성능 평가는 압축 포맷인 GIF(Graphics Interchange Format) 포맷, 국내 공간 데이터를 관리하는 역할을 담당하고 있는 국립지리정보원에서 사용하고 있는 DXF 포맷, 그리고 MVF 포맷에 대하여 비교·분석한다.

GIF는 JPEG(Joint photographic Experts Group)과 함께 모바일 환경에서 지원되는 그래픽 파일 포맷 중의 하나이다. 실제로 이 포맷에 대한 권리는 Compuserve에 의해 소유되고 있으며, 기술적인 측면을 보면 GIF는 2차원 라스터 그래픽 포맷으로, 바이너리로 표현되며, 압축 기술은 LZW(Lempel-Ziv-Welch)를 이용한다. GIF에는 87a와 89a의 두 가지 버전이 있는데, 1989

년 7월에 발표된 89a는 하나의 파일 내에 짧은 순서를 갖는 일련의 이미지들을 포함시킴으로써 animated GIF를 만들 수 있는 기능과 interlaced GIF 표현 기법을 지원한다. 본 논문에서 사양을 정한 MVF에 대하여 타포맷과 정성적으로 비교하여 보면 <표 1>과 같다.

<표 1> 기존 포맷과의 정성적 비교

	모델	위상정보	한글처리	객체지향	데이터 손실	속성 데이터	레이어
GIF	래스터	×	●	×	있음	×	1
DXF	벡터	×	●	○	없음	×	>=1
MVF	벡터	×	○	○	없음	×	>=1

○ : 지원됨 ● : 경우에 따라 다름 × : 지원안됨

첫째, GIF는 래스터 모델을 사용하며, DXF와 MVF는 벡터 모델을 사용한다. 모바일 GIS에서는 다양한 공간 객체를 원활하게 처리할 수 있고, 확대, 축소와 같은 공간 처리를 효율적으로 처리하기 위해서는 벡터 모델이 바람직하다.

둘째, GIF, DXF, MVF는 위상 정보와 속성 데이터를 담고 있지 않다. 반면에 공간 객체 변환 표준 포맷인 SDTS의 경우에는 위의 정보가 포함되어 있어 상대적으로 용량이 매우 크다. 또한 각 공간 객체들이 서로 다른 파일에 저장됨으로 모바일 인터넷에서 사용자의 질의를 실시간으로 결과 맵을 생성하는 경우에 여러 파일을 참조해야 함으로 SDTS를 바로 적용하기에는 무리가 따른다.

셋째, 실시간 맵을 생성하기 위해서는 공간 객체들을 서로 분리할 수 있어야 함으로 객체 지향 모델이 적합하며 도로, 하천, 시설물과 같은 공간 객체들을 서로 다른 레이어 별로 구분하여 저장하는 것이 바람직하다. 예를 들면, 현재 많이 사용되고 있는 서버 중심의 시스템에서는 래스터 포맷을 사용함으로 클라이언트에서 사용자의 재처리 요구는 사실상 한계가 있다.

마지막으로 데이터의 손실 문제이다. 여기에는 Tradeoff가 있다. 예를 들면, 모바일 환경에서 대표적으로 포맷인 GIF는 LZW 손실 압축 알고

리즘을 사용하고 있어 결과 맵을 손실해서 클라이언트로 전송함으로 인해 사용자 응답 시간은 짧아지는 장점을 가지지만 데이터의 손실로 인해 정확한 정보를 제공할 수 없다는 단점을 가진다. 반면에 데이터의 손실이 거의 발생하지 않는 포맷의 경우에는 이와 반대로 질 높은 서비스를 제공할 수 있지만, 네트워크의 트래픽을 증가시킨다.

6. 결론

최근 모바일 인터넷의 급속한 확산과 휴대장치 기술의 발전으로 모바일 지도 서비스가 보편화되고 있다. 이동이 자유로운 모바일 컴퓨터는 향후 정보 소비의 주체가 된다는 점을 감안할 때 모바일 GIS는 GPS(Global Positioning System)와 연동한 LBS(Location Based System), 물류 관리 시스템, 관광지 안내 등 많은 응용 분야를 창출하고 있다. 특히 보급률이 매우 높은 스마트폰에서의 지도 서비스는 대표적인 콘텐츠가 될 것이다.

본 논문에서는 지리 정보 시스템에서 유용하게 사용할 수 있고, 특히 GIS 개발자들에 의해 개척되고 있는 가장 최선의 기술인 모바일 GIS에서 응용하기 적합한 공간 객체를 위한 파일 포맷인 MVF에 대하여 설명하였다. 먼저 MVF에 관한 문법을 정의하고 애플리케이션을 위한 API를 구현하였다.

향후 연구과제로는 좀더 유용한 MVF 공간 파일을 만들기 위해 공간 객체 변환기를 만들어 다른 공간 파일들과의 호환성을 높이며, MVF 공간 포맷에 대한 일반 사용자들의 다양한 욕구를 충족시키기 위하여 저작 도구를 개발하여야 할 것이다.

참고 문헌

[1] Adobe Systems, "Adobe PostScript Overview", <http://www.adobe.com/prodindex/postscript/main.html>.
 [2] Autodesk, "Drawing eXchange Format", <http://www.autodesk.com/>.
 [3] W. Abdelsalam, "Maintaining Quality of Service for Adaptive Mobile Map Clients," Master thesis of Mathematics in Computer Science in University of Wate

rloo, Canada, 2001.

[4] Environmental Systems Research Institute Inc.(ESRI), "A Study on European Trends in Standards for Geodata Exchange," ESRI internal report, 1994.

[5] R. G. Fegeas, J. L. Cascio, and R. A. Lazar, "An Overview of FIPS 173, The Spatial Data Transfer Standard," Vol.19, No.5, 1992.

[6] National Institute of Standards and Technology, "The Spatial Data Transfer Standard, Federal Information Processing Standard Publication 173," U.S. Dept. of Commerce, 1992.

[7] Open GIS Consortium, "Open GIS Homepage," <http://www.opengis.org>.

[8] S. Shekhar, Y. Huang, J. Djugash, C. Zhou, "Location-based services and mobile computing: algorithms: Vector map compression: a clustering approach," Proc. of the 10th ACM Int'l Symp. on Advances in GIS, 2002.

[9] USGS, "Spatial Data Transfer Standard," <ftp://sdts.eer.usgs.gov/pub/sdts/>.

[10] S. C. Wang and C. C. Joyce, "GIS Major Data Structures vs. Application," Urban & Regional Systems Association, Vol.2, 1992.

[11] Yahoo! GIS Page, http://www.yahoo.com/Science/Geography/Geographic_Information_Systems_GIS_/.

[12] 김명삼, 정영지, "모바일 WMS를 위한 서버기반 GIS 컴퓨팅 플랫폼 설계 및 구현," 한국정보과학회 논문지, 제12권 제1호, pp.9-20, 2006.

[13] 배상근, 박영무, 김병국, "모바일 매핑시스템에서의 실시간 지리정보 전송을 위한 연구," 한국GIS학회지, 제13권 제1호, pp.91-101, 2005.

[14] 윤성균, "모바일 벡터이미지 처리기술 및 솔루션 동향," 한국정보처리학회지, 제12권 제1호, 2005.

[15] 홍일영, "지리정보로 열어가는 모바일 웹2.0," SW Insight 정책리포트, 정책연구센터@한국소프트웨어진흥원, pp.5-31, 2007.



문진용

1998년 : 건국대학교 대학원 (공학 석사)

2001년 : 수원대학교 대학원 (이학 박사)

2001년~현재 : 극동정보대학 방송영상미디어과 부교수
 관심분야 : 모바일 인터넷, 데이터베이스, 디지털 콘텐츠 등