

모바일 단말기를 위한 MPEG-7 기반의 비디오 관리 시스템의 설계 및 구현

안병태[†], 민정훈^{**}, 강병수^{***}, 강현석^{****}

요 약

본 논문은 모바일 단말기를 위한 MPEG-7 기반의 비디오 관리 시스템 MVMS(MPEG-7 Based Video Management System for Mobile Device)를 제안한다. 기존의 MPEG-7 응용들은 대부분 데스크탑 환경에서 MPEG-7 문서를 파일 시스템 기반으로 관리한다. 이에 비해, MVMS는 모바일 단말기 환경에서 MPEG-7 문서를 내장형 데이터베이스 시스템을 이용하여 필요한 정보만 관리하기 때문에 리소스가 부족한 모바일 단말기 환경에서 효율적으로 활용할 수 있다. MVMS는 비디오 세그먼트별 키 이미지 추출과 MPEG-7 문서 편집 기능을 지원한다. 또한, 내장형 데이터베이스를 통해서 MPEG-7 데이터를 관리하기 때문에 다중 문서의 검색과 관리가 가능하다.

Design and Implementation of MPEG-7 Based Video Management System for Mobile Devices

Byeong-Tae Ahn[†], Jeong-Hoon Min^{**}, Byeong-Shoo Kang^{***}, Hyun-Syug Kang^{****}

ABSTRACT

We propose video management system based on MPEG-7 for mobile device(MVMS). Most of existing MPEG-7 applications manage MPEG-7 document based on file system in the desktop environment. To contrast them, because the MVMS manages only necessary information using embedded database system for MPEG-7 documents in mobile devices, it can use efficiently lack of the resource in mobile device environment. MVMS supports key image extraction and MPEG-7 document edition function. Moreover, this is possible search and management of multi document because of MVMS manage MPEG-7 data through embedded DB.

Key words: MPEG-7, Embedded Database(내장형 데이터베이스), Video Management System(비디오 관리 시스템)

1. 서 론

무선 인터넷의 발달로 PDA, 포켓 PC, 휴대폰과

같은 다양한 모바일 단말기들이 널리 보급되면서 여러 가지 모바일 응용들이 사용되고 있다. 최근 들어서는 이러한 모바일 단말기들의 성능이 더욱 강력

※ 교신저자(Corresponding Author) : 강현석, 주소 : 경남 진주시 가좌동 900번지(660-701), 전화 : 055)751-5998, FAX : 055)762-1944, E-mail : hskang@nongae.gsnu.ac.kr
접수일 : 2005년 10월 4일, 완료일 : 2006년 1월 5일

[†] 정회원, 경상대학교 컴퓨터학과

(E-mail : ahnbt@hanmail.net)

^{**} 준회원, 경상대학교 컴퓨터학과

(E-mail : acmask@hanmail.net)

^{***} 준회원, 경상대학교 컴퓨터학과

(E-mail : shoo99@empal.com)

^{****} 종신회원, 경상대학교 컴퓨터과학부 교수, 컴퓨터.정보통신 연구소 연구원

※ 본 연구는 2005 정보통신 기초 기술연구지원 사업 지원으로 수행되었음

해지면서 단순 자료 처리 기능을 뛰어 넘어 다양한 멀티미디어를 활용하는 응용들이 나타나기 시작하였다. 따라서 이제는 모바일 환경에서도 멀티미디어 데이터의 관리는 매우 중요한 연구 주제가 되었다.

최근에 멀티미디어를 보다 효과적으로 다룰 수 있도록 멀티미디어 콘텐츠 기술 방법에 대한 국제 표준으로 MPEG-7(Moving Picture Experts Group)이 채택되었다[1,2]. 이러한 MPEG-7은 멀티미디어에 대한 메타데이터를 XML 형태로 정의하는데, 이를 이용하면 멀티미디어 데이터에 대해 다양한 검색이 가능하다[3]. 예를 들어, 월드컵 축구 경기에서 이영표의 멋진 어시스트 장면을 찾는 내용 기반 검색이나 장르, 배우, 방송 시간 등의 속성으로 TV 채널을 검색하는 속성 기반 검색 등이 가능하게 된다.

하지만 최근까지도 모바일 단말기에서 MPEG-7을 기반으로 원하는 멀티미디어를 다루는 응용은 매우 미미한 실정이다[4,5]. 그 이유는 멀티미디어가 갖는 대용량성 때문에 따른 모바일 단말기에서 MPEG-7 데이터를 효과적으로 다루기가 어렵기 때문이다. 즉, 멀티미디어 데이터를 모바일 환경에서 효과적으로 활용하기 위해서는 MPEG-7 기반의 멀티미디어 데이터를 효과적으로 관리할 수 있어야 한다[6]. 이러한 시스템의 개발을 위해 내장형 데이터베이스 기술을 이용할 수 있다.

본 논문에서는 PDA와 같은 모바일 단말기에서 MPEG-7 기반의 비디오를 효과적으로 관리할 수 있는 MVMS(MPEG-7 Based Video Management System for Mobile Device)를 설계 및 구현하였다. 제안하는 MVMS는 모바일 환경에서 비디오에 대해 MPEG-7 문서를 생성하고 검색할 수 있는 통합 환경을 제공한다. MVMS는 기본적으로 클라이언트측 모바일 단말기에서의 지역 DB와 서버측 데스크탑 컴퓨터에서의 전역 DB를 통해 비디오를 검색한다. 검색된 해당 비디오는 모바일 단말기에서 검색 인터페이스를 통해 보여진다.

논문의 구성은 다음과 같다. 2장에서는 비디오와 관련된 MPEG-7 연구들을 살펴본다. 3장에서는 제안하는 시스템인 MVMS의 설계, 그리고 4장에서는 이의 구현을 다룬다. 5장에서는 MVMS를 기존의 유사 시스템과 비교해 본다. 끝으로 6장에서는 결론과 향후 과제를 논한다.

2. 관련 연구

이 장에서는 제안하는 MVMS의 개발에 관련된 MPEG-7을 이용한 비디오 연구들에 대해 알아본다.

멀티미디어 데이터로부터 추출된 메타데이터를 MPEG-7 기술자와 기술 스킴을 이용하여 표현하게 되면, XML 형태의 MPEG-7 문서가 생성된다. 따라서 이러한 MPEG-7 문서를 토대로 하여 멀티미디어에 대한 다양한 검색을 수행할 수 있다[4]. 본 논문에서는 비디오를 그 대상으로 하는 대표적인 MPEG-7 도구들인 Richos사의 MovieTool, IBM사의 VideoAnnEx, SingingFish사의 SingingFish, TruVideo사의 Truvideo를 간략하게 살펴보도록 한다.

MovieTool[7]은 비디오를 계층 구조의 세그먼트 별로 분류하여 MPEG-7 문서를 생성하는 도구이다. 사용자가 비디오를 재생시키는 동안에 직접 비디오를 계층 구조의 세그먼트들로 분리할 수 있고, MPEG-7 문서의 기술들을 직접 편집할 수 있도록 하고 있다. MovieTool의 가장 큰 장점은 생성된 MPEG-7 문서를 통해서 비디오를 검색함으로써 사용자가 원하는 장면에 바로 접근할 수 있다는 것이다. 하지만 MovieTool은 비디오에 대한 MPEG-7 문서의 생성 기능만을 지원하고 생성된 MPEG-7 문서를 관리하는 기능은 지원하지 않는다. 따라서 다수의 비디오를 검색하기 위해서는 사용자가 MPEG-7 문서마다 MovieTool을 이용하여 개발한 후 검색해야 하는 번거로움이 있다.

VideoAnnEx[8]은 비디오에 대해 주석 기반으로 MPEG-7 문서를 생성하는 도구이다. 비디오를 순차 구조의 세그먼트들로 분류하고 각각의 세그먼트에 대해서 사용자가 주석을 입력할 수 있도록 한다. 이 도구의 장점은 비디오를 분석하여 자동적으로 순차적인 구조의 장면/샷(scene/shut)을 생성해 준다는 것이다. 이렇게 생성된 비디오의 순차 구조는 분리와 병합 과정을 통해서 재생이 가능하다. 뿐만 아니라 비디오에서 다양한 키 이미지를 자동으로 추출하여 파일로 생성해 준다. 하지만 VideoAnnEx는 생성된 MPEG-7 문서와 키 이미지에 대한 관리는 지원하지 않아 사용자들이 다른 MPEG-7 관리 시스템을 이용하여 새롭게 관리해야 한다.

SingingFish[9]는 인터넷을 기반으로 하여 오디오와 비디오에 대한 스트리밍 미디어의 검색을 지원한

다. 미디어에 대한 주석 기반 검색 뿐만 아니라 검색하고자 하는 미디어의 일부를 서버 측에 전송하면 서버에서 자동적으로 전송된 미디어에 대한 특징을 추출하여 저장된 MPEG-7 문서를 검색하여 사용자에게 그 결과를 반환해 줌으로써 많은 노력을 들어준다. 하지만 SingingFish는 데스크탑 환경을 기반으로 하고 있고, 검색된 결과를 모바일 환경의 클라이언트 측에서 저장 관리할 수 있는 형태는 지원하지 않는다.

TruVideo[10]는 이종의 모바일 환경에서 비디오 압축, 전송 및 녹화재생 소프트웨어 솔루션을 제공하며, 인코딩에서 서버 기반 주문형 비디오, 장치형 스토리지 및 녹화재생에 이르는 다양한 모바일 비디오 솔루션을 제공한다. 또한 10KB의 적은 용량의 자바 플랫폼을 이용하여 메모리 효율성이 높고 단일 CPU에서 수천 명이 동시 접속해도 데이터 전송 시 발생하는 버퍼링 문제가 거의 발생하지 않는다. 따라서 현재 모바일 환경에서의 멀티미디어 사용은 TruVideo가 전체의 95%를 차지하고 있다. 하지만, 이러한 TruVideo도 파일 시스템 기반으로 동영상을 관리하고 있으며 생성된 MPEG-7 문서를 데이터베이스 기술을 이용하여 관리하는 기능은 지원하지 않는다.

따라서 본 논문에서는 모바일 기기에서 멀티미디어를 효과적으로 활용하기 위해서 내장형 데이터베이스 기술을 이용해 MPEG-7 문서를 관리하는 시스템을 제안한다.

3. MVMS의 설계

앞 장의 기존 연구들에 대한 분석을 바탕으로 본

논문에서는 모바일 환경에서 MPEG-7 기반의 비디오 검색과 관리를 효과적으로 수행할 수 있는 MVMS(MPEG-7 Based Video Management System)를 개발하였다. (그림 1)은 MVMS의 전체적인 구조로써 모바일 환경을 고려한 클라이언트/서버 구조이다.

서버 측에 비디오가 들어오면 MPEG-7 문서 생성 부시스템을 통해서 비디오에 대한 특징과 메타데이터를 추출하여 MPEG-7 문서를 생성하고, 이렇게 생성된 MPEG-7 문서를 MPEG-7 문서 관리 부시스템을 통해서 하부의 데이터 저장소에 비디오와 연관시켜 저장 관리하게 된다.

클라이언트 측은 다양한 비디오 검색을 위한 검색 인터페이스를 이용하여 서버와의 통신으로 비디오를 검색하게 된다. 검색한 결과인 비디오와 MPEG-7 문서를 클라이언트 측의 MPEG-7 문서 관리 부시스템을 통해서 하부의 데이터 저장소에 저장 관리하게 된다. 이렇게 검색된 결과인 비디오와 MPEG-7 문서를 클라이언트 측에서도 저장 관리함으로써 서버와의 통신이 단절되어도 클라이언트에 저장된 결과를 이용하여 비디오의 사용이 가능하게 된다.

3.1 MPEG-7 문서 생성 부시스템

(그림 2)는 MPEG-7 문서 생성 부시스템에서 사용하는 MPEG-7 비디오 계층 구조이다. MPEG-7은 이미지, 오디오, 비디오 등의 객체에 대한 다양한 형태의 정보를 표현할 수 있는 구조를 가지고 있다[2]. 하지만 본 논문은 비디오를 대상으로 하고 있기 때문에 관련된 요소들만을 추출하여 사용하였다[11].

VideoSegment는 분리된 비디오 영역에 대한

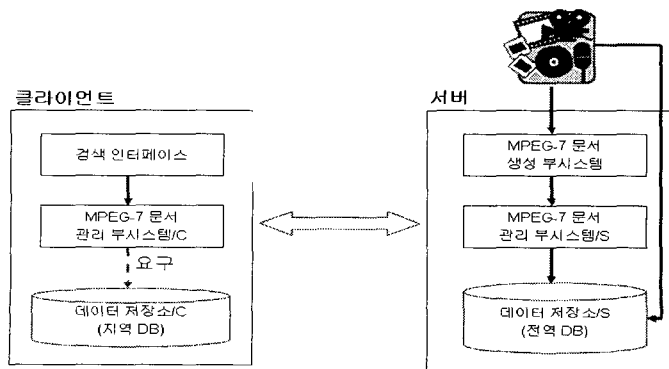


그림 1. MVMS의 전체 구조

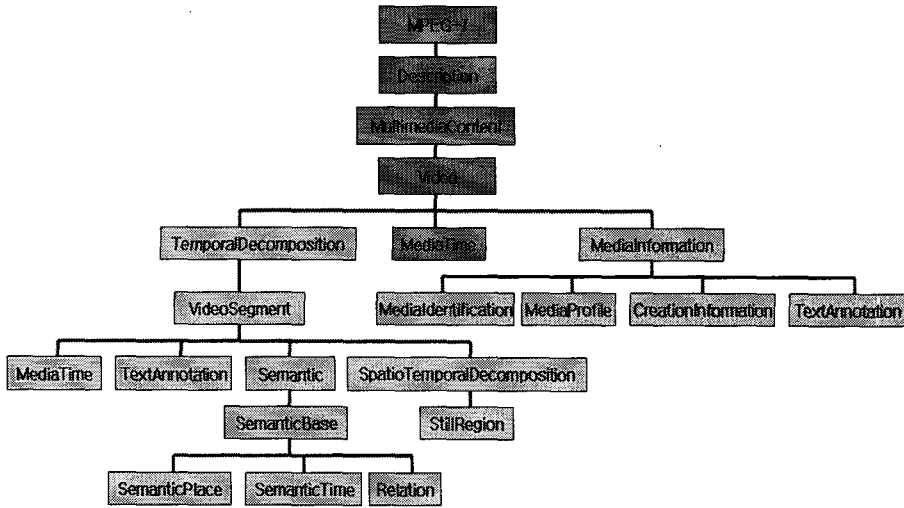


그림 2. MPEG-7 비디오 계층 구조

MediaTime, TextAnnotation, Semantic, Spatio-TemporalDecomposition 등을 가지게 된다. Semantic은 비디오 내용에 대한 SemanticTime, SemanticPlace 정보를 기본적으로 가질 수 있으며, SemanticBase 요소의 속성값을 이용하여 객체, 이벤트, 행위자 등에 대한 정보를 확장하여 표현할 수 있다. 뿐만 아니라 Semantic에서 표현된 정보들에 대한 Relation 설정을 통해서 해당 비디오 세그먼트 영역에 대한 정보를 가지게 된다. StillRegion은 정지 영상에 대한 정보를 표현할 수 있는 요소이다. 본 연구에서는 비디오 세그먼트에서 추출한 키 이미지에 대한 위치 정보를 표현하

기 위해서 StillRegion을 사용한다. MediaInformation은 비디오 정보를 표현하는 요소이다. 하위 노드인 MediaIdentification은 비디오의 압축 형식, 포맷 방법, 초당 프레임수, 재생 크기 등의 정보를 가지며, MediaProfile은 비디오의 위치, 파일명 등의 정보를 가지게 된다. CreationInformation은 비디오의 제작자, 제작사 등에 관련된 정보를 가지게 되며, TextAnnotation은 비디오 전체에 대한 주석 정보를 가지게 된다.

(그림 2)의 MPEG-7 비디오 계층 구조를 바탕으로 MPEG-7 문서를 생성하는 MPEG-7 문서 생성 부시스템의 구성은 (그림 3)과 같다.

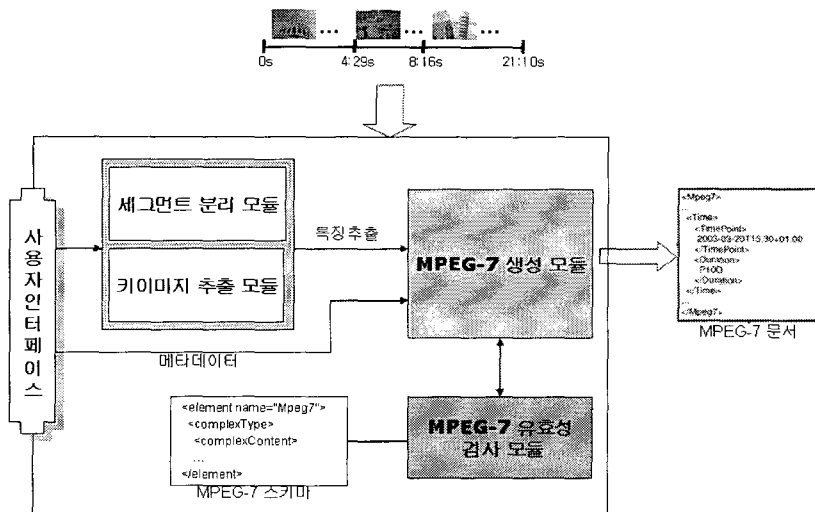


그림 3. MPEG-7 문서 생성 부시스템의 구성도

MPEG-7 문서 생성 부시스템에 비디오가 들어오게 되면 사용자 인터페이스를 통해서 수동적인 방법으로 비디오를 순차 세그먼트별로 나누고, 각 세그먼트별로 키 이미지를 추출하게 된다. 그리고 사용자가 세그먼트에 해당하는 메타데이터들을 입력하게 된다. 이러한 정보를 바탕으로 MPEG-7 문서를 생성하게 된다. 이때 MPEG-7 스키마(Schema)를 바탕으로 MPEG-7 문서에 대한 데이터타입과 구조 등의 유효성 검사를 병행하게 된다.

세그먼트 분리 모듈은 비디오 전체에 대한 시간 정보를 추출한 후, 비디오가 재생되는 동안 사용자가 세그먼트를 분배하기 원하는 위치를 선택하면 그 위치 정보를 추출하여 분리된 세그먼트의 시간 정보를 기록한다. 키 이미지 추출 모듈은 분리된 세그먼트에 대해서 키 이미지를 추출하여 이미지 파일을 생성하고 키 이미지를 추출한 위치 정보를 기록하게 된다. MPEG-7 생성 모듈은 각 모듈에서 추출된 비디오 정보와 사용자가 입력한 메타데이터를 모아 MPEG-7 기술자와 기술 스킴을 이용하여 MPEG-7 문서를 생성한다. MPEG-7 유효성 검사 모듈은 MPEG-7 문서가 MPEG-7 표준을 따르는지 검사한다. MPEG-7 표준 스키마를 토대로 하여 생성된 MPEG-7 문서의 구조와 요소들의 데이터타입이 유효한지를 검사한다.

3.2 MPEG-7 문서 관리 부시스템

(그림 4)는 MPEG-7 문서 관리 부시스템의 구성

도이다.

MPEG-7 문서 생성 부시스템으로 부터 MPEG-7 문서가 입력되면 MPEG-7 문서 관리 모듈을 통해서 하위의 전역 DB에 저장되고, 비디오 데이터와 추출된 키 이미지는 비디오 데이터 관리 지원 모듈을 통해서 전역 DB에 저장된다. 또한 클라이언트로부터 검색 요청이 들어오게 되면, 검색 키워드를 바탕으로 XQuery 생성 모듈을 통해서 XQuery로 생성한 후, MPEG-7 문서 관리 모듈을 통해서 전역 DB에서 해당 데이터를 검색하여 클라이언트로 전송하게 된다. 한편, 클라이언트 측으로 검색 결과를 전송할 때는 MPEG-7 전송 모듈을 통해서 MPEG-7 문서를 이진 형태로 변환하여 전송한다. 클라이언트 측에도 검색된 비디오를 저장 관리하기 위해서 이러한 MPEG-7 문서 관리 부시스템을 가지게 되는데, 클라이언트 측에서는 모바일 단말기의 환경을 고려하여 최소한의 기능만을 가지도록 설계하였다.

XQuery 생성 모듈은 검색 인터페이스를 통해서 들어온 검색 요구를 XQuery 표현으로 변환한다. MPEG-7 문서 관리 모듈은 하부의 내장형 XML DB[12]에서 MPEG-7 문서를 관리하기 위한 모듈이다. 뿐만 아니라 클라이언트 측의 검색 요청인 XQuery를 실행하여 그 결과를 반환하게 된다. 비디오 데이터 관리 지원 모듈은 입력된 비디오와 추출된 키 이미지의 저장 위치를 관리하고, 하부의 내장형 DB에 MPEG-7 문서에 해당하는 키 이미지를 연결시켜 주는 매핑 정보를 관리하게 한다. MPEG-7 전송 모듈은 전체 MPEG-7 문서와 XQuery의 검색 결

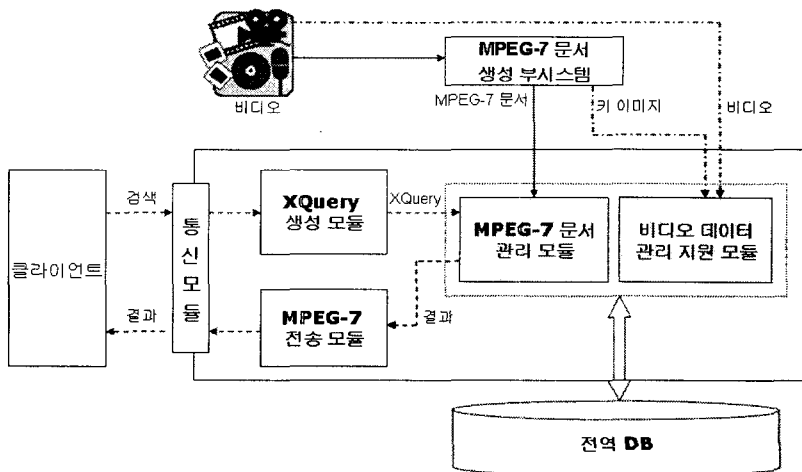


그림 4. MPEG-7 문서 관리 부시스템의 구성도

과를 데이터 손실을 막기 위해 이전 형태로 변환하여 클라이언트 측에 전송하게 된다.

3.3 내장형 데이터 저장소

내장형 데이터 저장소는 MPEG-7 문서를 저장하기 위해 내장형 XML 데이터베이스 시스템인 버클리 DB XML[13]을 사용하고 키 이미지의 매핑 정보를 관리하기 위해 레코드 기반의 내장형 데이터베이스 시스템인 버클리 DB[13]를 사용하였다. 내장형 XML 데이터베이스에서는 윈도우의 디렉토리 개념인 컨테이너(Container)를 사용하여 체계적으로 XML 문서를 관리하고 내장형 데이터베이스에서는 키-데이터(Key-Data)의 쌍으로 모든 데이터를 관리한다. (그림 5)는 MPEG-7 문서, 비디오, 키 이미지 정보를 저장하기 위한 데이터 저장소의 구성이다

(그림 5)에서 보는 바와 같이 내장형 데이터 저장소에서는 내장형 XML 데이터베이스를 이용하여 최상위에 분류 컨테이너를 두고 그 하위에 각 분류마다 해당 컨테이너를 두어 동일한 분류의 MPEG-7 문서는 같은 컨테이너에서 저장 관리되도록 하였다. 분류 컨테이너에는 (그림 7)과 같은 XML 문서가 들어가게 된다. 여기서, <subset> 엘리먼트의 id 속성값이 'grouping'이면 분류 컨테이너에서 계속해서 <location>

엘리먼트가 가리키는 하위 노드들을 찾게 되고, <subset> 엘리먼트의 id 속성값이 'MPEG' 이면 <location> 엘리먼트가 가리키는 컨테이너의 위치를 찾아 해당 MPEG-7 문서를 저장하게 된다. 이러한 분류 개념을 사용함으로써 다양한 형태의 사용자 정의 분류가 가능하게 된다. 예를 들어, 하나의 비디오에 여행이라는 분류 뿐만 아니라 문화, 축제 등과 같은 다양한 분류를 접목시켜서 저장 관리할 수 있게 된다.

MPEG-7 문서 생성 부시스템에서 생성된 키 이미지와 MPEG-7 문서의 매핑 정보를 관리하기 위해서 내장형 데이터베이스를 이용하였다. (표 1)은 키 이미지 매핑 정보를 관리하기 위한 내장형 데이터베이스의 테이블 구조이다.

표 1. 키 이미지 매핑 정보 테이블 구조

필드명	데이터타입	비고
Mpeg_id	char[50]	Not null
Vseg_id	char[10]	Not null
Ifile_location	char[200]	null
Ifile_name	char[50]	null

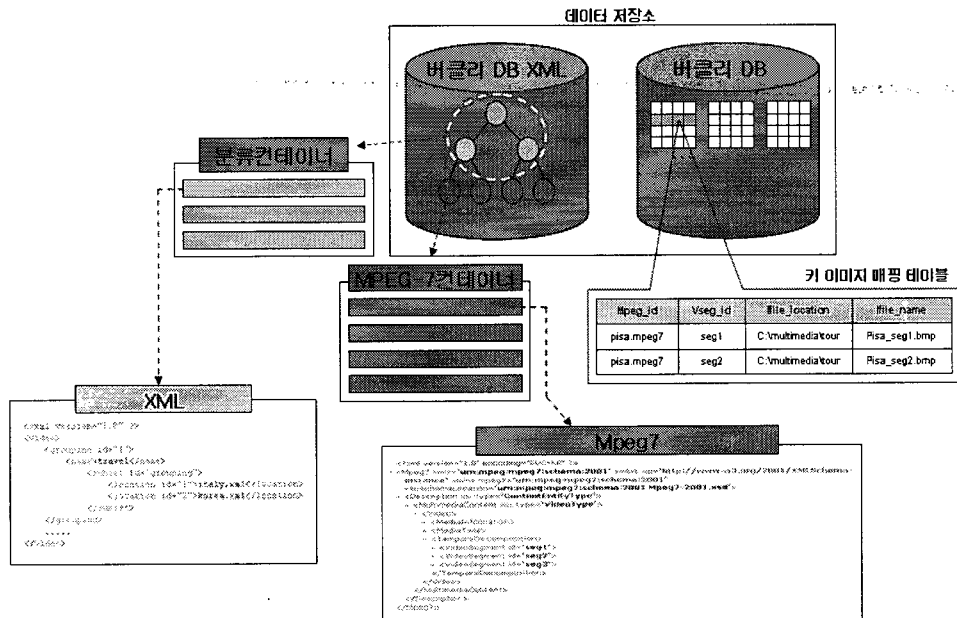


그림 5. 데이터 저장소의 구성

```

<?xml version="1.0" ?>
- <video>
- <grouping id="1">
  <name>travel</name>
  - <subset id="grouping">
    <location id="1">italy.xml</location>
    <location id="2">korea.xml</location>
  </subset>
</grouping>
+ <grouping id="2">
</grouping>
</video>
    
```

상위 노드 XML

```

<?xml version="1.0" ?>
- <video>
- <grouping id="1">
  <name>italy</name>
  - <subset id="MPEG">
    <parent>travel</parent>
    <location>tour.dbxml</location>
  </subset>
</grouping>
</video>
    
```

최하위 노드 XML

그림 6. 분류 컨테이너의 XML 문서 형태

Mpeg_id 필드에는 MPEG-7 문서의 이름이, Vseg_id에는 키 이미지를 추출한 비디오 세그먼트의 id 속성 정보가 저장되며, Ifile_location에는 실제 키 이미지들이 저장되어 있는 경로를 가리키는 정보가 저장되고, Ifile_name에는 키 이미지 파일명 정보가 저장된다. 이러한 매핑 테이블을 사용함으로써 모바일 단말기에서 비디오 검색이 이루어질 때 검색 결과로 MPEG-7 문서 뿐만 아니라 해당 키 이미지를 함께 전송함으로써 검색 결과에 대한 신뢰성을 높여 줄 수 있게 된다.

클라이언트 측 데이터 저장소는 서버 측의 데이터 저장소와 같은 형태로 구성되지만 클라이언트 측은 개인 사용자를 위한 모바일 단말기 환경을 고려하여

트랜잭션 처리, 병렬 처리와 같은 부분들을 배제한 내장형 데이터베이스 시스템으로 구성하였다.

3.4 검색 과정

비디오 데이터를 검색하는 과정을 좀 더 자세히 살펴보도록 하겠다. (그림 7)는 전체적인 비디오 검색 과정을 나타내고 있다.

기본적으로 내장형 데이터베이스에서의 검색은 5 단계 과정을 거치게 된다. ① 검색 요청이 들어오면 XQuery 생성 모듈을 통해서 검색 요청 데이터를 이용하여 XQuery를 생성한다. ② 검색 요청 데이터 중 분류 정보가 있다면 내장형 XML 데이터베이스의 분류 컨테이너를 검색한다. 분류 컨테이너의 최상위 노

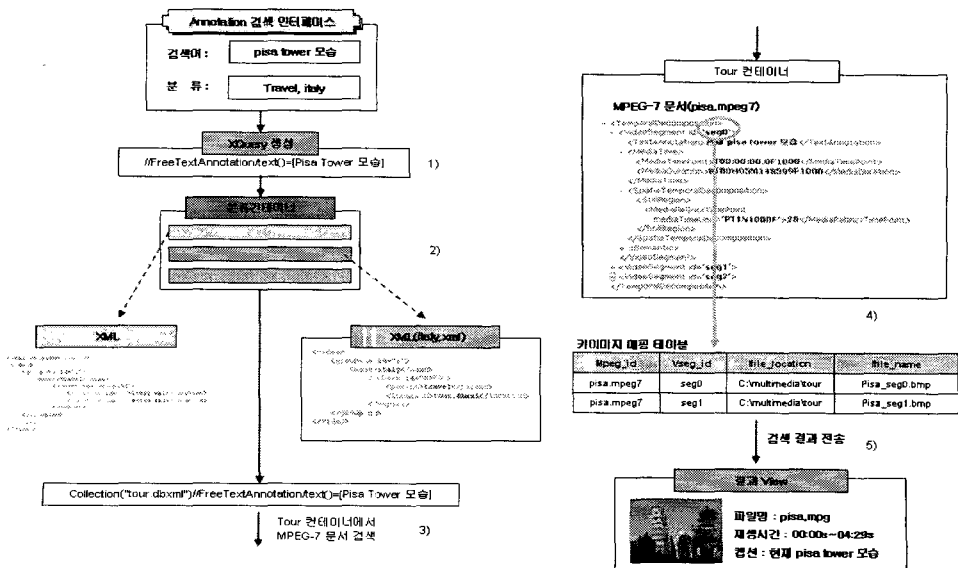


그림 7. 비디오 검색 과정

드의 XML 문서를 이용하여 하위 노드의 XML 문서를 검색한다. XML 문서의 <subset> 요소의 id 속성값이 'grouping'이면, <location> 요소가 가리키는 XML 문서를 계속 검색한다. 그리고 id 속성값이 'MPEG'이면, <location> 요소가 가리키는 컨테이너의 위치를 반환한다. ③ (그림 11)에서와 같이 검색된 컨테이너에 XQuery를 실행하여 조건을 만족하는 MPEG-7 문서를 찾는다. ④ 검색 조건을 만족하는 MPEG-7 문서를 발견하면 <Segment> 요소의 id 속성값을 이용하여 내장형 데이터베이스에 저장되어 있는 키 이미지의 위치 정보를 검색한다. ⑤ XQuery의 결과와 키 이미지를 반환한다.

4. MVMS의 구현

이 장에서는 앞 장의 설계를 바탕으로 MVMS를 구현하였다.

서버 측의 데스크탑 컴퓨터에서는 MPEG-7 문서를 관리하기 위해서 버클리 DB XML, 키 이미지 매핑 정보를 관리하기 위해서 버클리 DB를 사용하였고 시스템 개발 언어는 Visual C++을 사용하였다. XML 유효성 검증 파서로는 Xerces-C++[14]를 사용하고 비디오 세그먼트 조작을 위해서는 DirectX 9.0[15]를 사용하였다. 클라이언트 측은 데스크탑 컴퓨터에서 윈도우 CE 에뮬레이터를 사용하여 구현하였으며 서버 측과 마찬가지로 내장형 데이터베이스를 사용하였고 개발 언어로는 Visual Studio.Net을

사용하였다.

4.1 생성 인터페이스

(그림 8)은 사용자 인터페이스를 통해서 비디오를 순차 구조의 세그먼트로 분리하고 각 세그먼트마다 주석 정보를 입력하는 생성 인터페이스이다. 이는 필요에 따라 생성된 세그먼트별 키 이미지를 생성할 수 있는 기능을 제공한다. 세그먼트 분리의 효율성을 높이기 위해서 앞뒤 3초 간격으로 비디오 탐색 가능하도록 하였으며 정확한 키 이미지 추출을 위해 1/24 초 단위로 비디오 탐색이 가능하도록 하였다. (그림 8)은 비디오를 3개의 세그먼트로 나누고 각 세그먼트마다 주석 정보와 키 이미지를 추출한 화면이다.

4.2 의미 정보 인터페이스

(그림 9)는 각각의 비디오 세그먼트에 의미 정보를 입력하기 위한 의미 정보 인터페이스이다.

화면에서 Semantic Type List에는 비디오에서 사용할 SemanticTime, SemanticPlace, Event, Object, Agent 등의 데이터를 입력할 수 있으며, 리스트의 데이터를 더블클릭하면 해당 데이터들이 Semantic View에 입력된다. 이를 통해서 비디오의 세그먼트에 대한 의미 정보를 추가할 수 있게 된다. (그림 9)의 Semantic View에 나타나 있는 화면은 "2002년 10월 14일 Pisa에서 홍길동이 pisa tower를 촬영했다"를 의미한다.

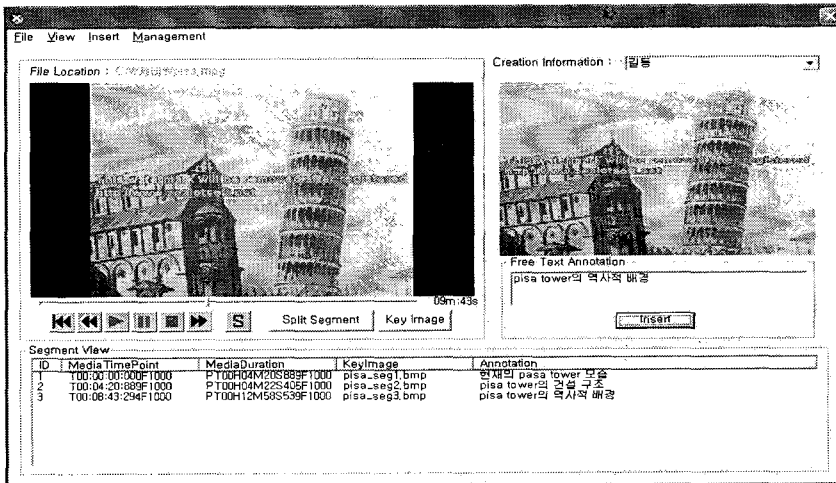


그림 8. MVMS의 생성 인터페이스

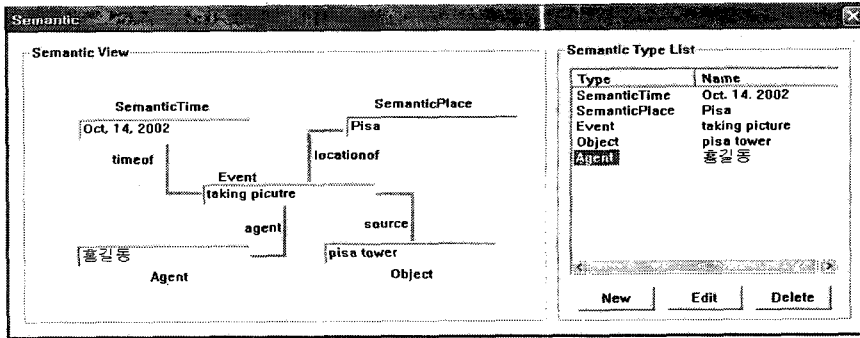


그림 9. 의미 정보 인터페이스

(그림 10)은 (그림 8)과 (그림 9)에서 본 생성 인터페이스와 Semantic 인터페이스 등을 통해서 MVMS에서 생성된 MPEG-7 문서의 예이다.

(그림 8)과 (그림 9)에서 살펴본 바와 같이 메인 인터페이스에서 분리한 3개의 <VideoSegment>, 키 이미지 정보를 가지는 <SpatioTemporalDecomposition>, 세그먼트의 시간 정보를 가지는 <MediaTime> 등을 확인할 수 있다. 그리고 Semantic 인터페이스에서 입력한 정보를 <SemanticPlace>, <SemanticTime>과

3개의 <SemanticBase>를 통해서 확인할 수 있다.

4.3 검색 인터페이스

클라이언트 측 모바일 단말기에서의 비디오 검색 과정은 (그림 11)과 같이 7단계로 구성된다. ① 클라이언트 측에서 검색 인터페이스를 이용하여 다양한 비디오 검색을 요청하면 ② 클라이언트측의 MPEG-7 문서 관리 부시스템이 구동되어 지역 DB에 질의를 요청한다. ③ 질의 결과는 검색 인터페이스를 통해

```

- <Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-Instance"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001" xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 Mpeg7-2001.xsd">
- <Description xsi:type="ContentEntityType">
- <MultimediaContent xsi:type="VideoType">
- <Video>
+ <MediaInformation>
+ <MediaTime>
- <TemporalDecomposition>
- <VideoSegment id="seg1">
- <TextAnnotation>현재 pisa tower 모습</TextAnnotation>
- <MediaTime>
- <MediaTimePoint>T00:00:00:0F1000</MediaTimePoint>
- <MediaDuration>PT00H04M20S889F1000</MediaDuration>
- <MediaTime>
+ <SpatioTemporalDecomposition>
- <Semantic>
- <Label>
- <Name />
- <Label>
- <SemanticBase xsi:type="EventType" id="TakingPhoto">
- <Label>
- <Name>taking picture</Name>
- <Label>
- <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:agent" target="#홍길동" />
- <Relation type="urn:mpeg:mpeg7:cs:SemanticRelationCS:2001:source" target="#Tower" />
- <SemanticPlace>
- <Label>
- <Name>pisa</Name>
- <Label>
- </SemanticPlace>
- <SemanticTime>
- <Label>
- <Name>October 14, 2002</Name>
- <Label>
- </SemanticTime>
- </SemanticBase>
- <SemanticBase xsi:type="ObjectType" id="Tower">
- <Label>
- <Name>Pisa Tower</Name>
- <Label>
- </SemanticBase>
- <SemanticBase xsi:type="AgentObjectType" id="홍길동">
- <Label>
- <Name>홍길동</Name>
- <Label>
- </SemanticBase>
- </Semantic>
- </VideoSegment>
+ <VideoSegment id="seg2">
+ <VideoSegment id="seg3">
- </TemporalDecomposition>
- </Video>
- </MultimediaContent>
- </Description>
</Mpeg7>
    
```

그림 10. MVMS에서 생성된 MPEG-7 문서의 예

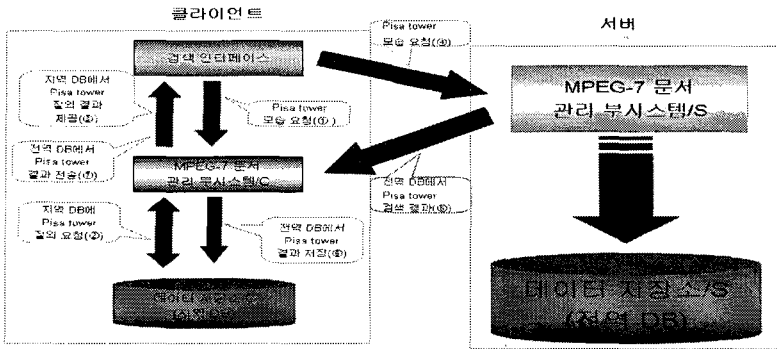


그림 11. 클라이언트측 모바일 단말기에서의 검색 과정

제공하며 ④ 지역 DB의 자료가 부족하거나 해당 자료가 없을 시 서버 측에 추가로 검색을 요청한다. ⑤ 서버 측에서는 전역 DB에 대해 질의를 처리해, 그 검색 결과를 클라이언트 측의 MPEG-7 문서 관리 부시스템에 제공하며 ⑥ 이렇게 제공된 전역 DB에 대한 검색결과는 지역 DB에 저장된다. 그리고 ⑦ 이는 계속해서 검색 인터페이스를 통해 사용자에게 추가로 보여준다. 이러한 7단계의 검색 과정을 거침으로써 클라이언트 측에서 서버와의 통신이 단절되어도 클라이언트 측에 있는 지역 DB에서 관리되고 있는 MPEG-7 문서 및 비디오를 사용하는 것은 가능하다. 뿐만 아니라 서버와의 통신 연결이 속개 되었을 때 지속적으로 전역 DB로부터 검색 결과를 가져올 수 있다. 이때 DB(전역이던 지역이던)에서의 검색은 3.4절에서 설명한 방법이 그대로 적용된다. 이러한

7단계 검색 과정을 통해 모바일 환경에서 비디오 검색이 보다 효과적으로 수행될 수 있다.

(그림 12(a))는 검색 인터페이스에서 검색어에 'pisa tower 모습'을 입력하고 '캡션에서 검색'을 선택한 후, travel/italy를 선택하여 검색을 요청했을 때, 우선 지역 DB에서 찾은 이미지(①~③까지 처리)를 나타낸 것이다. 검색 인터페이스는 지역 DB에서의 이미지 정보를 보여주면서 이미지 아래에 서버와의 접속 정보 상태를 나타내고 있다. 그 후, (그림 12(b))는 서버로 접속하여 전역 DB에서 해당 정보를 추가로 찾아 클라이언트 측의 검색 인터페이스에 전송한 결과(④~⑦까지 처리)이다. 이는 인터페이스의 추가(more) 버튼을 눌렀을 때 화면에 추가로 나타나게 된다. (그림 12(b))의 윗 부분은 클라이언트 측의 지역 DB에 있는 검색 결과가 나타난 것이며 아래 부분



(a) 지역 DB에서의 검색 결과 (b) 전역 DB에서의 검색 결과 추가

그림 12. 검색 결과

은 추가로 서버와의 통신 연결에 의해 서버측 전역 DB에서 검색한 결과가 나타난 것으로써 이는 지역 DB에서의 검색 결과를 제외한 나머지가 나타나게 된다.

5. 비교 분석

현재 모바일 환경에서 MPEG-7 기반의 내장형 데이터베이스를 이용한 비디오를 관리하는 시스템으로 기 발표된 것이 없기 때문에 우리 시스템과의 비교 대상을 찾기가 쉽지 않다. 따라서 이 장에서는 제안하는 MVMS와 기능 측면에서 유사한 DeskTop 환경에서의 MovieTool, VideoAnnEx과 모바일 환경에서의 파일 시스템을 이용한 TruVideo를 비교 대상으로 하였다. (표 2)는 MVMS, MovieTool, VideoAnnEx, TruVideo의 기능 및 성능을 비교 분석한 결과이다.

MPEG-7 기술자 항목에서 MVMS는 비디오의 순차 구조를 기술하는 VideoSegment를 지원하고, 고정 관계 설정을 통해 의미 정보를 기술하는 Semantic을 일부 지원한다. 비디오 정보를 기술하는 MediaInformation과 주석 정보를 기술하는 FreeTextAnnotation은 지

원하지만 StructureTextAnnotation을 지원하지 않아 주석 정보를 구조적으로 기술하기는 불가능하다. MovieTool은 비디오의 순차 구조와 계층 구조를 기술하는 VideoSegment를 지원하고, StructureTextAnnotation은 지원하지만 의미 정보를 기술하는 Semantic을 지원하지 않는다. VideoAnnEx는 MPEG-7 기술자 항목에서 다양한 기능을 지원하지만 의미 정보를 기술하는 Semantic은 지원하지 않는다. TruVideo는 VideoSegment, MediaInformation, FreeTextAnnotation은 지원하지만 Semantic 및 StructureTextAnnotation은 지원하지 않는다.

검색 항목에서 MVMS는 MPEG-7 기술자를 바탕으로 비디오에 대한 세그먼트별 검색을 지원하고 의미 검색과 주석 검색의 일부를 지원한다. MovieTool은 세그먼트별 검색과 주석 검색을 지원하지만 의미 검색은 지원하지 않는다. VideoAnnEx는 주석 검색을 지원하지만 세그먼트별 검색은 일부만 지원하고 의미 검색은 지원하지 않는다. TruVideo는 주석 검색은 지원하지만 의미 검색과 세그먼트별 검색은 지원하지 않는다.

MVMS는 비디오 세그먼트별 키 이미지 추출과 MPEG-7 문서 편집 기능을 지원하고 내장형 데이터

표 2. MVMS, MovieTool, VideoAnnEx, TruVideo의 비교

항목		MVMS	MovieTool	VideoAnnEx	TruVideo
MPEG-7 기술자	VideoSegment	△	○	○	○
	Semantic	△	×	×	×
	MediaInformation	○	○	○	○
	FreeTextAnnotation	○	○	○	○
	StructureTextAnnotation	×	○	○	×
검색	의미 검색	△	×	×	×
	주석 검색	△	○	○	○
	세그먼트 검색	○	○	△	×
편집	키 이미지 추출	○	×	○	△
	MPEG-7 문서 편집 기능	○	○	○	×
관리	다중 문서 관리	○	×	×	×
	통합 환경	Mobile	Desktop	Desktop	Mobile
	사용 DB/File System	(Embedded) DB	특화된 File System	DB	특화된 File System
성능	처리 속도	보통	빠름	빠름	매우 빠름
	필요 저장 공간	보통	보통	많음	적음

(○ : 지원, △ : 일부 지원, × : 지원 안함)

베이스를 통해서 MPEG-7 데이터를 관리하기 때문에 다중 문서의 검색과 관리가 가능하며 저장은 내장형 데이터베이스를 사용한다. 하지만 MovieTool은 MPEG-7 문서의 편집 기능을 제공하기는 하지만 키 이미지 추출은 지원하지 않고, 파일 시스템을 기반으로 하나의 비디오에 대한 검색을 목적으로 하기 때문에 다중 문서의 검색을 지원하지 않는다. VideoAnnEx는 생성된 MPEG-7 문서와 키 이미지에 대한 관리는 지원해 주지 않으므로 사용자들이 다른 MPEG-7 관리 시스템을 이용하여 새롭게 관리해야 하며 저장은 일반 데이터베이스를 사용한다. TruVideo는 모바일 환경의 특화된 파일 시스템 기반으로 하나의 비디오에 대한 검색을 목적으로 하기 때문에 부분적인 키 이미지 추출은 가능하나 문서 편집 기능 및 다중 문서 관리는 지원하지 않으며 저장은 특화된 파일 시스템을 사용한다. 본 연구에서는 환경 기반이 다르고 유사 제품이 없는 관계로 일반적 형태의 성능을 비교 분석을 하였다. TruVideo는 처리 속도 면에서 가장 빠르고 필요 저장 공간도 가장 적게 차지한다. 필요 저장 공간은 클라이언트 측에서의 문서 관리 시 필요 공간으로 TruVideo가 가장 적은 공간을 차지한다. 그리고 TruVideo는 모바일 환경 기반의 상용 톨로써 전송 속도 및 버퍼가 거의 없으며 모바일 시장의 95% 이상을 차지하고 있지만 특화된 파일 시스템으로 전송되고 있다. 하지만, MVMS는 자체 내장형 데이터베이스를 기반으로 하고 있어 파일 시스템 기반의 TruVideo와는 성능 비교 분석이 어렵다. 즉, 기본적으로 특정 항목 검색 등의 일부 검색 처리 성능에 있어 보다 특화된 파일 시스템을 사용하는 TruVideo가 MVMS 보다 우수하다. 하지만 MVMS는 내장형 XML DBMS 기술에 기반하여 개발되었기 때문에 보다 일반적이고 다양한 질의가 가능한 질의어 XQuery, XPath 등을 사용할 수 있고 기타 DBMS 기본 기능인 트랜잭션 기능 등이 적용된다. 따라서, MVMS는 기존 시스템들과의 성능 분석을 토대로 장·단점을 제시하였으며 시스템에 대한 신뢰성을 높였다.

6. 결론 및 향후 과제

본 논문에서는 모바일 단말기를 위한 MPEG-7 기반의 비디오 관리 시스템 MVMS(MPEG-7 Based

Video Management System for Mobile Device)를 제안하였다. MVMS는 기존 시스템들과 달리 모바일 단말기에서 비디오의 효과적인 검색과 관리가 가능한 시스템으로 주요 특징은 다음과 같다.

첫째, MPEG-7 문서 생성과 관리의 통합 환경을 제공함으로써 많은 양의 MPEG-7 문서 검색의 일괄 처리가 가능하다.

둘째, 모바일 환경에서 MPEG-7 기반의 비디오 검색이 가능하기 때문에 시간과 장소에 구애받지 않고 원하는 비디오를 검색할 수 있다.

셋째, 모바일 단말기에서 검색된 비디오만의 관리를 지원하기 때문에 리소스가 부족한 모바일 단말기에서 효율적으로 비디오를 관리할 수 있다.

현재 MVMS는 고정적인 관계 설정을 통한 의미 기반 검색의 일부분만을 지원하고 있다. 따라서 완전한 의미 기반의 검색을 지원하기 위해서는 다양한 관계 설정이 가능하도록 추가 개발이 필요하다. 그리고 클라이언트로부터 비디오의 일부를 전송받아 서버 측의 비디오를 검색할 수 있는 검색 기법들의 개발이 필요하다. 또한 MPEG-7 문서의 효율적인 저장을 위해서는 MPEG-7의 멀티미디어 기술 스킴(MDS)을 기반으로 MPEG-7 저장 기법을 연구할 필요가 있다.

참고 문헌

- [1] F. Nack, and A. Lindsay, "Everything You Want to Know About MPEG-7: Part1 and Part2," *IEEE Multimedia*, Vol. 6, No. 3, pp. 65-77, July 1999.
- [2] J. Martínez, "Overview of the MPEG-7 Standard, ISO/IEC JTC1/SC29/WG11," N4980 (Klagenfurt Meeting), July 2002, <http://www.chiariglione.org/mpeg/>.
- [3] H. Kosch, "Distributed Multimedia Database Technologies Supported by MPEG-7 and MPEG-21," *Auerbach Publications*, 2004.
- [4] J. Smith, "MPEG-7 Industry Use V.2.0. ISO/IEC JTC1/SC29/WG 11/N5928," October 2003, <http://www.chiariglione.org/mpeg/>
- [5] N. Day, "MPEG-7 Projects and Demos ISO/IEC JTC1/SC29/WG11/N4034," March

2001, <http://www.chiariglione.org/mpeg/>

[6] P. Beek, A. Benitez, J. Heuer, J. Martinez, P. Salembier, J. Smith, and T. Walker, "MPEG-7: Multimedia Description Schemes, ISO/IEC FDIS 15938-5:2001," *International Standard Document*, 2001.

[7] Richo MovieTool Home, <http://www.ricoh.co.jp/src/multimedia/MovieTool/index.html>

[8] AlphaWorks, "IBM MPEG-7 Annotation Tool," <http://www.alphaworks.ibm.com/tech/videoannex>

[9] Singingfish Home, "<http://www.singingfish.com>".

[10] TruVideo Home, <http://www.truvideo.com/products/index.html>

[11] T. Sikora, "The MPEG-7 Visual Standard for Content Description - An Overview," *IEEE Trans. Circuits System and Video Tech.* 11, pp. 703 - 315, 2001.

[12] Sleepycat사, "Berkeley DB," http://sleepycat.com/products/pdfs/wp_perf_0705c.pdf

[13] Sleepycat사, "Berkeley DB XML, Berkeley DB," http://www.sleepycat.com/xmldocs/intro_xml/BerkeleyDBXML-Intro.pdf

[14] Apache사, "Xerces-C++", <http://xml.apache.org>

[15] 마이크로소프트사, "DirectX", <http://www.microsoft.com/korea/directx>



안 병 태

1999년 국민대학교 컴퓨터과학부 졸업
 2001년 경남대학교 컴퓨터공학부 졸업(석사)
 2003년~현재 경상대학교 컴퓨터 과학과 (박사 수료)

관심분야 : MPEG-7, 멀티미디어 데이터베이스, XML, 유비쿼터스, 내장형 DBMS



민 정 훈

2003년 경상대학교 컴퓨터과학과 졸업
 2005년 경상대학교 컴퓨터과학과 석사졸업
 관심분야 : MPEG-7, 멀티미디어 데이터베이스, 대용량 데이터베이스



강 병 수

2004년 경상대학교 컴퓨터과학과 졸업
 2006년 경상대학교 컴퓨터과학과 졸업(석사)
 2006년~현재 경상대학교 컴퓨터 과학과 박사과정

관심분야: MPEG-7, 내장형 DBMS, 데이터마이닝, USN



강 현 석

1981년 동국대학교 전자계산학과 졸업
 1981년 서울대학교 계산통계학과 졸업(석사)
 1986년 서울대학교 계산통계학과 졸업(박사)
 1981년~1984년 한국전자통신연

구원

1984년~1993년 전북대학교 부교수
 1993년~현재 경상대학교 컴퓨터과학과 교수
 1993년~현재 경상대학교 컴퓨터.정보통신 연구소 연구원
 관심분야 : MPEG-7, 멀티미디어 데이터베이스, 내장형 데이터베이스