

J2ME기반 모바일 단말기에서의 MPEG4 동영상 재생 설계 및 구현

이봉하⁰ 김영만
 국민대학교 컴퓨터학부
 {bhlee⁰, ymkim}@ccclab.kookmin.ac.kr

Design and Implementation of the MPEG4 player on the J2ME

Bong Ha Lee⁰ Young Man Kim
 School of Computer Science, Kookmin University

요 약

본 논문에서는 MPEG4 표준에 의거한 멀티미디어 파일을 동영상 디코더가 장착되지 않은 모바일 단말기에서 재생이 가능하도록 설계 구현 하였다. 모바일 단말기는 제한된 리소스의 환경이므로 단말기 내에서 동영상을 처리하는 것에는 많은 한계가 있다. 본 논문에서는 MPEG4파일의 음성부와 영상부로 구분하여 모바일 단말기에 알맞은 파일로 변화시켜 실시간으로 데이터를 가져와 모바일 단말기 상에서 플레이를 해준다. 비디오부는 PNG로 음성부는 SMAF(Synthetic Music Mobile Application Format)로 변화를 시켜 설계 구현하였다. 동영상 디코더가 장착되지 않은 단말기에서도 다양한 멀티미디어 서비스를 제공하는데 그 목적이 있다.

1. 서 론

MPEG1, MPEG2 표준이 만들어짐에 따라 기존의 방송, 통신과 컴퓨터 산업 간의 경계가 없어지고 서로 융합되어 복잡적이고 다양한 서비스들을 제공하고 있다. 그러나 호환성에 문제가 생기자 이러한 문제점을 극복하고자 1993년부터 연구가 시작되어 MPEG4가 생겨났다. MPEG4는 대역폭이 적은 통신매체에서도 전송이 가능하고 양방향 멀티미디어도 구현할 수 있는 오디오/비디오 표준 부호화 방식이다. MPEG4는 기존 H.261, JPEG, MPEG1, MPEG2에서 쓰던 블록 단위의 변환 부호화 방법을 탈피하고 영상 내용에 근거하여 영상 신호를 부호화하는 새로운 방법을 추구하고 있으며, MPEG1, MPEG2가 압축율을 높여 빠른 전송을 하는 데 중점을 둔 반면 64kbps급의 초저속 고압축을 실현을 목적으로 하고 있다. 이러한 MPEG4는 기존의 방송, 가전, 컴퓨터, 통신 분야가 각기 멀티미디어 정보를 처리할 수 있도록 디지털화됨으로써 궁극적으로 이 모두가 융합되어 진정한 멀티미디어시대가 이루어 질 것이다[2][3].

본 논문에서는 MPEG4 표준[1]에 의거한 멀티미디어 파일을 유선이 아닌 모바일 단말기 상에서 재생이 가능하도록 변화를 주어 구현 설계 하였다. 본 논문의 2장에서는 관련 내용과 프로그램 설계에 대해 살펴보고 3장에서는 프로그램 구현에 대해서 알아본다. 마지막으로 4장에서는 본 논문의 결론과 향후 과제에 관해서 기술한다.

2. 관련 내용과 설계

국내 모바일 단말기에는 자바 기반의 KVM과 C 기반의 GVM들이 사용되고 있다.

본 논문에서는 자바 기반의 SK-VM을 사용하고 있다. 그리고 서버부는 MPEG4 표준[1]에 의거한 고화질 압축 파일을 음성부와 영상부로 나누어 각기 다른 방법을 적용하여 모바일 단말기에 알맞은 파일들(음성부는 SMAF, 영상부는 PNG)로 변환한다. 이 파일들은 실시간으로 모바일 단말기 환경(SK-VM Phone Emulator ver1.1d)으로 전송이 되어진다.

2.1 J2ME

J2ME는 1999년 Java One Conference에서 처음 발표되었다. 이것은 Java 2의 가전 제품과 임베디드 장치용 버전이다. J2ME에서 Connected Limited Device Configuration(CLDC)와 Mobile Information Device Profile(MIDP)가 무선 기기를 위해 설계되었다. 그림1은 J2ME의 전반적인 구조를 보여주고 있다.

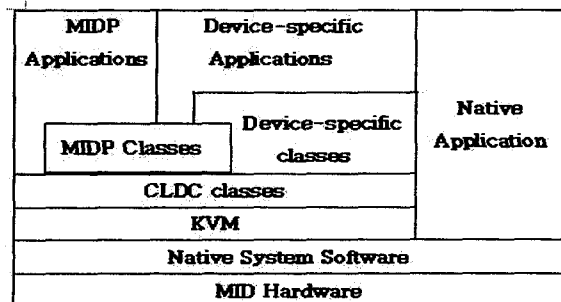


그림 1 MIDP architecture

그림의 구조를 살펴보면 MID hardware layer 은 휴대

전화나 양방향 무선 호출기의 하드웨어 부분에 해당한다. Native system software layer는 장치 제조업체가 제공하는 native 운영체제와 시스템 라이브러리들을 포함하고 있다. KVM layer는 애플리케이션을 위한 런 타임 환경을 제공한다. CLDC layer는 무선 애플리케이션들을 위한 순수 자바 API들을 제공한다. MIDP layer는 GUI 라이브러리, persistent storage 라이브러리, 네트워크 라이브러리, 타이머 클래스 등을 제공한다[4].

2.2 SK-VM

SK-VM은 SK Telecom에서 clean room 기반으로 자체 개발한 J2ME(Java2 Micro Edition) 자바 실행 환경이다. SK-VM에서 M-Configuration Package는 CLDC를 구현한 것인데 가상 머신 및 단말 확장 UI, 네트워크, IO를 포함하고 있다. M-Profile Package는 MIDP를 구현한 것이다. SKT Class Package는 SK Telecom에서 무선인터넷 서비스에 필요한 API를 정의한 것으로 J2ME 구성요소 중에서 OEM-Specific Classes 부분에 해당된다[5].

2.3 MMF

MMF파일은 YAMAHA에서 정의한 SMAF(Synthetic Music Mobile Application Format) 파일로서, MIDI 화일을 모바일 디바이스에 맞게 YAMAHA에서 제공되는 변환 형식을 이용해서 생성한 파일을 말한다[5].

2.4 RMS(Record Management System)

RMS는 J2ME MIDP에서 정의한 레코드 지향적인 간단한 데이터베이스 시스템이다. 출력, 저장, 조작, 액세스 등의 작업을 제공하는 크고, 지속적이며, 통합적인 동적 데이터의 집합체라고 할 수 있다[4].

2.5 설계

본 논문은 서버부와 클라이언트부로 크게 나누어진다. 그림2는 전체 흐름을 데이터 위주로 나타내고 있다. 서버부에서는 MPEG4의 고화질 파일을 음성부와 영상부로 나누어 모바일 단말기 환경에 맞도록 변화를 시켜주는 역할을 수행한다[5][6]. 음성부는 현재 SK-VM에서 지원하는 SAMF파일로 변화를 시키고 영상부는 PNG파일로 변화를 시킨다. 변화된 파일의 정보들이 XML 형식으로 모바일에 보내어지고 이 정보를 토대로 모바일 단말기는 실시간으로 파일들을 가져오게 되고 실시간으로 플레이가 이루어진다. XML정보에는 파일의 위치, 저장장소, 음성과 영상의 각 파일 수를 담고 있다.

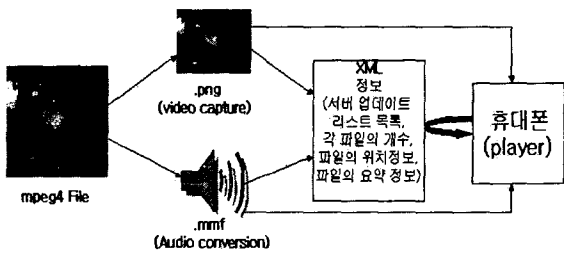


그림 2 모바일 동영상 플레이어 전체 흐름도

2.5 설계 구현 이유

일반 컴퓨터 환경에서는 간단한 코덱을 설치하여 다양한 동영상 파일들을 접할 수 있다. 하지만 모바일 단말기에서는 각 기기들의 특색에 따라 동영상 파일을 제공할 수 있다. 하지만 현재 유통되고 있는 휴대폰의 절반 이상은 이러한 서비스가 제공 되지 않는다. 이러한 제한된 환경에서도 동영상 서비스를 접할 수 있게 하고자 설계 및 구현을 하였다.

3. 구현

구현부에서는 서버와 클라이언트로 나뉘어 지는데 서버 프로그램은 데이터에 변화를 주고 클라이언트에게 변환된 정보들을 제공해 주는 역할을 하고, 클라이언트는 서버의 정보를 이용해서 데이터를 받아 버퍼링과 동적 데이터베이스를 이용하여 플레이 후 데이터를 파기 시킨다.

3.1 구현 환경

서버 프로그램은 윈도우98 이상의 환경에서 Visual C++ 6.0 Tool을 이용하여 구현하고 클라이언트 프로그램은 J2ME로 구현하고 실행은 SK-VM Phone Emulator Ver 1.1d와 휴대폰 Samsung SCH-250X을 사용한다.

3.2 서버 프로그램 구현

서버부에서는 음성부와 영상부로 나누어 음성부는 연속적인 파일(wave)로 변환 후 5초씩 잘라 SMAF 포맷으로 변환을 하고 영상부는 음성 10초에 한 장면씩 PNG 포맷으로 캡처 한다.

그림3은 서버 프로그램의 전반적인 흐름을 보여주고 있다. 서버부는 사용자 인터페이스를 통해서 MPEG4 파일이 음성부와 영상부로 나누어서 처리가 되어진다. 영상부는 IP프레임이 BMP형식으로 캡처 되어지고 이것은 다시 모바일 디바이스에 맞도록 PNG 파일로 만들어진다. 음성부는 wave파일로 변환된 후 Yamaha에서 제공하는 coder를 이용해서 SMAF로 변경된다.

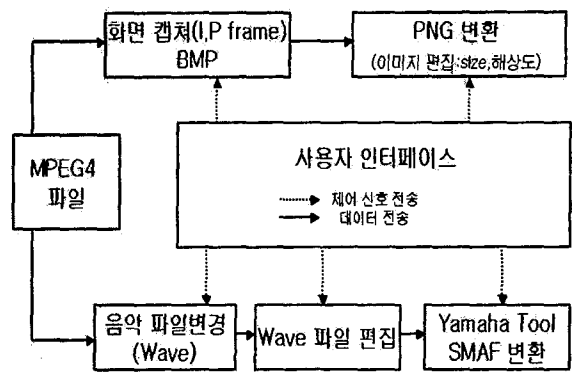


그림 3 서버 프로그램 구현도

3.3 클라이언트 프로그램 구현

클라이언트부에서는 제한된 환경에서 구현하기 위해서 최적화에 관심을 기울여야 한다. 본 논문에서는 SK-VM를 사용하고 있는데 이것 또한 제약이 많이 있다. SK-VM에서는 네트워크 환경으로 소켓을 하나만 열수 있고, HTTP, TCP, UDP를 지원하는데 여기서는 HTTP를 이용하여 바이트 스트림으로 각 파일들을 가지고 온다. 응용 프로그램에 할당된 메모리도 매우 작아서 멀티미디어 파일을 담아 내기에는 역부족이다. 더욱이 MPEG4는 고화질이라 이것을 변환한 파일은 그 정보를 최대한 살려야 하기 때문에 용량이 매우 크다. 그래서 파일을 소비한 후 바로 버리고 계속 받아오고 소비해야 한다. 이런 부분은 멀티 스레드 방식을 사용한다.

그림4는 이러한 클라이언트부의 구현도를 대략적으로 보여준다. 영상부는 서버 접속과 동시에 실시간으로 받아와서 RMS에 저장하고 플레이 시 호출하여 사용한다. 오디오부는 모바일 단말기의 애플리케이션 실행에 할당된 메모리의 크기에 맞게 일정량의 데이터를 버퍼에 저장하고 저장한 데이터들을 실시간으로 플레이해주고 플레이가 끝난 파일은 바로 파기시킨다.

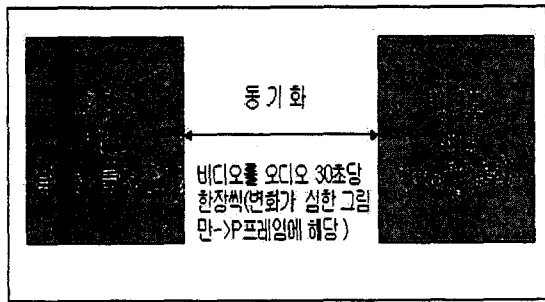


그림 4 클라이언트 프로그램 구현도

3.3 서버와 클라이언트 부분

위에서 서버부와 클라이언트부의 구조를 살펴 보았다. 지금부터는 전체적인 서버와 클라이언트 간의 데이터 흐름을 살펴 본다.

서버는 데이터를 준비해서 클라이언트에게 제공할 정보를 만들어 놓고 클라이언트의 접속을 기다린다.

클라이언트는 서버에게 접속을 시도하게 되고 서버측에서는 갱신된 멀티미디어 목록을 XML형식으로 클라이언트에게 보내주게 된다.

XML이 가지고 있는 정보에는 멀티미디어 파일의 이름과 각 파일들의 저장위치, 저장된 영상부, 음성부 파일들의 개수 정보, 각 정보의 세부 정보들이 들어 있다. 이 정보들을 토대로 데이터들의 위치에 접속해서 실시간 바이트 스트림 방식으로 영상부, 음성부 데이터들을 가져온다. 그 후에 클라이언트에서 서버로부터 받은 데이터를 플레이 해준다. 그림5는 위에서 설명한 내용을 표현한 데이터 흐름도이다.

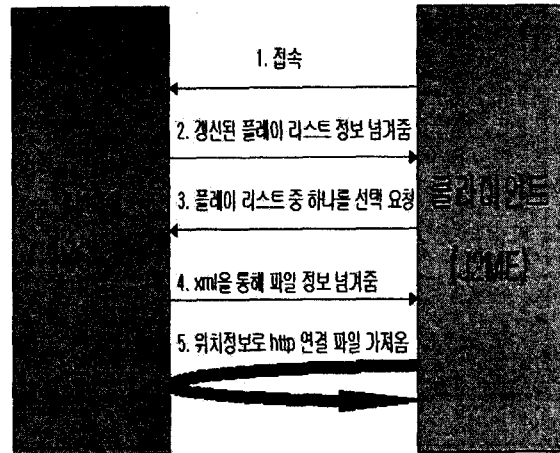


그림 5 서버 클라이언트간의 데이터 흐름도

4. 결론 및 향후 과제

모바일 단말기의 환경이 일반 컴퓨터 성능에 비해 훨씬 떨어진다. 낮은 성능의 CPU와 적은 메모리 용량으로 인하여 데이터의 로딩 시간이 길어지며 음성부의 끊김 현상을 일으키고 있다. 소켓도 휴대폰에서 하나만을 지원하고 있고 멀티 스레드 지원도 거의 잘 안되고 있는 상태이다. 향후 연구 방향으로 클라이언트 쪽 프로그램을 최적화하여 데이터들이 원활하게 플레이 될 수 있도록 데이터의 용량을 줄이고 버퍼링 프로그램을 최적화 시키는데 중점을 주어야 한다. 또한 현재 로컬환경에서 애플레이터로 돌리는 데도 데이터 로딩 시간이 많이 걸린다. 다시말하면 애플레이터보다 더 열악한 환경인 모바일 단말기에서도 끊어짐 없이 자연스럽게 플레이 되도록 해야한다.

5. 참고 문헌

- [1] ISO/IEC, "Information Technology - Generic Coding of Audio-Visual Objects", ISO/IEC 14496-1 ~ ISO/IEC 14496-5, Dec.1998.
- [2] Weiping Li, Jens-Rainer Ohm, Mihaela van der Schaar, Hong Jiang, and Shipeng Li, "MPEG-4 Video Verification Model version 17.0", July.2000.
- [3] Joan L.Mitchell, Willam B. Pennebaker, Chad E. Fogg, and Didier J. LeGall, "MPEG video compression standard", 1996.
- [4] Java2 Platform, Micro Edition, "KVM White Paper", "CLDC Specification, V1.0", "MIDP Specification, Final, V1.0", "<http://java.sun.com/j2me/docs/>", August 2002
- [5] XCE, "XCE Technical Document", <http://developer.xce.co.kr/>, August 2002
- [6] W3C, "PNG(Portable Network Graphics) Specification", <http://www.w3.org/TR/REC-png.html>, August 2002