

# 내용기반 검색을 위한 비디오 데이터베이스 질의처리기 및 브라우저의 설계 및 구현

이 훈 순<sup>†</sup> · 김 용 겔<sup>\*\*</sup> · 배 영 래<sup>\*\*\*</sup> · 진 성 일<sup>\*\*\*\*</sup>

## 요 약

컴퓨팅 기술의 발달 및 보편화로 인해 사용자들의 양질의 정보에 대한 요구가 증가하였고, 이러한 요구를 충족시키기 위해서는 단순 텍스트 형태의 데이터가 아닌 복잡한 멀티미디어 데이터, 특히 비디오 데이터에 대한 저장, 관리 및 검색하는 기능을 가진 시스템의 개발이 필요하다.

본 논문에서는 비디오데이터베이스에서 비디오 데이터에 대한 효과적인 내용기반 검색을 지원하기 위한 메타데이터 모델을 제안하고, 사용자가 비디오 자료를 효율적으로 검색할 수 있도록 질의 입력 기능과 검색 결과 브라우징 기능을 통합한 사용자 통합 인터페이스와 사용자의 다양한 내용기반 검색 요구를 처리할 수 있는 질의처리기 및 브라우저를 설계 및 구현하였다. 본 논문에서 제안하는 질의처리기 및 브라우저는 이미지 특징, 시공간 특징, 그리고 추석을 이용한 질의를 통합하여 지원함으로써 매우 다양한 사용자 질의를 지원할 수 있고, 검색된 결과에 대한 구조 브라우징 기능을 지원하여 사용자가 원하는 정보에 보다 정확하고 효율적으로 접근할 수 있다. 또한 브라우저에 스토리보드 기능을 추가함으로써 비디오 전체를 보지 않고도 내용을 파악할 수 있다.

## Design and Implementation of the Query Processor and Browser for Content-based Retrieval in Video Database

Hun-Soon Lee<sup>†</sup> · Yong-Keol Kim<sup>\*\*</sup> · Young-Lae Bae<sup>\*\*\*</sup> · Seong-Il Jin<sup>\*\*\*\*</sup>

## ABSTRACT

As computing technologies are rapidly progressed and widely used, the needs of high quality information have been increased. To satisfy these needs, it is essential to develop a system which can provide an efficient storing, managing and retrieving mechanism of complex multimedia data, esp. video data.

In this paper, we propose a metadata model which can support content-based retrieval of video data. And we design and implement an integrated user interface for querying and browsing, a query processor and browser for content-based retrieval in video database which can efficiently access and browse the video clip that user want to see. Proposed query processor and browser can support various user queries by integrating image feature, spatial temporal feature and annotation. Our system supports structure browsing of retrieved result, so users can more exactly and efficiently access relevant video clip. Without browsing the whole video clip, users can know the contents of video by seeing the storyboard. This storyboard facility makes users know more quickly the content of video clip.

† 준 회원 : 한국전자통신연구원 연구원  
\*\* 정 회원 : 대전보건대학 사무자동화과 교수  
\*\*\* 정 회원 : 한국전자통신연구원 책임연구원  
\*\*\*\* 종신회원 : 충남대학교 컴퓨터학과 교수  
논문접수 : 1998년 1월 15일, 심사완료 : 1999년 7월 6일

## 1. 서 론

최근 컴퓨팅 기술의 발달 및 보편화로 인해 사용자들의 양질의 정보에의 다양한 요구가 증가하였고, 이러한 증가하는 사용자들의 다양한 요구에 대하여 효율적인 서비스 제공을 하기 위해서는 단지 텍스트 형태의 데이터만을 주로 처리해 왔던 기존의 데이터베이스 시스템에서 다루지 않았던 이미지, 오디오, 비디오 등과 같은 비정형 데이터를 수용할 수 있는 멀티미디어 데이터베이스 시스템을 구축하는 것이 필요하다[1,14]. 이에 따라 멀티미디어 정보 서비스에 대한 요구를 충족시키기 위한 멀티미디어 데이터베이스에 대한 연구가 활발하게 진행되고 있다. 특히 다양하고 방대한 비디오 데이터를 효율적으로 저장, 관리 및 검색하는 기능은 비디오 데이터베이스의 중요한 요소로 대두되고 있다. 개발된 비디오 데이터베이스 시스템의 사례로는 노르웨이 공대의 VideoSTAR[1], 일본 고베 대학의 OVID[2], 미국 메릴랜드 대학의 AVIS[3], 대만 청화 대학의 Venus[4], 미국 카네기멜론 대학의 Informedia[5], 미국 콜롬비아 대학의 WebSEEk[6], VideoQ[7], WebClip[8] 등이 있다.

비디오 정보는 방대할 뿐만 아니라, 내포된 정보 내용들이 다양하여 기존의 데이터베이스 검색 방법으로는 사용자가 원하는 형태와 의미를 정확하게 처리하기 어려우므로, 비디오 데이터베이스에 저장된 객체들의 의미를 기반으로 하는 내용 기반 검색방법이 필요하다. 또한 비디오 데이터의 경우 스키마가 복잡하고 내용기반 검색 능력의 부족으로 기존의 질의 방법만으로는 원하는 비디오 데이터를 찾아내기 어려우므로 이를 보완하는 방법으로 비디오 브라우징 기술이 쓰인다. 비디오 데이터베이스의 내용 기반 검색을 제공하기 위해서는 비디오 데이터의 내용을 구조적으로 체계화한 데이터가 필요하다. 이러한 데이터를 메타데이터라고 하며, 검색 시 이정보와 같은 역할을 수행하게 되므로 메타데이터는 내용 기반 검색의 가장 기본적이고 필수적인 데이터로 모든 검색이 이를 통해 이루어진다[9,12].

본 논문에서는 비디오 데이터에 대한 내용 정보를 효과적으로 저장/검색하기 위해 메타데이터 모델링을 하고, 효과적으로 질의를 할 수 있도록 하는 질의어에 대한 선행 연구[18,19]를 바탕으로 원하는 비디오 데이터를 검색 브라우징할 수 있는 사용자 통합 인터페이스

스와 이에 따른 질의처리기 및 브라우저의 설계 및 구현을 하였다.

본 논문의 구성은 다음과 같다. 2장에서 비디오 데이터베이스 시스템의 사례 연구에 대해 알아보고, 3장에서는 내용기반 비디오 데이터베이스 시스템의 구조와 메타데이터 모델 및 질의어에 대해 설명한다. 4장에서는 내용기반 비디오 데이터베이스 시스템 구축의 핵심이 되는 질의처리기 및 브라우저에 대한 설계를 하고, 5장에서는 4장에서의 설계를 바탕으로 질의처리기 및 브라우저의 구현에 대해 설명한다. 6장에서는 질의처리 및 브라우징 관점에서 2장의 비디오 데이터베이스 시스템과 본 논문에서 제시하는 질의처리기 및 브라우저를 비교 분석하고, 마지막으로 7장에서 결론 및 향후 연구방향에 대해 다룬다.

## 2. 관련 연구

본 장에서는 기존 비디오 데이터베이스 시스템의 특징을 질의처리 및 브라우징 관점에서 살펴본다.

AVIS(Advanced Video Information System)[3]는 1995년에 미국 메릴랜드 대학에서 개발된 비디오 데이터베이스 시스템으로 비디오 내용에 나타나는 객체, 사건, 행위 유형에 대한 메타데이터를 정의하고, 이들을 비디오 세그먼트 트리에 연계 시킴으로써 효율적인 검색 방법을 제안하고 있다. AVIS 는 상위 레벨의 내용을 중심으로 하는 주석 기반의 메타데이터 유형만을 제공한다. 비디오에 대한 데이터 모델은 없으며, 사건과 객체 위주의 데이터 구조를 정의하여 데이터베이스를 구축하였다. 하위 레벨의 내용을 나타내는 특징들에 대한 메타데이터가 없다는 단점이 있다. AVIS의 질의는 SQL과 비슷하게 FIND절, FROM절과 WHERE 절로 구성된다. 주석 기반의 질의 유형만을 제공하며, INGRES라는 관계형 데이터베이스 관리시스템(DBMS : Database Management System)에 목적정보나 메타데이터에 저장할 수 없는 정보를 저장하고, 이와 연계하여 질의처리를 한다.

Informedia[5]는 미국의 디지털 라이브러리 관련 사업의 일환으로 카네기 멜론 대학에서 개발되었다. TV 뉴스를 실험데이터로 사용하였는데 음성으로 질의를 하게 되면 이 내용이 텍스트로 표시된다. 이후에 관련 비디오 클립들이 나타나게 되고 여기에서 원하는 클립을 선택하면 이 비디오 데이터와 관련 텍스트도 아울러

려 재생된다. 이 시스템은 기존의 다른 시스템의 기능에 음성 인식 기능을 추가하여 제공하는 것과 브라우저(User Interface)가 추상화(Video Abstraction) 능력을 지녔다는 점이 특징이다. 추상화 능력이란 사용자가 직접적으로 접촉하면서 질의 결과로 나온 비디오 클립 전체를 보지 않고도(약 10% 크기로 움축됨) 원하는 결과 선택을 빠르게 결정하도록 지원하는 기술이다.

**Venus[4]**는 1996년 대만 청화 대학에서 개발된 비디오 데이터베이스 시스템으로 이미지 처리 기술을 사용하여 비디오의 각 프레임에 나타나는 객체들과 그 객체들 사이의 시간 관계, 공간 관계들을 메타데이터로 구축하여 질의에 이용했다. Venus 시스템에서는 비디오의 특징들을 프레임 단위로 추출하기 때문에 데이터베이스 구성 시에는 많은 양의 메모리가 필요하게 된다. 또한, 메타데이터는 추출된 특징들만을 기반으로 하여 구성되므로 비디오의 내용인 사건, 장소 등의 상위 레벨의 의미들은 메타데이터로 구성되지 못하며, 한정적인 검색만을 지원하게 된다. Venus는 비디오 관리자, 질의 관리자, 색인 톨의 3가지 구성요소를 갖는데 비디오 관리자를 통해 비디오 객체를 관리하고 구조적 브라우저를 지원한다. Venus에서는 CVQL(Content-based Video Query Language)이라는 {range; predicate; target}의 형태를 가지는 비디오 질의언어를 제안했는데 검색 결과가 만족해야 할 조건을 명시하기 위해 비디오 함수와 수식어구를 이용한다. 비주얼한 스케치보드 형태의 질의 인터페이스를 제공하며 비디오 제거, 프레임 제거, 비디오 함수 제거의 3단계를 거치는 제거 기반(Elimination-Based) 질의 처리를 한다.

**VideoQ[7]**는 미국의 콜롬비아 대학에서 개발한 객체지향 내용기반 비디오 검색 시스템으로 기존의 다양한 검색 방법을 확장/통합했고 웹 상에서 동작한다. 데이터베이스에 저장된 특성들은 비디오 스트림의 자동적 분석으로부터 생성된 것들이다. 질의는 비주얼한 속성을 가진 것들로 만들어진다. PhotoBook, VisualSEEK, Virage 같은 기존의 시스템들도 이러한 패러다임을 지원했으나 정지화상에 대해서만 검색이 가능했다. 하지만 VideoQ에서는 객체에 모양, 색, 질감뿐 아니라 움직임과 시간적 지속, 시공간적 순서까지 할당하여 질의를 만들 수 있는 "animated sketch"라는 기법을 이용하고 있으며 명시한 각 특성들에 그들의 질의에서의 중요도를 나타내는 가중치를 부여할 수 있다. 텍스트에 의해 비디오 샷을 브라우징 할 수 있다. 비디오 샷

은 주제별로 분류되어 있어서 사용자가 쉽게 항해할 수 있다.

**VideoSTAR[1]**는 비디오 데이터와 메타데이터의 공유와 재사용을 지원하도록 노르웨이 공대에서 만든 범용 비디오 데이터베이스 시스템으로, basic context, primary context, secondary context 개념을 이용하여 데이터를 공유했다. 선택, 집합, 필터링, 합성, 브라우징 연산을 가지는 대수식을 사용하여 질의와 브라우징을 했으며 스트림 구간 사이의 시간적인 관련성을 기반으로 복잡한 질의들을 정형화 시켰다. 질의처리는 먼저 메타데이터의 내용을 검색하고, 메타데이터와 연관된 비디오 스트림을 대응시키는 단계적인 접근 방법을 취한다. 전체적인 비디오의 계층적인 구조를 보여주는 구조 브라우징(Structure Browsing)과 각 문서의 특정 부분과 관련된 메타데이터를 보여주는 내용 브라우징(Content Browsing)을 지원한다.

### 3. 내용기반 비디오 데이터베이스 시스템

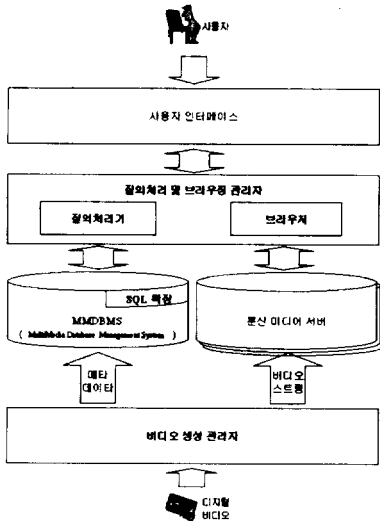
본 장에서는 본 논문에서 제시하는 내용기반 비디오 데이터베이스 시스템의 구조와 메타데이터 모델과 비디오 질의어에 대해 설명한다.

#### 3.1 시스템 구조

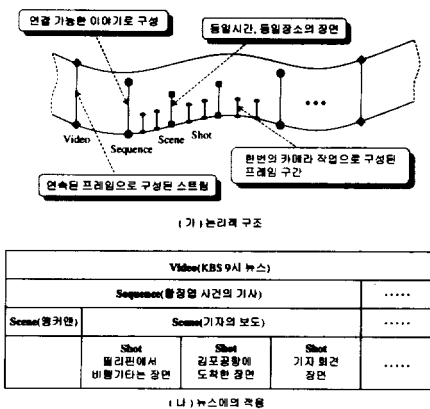
본 논문에서 비디오 데이터베이스 시스템은 대용량의 디지털 비디오에 대해 메타데이터를 이용하여 효율적으로 관리하고, 이를 이용하여 비디오의 내용에 근거하여 효과적으로 검색할 수 있도록 구성되어 있다. 비디오의 효율적인 검색을 위하여 비디오 데이터의 색인기능을 가진 메타데이터는 데이터베이스 서버에, MPEG 형태의 비디오 데이터는 분산 미디어 서버에 저장한다.

비디오 데이터베이스 시스템은 분산 클라이언트/서버 구조를 가지는데 (그림 1)과 같이 구성되어있다. 비디오 생성 관리자는 사용자가 원하는 정보를 쉽게 찾을 수 있도록 하기 위하여 디지털 비디오를 (그림 2)와 같이 논리적인 단위인 Shot, Scene, Sequence로 나누고(비디오 분할), 각각의 단위에 대해 적절한 주석을 입력하는 기능을 수행한다. 또한 대표 프레임을 뽑은 후 대표프레임에서 이미지 특징 정보를 추출하여 메타데이터베이스에 저장하는 역할을 하는 블록이다. 질의 처리 및 브라우징 관리자는 사용자의 원하는 정보에

대한 요구를 받아들여 이에 가장 근접한 정보를 제공하기 위한 블록이다. 이 블록에서는 사용자의 다양한 질의 요구를 받아서 질의어를 만들고, 질의어를 데이터베이스 관리시스템에 전달하여 수행시킨 후 결과를 사용자에게 보여주는 기능을 한다. 또한 질의 처리 결과로 검색된 클립 관련 구조 정보를 사용자에게 보여주며, 구조 정보를 이용하여 사용자가 관련 비디오를 향해 하면서 보다 정확한 클립을 검색할 수 있도록 한다.



(그림 1) 비디오 데이터베이스 시스템 구성도



(그림 2) 비디오의 논리적 구조

본 논문에서는 본 절에서 설명한 비디오 데이터베이스 시스템 중 질의 처리 및 브라우저 관리자를 구성하

는 질의처리기 및 브라우저의 설계 및 구현에 대해서만 4장과 5장에서 다룬다.

### 3.2 메타 데이터 모델

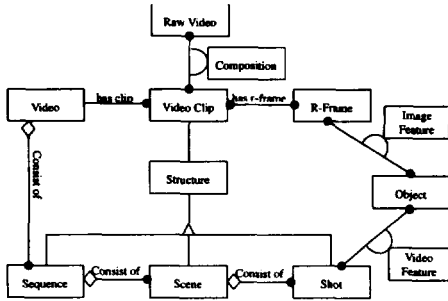
비디오 데이터에 대한 정보의 추출 뿐만 아니라 추출된 정보를 체계적으로 구축하여 관리하는 기능은 내용 기반 검색을 지원하기 위한 중요한 기능이다. 모든 비디오 데이터에 대한 정보는 메타데이터를 통하여 표현되고, 사용자는 어떠한 질의를 수행하더라도 메타데이터를 통하여 실제 데이터에 접근하게 됨으로써 메타데이터는 사용자와 비디오 데이터를 연결하는 매개자 역할을 수행한다. 따라서 효율적인 메타데이터의 구축은 비디오 데이터베이스의 성능을 높이고 가용성을 향상 시킨다[9,12].

메타데이터 모델은 선행연구[18,19,20]의 메타데이터 모델을 비디오 구조 정보 브라우저의 편의를 위해 (그림 3)과 같이 수정하였다. 선행 연구의 메타데이터 모델에서는 *Video*라는 객체가 하나의 논리적인 비디오를 나타내는 것이 아니라 *Composition* 상의 여러 값들이 모여서 하나의 논리적인 비디오를 이룬다. 따라서 구조 브라우저시 테이블들을 여러 번 참조하는 불편함이 있었다. 그러나 본 논문에서 제시하는 수정된 메타데이터 모델은 *Video*라는 객체가 하나의 논리적인 비디오를 의미하므로 구조 브라우저시 *Video* 테이블을 한번만 참조하면 된다. 또한 비디오 합성을 나타내는 *Composition*을 *Video*와 *RawVideo* 사이에 두는 선행연구[18,19,20]에 비해 *VideoClip*과 *RawVideo* 사이에 둬으로써 하나의 긴 비디오가 아닌 짧은 비디오 클립을 공유한다는 면에서 *Raw Data*의 공유 및 비디오 합성이 용이하다.

메타데이터 모델을 표현하기 위하여 객체지향 모델링 기법(Object Modeling Technique)[13]의 표기법을 사용하였다. 각 객체에 대한 설명은 다음과 같다.

한 편의 논리적인 *Video*는 연결 가능한 이야기인 *Sequence*로, *Sequence*는 동일 장소동일 시간의 장면인 *Scene*으로, *Scene*은 한 번의 카메라 작업으로 구성된 프레임 구간인 *Shot*으로 구성된다. 이러한 하나의 구조 정보는 *VideoClip*이라는 주석정보를 가지고, *Composition*이라는 객체를 통해 실제 비디오 데이터인 *RawVideo*와 매핑된다. *VideoClip*은 자신을 대표하는 하나 이상의 *R-Frame*(Representative Frame : 대표프레임)을 가지는데 대표프레임에 나타나는 *Object*

에 대해 *ImageFeature*라는 특성정보를 가진다. 또한, 각 *Shot* 마다 나타나는 *Object*의 움직임 같은 *Video-Feature*를 가진다. 시공간 정보를 나타내는 *Image-Feature*나 *VideoFeature*는 영상처리기술을 이용하여 추출하는데 벡터형식의 값을 가진다.



(그림 3) 메타데이터 모델

### 3.3 비디오 질의 언어

비디오 데이터베이스 시스템에서 내용기반 검색 기능을 제공하기 위해서는 의미적인 데이터 검색을 지원할 수 있는 질의언어가 필요하다[14]. [18,19,20]에서는 내용기반 비디오 데이터베이스 시스템에서 검색을 위해 키워드를 이용한 주석기반 질의와 색상, 질감, 유형 등 이미지 특징과 객체의 시간적, 공간적 관계 및 시공간적 관계를 이용한 특징기반 질의 표현을 할 수 있도록 비디오 질의 언어(Video Query Language : VQL)를 다음과 같이 정의하였다.

VQL = <Result\_Expression(검색항목)> (;)  
 <DB\_Expression(검색대상)> (;)  
 <Condition\_Expression(검색조건)> (;)

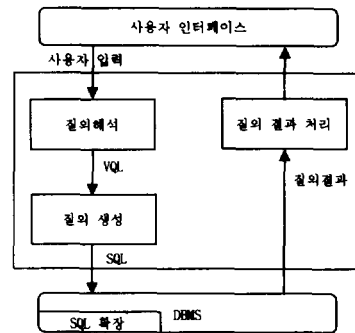
검색항목은 사용자가 검색하기 원하는 항목, 즉 메타데이터의 필드를 정의한다. 여기에 정의된 것들에 한해서 질의 수행 후 메타데이터베이스로부터 검색결과로 가져오게 된다. 검색대상은 질의의 탐색 영역을 정의한다. 개별 비디오 또는 전체 비디오를 대상으로 검색할 수 있다. 검색조건은 질의의 내용, 즉 검색하고자 하는 조건을 항목별로 명시하는 부분으로 비디오의 구조, 속성들을 포함하는 비디오 표현식, 사건 표현식, 객체의 속성을 포함하는 객체 표현식, 이미지 특성 표현식, 객체간의 공간 관계를 포함하는 공간 표현식, 사건의 시간 관계를 포함하는 시간 표현식과 객체의 시·공간 관계를 포함하는 시공간 표현식 등으로

질의의 상세한 내용이 명세 된다.

## 4. 질의처리기 및 브라우저의 설계

### 4.1 질의처리기

질의처리기는 (그림 4)와 같이 질의해석, 질의 생성, 질의결과 처리, SQL 확장의 4개의 모듈로 구성된다. 이 블록에서는 사용자의 다양한 질의 요구를 입력 받아서 해석하여 적절한 질의어를 만든 후 데이터베이스 관리시스템에 전달하여 수행시키고, 이 결과를 사용자에게 보여주는 기능을 수행한다. 또한 사용자가 초기 질의에 대한 검색 결과에 만족하지 못하는 경우 검색 결과에 대해 반복적으로 재질의를 할 수 있는 기능을 제공하고, 메타데이터를 구성하는 객체에 대해 특정 연산을 하는 함수를 만들어 데이터베이스 엔진에 프로시저 형태로 삽입함으로써 질의 처리를 위한 SQL 확장 기능을 제공한다.



(그림 4) 질의처리기의 구성도

각 모듈에 대한 설명은 다음과 같다.

#### (1) 질의 해석 모듈

사용자로부터 질의의 입력을 받아 사용자가 요구한 질의가 어떤 유형인지를 알아내고, 이를 이용하여 중간언어인 VQL을 만드는 모듈로 질의 유형 파악 서브 모듈과 VQL 생성 서브 모듈로 구성된다.

질의 유형 파악 서브 모듈에서는 초기질의인지 재질의인지 구분하고, 초기 질의인 경우에는 사용자가 입력한 검색 조건들이 주석기반 질의를 위한 것인지, 이미지 속성 질의를 위한 것인지, 시공간 질의를 위한 것인지를 파악한다.

VQL 생성 서브 모듈에서는 앞에서 분석한 질의 유형 및 사용자가 입력한 값을 바탕으로 초기 질의의 경

우에는 메타데이터베이스에 대해 질의를 하기위한 VQL을 만들고, 검색 결과가 많은 경우에 검색 조건을 더 자세히 명시함으로써 검색 결과를 줄이는 과정이 재질의이므로, 이미 검색되어 나온 결과인 임시테이블을 이용하여 질의를 정밀하게 할 수 있도록 VQL을 만든다.

**알고리즘 1 VQL 생성 알고리즘**

- 단계 1. 사용자가 검색되어지기를 선택한 검색항목을 VQL 문자열에 추가
- 단계 2. VQL 문자열에 검색 대상 추가
- 단계 3. 검색 조건 항목을 VQL 문자열에 추가
  - 단계 3.1 재질의의 경우
    - 조건 값과 임시테이블 및 필드명을 VQL문자열에 추가
  - 단계 3.2 주석기반 질의의 경우
    - 조건 값과 일치되는 테이블 및 필드명을 VQL 문자열에 추가
  - 단계 3.3 이미지 특징기반이 포함된 경우
    - 선택한 특징과 특징 값을 VQL문자열에 추가
- 단계 3.4 시공간 질의인 경우
  - 단계 3.4.1 절대공간질의이면 절대위치정보와 객체 정보를 VQL문자열에 추가
  - 단계 3.4.2 상대공간질의이면 상대위치정보와 객체 정보를 VQL문자열에 추가
  - 단계 3.4.3 시공간질의이면 시공간 관계정보와 객체 정보를 VQL문자열에 추가
  - 단계 3.4.4 시간질의이면 시간 관계정보와 사건정보를 VQL문자열에 추가

**(2) 질의 생성 모듈**

질의 생성 모듈은 사용자 질의 해석 모듈에서 생성한 VQL을 입력으로 하여 SQL문으로 변환한 후 DBMS에 질의를 요청하는 기능을 수행하는 모듈로 질의어(SQL) 생성 서버 모듈과 질의 요청 서버 모듈로 구성되어 있다.

질의어 생성 서버 모듈에서는 중간 언어인 VQL을 Lex와 Yacc을 이용하여 메타데이터가 저장된 특정 DBMS(본 논문에서는 Informix Universal Server 9.12T)에 맞는 SQL로 바꾸어주는 서버 모듈이다. 이미지 특성에 대한 검색 조건이 들어간 경우에는 이미지 유사

도에 의한 우선순위를 부여할 수 있도록 질의를 만든다. 질의 요청 서버 모듈에서는 질의어 생성 서버 모듈에서 생성한 질의를 데이터베이스에 실행하게 한다.

**알고리즘 2. SQL 생성 알고리즘**

- 단계 1. VQL을 Lex와 Yacc을 이용하여 AST(Ab-stract Syntax Tree) 생성
- 단계 2. AST를 이동(traverse)하면서 특정 DBMS 질의어로 변환
  - 단계 2.1 질의 유형에 해당하는 단순 질의를 생성
    - 단계 2.1.1 텍스트 기반 질의
      - 검색 조건을 AND로 연결
    - 단계 2.1.2 이미지 특징 질의
      - 해당 이미지 특징 비교 함수 호출을 포함하는 질의 문자열 추가
    - 단계 2.1.3 시공간 질의
      - 해당 시공간 질의 함수 호출을 포함하는 질의 문자열을 추가
  - 단계 2.2 복합질의 생성을 위해 질의 문자열을 추가
    - 단계 2.2.1 첫 번째 단순질의이면 사용자에게 대표프레임을 보여주기 위한 질의 문자열을 추가
    - 단계 2.2.2 첫 번째가 아니면 클립의 키가 되는 정보를 나타내는 필드(클립식별자)를 넘겨주기 위한 문자열을 추가
  - 단계 2.3 AST의 검색 조건 필드의 끝에 달할 때까지 단계 2.1과 단계 2.2를 반복 수행
- 단계 3 생성된 질의를 반환

**(3) 질의 결과 처리 모듈**

질의 결과에 따른 대표 프레임들을 제한된 사용자 화면의 검색 결과 표시 부분에 나타내고 사용자가 선택적으로 내용정보를 볼 수 있도록 한다. 사용자가 대표 프레임을 선택하면 관련 비디오의 식별자를 구조 브라우저에게 넘겨주어 트리 모양의 구조 정보를 보여 주게 한다. 또한 재질의의 편의를 위해 검색되어 나온 값들에 대한 각 항목별 리스트를 미리 만들어 주어 사용자가 재질의를 수행하고자 할 때에는 검색된 결과 값 중에서 검색 조건을 선택해서 재질의를 쉽게 할 수 있도록 한다. 재질의를 위하여 검색 결과를 임시 테이블

블로 만들어서 저장하고, 임시테이블에 대한 정보(테이블 이름)를 특정 사용자 프로파일 테이블에 저장한다. 세션 종료시 임시 테이블과 사용자 프로파일 테이블에서 관련 튜플을 삭제한다.

**(4) SQL 확장 모듈**

사용자가 입력한 값을 이용하여 생성한 질의의 보다 쉬운 처리를 위하여 메타데이터를 구성하는 객체에 대해 특정 연산을 하는 함수를 만들어 데이터베이스 엔진에 프로시저 형태로 삽입하는 SQL 확장 기능을 이용한다. 특히 이미지 특성 추출기로 추출된 정보들의 비교는 일반 숫자나 문자의 비교와는 다른 방법이 이용되는데 이를 위한 비교 함수를 만들어서 삽입한다. 또한, 시공간 질의처리를 위한 여러 시공간 관계 함수들을 정의해서 넣음으로써 시공간 질의의 편의를 도모한다.

본 논문의 비디오 데이터베이스 시스템에서는 이미지 특성 질의를 지원하기위한 함수로 Color Histogram의 통계량에 의한 방법을 사용하여 대표프레임과 샘플 이미지의 색에 대한 정보를 비교하여 두 특징 값 사이의 차이를 구해서 반환하는 함수인 GetColorPIMDiff(), Wavelet 변환방법을 사용하여 대표프레임과 샘플 이미지의 색에 대한 정보를 비교하여 두 특징 값 사이의 차이를 구해서 반환하는 함수인 GetWaveletDiff()와 대표프레임이나 샘플 이미지의 모양에 대한 정보를 비교하기 위하여 두 특징 값 사이의 차이를 구해서 반환하는 함수인 GetEdgemapDiff()를 정의해서 사용했다. 시공간 질의를 위해 특정 사건이 정의된 프레임 구간 정보를 이용하여 Allen이 제안한 7가지 시간 관계[15]를 바탕으로 BEFORE, MEETS, OVERLAPS, DURING, STARTS, FINISHES, EQUAL, BEFORE<sup>1</sup>, MEETS<sup>1</sup>, OVERLAPS<sup>1</sup>, DURING<sup>1</sup>, STARTS<sup>1</sup>, FINISHES<sup>1</sup>의 13가지 시간관계를 나타내도록 했다. 공간 관계는 위치관계, 위상관계, 거리관계의 3가지를 고려할 수 있으나 본 논문에서는 위치 정보에 대해서만 다룬다. 공간 관계는 대표 프레임상에 나타나는 객체의 위치 정보를 최소 범위 직사각형(Minimum Bounding Rectangular : MBR) 방법[16]으로 나타낸 값을 이용하여 대표 프레임상에서의 한 객체의 위치 및 두 객체 간의 위치 관계를 LEFT, RIGHT, ABOVE, BELOW, CENTER, LEFT\_ABOVE, LEFT\_BELOW, RIGHT\_ABOVE, RIGHT\_BELOW의 9가지 값으로 나타낸다. 객체의 클

립 내에서의 움직임 정보를 나타내는 시공간 관계는 STILL, UPPER, RIGHT\_UPPER, RIGHT, RIGHT\_LOWER, LOWER, LEFT\_LOWER, LEFT, LEFT\_UPPER의 9가지 값으로 나타낸다. 시공간 질의를 지원하기 위해 두 사건 사이의 시간 관계를 위한 Temporal(), 대표 프레임상의 한 객체에 대한 절대 공간 관계를 위한 AbsoluteSpatial(), 대표 프레임 상의 두 객체의 상대 공간 관계를 위한 RelativeSpatial()을 정의하여 사용했다.

**(5) 질의 표현 예**

사용자가 명시한 질의가 어떻게 표현되어 처리되어 지는 지를 하나의 예를 가지고 설명을 한다. (그림 7)의 (가)는 "정태수가 왼쪽으로 움직이는 장면을 찾으라"는 시공간 질의를 하는 사용자 인터페이스이다. 이를 알고리즘 1을 적용하여 VQL로 나타내면 다음과 같다. *R\_FRAME\_image, R\_FRAME\_video\_id, R\_FRAME\_r\_f\_id, VIDEO\_CLIP\_clip\_id, VIDEO\_CLIP\_my\_type, VIDEO\_CLIP\_interval\_s\_frame, VIDEO\_CLIP\_interval\_f\_frame; NEWS.DB; OBJECT\_name='정태수' MOVE\_TO LEFT ;*

이 VQL을 알고리즘 2를 적용하여 변환하면 아래와 같은 SQL로 표현된다.

```

SELECT LOTOFILE( RF_0_IMAGE, 'FRAME',
'LOCALHOST'), RF_0_VIDEO_ID, RF_0_R_F_ID,
VC_0_CLIP_ID, VC_0_MY_TYPE, VC_0_INTERVAL_
S_FRAME, VC_0_INTERVAL_F_FRAME

FROM video_clip VC_0, vc_has_rf VCHRF_0, r_frame
RF_0, image_feature IF_0, object O_0, video_feature
VF_0, SHOT SH_0

WHERE (O_0_name='정태수') and (VC_0_clip_id=
VCHRF_0_clip_id) and (RF_0_video_id=VCHRF_0_
video_id) and (RF_0_r_f_id=VCHRF_0_r_f_id) and
(RF_0_video_id=IF_0_video_id) and (RF_0_r_f_id=
IF_0_r_f_id) and (O_0_video_id=IF_0_video_id) and
(O_0_object_id=IF_0_object_id) and (O_0_video_id=
VF_0_video_id) and (O_0_object_id=VF_0_object_id)
and (VC_0_my_type=4) and (VC_0_clip_id=SH_0_
clip_id) and (SH_0_shot_id=VF_0_shot_id) and
(VF_0_motion= 8) ;
    
```

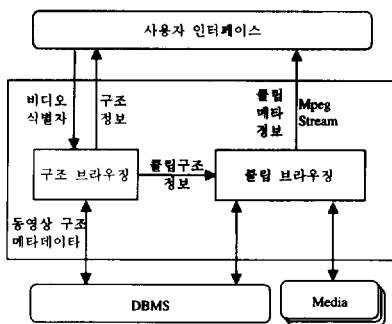
사용자가 입력한 질의는 위의 VQL, SQL 변환 단계를 거친 후 DBMS에 전달되어 조건을 만족하는 결과를 가져오게 된다.

4.2 브라우저

브라우저는 (그림 5)와 같이 하나의 비디오의 구조를 보여주는 구조 브라우징과 특정 비디오 클립의 내용 정보, 즉 특정 클립과 관련해서 메타데이터와 그리고 실제 비디오 클립을 재생시켜 보여주는 클립 브라우징의 2개의 모듈로 구성된다. 이 블록에서는 질의처리기에서의 질의 처리 결과로 검색된 클립 관련 구조 정보를 사용자에게 보여주고, 사용자가 항해하면서 보다 정확한 클립을 검색할 수 있도록 한다. 또한 구조 항해시 현재의 클립과 관련된 메타데이터 정보를 보여주는 역할을 한다.

(1) 구조 브라우징 모듈

하나의 비디오는 (그림 2)와 같이 Sequence, Scene, Shot의 개념적 구조를 갖게 된다. 이러한 구조 정보를 이용하여 한 편의 비디오를 모두 보지 않고 전체적인 내용을 파악할 수 있다. 비디오에 대한 구조를 쉽게 파악할 수 있도록 트리 구조로 표현된 비디오 구조 정보를 제공하는데 클립 브라우징 모듈과 연계하여 하나의 구조, 즉 클립에 대한 해당 대표 프레임과 메타데이터를 자세히 볼 수 있다.



(그림 5) 브라우저의 구성도

(2) 클립 브라우징 모듈

사용자가 검색된 비디오와 관련하여 구조 브라우징을 할 때에 이 특정 구조와 관련된 내용정보, 즉 메타데이터를 보여주는 기능과 관련 동영상을 재생시켜주는 기능을 하는 모듈로 대표 프레임 표시 서버 모듈과

메타데이터 표시 서버 모듈, 재생 서버 모듈로 구성된다.

대표 프레임 표시 서버 모듈은 구조 브라우징 모듈에서 특정 클립이 선택된 경우, 선택된 클립의 식별자를 이용하여 관련 대표프레임을 보여줌으로써 그 장면의 내용을 파악할 수 있도록 해주는 서버 모듈이다. 또한 사용자가 선택한 특정 구조의 하위 구조에 대한 대표 프레임들을 보여줌으로써 간단한 스토리 보드 기능을 할 수 있도록 한다.

메타데이터 표시 서버 모듈은 구조 브라우징 모듈에서 특정 클립이 선택된 경우, 선택된 클립의 식별자, 비디오 식별자, 대표프레임 식별자, 자기의 구조형태를 이용하여 각 장면에 대한 내용을 이해할 수 있도록 텍스트 정보를 사용자에게 보여주는 서버 모듈로, 사용자가 보여지기를 선택한 메타데이터 항목에 대해서만 속성명과 속성값을 보여준다.

재생 서버 모듈은 실제 사용자가 선택한 클립의 실제 동영상상을 보여주는 기능을 하는 서버 모듈로 기본적인 기능들인 재생, 뒤로 가기, 앞으로 가기, 빨리 앞으로 가기, 빨리 뒤로 가기, 멈춤, 잠깐 멈춤, 스틸 재생 등의 기능을 갖는다.

5. 질의처리기 및 브라우저의 구현

5.1 구현 환경

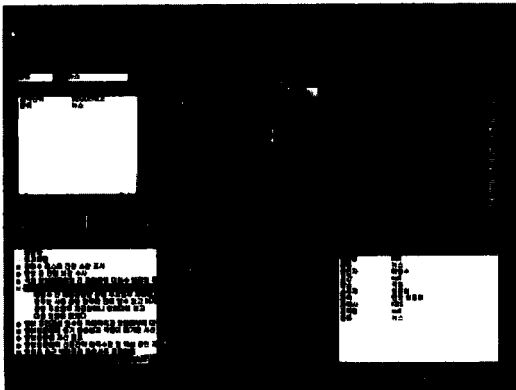
내용기반 검색 지원을 위한 비디오 데이터베이스 시스템의 프로토타입을 구현하기 위하여 메타데이터를 저장하기위한 DBMS로는 WINDOWS NT용 INFORMIX UNIVERSAL SERVER(Ver.9.12)를 이용하였고, 네트워크를 통하여 WINDOWS95/98이나 WINDOWS NT 환경의 클라이언트에서 내용기반 검색이 가능하도록 구현하였다. 일반 사용자에게 보이는 검색 및 브라우징을 위한 인터페이스는 Visual C++을 이용하여 구현하였으며, 데이터베이스와의 연결을 위해서는 INFORMIX사에서 제공하는 C++ Object Interface[17]를 사용하였다. MCI(Media Control Interface)를 이용하여 검색된 동영상상을 재생하기 위한 동영상 플레이어를 구현하였다. 또한, VQL을 SQL로 바꾸기 위한 모듈을 구현하기 위해 Lex와 Yacc을 이용하였고, SQL 확장을 위한 이미지 특성 비교 함수를 데이터베이스 관리시스템의 엔진에 삽입하기위해 INFORMIX사의 데이터블레이드(DataBlade) 개념을 이용하였다. 데이터블레이드를 개발하기위해 INFORMIX사에서 제공하는 데이터블레이



드 개발 키트(DataBlade Development Kit)[17]에 들어 있는 BladeSmith와 BladePack이라는 툴과 한국 전자통신 연구원의 영상처리 팀이 개발중인 이미지처리 라이브러리를 이용하여 Visual C++로 구현하였다. 또한 시공간 질의처리를 위한 함수는 SQL의 프로시저를 이용하여 정의했다.

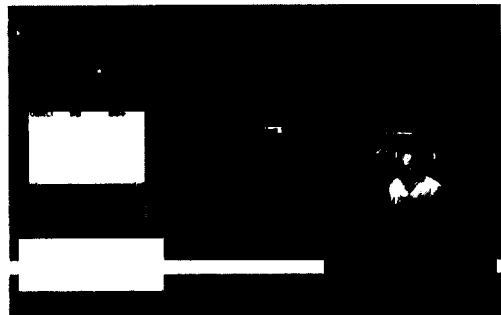
5.2 사용자 인터페이스

질의 입력 및 브라우징을 위한 인터페이스는 (그림 6)과 같다. 커다란 정보 공간을 가진 데이터베이스에서 원하는 자료에 접근하기위한 방법으로 브라우징은 효과적이지 못하다. 또한, 비디오 데이터의 경우 스키마가 복잡하고 내용기반 검색 능력의 부족으로 기존의 질의 방법으로는 원하는 비디오 데이터를 찾아 내기가 어렵다. 본 논문에서는 사용자가 원하는 정보에의 효과적인 검색을 지원하기위한 방법으로 질의 입력과 브라우징의 두 방법을 이용했다. 먼저 질의 방법을 이용하여 브라우징의 대상을 줄인 후에 검색된 정보를 브라우징하도록 함으로써 서로의 단점을 보완토록 했다. 질의 입력과 이에 따른 질의 결과의 브라우징이 매우 밀접한 관계를 가지고 있어 질의 입력과 검색결과에 대한 다양한 브라우징을 지원하기 위한 통합 인터페이스로 구현하였다. 질의 입력 및 브라우징을 위한 인터페이스는 사용자의 선택에 따라 기능들을 활성화/비활성화(enable/disable) 시킴으로써 보다 편리하게 질의할 수 있도록 하였는데 검색 조건을 입력하는 부분, 검색 결과를 보여 주기위한 부분, 구조 브라우징 및 클립 브라우징을 위한 부분으로 구성된다.

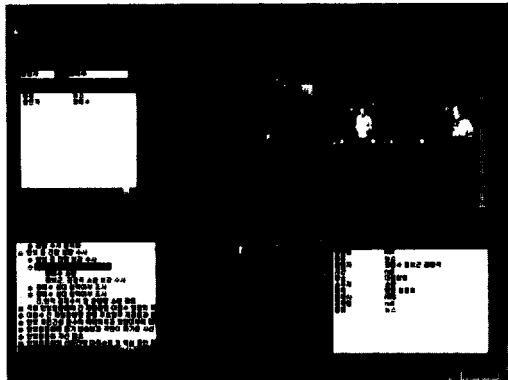


(그림 6) 사용자 인터페이스

(그림 7)은 "정태수가 왼쪽으로 움직이는 장면을 찾아라"라는 질의를 한 후 그 결과를 보는 화면이다. 이와 같은 질의를 위해 시공간 질의 인터페이스에서 Object를 선택하여 그에 대한 이름을 입력한 후 객체의 움직임을 그림으로 그린다. (그림 8)은 "정태수 관련 뉴스를 검색하라"는 질의를 한 후 브라우징하는 화면이다. 사용자는 구조 브라우징 영역을 클릭하여 항해하면서 그 구조와 관련된 클립의 내용을 빠르게 파악하면서 브라우징한다.



(그림 7) 시공간 질의 예



(그림 8) 브라우징 예

6. 비교 분석

본 장에서는 데이터 모델, 질의처리 및 브라우징의 관점에서 2장의 관련 연구에서 소개된 시스템 중 주식 기반인 VideoSTAR, 특징 기반인 Venus, 주석 및 특징 기반 혼용인 VideoQ와 본 논문에서 제시하는 질의 처리기 및 브라우저를 비교하였다.

VideoSTAR[1]는 데이터모델을 가지고 있으며 선택, 집합, 필터링, 합성, 브라우징 연산을 가지는 비디오 대수식(Video Algebra)을 이용하여 일반 주석기반 질의와 프레임 시퀀스의 전후관계를 비교하는 시간 질의는 가능하나 특징기반 질의를 지원하기 위한 정보가 메타데이터에 들어가 있지 않아 이를 지원하지 못한다. 하지만 구조 브라우징 및 내용 브라우징을 지원한다. Venus[4]는 객체와 객체들의 시공간 관계를 위해 객체가 나타나는 프레임 기간(frame duration), 프레임에서의 객체의 위치 등의 정보를 테이블 형태로 저장하였다. 따라서 이러한 시공간 관계에 대한 질의만을 비디오 함수와 수식어구를 이용하는 CVQL을 통해 지원할 뿐 비디오의 내용인 사건, 장소 등의 상위 레벨에 대한 데이터가 없으므로 한정된 검색만 지원한다. 또한 Venus에서 브라우징은 단순히 검색 범위를 줄이는 의미의 브라우징만을 지원한다. VideoQ[7]는 웹 기반 비디오 검색 시스템으로 이미지 및 동영상에 대한 주석 및 특징 기반 질의를 지원한다. 또한 데이터모델을 고려하지 않고 단순히 속성 값의 쌍을 데이터베이스에 저장하고 있으며 주제 분류를 이용한 브라우징을 지원한다.

본 논문에서 구현한 질의처리기 및 브라우저는 주석 및 특징 기반을 혼합하고, 브라우징을 통한 검색을 지원하므로 주석기반[1,2,3]이나 특징 기반[4,7]만을 다룬 다른 선행 시스템들에 비해 기능이 강력하다. 또한 검색한 결과에 대해 구조 및 클립 브라우징을 하면서 좀 더 사용자의 요구에 근접한 정보를 검색할 수 있도록 하고 스토리 보드 기능을 돕으로써 덜 관련된 정보를 검색할 확률이 상대적으로 작을 것으로 기대된다. 시

공간 질의를 위해 객체의 위치나 움직임 정보를 벡터 형태로 표현하여 이용했는데 이러한 정보는 영상 처리의 자동 추출하는 기술에 많이 의존한다. 현재는 테스트를 위해서 사람(Human Annotator)이 대표적인 객체에 대해 위치 및 움직임 정보를 입력하였다. 사례 시스템과 비교표는 <표 1>과 같다.

### 7. 결 론

본 논문에서는 비디오 데이터베이스에서 비디오 데이터에 대한 효과적인 내용기반 검색을 지원하기 위하여 메타데이터 모델링을 하고, 질의어에 대한 선행연구를 바탕으로 원하는 비디오 자료를 검색/브라우징할 수 있는 사용자 통합 인터페이스와 이에 따른 질의처리기 및 브라우저의 설계 및 구현을 하였다.

본 논문에서 설계 및 구현한 내용기반 검색을 위한 비디오 데이터베이스 질의처리기 및 브라우저의 특징은 다음과 같다. 첫째, 이미지 특징, 시공간 특징, 그리고 주석을 이용한 질의를 통합하여 지원함으로써 매우 다양한 사용자 질의를 지원할 수 있다. 둘째, 주석이나 이미지 특징에 의해 검색된 결과에 대한 구조 브라우징 기능을 지원하여 사용자가 원하는 정보에 보다 정확하고 효율적으로 접근할 수 있다. 셋째, 이미지/시공간 특성 질의를 함수 레벨에서 지원함으로써 사용자 질의 처리에 사용할 수 있는 SQL을 확장했다. 넷째, 비주얼한 사용자 인터페이스를 통해 사용자가 보다 쉽게 원하는 질의를 할 수 있다. 다섯째, 브라우저에 스토리보드 기능을 추가함으로써 비디오 전체를 보지 않고도 내용을 파악할 수 있고 보다 정확한 검색을 지원

<표 1> 사례 시스템과 비교

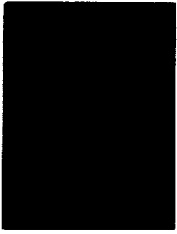
구 분	VideoSTAR	Venus	VideoQ	제 안	
데이터 모델링	있음	없음	없음	있음	
질의처리	질의언어	Algebra	없음	VQL	
	주석기반	가능	불가	가능	
	이미지특징기반	불가	불가	가능	
	시공간	시간	가능	가능	
	우선순위	불가	불가	가능	부분 가능
브라우징	가중치	불가	불가	가능	불가
	구조	가능	불가	불가	가능
	클립	가능	가능	가능	가능
	빠른 내용과악	불가	불가	불가	스토리보드
비주얼 질의 인터페이스 기능	없음	Sketch board	Animated Sketch	Sketch	
비 고			Web 환경		

한다.

향후 연구방향으로는 좀 더 빠른 이미지 특성비교를 위한 이미지특성 정보에의 고차원 색인 기능 적용 기술에 대한 연구, 음성인식 및 자연어 이해 기술을 통합하여 오디오 정보를 포함한 메타데이터를 생성하고 내용기반 질의를 할 수 있는 방법에 대한 연구 그리고 사용자 프로파일 기능을 이용한 사용자의 특성에 맞는 편리한 사용자 인터페이스를 제공하고 질의에 대한 안내자 역할을 하여 불필요한 검색을 줄임으로써 검색의 효율을 높일 수 있는 방법에 대한 연구가 수행되어야 한다.

### 참 고 문 헌

- [1] R. Hjelmsvold, "VideoSTAR-A Database for Video Information Sharing," *Ph.D. Thesis*, Norwegian Institute of Technology, Nov. 1995.
- [2] E. Oomoto and K. Tanaka, "OVID: Design and Implementation of a Video-Object Database System," *IEEE Trans. on Knowledge and Data Engineering*, Vol.5, No.4, pp.629-643, 1993.
- [3] Sibel Adali, et. al., "The Advanced Video Information System : data structures and query processing," *Multimedia Systems*, pp.172-186, 1996.
- [4] Tony C.T.Kuo and Arbee L.P.Chen, "A Content-Based Query Language for Video Databases," In *Proc. of IEEE MULTIMEDIA '96*, pp.209-214, 1996.
- [5] A. Hauptmann and M. Witbrock, "Informedia News-On-Demand: Using Speech Recognition to Create a Digital Video Library," May 1997. <http://www.informedia.cs.cmu.edu/>
- [6] <http://www.ctr.columbia.edu/webseek>
- [7] Shih-Fu Chang, William Chen, Horace Meng, Hari Sundaram, and Di Zhong, "An Automated Content-Based Video Search System Using Visual Cues," *ACM Multimedia*, 1997 <http://www.ctr.columbia.edu/videoq>
- [8] <http://www.ctr.columbia.edu/webclip>
- [9] S. Smoliar and H. Zhang, "Content-Based Video Indexing and Retrieval," *IEEE Multimedia*, pp.62-72, 1994.
- [10] W.I. Grosky, "Multimedia Information Systems," *IEEE Multimedia*, pp.12-24, 1994.
- [11] Arun Hampapur, Designing Video Data Management Systems, *Ph.D. Thesis*, Computer Science and Engineering in the University of Michigan 1995.
- [12] W. Klas and A. Sheth, "Metadata for digital media : Introduction to the special issue," *SIGMOD Record*, Vol.23, No.4 , pp.19-20, Dec. 1994.
- [13] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen, Object-Oriented Modeling, General Electric Research & Development Center Schenectady, New York, Prentice Hall, 1991.
- [14] John Z. Li, M. Tamer Ozsu, and Duane Szafron, "Querying Languages in Multimedia Database Systems," TR95-12, The university of Alberta, Canada, 1995.
- [15] J. F. Allen. "Maintaining knowledge about temporal intervals," *Communications of ACM*, Vol.26, No.11, pp.832-843, 1983.
- [16] John Z. Li, M. Tamer Oisu, and Duane Szafron, "Modeling video spatial relationships in an object model," TR96-06, The university of Alberta, Canada, 1996.
- [17] Informix Universal Server 9.12 Manuals, <http://www.informix.com/>
- [18] 진성일 외, "동영상 메타 정보 구조화 및 검색 방법에 관한 연구"의 최종 연구 보고서, 시스템공학 연구소, 1997. 11.
- [19] 진성일 외, "동영상 메타 정보를 이용한 검색기의 구현"의 최종 연구 보고서, 한국전자통신 연구원, 1998. 11.
- [20] 조은희 외, "내용기반 질의처리를 위한 동영상 질의처리기의 설계 및 구현", 한국정보처리학회 논문지 제6권 3호(99.3), pp.603-614.



**이 훈 순**

e-mail : hunsoon@etri.kr  
1997년 충남대학교 컴퓨터과학과 졸업(학사)  
1999년 충남대학교 컴퓨터과학과 졸업(이학석사)  
1999년~현재 한국전자통신연구원 연구원

관심분야 : 데이터베이스, 멀티미디어, 자료저장시스템



**김 용 굴**

e-mail : ykkim@cs.chungnam.ac.kr  
1985년 충남대학교 계산 통계학과 졸업(학사)  
1987년 충남대학교 계산 통계학과 졸업(석사)  
1992년~현재 대전보건대학 사무 자동화과 조교수

1997년 충남대학교 전산학과 졸업(박사)

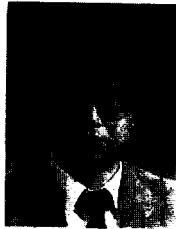
관심분야 : 데이터베이스, 멀티미디어, 성능분석



**배 영 래**

e-mail : ylijb@etri.re.kr  
1976년 서울 대학교 해양물리학과 졸업(학사)  
1976년~1979년 해군 장교로 군복무  
1986년 한양 대학교 전산학과 졸업(석사)

1992년 영국 Kent 대학 전자공학과 졸업(공학 석사)  
1995년 영국 Kent 대학 전기 공학과 졸업(공학 박사)  
1980년~현재 한국전자통신연구원 책임 연구원  
관심분야 : 영상처리, 컴퓨터 비전, 병렬처리



**진 성 일**

e-mail : sjjin@cs.chungnam.ac.kr  
1978년 서울대학교 계산통계학과 졸업(학사)  
1980년 한국과학기술원 전산학과 졸업(석사)  
1994년 한국과학기술원 전산학과 졸업(박사)

1983년~1989년 충남대학교 전산학과 조교수  
1987년~1989년 Northwestern대학 객원 교수  
1989년~1994년 충남대학교 전산학과 부교수  
1994년~현재 충남대학교 컴퓨터과학과 교수  
1996년~현재 충남대학교 소프트웨어연구센터 소장  
관심분야 : 데이터베이스, 멀티미디어, 성능분석, CALS