

멀티미디어 검색 시스템의 설계 및 구현

(Design and Implementation of Multimedia Retrieval a System)

노 승 민 [†] 황 인 준 ^{**}
(Seung Min Rho) (Een Jun Hwang)

요 약 최근 들어 멀티미디어 정보의 사용량이 증가하면서 멀티미디어 데이터베이스로부터 오디오나 비디오, 이미지 등 다양한 형태의 멀티미디어 콘텐츠를 효과적으로 찾아내는 멀티미디어 검색 시스템의 필요성이 증가하였다. 본 논문에서는 기존의 주석 및 내용 기반 검색 기법을 상호 보완하고 효과적인 멀티미디어 데이터 검색을 지원하는 XML 기반의 새로운 검색 기법과 이를 위한 데이터 모델을 제시한다. 이미지 및 비디오에 대한 데이터 모델은 MPEG-7 표준에 정의되어 있는 멀티미디어 기술 구조(MDS)와 기술 정의 언어인 XML Schema를 사용하여 멀티미디어 데이터의 특성 및 계층구조를 표현하였고 오디오 데이터의 경우 음향 특징들로부터 추출된 음높이를 분석하여 UDR 스트링으로 변환하고 자주 검색된 멜로디의 관리를 통해 검색 성능을 향상하였다. 본 논문에서는 제안된 모델을 기반으로 검색 시스템을 구현하였으며 다양한 실험을 통하여 성능 평가를 하였다.

키워드 : 멀티미디어 데이터베이스, 이미지, 비디오, 오디오, 인덱싱, XML, MPEG-7

Abstract Recently, explosive popularity of multimedia information has triggered the need for retrieving multimedia contents efficiently from the database including audio, video and images. In this paper, we propose an XML-based retrieval scheme and a data model that complement the weak aspects of annotation and content based retrieval methods. The property and hierarchy structure of image and video data are represented and manipulated based on the Multimedia Description Schema (MDS) that conforms to the MPEG-7 standard. For audio contents, pitch contours extracted from their acoustic features are converted into UDR string. Especially, to improve the retrieval performance, user's access pattern and frequency are utilized in the construction of an index. We have implemented a prototype system and evaluated its performance through various experiments.

Key words : Multimedia Database, Image, Video, Audio, Indexing, XML, MPEG-7

1. 서 론

최근 웹의 급속한 발전으로 인한 웹 콘텐츠의 증가와 네트워크의 고속화 그리고 압축기술의 발전에 힘입어 멀티미디어 정보의 활용이 일반화 되었다. 멀티미디어 정보 사용의 증가와 함께 멀티미디어 정보를 생성하고, 처리하며 필요한 정보를 효과적으로 검색하는 등의 요구가 증가하게 되었다. 사용자가 원하는 멀티미디어 정보를 보다 효과적으로 찾을 수 있는 멀티미디어 검색기능은 VoD(Video on Demand) 서비스나 원격 교육, 전자 도서관, 디지털 방송 등의 많은 응용분야에서 필수적이다.

멀티미디어 데이터의 검색은 사용되는 특징에 따라 주석 기반 검색과 내용 기반 검색으로 나누어진다. 내용 기반 검색(content-based retrieval)은 멀티미디어 데이터에서 모양, 색상, 질감, 움직임 등의 특징적인 내용을 표현하는 데이터를 추출하여 저장하고 이를 바탕으로 이미지나, 비디오 클립, MIDI, 허밍(humming) 등에 대한 질의를 처리하는 방법이다. 따라서 사용자의 주관이 개입되지 않아 객관적인 검색이 가능하고, 특정 도메인에서 좋은 결과를 나타내지만 데이터의 정확한 내용 표현이 어렵다. 이미지의 경우 색상, 질감, 형태, 위치 등 시각적인 정보, 히스토그램, 윤곽선 검출(edge detection)을 사용한 검색을 한다. 비디오의 경우 미디어 특성상 장면(scene), 샷(shot), 프레임(frame) 등의 계층적인 구조로 표현될 수 있기 때문에 비디오 분석을 통하여 그 경계를 검출하는 것이 필요하다. 내용 기반의 비디오 검색은 이렇게 검출된 샷 정보와 프레임에 대한 색상 히스토그램과 대표 색상을 이용한 검색을 한다. 오

· 본 연구는 한국과학재단 목적기초연구 R05-2002-000-01224-0 지원으로 수행되었음

[†] 학생회원 : 아주대학교 정보통신 전문대학원
anycall@ajou.ac.kr

^{**} 종신회원 : 아주대학교 정보통신전문대학원 교수
ehwang@ajou.ac.kr

논문접수 : 2002년 11월 18일

심사완료 : 2003년 5월 7일

디오 검색은 마이크로 폰을 통하여 입력 받은 사용자의 허밍이나 MIDI 파일로부터 음의 크기, 높이, 대역폭 등의 음향 특징들을 비교하여 검색 한다.

반면에 주석 기반 검색(annotation-based retrieval)은 멀티미디어 데이터에 대한 의미정보를 사용자가 결정하고 이를 주석으로 작성하여 저장한 후에 질의에 나타난 내용들을 주석과 비교하여 관련된 데이터를 찾는 방법이다. 이 방법은 사람이 인식할 수 있는 모든 내용을 주석으로 처리할 수 있기 때문에 여러 복잡한 개체간의 관계들로 이루어진 자료의 의미를 쉽게 모델링 할 수 있다는 장점이 있다. 하지만, 대용량의 데이터에 대하여 사람이 일일이 주석을 작성해야 하며 사용자의 관점에 따라 내용을 다르게 해석하거나 주석을 다르게 부여할 수 있으므로 데이터에 대한 일관성을 잃기 쉽다.

따라서 이 두 가지 방법이 가지는 장점을 함께 수용하고 단점을 상호 보완하여 사용한다면 효과적인 검색을 할 수 있을 것이다. 본 논문에서는 MPEG-7을 기반으로 하여 멀티미디어 데이터에 대해 XML로 주석 처리하여 저장하고 검색하는 시스템을 구현하였다. 이미지에 대한 주석을 처리하기 위해 색상 히스토그램과 사용자가 원하는 오브젝트를 분리하고, 오브젝트간의 공간 관계를 분석[1]하여 XML로 표현 하였다. 오디오 데이터는 FAI(Frequently Accessed Index) 기반의 새로운 내용 기반 오디오 검색 기법[2]을 사용하여 데이터베이스 전체를 검색하기 이전에 사용자들의 검색 패턴을 활용하여 검색 처리 시간을 줄였다. 비디오 데이터는 샷 검출과 자막을 이용한 분석과 함께 스키마에서 정의된 각각의 도메인에 따른 유연한 인터페이스를 통하여 주석 및 검색을 할 수 있도록 구현하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 통하여 기존의 멀티미디어 검색 및 주석 시스템과 그에 대한 여러 검색 및 주석 방법들에 대한 기술과 MPEG-7에 대해 간단히 기술하였으며 3장에서는 본 논문에서 구현한 시스템의 개략적인 구조를 설명한다. 4장에서는 주석 기반과 내용 기반 검색 기법과 이를 보완하기 위한 새로운 데이터 검색 기법들을 통합하는 메타 데이터 모델링 방법과 주석 처리기법을 이미지와 비디오를 중심으로 설명하며, 5장에서는 오디오 데이터에 대한 FAI 기반의 인덱싱 메커니즘에 대해서 알아본다. 6장에서는 스키마를 이용한 통합 검색 및 주석 시스템의 구현과 실험에 대해서 기술하며 마지막으로 7장에서는 결론과 향후 계획에 대해서 논의한다.

2. 관련 연구

멀티미디어 검색 시스템은 크게 이미지, 오디오 그리고 비디오 검색 시스템으로 나눌 수 있다. 우선 이미지

검색 시스템으로는 POSTGRES 데이터베이스를 기반으로 하는 Chabot[3]이 있으며, 각 이미지가 가질 수 있는 속성들과 개체들에 대한 설명을 이용한 텍스트 기반의 검색과 사용자에게 의해 다양한 형태로 정의될 수 있는 concept query, 그리고 색상 정보를 이용한 내용 기반 검색을 지원한다. MIT Media laboratory의 Photo-book[4] 프로젝트는 foreground extraction을 통해 이미지 로드시에 얼굴, 모양, 텍스트 등으로 분류하고, 이를 이용하여 이미지로부터 얼굴을 찾아낸다. Columbia 대학에서 개발된 VisualSEEK[5] 시스템은 색상과 공간 질의를 제공하기 위한 이미지 데이터베이스 시스템이다. 이미지의 구분은 색상 히스토그램과 같은 특성에 의하여 이루어지는데, 이미지 비교를 위해서 이미지의 영역과 색상, 크기, 공간적 위치들을 추가적으로 사용하는 방식이다. Blobworld[6]는 Berkeley 대학의 디지털 도서관 프로젝트의 하나이며 이미지내의 개체들로 구분되는 영역들을 이용한 내용 기반 검색 시스템이다. 우선 사용자로 하여금 카테고리들 선택함으로써 검색 범위를 제한하고 blob이라고 불리우는 영역을 선택하고, 선택된 blob의 색상, 질감, 위치나 형태 등의 추가 정보를 가지고 검색한다.

오디오 검색 분야에서 선구적인 연구 내용을 소개하고 있는 Ghias[7]는 마이크로 폰을 통하여 받은 사용자의 허밍에서 음높이 변화(pitch contour)를 감지하여 UDR(up, down, repeat) 스트링으로 표현하고 오디오 데이터베이스의 콘텐츠와 비교해서 유사한 멜로디를 찾아내는 시스템을 소개하고 있다. 웹기반 오디오 검색 시스템인 MELody index[8] 역시 마이크로 폰을 이용해서 사용자의 질의를 받고 질의 멜로디와 오디오 데이터베이스 내의 멜로디를 비교해서 유사한 정도에 따라 후보 멜로디의 리스트를 보여 주고 재생해볼 수 있도록 하였다. 오디오 검색에서는 멜로디의 음향, 간격, 리듬을 이용했고 멜로디 UDR 스트링 매칭을 위해서 다이나믹 프로그래밍 알고리즘을 이용한 유사 검색 기법을 사용하였다. Themefinder[9]는 16세기의 서양 클래식 음악, 포크송들을 대상으로 Humdrum 명령을 사용하여 웹상에서 사용자가 원하는 곡의 테마를 찾을 수 있는 시스템이다.

QBIC[10]은 IBM에서 개발한 정지영상 및 동영상 검색 엔진이며 QBE(Query by Example)를 통한 유사도 질의와 색상이나 질감 등을 이용한 사용자의 스케치를 통해서 질의를 할 수 있게 해주는 상용화된 영상 검색 시스템이다. TIVRON[11]은 OODBMS(Object-Oriented Database Management Systems) 상에 구축된 MPEG 비디오 데이터베이스로부터 사용자가 원하는 비디오를 보다 효율적으로 찾기 위해 주석 기반 검색과

내용 기반 검색을 독립적으로 지원할 뿐만 아니라 두 기법들을 함께 사용한 복합 비디오 검색까지 지원하는 통합 비디오 검색 시스템이다. SMOOTH Video DB [12]는 Klagenfurt 대학에서 제안한 비디오 인덱싱 모델인 VIDEX를 기반으로 하여 인덱스 데이터베이스를 구축하였고, 자바 기반의 주석, 질의 및 브라우징을 할 수 있는 클라이언트와 UDP와 RTP를 지원하는 비디오 서버로 구성되어 있다. SMOOTH는 이벤트, 개체, 사람, 위치와 같은 high-level의 의미 정보들과 히스토그램과 같은 물리적인 low-level 정보들을 저장하여 텍스트 기반의 질의를 지원한다. Vane[13]은 Tcl/Tk를 이용한 유연한 질의 및 주석 인터페이스와 SGML DTD를 이용한 데이터 모델을 구현하였다. 이때 사용된 비디오 데이터는 교육용 비디오와 뉴스이며, 이들 비디오 데이터로부터 DTD를 생성하거나 기존의 DTD를 이용하여 주석 및 질의를 하게 된다.

사용자가 원하는 멀티미디어 정보를 효율적으로 검색할 수 있도록 하기 위해서는 멀티미디어 데이터로부터 콘텐츠를 표현할 수 있는 특징들을 추출하고 표준화된 방식으로 기술하여 해당 콘텐츠와 함께 저장해야 한다. 이러한 동향의 하나로 현재 MPEG-7 표준화가 진행되고 있으며, 다양한 분야에서 필요로 하는 여러 특징들을 묘사하는 기술자(Descriptor: D)들의 집합을 규정하고 이들 기술자와 기술자들에 대한 구조, 그들 간의 연관성의 의미를 정의하는 기술구조(Description Scheme: DS)의 집합을 규정하며 이와 함께 새로운 기술자나 기술구조를 생성, 기존의 것을 확장하거나 수정할 수 있도록 하는 기술 정의 언어(Description Definition Lan-

guage: DDL)를 표준화 한다.

MPEG-7 표준을 따르는 멀티미디어 주석 및 검색 시스템을 구축하기 위해서는 우선 MPEG-7 기술자와 멀티미디어 기술구조(Multimedia Description Scheme: MDS)를 결정해야 한다. MPEG-7 기술자는 멀티미디어의 특징(feature)을 표현하는 가장 기본적인 구성요소로서 visual과 audio 부분으로 나누어지며 이미지와 같은 정지영상에서는 시각 기술자(visual descriptor)를 동영상에서는 시각과 음성 기술자(audio descriptor)를 사용한다. MPEG-7 기술구조는 멀티미디어를 표현하는 문서의 구조를 정의하며, XML 형태의 문서로서 기술자나 다른 기술구조를 이용해서 새로운 기술구조를 만들 수 있다.

본 논문에서는 이미지와 동영상의 메타데이터 구축에 직접적으로 관련된 메타 정보 기술구조(Meta Information DS)와 의미 기술구조(Semantic DS)를 중심으로 분석하여 정의 하였다.

3. 멀티미디어 검색 시스템

3.1 시스템 구조

그림 1은 본 논문에서 구현한 멀티미디어 검색 시스템의 전체적인 구조를 보여준다. 시스템은 크게 이미지, 오디오, 비디오 데이터와 같은 멀티미디어 데이터를 저장할 수 있는 데이터베이스와 질의 입력 처리기(Query Input Interface), 각각의 멀티미디어 데이터들을 분석하는 분석기(Data Analyzer), 멀티미디어 데이터로부터 추출된 특징들과 주석 처리된 내용들을 처리하는 주석 처리기(Annotation Interface), 분석 및 주석 데이터들

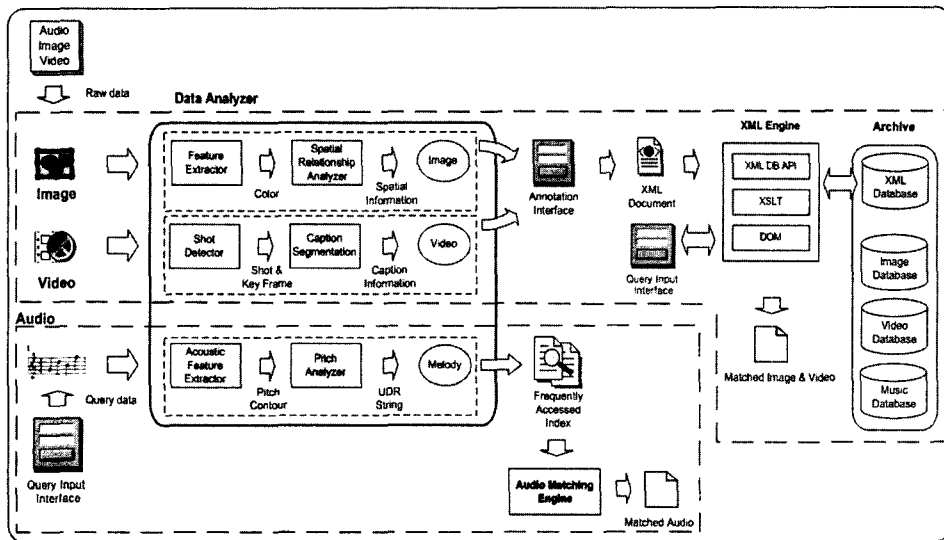


그림 1 멀티미디어 검색 시스템의 구조도

의 저장과 추출을 위한 XML DB API, XSLT, DOM 등을 포함하는 XML Engine, 저장된 오디오 콘텐츠와 질의를 비교하는 오디오 매칭 엔진(Audio Matching Engine)들로 구성되어 있다.

데이터베이스(Database): 우선 사용자들의 질의 입력을 받기 전에 멀티미디어 데이터의 특징들을 추출하여 콘텐츠와 함께 데이터베이스에 저장한다. 이미지의 경우는 이미지 데이터로부터 추출된 색상 히스토그램과 관심있는 오브젝트들간의 공간관계를 분석한 정보를 저장하며, 비디오는 검출된 샷 정보와 주석처리된 정보 및 자막을 저장한다. 오디오의 경우에는 오디오 콘텐츠를 미리 UDR 스트링으로 변환하여 실제 오디오 콘텐츠에 대한 포인터 정보와 함께 저장한다.

질의 입력 처리기(Query Input Interface): 질의 입력 처리기는 세부적으로 이미지, 비디오, 오디오로 나뉘며 이미지 질의는 캔버스(canvas)와 키워드를 이용하여 공간관계에 대한 질의를 하며 비디오의 경우 질의는 색상, 위치정보, 키워드 등을 이용하여 질의를 한다. 오디오 질의는 마이크로 폰을 이용한 허밍이나, 질의하고자 하는 멜로디를 오선지에 기입하는 CPN(Common Practice Notation)기반의 방식으로 표현한다. 사용자가 작성한 질의 멜로디는 MIDI 포맷으로 변환되어 저장되고 오디오 분석기(Audio Analyzer)에서 음향 특징을 추출하고 UDR 스트링으로 변환되는 데 사용된다.

데이터 분석기(Data Analyzer): 데이터 분석기는 크게 이미지 분석기, 오디오 분석기, 비디오 분석기로 나누며 이들 분석기들은 모두 원시 데이터(raw data)를 입력 받아서 색상 히스토그램, 키 프레임(key frame) 추출을 통한 샷 경계 검출(shot change detection), 음향 특징(acoustic feature)추출을 통한 음높이(pitch)의 분석 등 다양한 기능을 가지고 있다.

이미지 분석기는 특징 추출기(Feature Extractor)와 공간 관계 분석기(Spatial Relationship Analyzer)의 두 모듈로 구성되어 있다. 특징 추출기는 색상 히스토그램을 통해 이미지 데이터의 특징 중 색상정보를 추출하며 공간 관계 분석기는 사용자의 질의에 의해 작성된 질의 이미지와 위치 연산자를 이용하여 질의시 선택된 오브젝트의 관심 영역(Region of Interest)간의 공간 관계를 분석한다.

비디오 분석기는 샷 검출기(Shot Detector)를 통해 키 프레임을 추출하여 샷 경계 검출과 자막 데이터를 기반으로 세그멘테이션 처리 과정을 거치게 된다.

오디오 데이터는 음향 특징 추출기(Acoustic Feature Extractor)와 음높이 분석기(Pitch Analyzer)의 두 가지 모듈을 통하여 분석된다. 음향 특징 추출기는 MIDI 파일 안의 음막 재생시 필요한 여러 음향 정보 중에서

음높이(pitch) 값을 뽑아내어 사용자가 작성한 질의 MIDI 파일을 입력받아 모든 음표에 대한 음높이 값을 추출하여 순서대로 임시공간에 저장한다. 음높이 분석기는 음향 특징 추출기가 저장한 음높이값을 하나씩 비교하며 앞 음표의 음높이와 현재 음표의 음높이를 비교해서 현재 음표의 음높이가 상대적으로 높으면 U(up), 낮으면 D(down), 같으면 R(repeat)로 표시하는 방식으로 UDR 스트링을 생성한다.

주석 처리기(Annotation Interface): 이미지의 경우 분석기를 통해서 추출된 이미지 데이터가 가지는 특징들과 공간관계 분석정보를 팔레트(palette)를 이용하여 백그라운드나 특정 오브젝트의 색상을 구체적으로 선택하거나 키워드를 이용하여 주석처리를 한다. 비디오 역시 주석 처리 이전에 분석기를 통하여 검출된 샷 정보와 이들 샷들을 보면서 특정 비디오의 시간 위치에 대한 이벤트와 풍부한 정보들에 대해 서술할 수 있다.

XML 엔진(XML Engine): XML 스키마로부터 엘리먼트와 속성 등을 분석하여 주석 처리기를 통해서 입력받은 주석 정보와 데이터 분석기를 통해서 추출된 멀티미디어 데이터의 특징들을 XML 문서로 변환하여 데이터베이스에 저장한다. 또한, 질의시 XML 문서에 스타일 언어인 해당 XSLT 파일을 적용하여 질의 입력을 받고 XPath와 XQuery를 통하여 결과 값을 얻어낼 수 있다.

오디오 매칭 엔진(Audio Matching Engine): 오디오 매칭 엔진은 미리 UDR 스트링으로 변환되어있는 오디오 콘텐츠와 질의 UDR 스트링을 비교한다. UDR 스트링끼리의 비교에는 사용자 질의의 오류를 허용하기 위해 유사검색을 이용할 수 있지만 질의 멜로디가 길거나 오디오 데이터베이스의 콘텐츠 수가 많으면 시스템 성능저하를 가져올 수 있기 때문에 본 시스템에서는 정확 검색(exact matching) 방법을 사용했다.

3.2 주석 및 검색 인터페이스

이미지의 경우는 이미지 분석기를 통해 추출된 특징들과 이미지 파일이 가지는 물리적인 정보, 사용자에게 의해서 선택, 입력되는 텍스트 정보들을 데이터베이스에 저장한다. 비디오의 경우 화면 왼쪽에서 비디오 데이터가 재생이 되고, 샷 검출기에 의해 검출된 샷의 시작 및 끝 프레임 정보 등이 테이블의 형태로 보여지게 된다. 하단의 주석 입력 인터페이스를 통해 샷에 대한 정보를 추가, 수정, 삭제할 수 있으며 입력된 정보는 데이터베이스에 저장된다. 그림 2는 이미지와 비디오 데이터에 대해 주석 정보를 기술할 수 있는 인터페이스이다.

이미지의 주석 입력 과정이 끝나면 이미지내의 개체들간의 공간 관계 분석 과정을 거치게 된다. 이미지 객체들의 시각적인 특징과 공간적인 특징 정보를 이용하

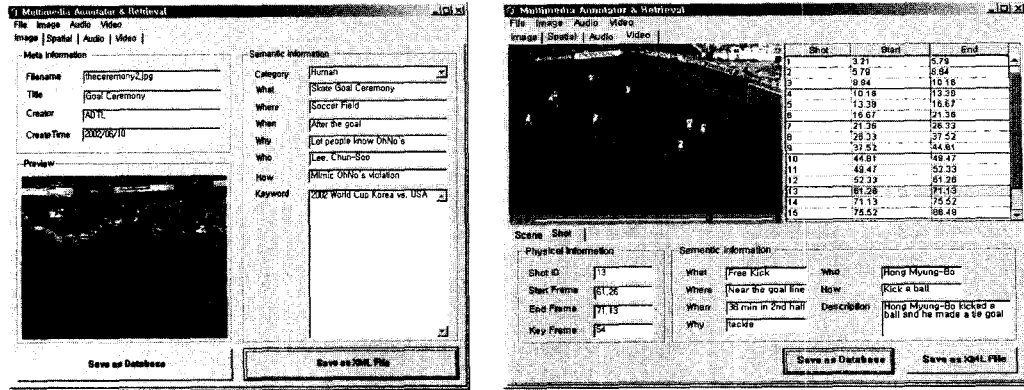


그림 2 이미지, 비디오를 위한 주석 처리 인터페이스

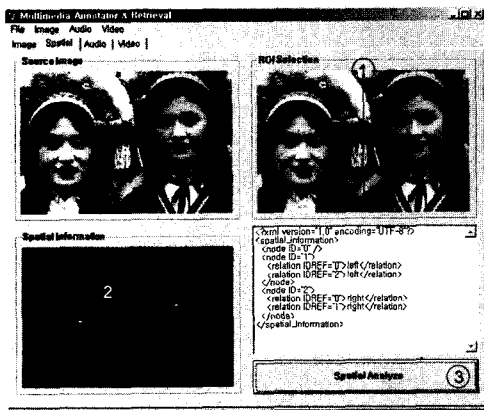


그림 3 ROI 선택 및 공간분석 인터페이스

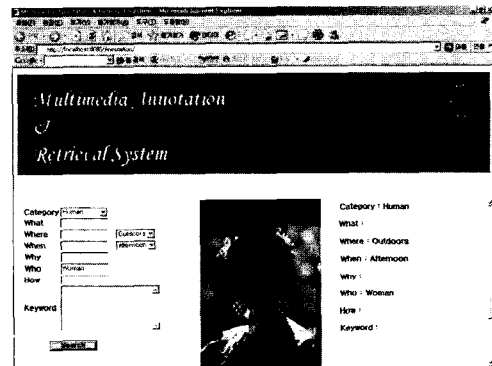


그림 5 검색 및 브라우징 인터페이스

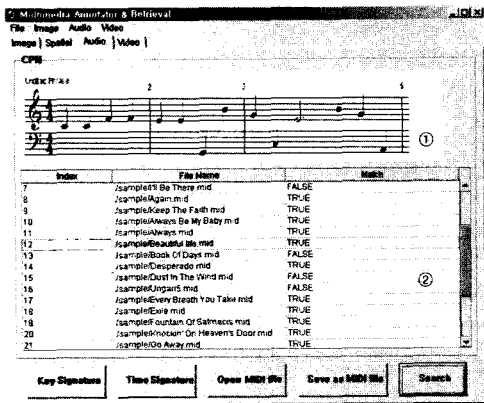


그림 4 CPN을 이용한 질의 및 검색 인터페이스

여 내용 기반과 주석 기반의 질의와 함께 공간질의를 포함하는 복합질의를 통하여 보다 정확한 질의를 할 수 있게 해준다.

그림 3은 선택된 이미지로부터 관심 영역을 선택하고 공간 유사도를 이용한 분석을 할 수 있는 인터페이스이다. 우선 분석하고자 하는 파일을 선택하고 ①과 같이 원본 이미지에서 사용자가 관심있는 오브젝트에 대한 영역을 선택한 뒤에 분석 버튼을 눌러 선택된 관심 영역 간의 공간 관계를 분석한다. ②는 분석된 공간 정보를 공간관계 그래프의 형태로 보여주며, 이를 ③과 같이 XML 문서로 자동 생성되어 XML 데이터베이스에 저장한다.

오디오의 경우 검색시 질의하고자 하는 멜로디를 오선지에 기입하는 CPN(Common Practice Notation)기반의 방식을 이용하여 사용자 정의 질의를 작성하며 그림 4의 ①은 오디오 검색 인터페이스를 보여준다. 사용자에 의해 작성된 질의 멜로디는 MIDI 파일 포맷으로 변환되어 저장되거나, 미리 작성된 샘플 MIDI 파일을 불러서 질의를 할 수도 있다. ②는 오디오 매칭 엔진을 통해서 얻어진 결과를 테이블의 형태로 보여주며 검색된 후보 곡을 선택하면 QuickTime Audio Player를 통해서 재생해 볼 수 있다.

4. 이미지 및 비디오 데이터의 검색

본 장에서는 이미지와 비디오 데이터 검색에 필요한 메타데이터 모델링과 주석을 처리하는 방법에 대해서 설명한다. 우선 이미지 메타데이터 기술구조(Image DS)의 정의를 통한 이미지 메타데이터 구축에 사용된 스키마에 대한 설명을 하고 비디오 데이터로부터 얻어낼 수 있는 영상 정보를 분석하는 방법들과 비디오 메타데이터 구축에 사용된 스키마에 대해 설명한다. 또한 공간 정보 분석에 기반한 질의 이미지와 이미지간의 유사도 검색을 이용한 이미지 오브젝트의 주석표현에 대해 알아본다. 마지막으로 주석처리시 효율적인 인덱싱을 위한 카테고리라이징 기법에 대해 설명한다.

4.1 이미지 메타데이터

이미지(정지영상)에서의 메타 데이터 기술구조는 이미지의 의미적인 표현이 가능한 의미 기술구조(Semantic DS)와 정형적인 데이터를 표현할 수 있는 메타 정보 기술구조(Meta Information DS)로 나눌 수 있다. 의미 기술구조는 이미지가 표현하고 있는 객체들에 대한 상세한 정보와 움직임이나 상태 등에 대한 구체적인 정보를 표현하기 위해서 5W1H(When, Where, Why, What, Who, How)와 이벤트 등에 대해서 기술하고 있다. 반면에 메타 정보 기술구조는 이미지의 정형적인 데이터를 기술하는 기술구조로서 의미 기술구조에 비해 객관적이며 이미지의 저작권, 제작자의 정보와 이미지에 대한 간단한 설명 정보를 기술할 수 있다.

MPEG-7에서는 데이터를 기술하는 언어로 XML을

선택하고 있으며, 이미지 메타데이터를 정의하고 표현하기 위해서 XML 스키마를 사용하였다. 그림 6은 이미지 메타데이터를 위한 XML 스키마 다이어그램이다.

실험에 사용된 이미지에 대해서는 전체 이미지를 앞서 기술되었던 이미지 기술구조와 일반적인 분류 체계에 맞추어서 사람(human), 동물(animal), 식물(plant), 예술(art), 스포츠(sport), 건물(architecture), 풍경(landscape) 등 7가지로 나누었다. 최상위 노드인 *MultimediaContentDescription* 엘리먼트는 *Content*라는 자식(child) 엘리먼트를 가지며, 이때 콘텐츠는 이미지, 오디오, 비디오인지를 구별해주는 *type* 속성을 가진다. 각 이미지는 *complex type*으로 구성되어 있으며 이미지의 일련번호, 이미지 파일 자체에 대한 물리적인 정보들과 저작권 등에 대한 정보를 표현한 메타 정보, 이미지로부터 추출될 수 있는 RGB 값과 색상 키워드를 정의한 색상 정보, 일반적인 텍스트 질의를 위한 내용 정보로 구성되어 있다.

그림 7은 본 논문에서 정의된 이미지 기술자와 이미지 기술구조에 따라 구축된 XML 형태의 이미지 문서를 보여준다.

4.2 비디오 메타데이터

비디오(동영상)에서의 메타데이터 기술구조 역시 이미지와 비슷하게 메타 정보와 의미 정보 그리고 물리적인 정보로 나눌 수 있으며, 이러한 정보들은 비디오 데이터를 장면, 샷, 프레임의 계층적 구조를 가지는 트리 형태로 표현한다고 볼 때 하나의 노드로 표현될 수 있다. 예

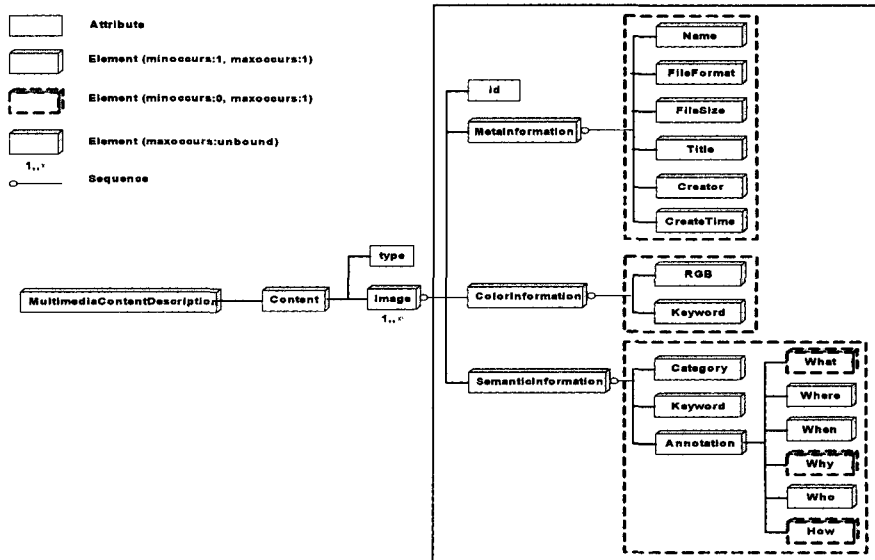


그림 6 이미지 메타데이터를 위한 XML 스키마 다이어그램

를 들어, 물리적인 정보는 비디오 세그먼트에 대한 시작 프레임과 마지막 프레임 정보나 샷을 대표하는 프레임 정보 등의 특성을 가지며, 의미적인 정보는 영상과 오디오 정보로부터 얻을 수 있는 내용 정보를 말한다. 메타 정보는 물리적인 정보와 마찬가지로 의미적인 정보에 비해 객관적인 정보인 저작권, 제작자의 정보 등에 대해 기술한다.

그림 8은 비디오 메타데이터를 위한 XML 스키마 다이어그램이다. 이미지와 유사하게 각 비디오는 complex type으로 구성되어 있고, 비디오 파일과 자막 정보 등에 관련된 메타정보와 샷 경계 검출 알고리즘을 통하여 검출된 각각의 샷들에 대한 프레임 정보들과 이벤트들에

대해 기술하고 이들을 다시 상위 엘리먼트인 장면에 대한 프레임과 이벤트 정보들과 묶어 의미적인 정보를 표현한다. 카테고리(category)는 비디오 파일이 어느 범주에 속해 있는지를 나타내며 크게 뉴스(news), 스포츠(sports), 영화(movie)와 교육(education)으로 나누었다. 각 범주는 하위범주(sub category)를 가지며 이는 동적으로 생성되는 인터페이스를 통해서 다양하게 주석을 입력할 수 있다.

비디오 주석처리는 샷 경계 검출과 자막 데이터를 기반으로 세그멘테이션 과정을 거치게 된다. 이때 사용되는 자막 데이터는 영화, 뉴스, 스포츠 등에서 사용되는 자료로써 대화 내용은 물론 대화의 시작위치와 지속 시

```
<?xml version="1.0"?>
<MultimediaContentDescription>
  <Content type="Image">
    <image id="1">
      <MetaInformation>
        <FileName>p20020610.jpg</FileName>
        <FileFormat>JPG</FileFormat>
        <FileSize>350KB</FileSize>
        <Title>Goal Ceremony</Title>
        <Creator>ADTL</Creator>
        <CreateTime>2002/06/10</CreateTime>
      </MetaInformation>
      <ColorInformation>
        <RGB>212 188 95</RGB>
        <Keyword>Red Green</Keyword>
      </ColorInformation>
      <SemanticInformation>
        <Category>Sports</Category>
        <Keyword>Ceremony</Keyword>
        <Annotation>
          <What>Skate Goal Ceremony</What>
          <Where>Field</Where>
          <When>After the goal</When>
          <Why>Let people know OhNo's meanness</Why>
          <Who>Lee, Chun-8oo</Who>
          <How>Mimic OhNo's violation</How>
        </Annotation>
      </SemanticInformation>
    </image>
  </Content>
</MultimediaContentDescription>
```

그림 7 이미지 문서의 예

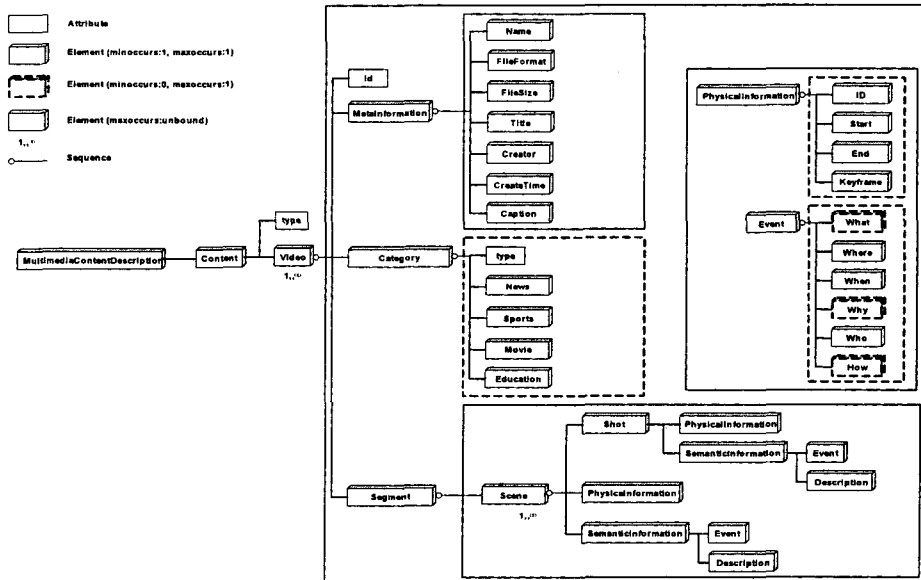


그림 8 비디오 메타데이터를 위한 XML 스키마 다이어그램

간정보를 가지고 있으며 XML 형태로 표현되어 있다. 하지만, 이러한 정보들만 가지고는 비디오 데이터를 인덱싱하기에는 부족하기 때문에 세그멘테이션 과정에 내용기반의 주석처리가 수반되어야 한다. 주석자에 의해서 처리된 각 세그먼트와 관련된 주석정보는 세그먼트와 함께 XML 데이터베이스에 저장된다.

4.3 공간 관계의 분석을 통한 이미지 오브젝트의 주석

모든 이미지들은 각각 독특한 특성을 가진 오브젝트들로 구성되어 있으며, 오브젝트들간에는 다양한 공간 관계를 가지고 있다. 이미지 오브젝트가 가지고 있는 이러한 특성들을 주석처리 하기 위해서는 공간 정보의 분석이 우선되어야 한다. 이미지 오브젝트는 하나의 공간 위치 좌표에 의해 표현이 되기 때문에 전체 이미지는 공간 위치 좌표의 집합으로 나타낼 수 있다. 이미지 내의 여러 오브젝트간의 공간 관계를 표현하기 위해서 오브젝트의 관심 영역을 사용하여 XY 좌표값을 가지는 이미지 오브젝트의 중심 좌표를 루트로 나타내고, 오브젝트간의 공간관계를 표현하기 위해 *Lupper, Llower, Rupper, Rlower, Upper, Below, Right, Left, Center, Overlap*의 10개 위치 연산자를 정의한다.

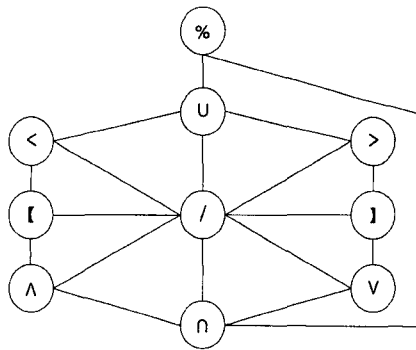
사용자가 질의 이미지를 작성하기 위해서 원본 이미지에서 자신의 관심영역을 선택한 뒤 위치 연산자를 이용하여 선택된 관심영역간의 공간 관계를 분석한다. 이렇게 분석된 데이터는 XML 형태로 데이터베이스에 저

장 되는데 그 이유는 XML을 이용하여 주석처리를 하게 되면 구조화된 질의가 가능하고 기존의 검색방법에 비해 효율적이기 때문이다. 공간 관계 분석이 끝나면 연산자 이웃 그래프(operator neighborhood graph)[1]에 의해서 유추된 유사정도(similarity degree)를 이용하여 연산자간의 유사도를 계산한다. 질의 이미지와 데이터베이스 내의 이미지가 유사할수록 유사도 값은 '1'에 가까워지게 되며, 유사도 임계값(similarity degree threshold)을 지정하여 보다 정확한 결과 이미지들을 얻을 수 있게 된다. 그림 9는 위치 연산자간의 연산자 이웃 그래프와 10개의 위치 연산자간의 유사도를 나타낸다. 연산자 이웃 그래프는 위치 연산자간의 거리(distance)를 나타낸 것으로 최대 거리는 3이고 최단 거리는 0이다. 연산자간의 거리가 짧을수록 유사도는 커지게 되며 거리가 길면 유사도는 0에 가까워지게 된다.

이미지에서 사용자에게 의해서 선택된 관심영역은 하나의 노드로 표현되며 각각의 영역 하나가 ID가 되며 이렇게 선택된 각 노드의 ID와 다른 오브젝트간의 공간 관계를 나타내기 위해서 IDREF라는 속성을 이용해서 다른 노드와 연결하게 된다. 그림 10은 질의 이미지와 이미지내의 관심 영역에 대해 XML로 주석처리 한 것이다.

4.4 카테고리라이징

정보의 양이 증가함에 따라서 검색에 소요되는 시간은



Notation	Operator	<	[^	>]	v	%	/	U	∩
<i>Lupper</i>	<	1	0.67	0.33	0.33	0.33	0.33	0.33	0.67	0.33	0.67
<i>Llower</i>	[0.67	1	0.67	0.33	0.33	0.33	0	0.67	0.33	0.33
<i>Rupper</i>	^	0.33	0.67	1	0.33	0.33	0.33	0.33	0.33	0.33	0.33
<i>Rlower</i>	>	0.33	0.33	0.33	1	0.67	0.33	0	0.67	0.33	0.67
<i>Upper</i>]	0.33	0.33	0.33	0.67	1	0.67	0	0.67	0.33	0.67
<i>Below</i>	v	0.33	0.33	0.33	0.33	0.67	1	0.33	0.67	0.67	0.33
<i>Right</i>	%	0.33	0	0.33	0	0	0.33	1	0.33	0.67	0.33
<i>Left</i>	/	0.67	0.67	0.33	0.67	0.67	0.33	1	0.67	0.67	0.67
<i>Center</i>	U	0.33	0.33	0.33	0.33	0.67	0.67	0.67	0.67	1	0.33
<i>Overlap</i>	∩	0.67	0.33	0.33	0.67	0.67	0.33	0.33	0.67	0.33	1

그림 9 연산자 이웃 그래프와 위치 연산자간의 유사도

```

<?xml version="1.0" encoding="UTF-8"?>
<spatial_info>
  <node ID="0"> // 기준점 R
  <node ID="1"> // node ID="1"은 A를 나타낸다.
  <relation IDREF="0">left_upper</relation>
  <relation IDREF="2">left_upper</relation>
  </node>
  <node ID="2"> // node ID="2"은 B를 나타낸다.
  <relation IDREF="0">right_lower</relation>
  <relation IDREF="1">right_lower</relation>
  </node>
</spatial_info>
    
```

그림 10 질의 이미지와 ROI를 이용한 주석 예제

더욱 증가하게 된다. 이러한 문제를 해결하고 효율적인 정보의 인덱싱과 추출을 위해 주석 정보에 대한 카테고리라이징을 수행한다. 시스템에서 문서의 총 개수를 N , k_i 를 인덱스 단어, n_i 는 k_i 가 나타나는 문서의 개수라 하고, 문서에서의 단어 k_i 의 빈도수를 $freq_{i,j}$ 라고 하면, 문서 d_j 에서 단어 k_i 의 빈도수는 (1)과 같이 주어진다[14].

$$f_{i,j} = \frac{freq_{i,j}}{\max_i freq_{i,j}} \quad (1)$$

(1)에서 최대값은 문서에 나타난 텍스트의 모든 단어를 계산한 것으로 만약 단어 k_i 가 문서에 나타나지 않으면 $f_{i,j} = 0$ 이다. k_i 에 대한 빈도수의 역을 이용하여 단어의 가중치를 (2)와 같이 정의한다.

$$W_{i,j} = \alpha(f_{i,j} \times \log \frac{N}{n_i}) + (1-\alpha) \times K_{i,j} \quad (2)$$

$K_{i,j}$ 는 문서에서 키워드 가중치이고 α 는 단어의 상대적 중요성이다. 실험에서는 카테고리라이징에서 도출된 단어의 가중치와 키워드의 가중치에 대한 중요성을 같은 비율로 두어 α 를 0.5로 정했다.

5. 오디오 데이터의 검색

일반적인 오디오 검색 시스템에서는 오디오 콘텐츠의 음향 특징들을 인덱스로 사용하는데, 본 시스템 역시 음향 특징 중 하나인 음높이의 변화를 이용해서 UDR 스트링을 만들고 검색하는데 이용하였다.

일반적으로 어느 노래에 대해서 사람들이 기억하는 부분은 반복되는 부분이나 클라이맥스 등 특정 부분에 한정되는 경우가 많다. 만약 사용자가 A라는 곡을 찾기 위해서 a라는 질의 멜로디를 작성하면 질의 멜로디 a의 위치를 따로 저장하여 이후 사용자의 질의가 a와 같은

질의 멜로디를 사용하게 되는 경우에 접근 빈도수값을 증가시키고 이 접근 빈도수값이 특정 임계값 이상이 되면 이 멜로디의 내용과 멜로디를 포함하는 오디오를 포인팅하는 링크를 FAI(Frequently Accessed Index)라는 인덱스 안에 저장한다. 이후 사용자의 질의를 가지고 해당 멜로디를 찾기 위해서 오디오 데이터베이스를 검색할 때는 모든 오디오 콘텐츠의 UDR 스트링을 검색하는게 아니라 FAI를 먼저 검색하고, 질의 멜로디가 FAI 엔트리 안에 없을 경우에만 나머지 부분을 검색하게 된다. 따라서 모든 오디오 콘텐츠를 검색해야 되는 불필요한 경우를 줄여준다.

그림 11은 FAI 기반 인덱싱 구조를 보여준다. ①은 FAI의 생성 단계를 보여주며 FAI 생성 초기에는 FAI 내에 엔트리가 비어있는 상태로 존재한다. ②는 사용자의 질의 멜로디가 들어오면, 그 멜로디를 오디오 데이터베이스에서 찾고 부합하는 오디오 콘텐츠가 있을 경우, FAI 엔트리에 오디오의 링크와 함께 질의 멜로디가 들어갈 공간을 하나 할당해 준다. 이 공간에는 질의 멜로디에 대한 정보뿐만 아니라 접속 카운트 값도 포함되어 있다. 이후에 다른 질의 멜로디가 오디오 콘텐츠와 매치하면 새로운 FAI 엔트리 공간을 할당해 준다. 초기의 FAI 엔트리에는 이런식으로 오디오 콘텐츠와 매칭되는 모든 질의 멜로디를 넣어준다.

사용자의 질의 회수가 증가하면 FAI 엔트리의 수도 비례해서 증가하고 축적되는 정보도 증가한다. 여러 사용자의 질의를 받다보면 한 노래에 대한 질의가 여러번 있을 수 있다. 이렇게 되면 그 노래를 가리키는 FAI 엔트리의 개수가 여러 개가 될 수도 있다. 하지만, 이때는 질의 멜로디를 무조건 FAI 엔트리에 넣는 것이 아니라, 기존의 FAI 엔트리와 비교해 보고 질의 멜로디와 중복

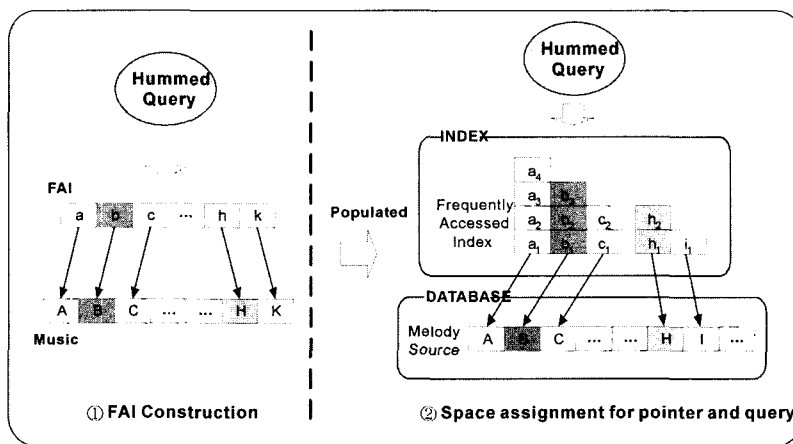


그림 11 FAI 기반 인덱싱 매커니즘

```

Algorithm Query Process

(1) Get a humming query from input device-microphone, CMN;

(2) While {any query comes into the system}
    If {melody q is equal to the melody string in FAI}
        Increase the frequency value of matched FAI entry;
        Return matched_melody;
    Else
        ResultSet ← search the whole music database;
        If {same melody is found in ResultSet}
            AddMelodyIntoFAI(musicID, melody);

(3) List up the candidate result melody ordered by the highest ranked melody first;

(4) Play the retrieved melody;

/* Definition of Functions */

FunctionAddMelodyIntoFAI(current_music_id n, melody m)

    max_FAI_num f = music[n].FAI.max;

    (1) Find the position of new entry;

    (2) Insert a melody m into music[n].FAI[f+1];

    (3) Increase the access_count variable by 1 at music[n].FAI[f+1];
    
```

그림 12 FAI에서 사용되는 질의 처리 알고리즘

되는 것은 추가하지 않는다. 그림 11의 ㉔는 이것을 설명하는 것이다. 오디오 데이터베이스 안에 있는 A라는 곡을 가리키는 멜로디가 a_1, a_2, a_3, a_4 로 네 가지가 존재한다. 이것들은 A곡의 서브 UDR 스트링이고 a_1, a_2, a_3, a_4 는 서로 중복되지 않는다. 이들의 관계는 다음과 같이 표현될 수 있다.

$i \neq j$ 에 대하여 $a_i \cap a_j = \emptyset$ 이고 $a_i, a_j \subset A$ 이다.

즉, 한 노래를 가리키는 FAI 엔트리는 여러 개가 될 수 있고 서로 중복되지 않으며, 그 노래에 대한 서브 UDR 스트링의 집합이라고 할 수 있겠다. 본 시스템에서는 UDR 스트링으로 저장된 오디오 콘텐츠와 질의의 UDR 스트링은 유사매칭 방법이 아닌 정확 매칭 방법을 사용한다. 이때 발생할 수 있는 문제점으로는 많은 질의들이 중복된 UDR 스트링을 가질 수 있고, 사용자들마다 자기 전혀 다른 질의 스트링을 만들어 FAI에 인덱싱되는 질의 패턴이 너무 많아진다는 점이다. 이를 해결하기 위해서, FAI 인덱스 생성시 저장될 인덱스 엔트리의 개수를 제한하고, 새로운 질의 멜로디가 엔트리에 저장된 질의 멜로디와 중복되는 경우 그들을 합하여(merge) 새로운 스트링을 생성한다. 그림 12는 사용자

의 질의 멜로디와 FAI내의 멜로디 스트링과의 비교 및 질의 처리과정을 표현한 알고리즘을 보여준다.

6. 실험

실험을 위해 1,000개의 코렐(corel) 이미지에 대해서 주석 입력 및 오브젝트간의 공간 관계에 대한 처리과정을 거쳐 엑셀론(excelon) 데이터베이스에 저장하였고 비디오 데이터는 영화 10편, 스포츠 5편, 뉴스 10편을 오디오는 5000곡의 MIDI 파일을 이용하여 본 시스템의 성능을 평가하였다. 실험에 사용된 모든 데이터는 전송률을 높이기 위해 RAID(Redundant Array of Inexpensive Disks)에 저장하였다. 이미지와 비디오에 대한 메타데이터를 구축하여 총 633,000개의 노드로 구성하였고, 실험 데이터에 대한 자세한 내용은 표 1과 같다.

이미지 검색의 경우 색상 정보와 키워드를 이용한 질의와 공간 정보를 포함하는 복합 질의를 수행하여 질의 결과를 비교하였다. 표 2는 “빨간색 자동차와 나무 옆에서 있는 사람의 이미지를 찾아라”라는 질의에 따라서 수행된 검색결과를 나타내며 이때 데이터베이스내에서 질의에 정확히 부합하는 이미지는 23개이고, 그들의 질의

표 1 실험 데이터

구분	저장형태	개수	사이즈	메타데이터의 노드수	
이미지	코렐 이미지	JPEG/BLOB	1000	15 MB	17,000
비디오	영화	MPEG-4	10	6.87 GB	34,850
	스포츠	MPEG-4	5	7.51 GB	5,250
	뉴스	ASF	10	660 MB	6,200
오디오	음악	MIDI	5000	175 MB	-

표 2 공간 관계를 이용한 이미지에 대한 질의 결과

No	Spatial Constraint	Color	Keyword	Retrieved	Relevant	Precision
1	-	-	person	53	23	43.4 %
2	-	red	car	366	23	6.2 %
3	-	red	tree	147	23	15.6 %
4	-	red	person, car & tree	32	23	71.9 %
5	left or right	red	person, car & tree	24	23	95.8 %

표 3 비디오에 대한 질의 결과

No	Type	Keyword	Retrieved	Relevant	Relevant shots in DB	Precision
1	movie	love	56	42	45	75.00 %
2	movie	car	35	24	25	68.57 %
3	news	ship	8	7	9	87.50 %
4	news	stock	4	4	4	100.00 %
5	sports	shoot	97	94	94	96.90 %

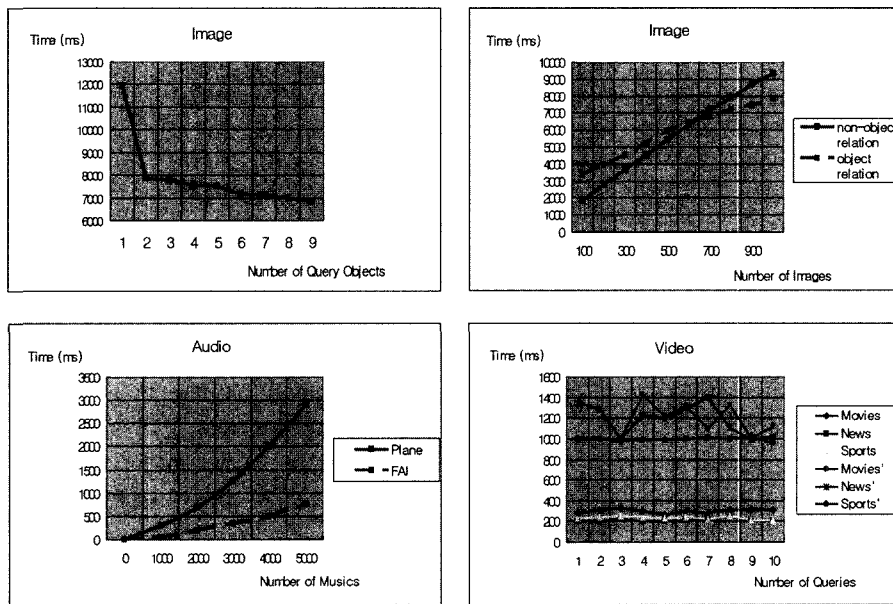


그림 13 이미지, 비디오, 오디오 데이터의 평균 질의 처리 시간

조건에 따라 검색된 이미지의 결과 수와 정확도를 보여 주고 있다. 표에서 알 수 있듯이 1~4번 항목은 공간정보를 이용하지 않은 기존의 방법이고, 5번 항목은 공간정보를 이용한 복합질의에 대한 검색 결과이고 키워드와 색상에 의존한 검색보다 정확도가 높음을 알 수 있다.

비디오 검색의 경우 영화, 스포츠, 뉴스에 대해서 각각 질의를 하였고 표 3은 각각의 질의에 대한 실험결과를 나타낸다. 영화에 대한 질의는 영화내에서 “사랑”이란 단어와 “자동차”라는 단어가 사용된 장면을 찾는 것이고, 뉴스에서는 “어선이 나오는 장면”과 “주식현황에 대해 설명하는 장면”을 검색하였으며, 스포츠에서는 “축구 경기에서 슈트를 하는 장면을 찾아라”라는 질의를 하였다.

이들 질의에 대한 실험 결과를 분석해 보면, 자막을 이용한 영화의 검색보다는 스포츠나 뉴스의 경우가 더 높은 정확도를 얻을 수 있었다. 이는 연기자간의 대화가 주를 이루는 영화에 비해서 스포츠나 뉴스는 사용되는 용어나 이벤트들이 한정되어 있기 때문이다.

오디오 검색의 경우 질의시 UDR 스트림에 대한 유사 매칭 방법이 아닌 정확한 스트링 매칭방법을 사용하기 때문에 정확도에 대해서는 고려하지 않는다. 하지만, FAI 기반의 검색은 오디오 콘텐츠의 수가 증가할수록 기존의 데이터베이스내의 모든 오디오 데이터를 검색하는 방법보다 빠른 질의 처리시간을 가지는 것을 알 수 있다.

그림 13은 이미지, 비디오 및 오디오 데이터에 대한

평균 질의 처리 시간을 보여준다. 이미지 데이터의 경우 공간관계 분석시 질의 오브젝트의 수가 증가하게 되면 응답시간이 감소함을 알 수 있다. 이는 질의 오브젝트의 수가 적으면 적을수록 질의 이미지와 데이터베이스내의 이미지와의 비교 연산이 증가하기 때문이다. 또한 공간 정보를 이용한 질의와 그렇지 않은 경우의 질의를 비교하였다. 이미지의 수가 증가함에 따라 두 경우 모두 평균 응답 시간이 증가하는데 이미지의 수가 증가할수록 공간정보를 이용한 질의가 보다 정확하고 빠른 검색 결과를 나타냄을 알 수 있다. 이때 이미지 내의 오브젝트 수는 일정하다고 가정한다. 비디오 데이터의 경우 영화, 뉴스, 스포츠에 대해서 두 가지 형식의 질의를 하였는데 XML 문서내의 특정 노드값만을 비교하는 방법이 특정 부분이 아닌 모든 노드의 값을 비교하는 방법보다 짧은 시간내에 질의를 처리할 수 있음을 확인할 수 있다. 오디오 데이터의 경우 실선으로 표시된 선이 FAI를 사용하지 않은 일반적인 오디오 검색 방법의 평균 질의처리 시간을 나타내고 점선으로 표시된 선이 FAI를 이용한 오디오 검색 질의 처리시간을 나타낸다. 두 실험 모두 콘텐츠의 수가 증가함에 따라 처리 시간도 같이 증가하지만 FAI를 기반으로 한 오디오 검색이 모든 데이터베이스를 검색하는 방법보다 성능이 뛰어나며 오디오 콘텐츠의 수가 증가할수록 유리함을 알 수 있다.

7. 결론 및 향후연구 방향

본 논문에서는 많은 이미지, 비디오, 오디오 데이터를 가지고 효율적인 검색을 하는 방안으로 멀티미디어 데이터에 대한 메타 데이터를 구축하고 사용자의 질의 패턴을 FAI라는 인덱스에 저장하고 검색하는 방법을 통해 좀 더 쉽고 편리한 인터페이스를 가지는 XML 기반의 멀티미디어 주석 및 검색 시스템을 구현하였다. 실험을 통해서 이미지의 경우 색상이나 키워드들의 특징들을 이용하여 질의를 했을때보다 공간정보를 복합적으로 이용하여 질의를 한 경우가 정확도가 높아짐을 확인하였다. 비디오의 경우 스키마에 정의된 모든 노드들을 검색하는 방법보다 XPath와 XQuery를 이용한 특정 노드들의 검색 방법이 짧은 질의 처리시간을 요구하였다. 그리고 오디오의 경우 선형적으로 모든 데이터베이스를 검색하는 방법보다 FAI를 기반으로 한 검색 방법이 훨씬 짧은 시간 내에 질의를 처리할 수 있음을 확인하였다. 향후 연구 과제로는 더욱 자동화된 멀티미디어 데이터 분석 기법과 효율적인 인덱싱 방법의 개발과 시스템의 성능 향상을 위한 최적의 FAI 엔트리 개수를 찾는 것이다.

참고 문헌

- [1] E. Hwang and S. Lee, "Spatial Similarity and

Annotation-Based Image Retrieval System," IEEE 4th International Symposium on Multimedia Software Engineering, California, December 2002.

- [2] E. Hwang and D. Park, "Popularity-Adaptive Index Scheme For Fast Music Retrieval," Proc. of IEEE Multimedia and Expo, Lausanne, Switzerland, August 2002.
- [3] V. Ogle and M. Stonebraker, "Chabot: Retrieval from a Relational Database of Images," IEEE Computer, Vol. 28, no. 9, pp. 40-48, 1995.
- [4] Alex Pentland, Rosalind Picard, and Stan Sclaroff, "Photobook: Tools for Content-Based Manipulation of Image Databases," SPIE PAPER 2185-05 Storage and Retrieval of Image and Video Databases II, San Jose, CA. February 6-10, 1994.
- [5] J. R. Smith and S.-F. Chang. "VisualSEEK: a fully automated content-based image query system," ACM Multimedia, Boston, May 1996.
- [6] C. Carson, M. Thomas, et al., "Blobworld: A system for region-based image indexing and retrieval," In Proc. Int. Conf. Visual Inf. Sys., 1999.
- [7] A. Ghias, J. Logan, D. Chamberlin, and B. Smith, "Query by humming - musical information retrieval in an audio database," Proc. of ACM Multimedia Conference, San Francisco, 1995.
- [8] R.J. McNab, L.A. Smith, D. Bainbridge, and I.H. Witten, "The New Zealand digital library MELody inDEX," D-Lib Magazine, May 1997.
- [9] Huron, D., C.S. Sapp, and B. Aarden, Theme-finder. 2000. <http://www.themefinder.org>
- [10] M. Flickner et al., "Query by Image and Video Content: The QBIC System," Computer, vol. 28, no. 9, pp. 23-32, 1995.
- [11] Song, Yong-Jun, Hyoung-Joo, et al., "TIVRON: The Integrated Video Retrieval System based on OODBMS," Journal of Korean Institute of Communication Sciences, April. 1999.
- [12] H. Kosch, R. Tusch, et al., "The SMOOTH Video DB - Demonstration of an integrated generic indexing approach," ACM Multimedia Conference, pp. 495-496, Los Angeles, USA, October-November 2000.
- [13] M. Carrer, L. Ligresti, and T.D.C. Little, "A Tcl/Tk-Based Video Annotation Engine," Proc. USENIX, Fifth Annual Tcl/Tk Workshop, Summer 1997.
- [14] B.Y. Ricardo and R.N. Berthier, Modern Information Retrieval, ACM press, 1999.
- [15] L. Wilcox and J. Boreczky, "Annotation and Segmentation for Multimedia Indexing and Retrieval," The 31st Annual Hawaii International Conference on System Sciences, Vol. II, pp. 259-267, Wailea, Hawaii, January 1998.
- [16] B. Emond, M. Brooks, and A. Smith, "A Broad-band Web-based Application for Video Sharing and Annotation," ACM Multimedia 2001, Ottawa,

- Canada, September 30-October 5, 2001.
- [17] S. Blackburn and D. DeRoure, "A Tool for Content Based Navigation of Music," Proc. of ACM Multimedia, 1998.
 - [18] J.T. Foote, "Content-Based Retrieval of Music and Audio," Multimedia Storage and Archiving Systems II - Proc. of SPIE, Vol. 3229, pp. 138-147, 1997.
 - [19] Guojun Lu, "Indexing and Retrieval of Audio: A Survey," Journal of Multimedia Tools and Applications, Vol. 15, pp. 269-290, 2001.
 - [20] N. Kosugi, Y. Nishihara, and T. Sakata, "A Practical Query-By-Humming System for a Large Music Database," Proc. of ACM Multimedia 2000 Conference, November 2000.
 - [22] E. Hwang and Y. Nam, "XVIDEO: XML Based Video Annotation and Retrieval," Proc. of Int'l Conf. on Internet Computing, Las Vegas, June 2002.
 - [23] C. Carson, S. Belongies, H. Greenspan and J. Malik, "Region-based image querying," Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries, June 1997.
 - [24] jMusic Java library, <http://jmusic.ci.qut.edu.au/>
 - [25] L. H. Rodrigues, Building Imaging Applications with Java Technology, Addison Wesley, 2001.
 - [26] K. Tan, B. Ooi and C. Yee, "An Evaluation of Color-Spatial Retrieval Techniques for Large Image Databases," Multimedia Tools and Applications, Vol. 14, pp. 55-78, 2001.



노 승 민

2001년 아주대학교 정보 및 컴퓨터 공학과(학사). 2001년~2003년 8월 아주대학교 정보통신 전문대학원(석사). 2003년 9월~현재 아주대학교 정보통신 전문대학원 박사과정. 관심분야는 데이터베이스, 멀티미디어 시스템, XML 응용, Water-

marking

황 인 준

정보과학회논문지 : 데이터베이스
제 30 권 제 2 호 참조