

비디오 주석 시스템의 설계 및 구현

(Design and Implementation of Video Annotation System)

김기욱[†] 김형주^{**}

(Ki-Wook Kim) (Hyoung-Joo Kim)

요약 다양하고 방대한 비디오 데이터를 효율적으로 저장, 관리 및 검색하는 기능은 멀티미디어 데이터베이스의 중요한 요소로 대두되고 있다. 본 논문은 객체 지향 데이터베이스에 비디오 데이터를 저장하고 이를 주석을 이용하여 효율적으로 검색하는 시스템 VIRON (Video Information Retrieval On Notation)을 설계하고 구현한 내용을 다루고 있다. 사용자간의 관점의 차이에서 비롯된 주석 공유의 난점을 극복하기 위해, 단일한 주석 시스템 구조를 속성 스키마, 개념적 주석 구조, 비디오 인덱스 스트림 구조로 분할하여 주석을 이용한 비디오 데이터 질의 시스템을 모델링하였다. VIRON은 주석을 작성하는 비디오 객체 관리기, 비디오 주석 작성기 및 질의 처리기로 구성되어 있다.

Abstract The mechanism for searching and retrieving video data from large archives becomes one of the main issues on multimedia database. In this paper we propose a video data model based on computer-supported human annotation on video data. With descriptor schema, we aim at providing framework for sharing and reusing annotations among users. In order to process query efficiently, annotated video units are mapped into a unified video annotation stream. Based on the proposed model, we design and implement the annotation-based video retrieval system VIRON (Video Information Retrieval On Notation). VIRON is composed of three tools' video object manager to manage and visualize conceptual video units; annotator to facilitate for user to annotate video data by offering interactive video player; and video query tool to pose and process video queries.

1. 서론

근래의 다양한 영상기술의 발전으로 인해 대규모의 영상 정보의 용이한 저장, 관리 및 전송이 가능해졌다. 이와 같이 사용자가 저장해놓은 영상정보에 관해 어떠한 방식으로 질의를 하고 이를 처리하는 가에 관한 문제가 새롭게 대두되고 있다. 이 문제의 해결책으로 다양한 방법의 검색방법과 모델이 제시되었다. 영상정보의 검색방법은 크게 내용을 기초로한 검색[3,4,19]과 주석을 이용한 검색[6,11,15,16,17,18]으로 나눌 수 있다.

첫째, “내용을 기초로한 영상정보 검색(content-based video retrieval)”은 비트맵에서 윤곽선등의 영상정보를 직접 자동으로 추출하고, 이를 기반으로 하여 영

상 정보에 담겨있는 의미를 추출하는 방법이다. 화면의 정지점 및 음성정보의 휴지점을 검색하여 영상정보를 분석하는 기법들이[20,21] 이용되고 있으나 세부적인 부분까지 내용이 파싱되어 검색에 사용되는 단계에는 아직 못미치고 있다. “주석을 이용한 영상정보 검색(annotationbased video retrieval)”은 세밀한 부분까지 사용자가 원하는 영상정보를 검색할 수 있다는 장점을 지니고 있다. 반면, 주석을 추가하는 작업이 방대할 뿐만 아니라 여러 사용자가 자신의 요구에 부응하도록 주석을 추가하기 때문에, 사용자간에 주석을 공유하기 어려운 단점을 내포하고 있다. 본 논문에서는 주석을 이용한 영상정보 검색을 바탕으로 사용자간에 주석을 공유할 수 있도록 일반적이며 주석간에 일관성을 유지할 수 있는 영상정보 모델을 제시하고 이에 기반한 시스템의 개발에 관한 내용을 기술하고 있다.

지난 십여년간 영상정보 검색에 관한 연구가 활발히 진행되어왔으며, 본 논문과 연관된 주석을 기반으로한 검색에 관한 연구는 다음과 같은 것들이 있다.

· 본 연구는 한국과학재단 목적기초연구과제인 No. 95-1022 “VOD Set-top Box에 관한 연구” 프로젝트의 일환으로 진행되었던 과제이다

[†] 비회원, Univ of Illinois

^{**} 종신회원: 서울대학교 컴퓨터공학과 교수

논문접수: 1996년 2월 8일

심사완료: 1997년 4월 17일

EVA[16,17] 는 Mackay 등이 개발한 주석시스템으로 사용자가 주석을 달고 이를 이용하여 검색할 수 있다. 그러나 이 시스템은 사용자에게 주석을 공유할 수 있는 메카니즘을 제시하고 있지 못하다.

Gibbs 등은[7] 객체지향 개념을 도입하여, 스트림 기반 멀티미디어 데이터를 모델링하였다. 이들의 주요 관심사는 시간관련 매체에 대해 일반적인 모델의 구성에 있었기에, 영상정보의 주석의 저장, 검색 및 관리에 대해서는 언급하고 있지 않다.

Oomoto와 Tanaka가 구현한 OVID 시스템[18]은 비디오 객체 개념을 도입하여, 의미있는 비디오 프레임을 정의하고 이를 구성할 수 있는 기반을 제공한다. 객체지향 개념을 도입하여 영상정보를 체계적으로 구성하고 있으나, 주석의 공유 및 효율적인 관리에 대해서는 언급하고 있지 않다.

Hjelsvold 및 Midtstraum[10]은 영상정보에 인덱스를 구성하는 메카니즘을 제시하였다. 이 모델은 인덱스를 작성하는 방법을 제시하였으나, 영상정보에 관한 저장, 공유 및 질의에 관해서는 주안점을 두고 있지 않다.

이외에 Athena Muse[11]는 다차원 정보에 관한 개념을 소개하고 있으며, HERMES[12]는 영상정보 질의를 다루고 있다. 또한, TGQL[9]에서는 비디오 객체간의 시간관계를 정의하고 있고, Media Streams[5]은 아이콘을 이용하여 주석을 작성하는 시스템을 소개하고 있다.

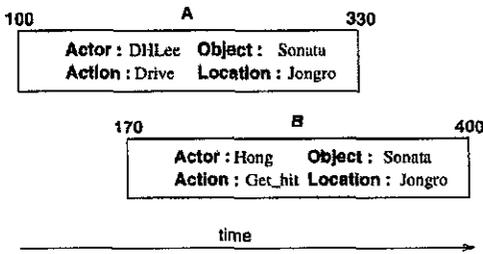


그림 1 비디오 객체의 예

지금까지 구현된 주석을 이용한 영상정보 검색 시스템들[5,11,16,18]은 '비디오 객체' (video object) 또는 '주석 객체' (annotation object)라고 불리는 영상처리 정보를 기본으로 구성되었다. 이 비디오 객체는 '내용 데이터' (descriptive data)와 '시간 데이터' (temporal data)를 모두 포함하고 있다. 내용 데이터는 사용자가

특정한 프레임 시퀀스의 내용에 대해 기술한 정보를 뜻하며, 시간 데이터는 주석을 부가한 프레임 시퀀스의 시작 프레임수와 끝 프레임 수를 나타낸다. 일례로 그림 1에서, 비디오 객체 A는 "Actor:DHLee"와 같은 내용 데이터와 프레임 100에서 330까지 걸쳐 있다는 시간 데이터를 모두 포함하고 있다. 이와 같이 비디오 객체가 이들 정보를 혼합하여 유지하고 있는 이상의 시스템들은 다음과 같은 단점들을 지니고 있다.

첫째, 주석의 일관성을 유지하기 어렵다. 영상정보에 관한 주석의 내용 데이터들이 각각의 비디오 객체에 저장되기때문에 이들을 효율적으로 관리하기 어렵다. 특히, 다른 사용자가 작성한 주석의 변경하는 작업은 이러한 시스템에서는 불가능하다.

둘째, 사용자에게 주석을 공유하기 어렵다. 각각의 사용자가 자신의 관점에 따라 주석을 부가하며, 이 주석이 개별적으로 비디오 객체에 포함되어 있으므로, 일견 주석구조를 파악하거나, 다른 사용자가 작성한 주석을 사용하여 검색할 수 없다.

셋째, 기존의 시스템들[5,12,18] 등에서 제공하는 질의는 단순한 비디오 객체를 검색하는 수준에 머무르고 있다. 반면, Hibino 등은[8,9]등에서 시간 관계에 입각한 질의를 구현하였다. 그러나 비디오 객체간의 의미관계를 정의하고 이를 질의에 이용하는 수준에 이르지 못하는 못하고 있다.

이상에서 기술한 문제점들을 해결하기 위해, 본 논문에서는 다음과 같은 접근 방법을 취하였다. 비디오 객체 내에 포함되어 있는 내용 데이터와 시간 데이터를 분리하여 이들을 각각 속성 스키마와 비디오 인덱스 스트림에 저장 관리한다. 이와 같은 구조를 통해 다음과 같은 특성을 지닌 시스템 VIRON을 구현하였다.

첫째, 내용 데이터의 효율적인 관리를 위해 속성 스키마를 정의하였다. 속성 스키마를 통해 내용 데이터를 일관성있게 유지할 수 있다. 다른 사용자가 작성한 속성의 구조를 파악할 수 있기 때문에 주석을 공유할 수 있게 된다.

둘째, 시간 데이터를 일괄적으로 관리하는 비디오 인덱스 스트림을 정의하였다. 시간 데이터와 속성 스키마에서 비롯한 속성값에 대한 참조값을 단일한 스트림으로 구성하여 질의처리의 효율적인 수행이 가능하게 되었다.

셋째, 속성 스키마와 비디오 인덱스 스트림을 기반으로 하여, 영상 정보에 관한 질의를 구성하였다. 특히 비디오 객체간의 의미관계를 질의에 도입함으로써 질의의 폭을 넓혔다.

본 논문의 구성은 다음과 같다. 2장에서는 비디오 테

1) 본 논문에서는 비디오 객체라는 용어를 사용한다.

서 oid를 취하며, OID 의 유한 부분집합이다. V 는 서술내용 d 의 '속성값'의 집합으로, 그 값으로는 원소값 (스트링, 숫자) 또는 서술내용의 oid를 갖는다.

▶ $Description_of \subseteq T \times T$. $Description_of(t,d)$ 는 d 가 서술자 타입 t 의 서술내용임을 표현하며, $d \leq t$ 로 표시된다.

직관적으로 공유불가능한 서술자 타입의 서술내용은 그 서술내용이 몇번 참조되었는가에 따라 그 수만큼의 oid를 갖는다. 반면, 공유가능한 서술자 타입의 서술내용은 단하나의 oid를 갖는다.

예제 2.1: 그림 2에서 $Object \in T^{msb}$ 이고 $Location \in T^{sk}$ 로 가정하면, $Car \leq Object$ 이므로 $Car \in T^{msb}$ 이다. 또한, $Sonata \leq Car$ 이므로, $Sonata$ 는 공유불가능한 서술자 타입의 서술내용이다. 따라서, $Sonata \leq \{ "Sonata-1", "Sonata-2", ("Sonata") \}$ 로 표시할 수 있는데, 그림 1에 있는 비디오 객체들이 모두 데이터베이스에 정의될 때, 비디오 객체 A 및 B가 모두 $Sonata$ 를 참조하기에 두개의 oid를 갖는다. 반면, $Jongro \leq Location$ 이므로 $Jongro$ 는 공유가능한 서술자 타입이다. 따라서 그림 1의 비디오 객체가 모두 참조했지만, 단 하나의 oid를 가지며 $\{ "Jongro-1", ("Jongro") \}$ 로 표시된다.

2.3 비디오 인덱스 스트림

제 2.2절에서와 같이 속성 스키마에 비디오 객체의 내용 데이터가 사상되었기에, 이제 비디오 객체에는 시간 데이터와 서술내용에 대한 참조값이 저장되어 있다. 이와같은 비디오 객체를 '비디오 단위'라 부르기로 한다. 그러면, 비디오 단위는 (Val, I) 의 순서쌍으로 표현된다. 여기에서 Val 는 '속성값 집합'이며, 속성 스키마에 저장되어 있는 서술내용에 대한 참조값을 나타낸다. 또한 I 는 비디오 단위의 '범위'이며, Adali[1]의 표시방법에 따라 $[i, j) = \{ n \mid 1 \leq n < j, \text{ 단, } i : \text{시작 프레임 수, } j : \text{끝 프레임 수} \}$ 로 표시할 수 있다. 비디오 단위의 범위의 상대적인 값에 따라, 다음과 같이 두개의 비디오 단위간의 시간적 관계를 정의할 수 있다. 우선, 두개의 비디오 단위 u_1 및 u_2 에 관해, $i_1 < j_1 \leq i_2 < j_2$ 의 관계가 있으면, u_1 과 u_2 는 '중첩 관계'에 있다고 표현하며, $u_1 \supseteq u_2$ 로 표시한다. 단, $[i_1, j_1)$ 는 u_1 의 범위들, $[i_2, j_2)$ 는 u_2 의 범위들 각각 나타낸다. 특히, $i_2 = j_1$ 의 관계에 있으면, u_1 과 u_2 는 '인접 관계'에 있다고 표현

한다. 한편, $u_1 \supseteq u_2$ 이고 $u_2 \supseteq u_1$ 이면, u_1 과 u_2 는 '중첩 관계'에 있다고 표현하며, $u_1 \propto u_2$ 로 표시한다.

이제 비디오 인덱스 스트림을 정의하기로 한다.

정의 2.2: STR로 표시되는 '비디오' 인덱스 스트림'은 다음과 같은 조건을 만족하는 비디오 단위의 집합이다. $\forall u_i, u_j \in STR, u_i \supseteq u_j, \text{ iff } i < j$.

직관적으로보아 비디오 인덱스 스트림은 모든 비디오 객체의 정보를 반영하는 비디오 단위들의 집합이다. 또한 이들 비디오 단위들은 준순서 관계에 있다. 즉, 비디오 주석 데이터베이스 하나당 단일하게 존재하는 순차적인 비디오 단위 집합이다.

비디오 인덱스 스트림의 동적인 구성

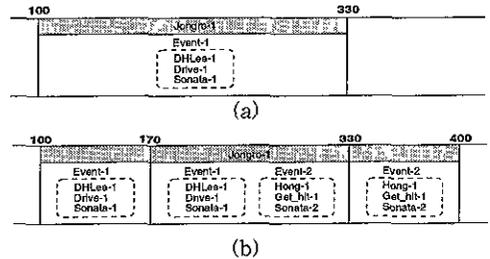
각각의 사용자들이 필요에 따라 주석을 작성하게 된다. 이렇게 임의로 작성된 주석이 기존의 데이터베이스의 일관성을 해치지 않으면서 비디오 인덱스 스트림에 반영되어야 한다. 이와 같은 동적인 비디오 인덱스 스트림의 구성을 위해 다음과 같은 네개의 휴리스틱 (heuristic)을 제시하였다. 우선, u_n 는 비디오 인덱스 스트림에 저장되어 있는 기존의 비디오 단위를, u_n 는 새롭게 저장될 비디오 단위를 나타내는 것으로 가정한다. 또한, I_s 및 I_n 는 각각 u_s 및 u_n 의 범위라고 가정한다.

Heuristic I 만일 $\forall u_s \in STR, u_s \not\propto u_n$ 이면, 단순히 u_n 를 STR 에 삽입한다.

예를 들면, 그림 3 (a)는 그림 1에 있는 비디오 객체 A가 삽입된 후의 상태를 나타낸 것이다.

Heuristic II 만일 $u_n \propto u_s$ 이면, u_s 와 u_n 모두 '중첩 부분'과 '잔여부분'으로 분할된다.

u_s 와 u_n 의 속성값 집합 V_s 및 V_n 은 각각 중첩부분과 잔여부분으로 복제된다.





(c)

그림 3 비디오 인덱스 스트림의 예

그림 1에 있는 비디오 객체 B를 비디오 인덱스 스트림에 저장할때, 이 비디오 객체는 중첩부분 [170,330]과 잔여부분 [330,400]로 분할되며, 각각의 부분에 속성값 집합이 복제되어 그림 3 (b)와 같이 된다. 위의 휴리스틱에 따라, 모든 비디오 객체는 단일하며 순차적인 비디오 단위의 집합으로 표현된다. 이렇게 구성된 비디오 인덱스 스트림으로 인해 암묵적인 질의의 결과를 얻을 수 있다. 일례로 그림 3 (b)에서 [170,330]의 비디오 객체는 “DHLee와 Hong이 동시에 출현하는 비디오 객체를 검색하라”는 질의의 결과로 출력될 수 있다. 즉, 사용자가 명시적으로 주석을 달지 않는 비디오 단위에 대해서도 질의의 결과로 출력되는 특징을 갖는다. 중첩부분에 있어서, V_s 와 V_n 는 다음의 휴리스틱에 따라 하나의 비디오 단위로 합병된다.

Heuristic III 공유가능한 속성값 $V^{sh} = V_s^{sh} \cup V_n^{sh}$ 이 되며, 공유불가능한 속성값 $V^{unsh} = \{V_s^{unsh}, V_n^{unsh}\}$ 이 된다.

그림 3 (b)에서, “Jongro”는 공유가능 속성이므로, 하나의 속성값 집합으로 통합되어 관리된다. 이 휴리스틱의 결과로 서술내용에 관한 참조값인 oid의 저장을 경제적으로 할 수 있는 특징을 갖게 된다. 또한 중첩부분에서 공유불가능한 속성값중, 서로 공통되는 속성값에 대해 다음 휴리스틱에 따라 의미적 연결을 형성할 수 있다.

Heuristic IV V_s 와 V_n 가 공통된 속성값을 포함하고 있으면, 사용자가 이 속성값이 동일한 OID를 갖게 조정하여 비디오 단위간의 의미적 연결을 형성할 수 있다.

예를 들면, 그림 1의 비디오 객체 B가 새로 삽입될 때, 비디오 인덱스 스트림의 “Sonata”와 동일한 속성값을 포함하고 있으므로, 의미관계 설정의 조건을 만족한다. 만일 사용자가 의미관계의 설정을 인준하면, 그림 3 (c)에서와 같이, “Sonata”의 참조값을 “Sonata-2”에서 “Sonata-1”으로 조정하여 이 비디오 단위간의 의미관계

가 성립된다. 이 휴리스틱에 따라, 제 3.2절의 의미관계 질의가 가능하게 된다.

서술내용 증가 및 객체 증가

본 절에서는 질의 조건에 사용될 수 있는 두개의 기본적인 비교 연산자에 대해 기술하기로 한다. 이상에서 언급한 바와 같이 공유가능한 서술내용은 두개 이상의 oid를 가질 수 있다. 따라서 비디오 단위간에 다음과 같은 관계가 성립한다. 첫째, 동일한 서술내용을 갖지만, oid는 다른 속성값과, 둘째, 서술내용 및 oid가 동일한 관계가 있을 수 있다. 이 관계를 질의에 반영하기 위해 OODB[13]의 값 증가 (value equality) 및 객체 증가 (object equality)와 다음과 같은 점들에서 상이한 ‘서술내용 증가 (description equality)’ 및 ‘객체 증가 (object identity)’를 설정하였다. 서술내용 증가는 두개의 속성값이 동일한 서술내용을 갖지만, oid가 다른 경우로서 “=”로 표시된다. 한편, 객체 증가는 “==”로 표시되며, 서술내용 및 oid가 동일한 속성값을 나타낸다. 일례로, 그림 3 (b)에서, “Sonata-1”와 “Sonata-2”는 서술내용 증가이지만, 객체 증가는 만족하지 못하는 관계가 된다.

3. 영상 정보에 관한 질의

속성 스키마와 비디오 인덱스 스트림을 이용하여 질의[14]를 구성할 수 있다. 본 장에서는 질의를 구분하고 각각의 질의에 대해 구조 및 처리과정을 기술하기로 한다. 영상 정보에 관한 질의는 질의의 조건에 따라 단순 질의, 관계질의 및 복합질의로 나뉜다.

3.1 단순 질의

단순질의는 단일한 비디오 단위에 대한 질의[12]이다. 비디오 단위에 부가되는 조건 및 검색 대상에 따라 단순 속성 질의, 단순 내용 질의 및 단순 출현 질의로 구분된다.

단순 속성 질의

단순 속성 질의는 특정한 서술자 및 서술내용을 조건으로 주고 이 서술내용을 포함하고 있는 비디오 단위에서 주어진 서술자를 타입으로 하는 서술내용을 검색하는 질의이다. 이 질의는 SQL식으로 다음과 같이 표현된다.

```
SELECT descriptions
FROM all_video_units
WHERE descriptor = description
```

일례로, “DHLee가 운전하는 차의 모델명을 검색하라”라는 질의가 있을 수 있다. 우선, 속성 스키마에서 조건절에 명시된 서술자의 oid를 찾고, 이 oid를 포함하고 있는 비디오 인덱스 스트림을 선택하여, 대상 서술내용을 검색한다.

단순 내용 질의

단순 내용 질의는 조건절에 명시한 서술내용을 가지고 있는 비디오 단위를 검색하는 질의이다. 이 질의는 다음과 같이 표현된다.

```
SELECT video_unit
FROM all_video_units
WHERE descriptor = description
```

일례로, “Hong이 등장하는 모든 비디오 단위를 검색하라”라는 질의를 들 수 있다. 이러한 질의를 처리하기 위해 우선 조건절에 등장하는 서술내용의 oid를 속성 스키마에서 검색을 한다. 검색한 oid를 포함하고 있는 모든 비디오 단위를 비디오 인덱스 스트림에서 검색한다. 만일 인접한 비디오 단위가 있으면, 이들 비디오 단위를 결합하여 단일한 비디오 단위를 만들어 재생을 위한 구조로 전환시킨다.

단순 출현 질의

단순 출현 질의는 주어진 프레임 범위내에 존재하는 모든 비디오 단위를 검색하는 질의이다. 다음과 같이 표현될 수 있다.

```
SELECT video_unit
FROM all_video_units
WHERE video_unit contained_in the_given_range
```

“프레임 수 380에서 700사이에 존재하는 모든 비디오 단위를 검색하라”라는 질의를 예로 들 수 있다. 이러한 질의의 처리를 위해 단순히 비디오 인덱스 스트림에서 비디오 단위를 검색하여 만일 인접한 비디오 단위가 있으면, 단순 내용 질의에서와 같이 사용자를 위한 재생을 위해 서로 결합하여 결과로 출력한다.

3.2 관계 질의

관계질의는 조건절에 비디오 단위간의 관계에 관한 조건을 포함하는 질의로서, 관계의 종류에 따라 시간 관계 질의 및 의미 관계 질의로 구분한다.

시간 관계 질의

시간 관계 질의는 Allen[2]이 제시한 13가지 상대적인 시간관계 - before, meets, overlaps, finishes, during, starts 및 이들의 역관계와 equal - 를 바탕으로, 비디오 단위간의 상대적인 시간 관계에 관한 질의를 처리하는 것으로 TGQL[9]의 개념을 포함하고 있다. 이 질의는 다음과 같이 표현될 수 있다.

```
SELECT video_unit
FROM (reference_video_unit A) and
      (target_video_unit set B)
WHERE A temporal_operation element_of_B
```

일례로 “비디오 단위 OBJ_12에 대해 overlapped 관계에 있는 모든 비디오 단위를 검색하라”라는 질의가 있을 수 있다.

의미 관계 질의

의미 관계 질의[14]는 제 2.3절에 기술한 휴리스틱에 따라 형성된 비디오 단위간의 의미관계에 관한 질의이다. 즉, 비디오 단위간에 서로 공유하고 있는 속성값의 비교를 통해 비디오 단위간의 연결관계에 관한 질의를 수행하는 것이다. 이 질의는 다음과 같이 표현될 수 있다.

```
SELECT video_unit
FROM (reference_video_unit A) and
      (target_video_unit set B)
WHERE (description in A) semantic_operation
      (description in element_of_B)
```

예를 들면, 한 사용자가 “DHLee가 종로에서 소나타를 운전한다”라고 주석을 달았다고 가정하자. 이후 다른 사용자가 그 비디오 프레임에 대해서 “Hong이 종로에서 소나타에 치였다”라는 주석을 부가했을 경우, 다음과 같은 두 질의는 동일한 결과를 출력하게 된다.

질의 1: “DHLee가 종로에서 운전을 하며, 또한 Hong이 (인의의) 소나타에 치이는 장면이 포함된 비디오 단위를 검색하라.”

질의 2: “종로에서 DHLee가 운전하는 소나타에 Hong이 치이는 장면이 있는 비디오 단위를 검색하라.”

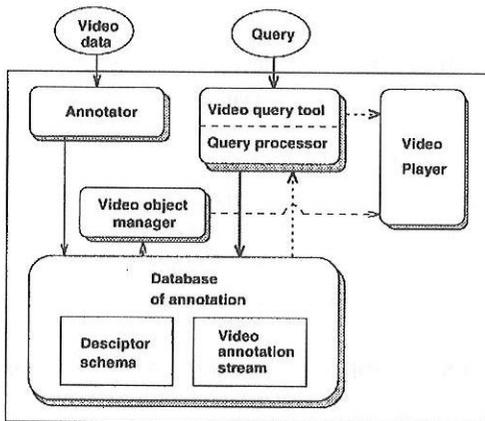
비디오 단위간의 의미관계를 정의하고 위에서 언급한 질의간의 의미를 좀더 명확히 하기 위해, 제 2.3절에서 서술내용 등가 및 내용 등가를 정의하고 이를 질의에 도입하였다. 내용 등가는 서술내용 및 이 서술내용을 참조하는 oid까지도 동일한 것을 나타낸다. 이 내용등가 “=”를 사용하여 위의 질의를 나타내면 다음과 같이 표현할 수 있다.

```

SELECT AnyObjects
FROM X := ( SELECT AnyObjects
            FROM AllObjects
            WHERE actor = 'DHLee' AND action
                  = 'Drive' AND
                  object = 'Sonata' AND location
                  = 'Jongro')
WHERE actor = 'Hong' AND action
      = 'Get_hit' AND
      object == X.object AND location
      == X.location
    
```

3.3 복합 질의

이상에서 언급한 단순질의 및 관계질의를 연결하여 질의를 구성할 수 있다. 단순질의 및 관계질의를 “AND”, “OR” 등으로 연결하여 질의의 결과를 복합처리할 수 있다. 일례로 “DHLee가 운전을 하고 있고, Hong이 거리에서 Park과 대화를 하는 장면이 포함되어 있는 비디오 단위를 검색하라”를 들 수 있다.



Legend:
 → annotation
 - - -> presentation of video units
 <- - - querying
 <- - - presentation of query result

그림 4 전체적인 시스템의 구조

4. 비디오 주석 시스템의 설계와 구현

4.1 전체적인 구조

그림 4는 전체적인 시스템의 구조를 나타낸다. 시스템은 크게 주석 작성 서브시스템과 질의 처리 서브시스템으로 구분할 수 있다. 주석 작성 서브시스템은 주석 작성기(annotator), 비디오 객체 관리기(video object manager)로 구성되고, 질의 처리 서브시스템은 질의 작성 및 질의처리기(video query tool)로 구성된다. 주석 작성기에 포함되어 있는 비디오 정보 재생기를 통해 비디오 프레임을 재생하여 보면서 사용자가 주석을 작성한다. 이렇게 작성된 주석은 데이터베이스 시스템에 속성 스키마와 비디오 인덱스 스트림의 형태로 변환되어 저장된다. 사용자가 주석이 추가된 비디오 객체를 관리할 수 있는 비디오 객체 관리기에 작성된 주석이 표현된다. 이상과 같이 작성된 주석 데이터베이스에 그래픽을 이용한 자유로운 질의를 할 수 있다. 질의를 입력받아 이를 처리하는 부분이 질의처리기이다. 비디오 주석을 저장하는 데이터베이스는 SOP⁴⁾상에서 구성되었다.

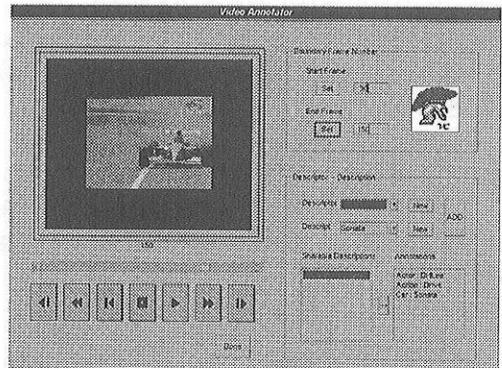


그림 5 주석 작성기

4.2 비디오 주석 작성기(Annotator)

비디오 주석 작성기는 그림 5와 같이 구성되었다. 주석 작성은 크게 세가지로 나뉜다. 첫째 이미 작성된 속성 및 그 값을 사용하여 주석을 작성하는 방법이다. 사용자가 모든 속성을 검색할 수 있으며, 검색한 속성을 새로운 주석에 포함시킬 수 있다. 일단 특정한 속성이 선택되면, 그 속성 값들을 사용자에게 보여준다. 사용자는 속성 및 속성값들을 새로운 객체에 부여하면 된다.

4) SOP는 Seoul National University OODBMS Platform의 약어이며, 서울대학교 OOPSLA 연구실에서 개발한 객체지향형 데이터베이스 시스템이다.

둘째로 기존의 속성을 이용하지만 새로운 속성값을 이 속성에 부여할 수 있다. 세번째로 완전히 새로운 속성을 생성하고 이 속성에 값을 부여할 수 있다. 이와 같은 방법을 따라 속성 및 그 값을 설정한다. 이미 작성한 속성을 중복하여 생성한다든가, 기존의 속성과 새로 생성된 주석이 그 이름만 다를뿐 내용이 동일하게 될 수 있다. 이러한 오류를 최소화하기 위해 새로 지정한 프레임과 시간상 중복되어 있는 모든 객체들의 속성을 리스트로 보여준다.

4.3 비디오 객체 관리기 (Video object manager)

비디오 객체 관리기는 그림 6과 같이 사용자 인터페이스를 구성하였다. 비디오 객체 관리기는 전체적인 시스템의 통괄기능, 주석 작성기 및 질의처리기로의 임의 전환 및 개념적인 주석 구조 현황의 표현 등의 기능을 가지고 있다. 사용자가 작성한 주석은 내부적으로 비디오 인덱스 스트림으로 매핑되고, 속성 스키마에 반영된다. 그러나 사용자에게는 사용자가 작성한 비디오 객체만이 의미있는 단위가 된다. 따라서, 사용자가 작성한 비디오 객체의 시작 프레임 번호, 종료 프레임 번호에 따

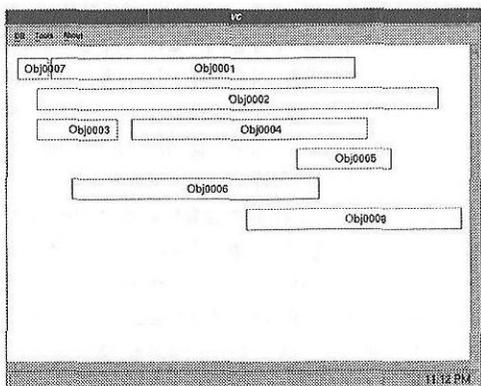


그림 6 비디오 객체 관리기

라 길이가 조정된 직사각형으로 각각의 비디오 객체를 표현하여 출력하였다. 따라서 사용자는 이 비디오 객체의 정보, 즉, 객체의 시작 및 종료 프레임번호, 객체가 가지고 있는 서술자와 서술내용에 관한 정보를 비디오 객체 관리기를 통해 볼 수 있다. 또한 비디오 객체 관리기를 이용하여 이미 작성한 비디오 객체간의 의미관계를 정의할 수 있다.

4.4 질의 처리기 (Video query tool)

질의 처리기의 사용자 인터페이스는 그림 7과 같다. 이 사용자 인터페이스의 윈도우를 마우스 등을 사용하여, 일정한 점을 누르면, 그림 8과 같은 단순 질의기가

화면상에 출력된다. 이와 같은 단순 질의기에 각각의 항목에 적합한 것을 선택함으로써 질의를 작성하게 된다. 질의의 조건절에 기술되는 서술자 및 서술내용은 시스템이 자동적으로 사용자에게 제시함으로써 질의의 작성이 매우 용이하다.

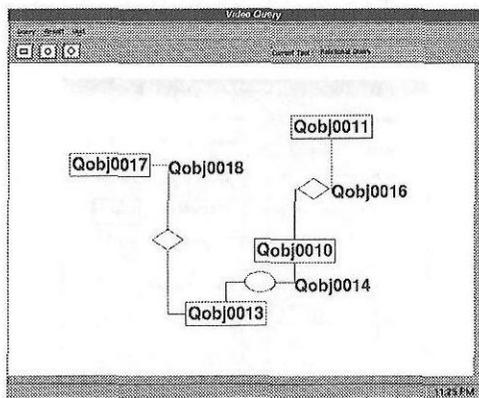


그림 7 질의 처리기

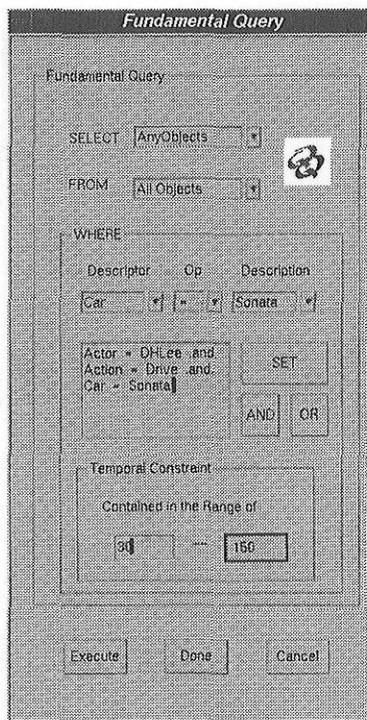


그림 8 단순 질의

그림 8에서 단순질의의 예로 “비디오 프레임수 30에서 150사이에 존재하는 비디오 단위중 DHLee가 Sonata를 운전하는 장면이 포함되어 있는 비디오 단위를 검색하라”는 질의이다. 이와 같은 단순질의가 수행되면, 그 결과로 그림 7에 직사각형 형태로 질의의 결과가 임시로 저장된다. 이들 단순질의의 결과간에 관계질을 수행할 수 있다.

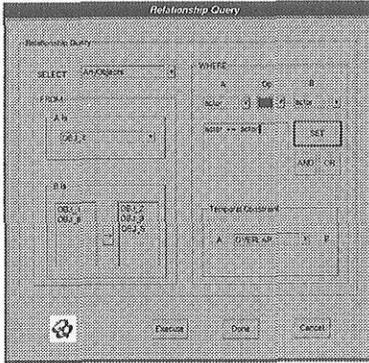


그림 9 관계 질의

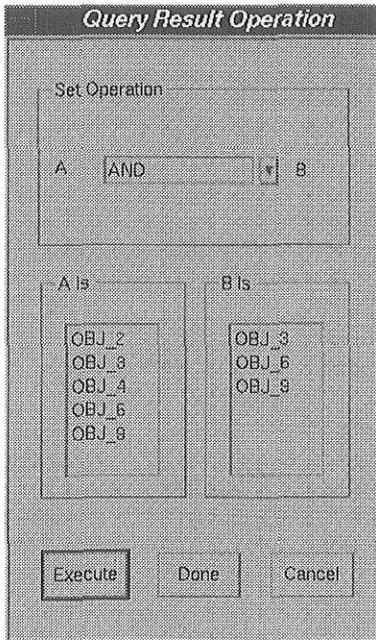


그림 10 복합 질의

그림 9에 나타난 바와 같이 질의를 작성할 수 있다.

“OBJ_2, OBJ_3, and OBJ_5중에서 기준 비디오 단위 OBJ_2와 시간적으로 중첩되어 있으며, 의미적으로는 동일한 actor를 포함하는 것을 검색하라”는 시간관계 및 의미관계에 관한 질의를 표시한 것이다. 이 예에서 기준 비디오 단위는 그림 7에서 단순질의의 결과인 Qobj0010에서 선택된 것이며, 대상 비디오 단위는 Qobj0011에서 선택된 것들이다. 이러한 관계질의의 결과로서 Qobj0016와 같은 마름모꼴의 질의 결과가 표시된다. 이와 같은 단순질의 및 관계질의의 결과에 대해 집합 연산을 수행케하여 복합질을 수행케하는 것이 그림 10에 표시되어 있다.

5. 결론

본 논문에서는 주석을 달아 비디오 데이터를 검색하는 방법을 설계 및 구현한 시스템에 관해 기술하였다. 비디오 데이터에 부가한 주석을 다른 사용자가 재사용할 수 있는 방법을 강구하기 위해 다음과 같은 특성을 지닌 시스템을 설계하였다. 개념적 주석인 비디오 객체의 내용 데이터와 시간 데이터를 분리하여 처리하였다. 따라서 사용자간의 주석에 관한 공유성을 극대화하고, 질의의 처리를 효율적으로 수행할 수 있는 기반을 제시하였다. 우선 내용 데이터를 서술자와 서술내용으로 구분하여 체계적으로 스키마를 구성한다. 이러한 구조를 통해 다른 사용자가 작성한 주석구조를 유동적으로 재편성할 수 있다. 또한 시간 데이터 및 서술내용에 관한 참조값을 비디오 인덱스 스트림에 저장함으로써 질의를 원활히 처리할 수 있게 하였다. 또한 비디오 인덱스 스트림의 동적인 구성을 이용하여 비디오 단위간의 의미 관계를 설정할 수 있으며, 사용자가 명시적으로 주석을 부가하지 않은 부분에 대해서도 질의의 결과로 출력해주는 특징을 갖는다. 이상의 모델을 바탕으로 하는 VIRON 시스템은 다음과 같은 구조를 갖는다. 우선 주석을 작성하는데 사용되는 주석 작성기와 개념적인 프레임들의 지도를 보여주며, 각각의 프레임이 갖고 있는 속성 및 값들의 집합을 출력하는 기능을 하는 비디오 객체 관리기가 있다. 또한 단순질의, 관계질의 및 복합질의를 자유롭게 수행할 수 있는 질의 처리기를 지원한다. 또한 본 논문에서는 비디오 질의를 분류하였으며, 특히 이전까지 없었던 의미관계 질의를 정의하고 이를 시스템에 구현함으로써 질의의 폭을 넓혔다.

참고 문헌

[1] S. Adali, K. Candan, S. Chen, K. Erol, and V.

- Subrahmanian. *Advanced Video Information System: Data Structures and Query Processing*. Technical report, Univ. of Maryland, 1995.
- [2] J. Allen. Maintaining knowledge about temporal intervals. *Communication of the ACM*, 26:832--843, Nov. 1983.
- [3] I. A. R. Center. Query by Image and Video Content: The QBIC System. *IEEE Computer*, 28(9):23--32, Sept. 1995.
- [4] T. Chiueh. *Content-Based Image Indexing*. Proc. of the 20st VLDB Conf.: Santiago Chile, pages 582--593, 1994.
- [5] M. Davis. *Media Streams: An Iconic Visual Language for Video Representation*. Readings in Human-Computer Interaction: Toward the Year 2000, pages 854--866, 1995.
- [6] M. Davis. *Media Streams: Representing Video for Retrieval and Repurposing*. Ph.D. Thesis, Massachusetts Institute of Technology, 1995.
- [7] S. Gibbs, C. Breiteneder, and D. Tsichritzis. Data Modelling of Time-Based Media. Proc. 1994 ACM SIGMOD Conf. on Management of Data, pages 91--102, 1994.
- [8] S. Hibino and E. Rundensteiner. A Graphical Query Language for Identifying Temporal Trends in Video Data. University of Michigan, EECS Technical Report CSE-TR-225-94, Dec. 1994.
- [9] S. Hibino and E. Rundensteiner. Interactive Visualizations for Exploration and Spatio-Temporal Analysis of Video Data. IJCAI'95 Workshop on Intelligent Multimedia Information Retrieval, 1995.
- [10] R. Hjelmsvold and R. Midtstraum. Modelling and Querying Video Data. Proc. 1994 Int'l. Conf. on Very Large Databases, pages 686--694, 1994.
- [11] M. Hodges, R. Sasnett, and M. Ackermann. A construction set for multimedia applications. *IEEE Software*, 6:37--43, Jan. 1989.
- [12] E. Hwang and V. Subrahmanian. *Querying Video Libraries*. Technical report, Univ. of Maryland, 1993.
- [13] W. Kim. *Introduction to Object-Oriented Databases*. The MIT Press, 1990.
- [14] K.W.Kim, K.B.Kim, and H.J.Kim. VIRON: An Annotation-based Video Information Retrieval System. submitted to COMPSAC, 1996.
- [15] A. M. Lab. *The visual almanac*. Apple Computer, 1989.
- [16] W. Mackay. EVA: An experimental video annotator for symbolic analysis of video data. *SIGCHI Bulletin*, 21:68--71, Oct. 1989.
- [17] W. Mackay and G. Davenport. Virtual video editing in interactive multimedia applications. *Communication of the ACM*, 32:802--810, July 1989.
- [18] E. Oomoto and K. Tanaka. *OVID: Design and Implementation of a Video-Object Database System*. *IEEE Trans. on Knowledge and Data Engineering*, 5(4):629--643, 1993.
- [19] S. Smoliar and H. Zhang. Content-Based Video Indexing and Retrieval. *IEEE Multimedia*, 1(2):62--72, 1994.
- [20] Y. Tonomura and et. al. VideoSpaceIcon: tools for anatomizing content. Proc. of INTERCHI'93 conf. on human factors in computing systems, pages 131--136, 1993.
- [21] H. Zhang, A. Kankanali, and S. Smoliar. Automatic partitioning of full motion video. *Multimedia Systems*, pages 10--28, Jan. 1993.

김기욱

1994년 2월 서울대학교 컴퓨터 공학과 졸업. 1996년 2월 서울대학교 컴퓨터 공학과 석사. 1996년 8월 ~ 현재 Dept. of CS, Univ. of Illinois at Urbana-Champaign 박사과정. 관심분야는 객체지향 시스템, 멀티미디어 시스템, 데이터베이스.

김형주

제 24 권 제 1 호(B) 참조.