

시간기반 비디오 검색 시스템

이지현*, 강오형*, 나도원*, 이양원*

*군산대학교 컴퓨터학과

Temporal_based Video Retrieval System

Ji-Hyun Lee*, Oh-Hyung Kang*, Do-Won Na*, Yang-Won Rhee*

*Dept of Computer Science, Kunsan National University

E-mail : jhlee@kunsan.ac.kr

요 약

기존의 비디오 데이터베이스 시스템들은 대부분 간단한 간격을 기반으로한 관계와 연산을 지원하는 모델을 이용하였다. 비디오 모델에서 시간을 지원하고 객체와 시간의 다양한 연산을 제공하며 효율적인 검색과 브라우징을 지원하는 비디오 데이터 모델이 필요하게 되었다. 비디오 모델은 객체 지향 개념을 기반으로 한 모델로서 비디오의 논리적인 스키마, 객체의 속성과 연산 관계, 그리고 상속과 주석을 이용한 메타데이터 설계를 통하여 비디오 데이터에 대한 전체적인 모델 구조를 제시하였다. 그리고 점 시간과 시간 간격을 정의하여 시간의 개념을 객체 지향 기반 모델에 부여함으로써 시간 변화에 따른 비디오 정보를 보다 효율적으로 활용할 수 있도록 하였다.

ABSTRACT

Traditional database systems have been used models supported for the operations and relationships based on simple interval. video data models are required in order to provide supporting temporal paradigm, various object operations and temporal operations, efficient retrieval and browsing in video model. As video model is based on object-oriented paradigm, I present entire model structure for video data through the design of metadata which is used of logical schema of video, attribute and operation of object, and inheritance and annotation. by using temporal paradigm through the definition of time point and time interval in object-oriented based model, we can use video information more efficiently by time variation.

키워드

비디오모델링, 시간지원, 객체지향, 메타데이터, 교양비디오

1. 서 론

시간지원 개체-관계 모델[1]은 이력 사항을 분석하는데 이용하는 모델로서, 시간에 따라 변하는 데이터베이스를 모델링하기 위하여 개체 사이의 관계와 속성 등에 관련된 이력사항을 이용한다. 일반적으로 시간지원 관계형 모델[2]과 시간지원 객체지향 모델[3]이 시간지원 데이터베이스의 논리적 스키마를 설계하기 위한 모델로 주로 사용된다. 시간지원 관계형 모델은 개체나 개체들간의 관계를 독립적인 릴레이션으로 표현하는 모델이고 시간지원 객체지향 모델은 관계의 객체 상태와 행위를 클래스로 표현하는 모델인데, 관계의 상태와 행위는 시간지원 객체지향 모델이 보다 효율적으로 표현한다[4].

비디오 데이터베이스에서 시간은 동적으로

변하는 데이터에 시간의 개념을 추가함으로써 질의를 통한 검색의 편의성을 제공하는데 있다. OVID[5], VideoStar[6] 등 대부분의 모델들이 시간 관계로써 간격을 사용하였으나 간격의 정의가 미흡하고, 간격 연산 또한 제공하지 못하는 부분들이 많이 존재하기 때문에 많은 문제점을 갖고 있다.

본 논문에서는 비디오 데이터의 시간 속성과 연산을 정의하고 교양비디오에 필요한 시간 간격과 관계 및 시점에 대한 시간 속성을 정의하여 비디오 데이터를 효율적으로 검색하고 공유할 수 있는 방법을 제공한다. 또한 객체 지향 모델을 기반으로한 스키마(schema)를 정의하고 클러스터와 롤을 사용하여 동적인 비디오 관리 기반을 마련한다. 스키마는 의미 있는 비디오 단위에 대하여 객체와 클래스, 시간 속성, 그리고 비디오에 필요한 다양한 속성들을 정의하고

록 한다.

II. 본 론

비디오 데이터베이스 시스템은 비디오의 효율적인 저장, 분석, 그리고 정보 검색과 같은 폭넓은 응용분야에 활용될 수 있어야 한다. 따라서 비디오 데이터의 내용에 대한 정보의 공유와 활용을 편리하고 원활하게 수행하기 위한 효율적인 비디오 모델을 필요로 한다.

2.1 시간개체

물리적 시간 개체의 선택은 점 시간과 시간 간격으로 표현한다. 점시간은 대부분의 시스템들이 사용하는 방법으로 가장 기본적인 시간을 표현하기 위해서 시간을 하나의 점으로 표현하며 시간간격은 연속적 시간을 주로 처리해야 하는 스케줄링 시스템에서 많이 사용하는 방법으로 가장 기본적인 시간 개체는 시간 간격이다.

2.2 비디오 모델 구조

물리적 구조인 Video_Source 클래스의 한 객체는 여러 가지 시간 관계를 이용하여 서로 다른 형식으로 표현될 수 있다. Temporal 클래스의 객체에서 생성되는 시간 관계성은 Explorer 클래스의 스키마 정보를 이용하여 정의한다. 이에 따라 데이터의 중복성을 피할 수 있기 때문에 데이터의 일관성을 갖게 해주며 물리적 저장 공간을 보다 효율적으로 사용 가능하게 해준다. 그림 1은 개념적 객체와 클래스 구조 모델로써, 시스템의 전체적 개념 모델을 보여주고 있다. 최상위 클래스로 Video_Object가 있으며, 서브 클래스로써 Video_Source, Explorer, 그리고 Temporal이 있다. 여기에서 물리적 구조와 논리적 구조의 상호 연관 관계를 나타내기 위하여 화살표를 사용하여 동적인 클러스터링을 표시하였다.

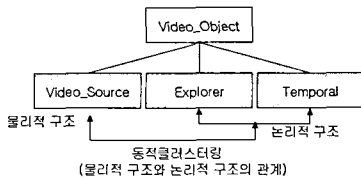


그림 1. 객체와 클래스 구조

본 논문에서는 클래스를 다음과 같이 정의하여 사용하도록 한다.

```

CLASS <클래스 이름>
{
    SUPER <수퍼 클래스>
    {
        { ts <시작 시간>
          te <종료 시간>
        }
        (<속성 리스트> )
    }
}
    
```

METHOD <인스턴스 메소드>

2.3 교양 비디오의 스키마

교양 비디오의 논리적 구조를 정의하기 위한 교양 비디오 영역의 클래스 구조는 그림 2와 같이 나타난다. 교양 비디오는 시간, 즉 이벤트에 해당하는 내용과 비-이벤트에 해당하는 내용으로 구성된다. 그리고, 교양비디오는 방송을 실시한 일자와 지역에 따라 단위 프로그램으로 구성된다.

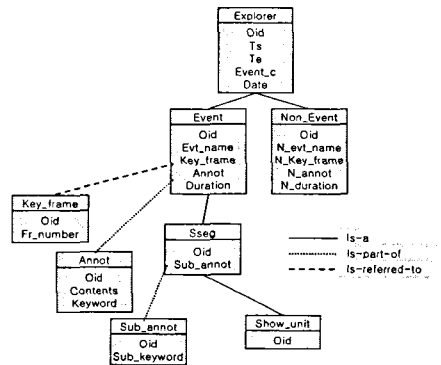


그림 2. 교양 비디오 영역의 클래스 구조

그림 2에서 Explorer 클래스의 Ts와 Te는 교양 비디오의 실제 내용에 대한 시작 시간과 종료 시간을 나타내고, Event_c는 발생한 이벤트의 수를 나타낸다. Date는 방송을 실시한 년, 월, 일을 표시하는 것으로써 타임스탬프 정보로 활용하도록 한다.

Event 클래스에서 Evt_name은 이벤트의 이름을 나타내고, Key_frame은 이벤트를 대표적으로 표현하기 위하여 하나의 장면에 대한 대표 프레임을 나타내는 프레임 번호를 설정하여 검색의 참조 관계로써 작용하게 된다. Annot는 구성 관계의 클래스와 연관되어 이벤트 제목, 키워드, 자막 및 행위 정보를 설명한다. Duration은 이벤트의 총 시간을 나타낸다.

Sseg 클래스는 교양 비디오에서 하나의 이벤트 장면을 구성하기 위한 다양한 샷들로 구성되는 일정한 의미와 구성 관계를 갖는다. 여기에서 Sub_annot는 이벤트를 구성하는 세부적인 행위들에 대한 묘사를 나타내는 것으로써 세부 이벤트 제목, 자막 및 행위 정보, 그리고 키워드 등이 묘사된다.

2.4 물리적 비디오 모델링

스트림 간격은 물리적 구조 단위와 논리적 구조인 메타데이터와 연결되기 위한 개념으로써 데이터 모델에서 중심적인 역할을 담당하는데, 스트림 간격이란 비디오 스트림에서 의미 단위를 나타내는 연속적인 프레임들의 간격이다.

비디오 데이터베이스에서 비디오의 물리적 저장 구조를 나타내기 위하여 다음과 같이 정의하도록 한다.

【정의 1】 스트림 간격(SI)은 비디오 스트림(VS)의 연속적인 부분들로 구성된다. 이를 식으로 표현하면,

$$SI = (oid, VS_ref, Ts, Te) \text{ 일때, } Ts \leq Te \wedge \exists VS \in SI_set (VS_ref = VS \wedge VS.Ts \leq Ts \wedge VS.Te \geq Te)$$

스트림 간격은 스트림 간격 집합의 요소이며, 각각은 시작 시간과 끝 시간의 시간 단위로 표시된다. VS_ref는 스트림 간격이 나타내는 비디오 스트림의 부분에 대한 참조를 나타낸다.

【정의 2】 비디오 스트림(VS)을 트랙(Track)이라는 매체의 기록 집합을 나타내는 속성과 연관시켜 다음과 같이 정의한다.

$$VS = (oid, Ts, Te, Track) \text{ 일때, } Ts \leq Te, \exists VS \in VS_set (VS.Track \in VS.Tracks)$$

여기에서 VS.Track은 비디오 스트림이 매체에 저장된 형태로 표시된 것이며, VS.Tracks는 VS.Track의 집합을 나타낸다. 여기에서 각 단위 구조를 갖는 비디오 스트림들은 시작 시간과 끝 시간의 범위를 갖는다.

【정의 3】 비디오 단위를 구성하는 sequence, scene, shot, 그리고 frame은 다음과 같이 정의된다.

$$VU = (oid, Type, a_1, a_2, \dots, a_m, Si_ref) \text{ 일때, } Si_ref \in Si_set \wedge VS \in VS_set (Si_ref.VS_ref = VS) \wedge Type \in \{sequence, scene, shot, frame\} \wedge a_1, \dots, a_m$$

비디오를 구성하는 각 단위는 스트림 간격의 집합에 포함되며, 네 개의 단위중 하나를 취하게 된다. 그리고 단위별 속성들을 가질 수 있다.

2.5 동적 객체 관계 표현

객체지향 데이터베이스 모델링 방법은 클래스들이 정적으로 미리 정의되어야 하며, 클래스들의 객체들은 성질상 동종 이어야한다는 점이다. 이러한 모델은 성질상 비정규적이고, 동적인 객체들 간의 관계를 포함하는 응용 표현에 어려움이 있다. 따라서 기존 객체지향 모델을 기반으로 이를 확장한 클러스터의 개념을 도입하는 것은 이와 같은 응용들을 효율적으로 수행하기 위하여 필요하다.

메소드는 클러스터와 볼에 의해서 실시될 수 있는데, 하나의 메소드는 여러 개의 다른 클러스터와 볼에 의해서 공유될 수 있고, 객체는 볼의 메소드에 의해서 운영된다. 볼은 비디오베이스와 메소드베이스의 연결구조에 의해 보여 지는데 이런 관계를 나타낸 것이 그림 3이다.

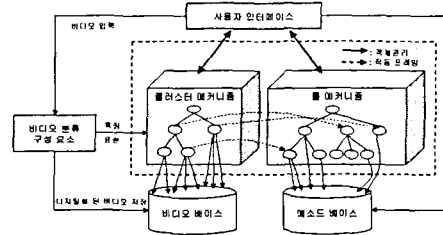


그림 3. 클러스터와 볼의 관리 구조

2.6 시간 모델

Temporal 클래스에서는 집합의 개념을 기본으로 하여 비디오 데이터들 간의 시간 관계를 표현하였다. 이를 나타내기 위하여 다음과 같이 Temporal 클래스를 정의하도록 한다.

```
CLASS Temporal
{ attr_list : set of
  (name : string,
   duration : integer,
   set_info : set_type)
  time_info : set of
  (first : set of string,
   last : set of string) }
```

2.7 비디오 데이터의 연산

스트림 간격은 비디오 스트림의 연속적인 부분을 나타내는 것으로써 결국 스트림 간격의 크기는 시작시간과 끝 시간으로 표시된다. 따라서 스트림 간격의 집합은 결국 시간 값들의 집합으로 표현된다. 따라서 스트림 간격에 대한 연산을 다음과 같이 정의하여 적용하도록 한다.

【정의 4】 스트림 간격 교집합은 두 개의 스트림 간격이 교차하는 부분을 나타내는 스트림 간격을 되돌려 주는 것으로 다음과 같이 정의한다.

$$c.vs_ref = a.vs_ref \wedge (c.Ts = \max(a.Ts, b.Ts)) \wedge (c.Te = \min(a.Te, b.Te))$$

스트림 간격 교집합의 예제로는 여러 회차의 식사 장면이나 만드는 장면들을 모두 보고자 하는 경우에 해당된다.

【정의 5】 스트림 간격 합집합은 두 개의 스트림 간격에서 하나의 스트림 간격과 나머지 스트림 간격을 병합하는 것으로써 다음과 같이 정

의한다.

$$c.v.s_ref = a.v.s_ref \wedge (c.Ts = \min(a.Ts, b.Ts)) \wedge (c.Te = \max(a.Te, b.Te))$$

스트림 간격 합집합의 예제로는 어떤 회차의 개요를 보고자 하는 경우에 해당된다.

2.8 질의와 검색

(1) 질의 1

"VJ특공대의 2004년 12월 10일부터 2004년 12월 24일까지 방영된 식사 장면을 검색하시오."이며, <SQL 문>은

```
SELECT 식사 장면 키 프레임
FROM explorer_video
WHERE (date Between "2004/12/10" AND
"2004/12/24") AND
sub_keyword="식사"
```



그림 4. 질의 1의 검색 결과 브라우저

질의1에서는 날짜를 기준으로 "2004/12/10" - "2004/12/24"까지의 시간 간격과 이벤트를 이용하여 질의를 수행하였다.

(2) 질의 2

"2004년 12월 24일 방영된 VJ특공대의 15분 이후의 내용을 검색하시오."이며, <SQL 문>은

```
SELECT 15분 이후 시점
FROM explorer_video
WHERE date="2004/12/24" AND
time="10:5" AND string="last"
```

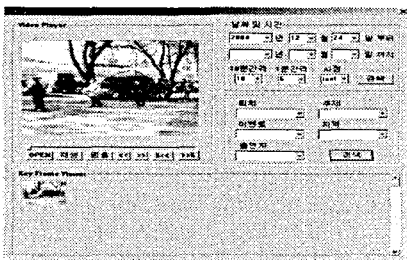


그림 5. 질의 2의 검색 결과 브라우저

질의2에서는 날짜 정보 "2004/12/24"과 시간 정보 15분("10:5") 그리고 시점 정보 이후

("last")를 이용하여 질의를 수행하였다.

III. 결론

기존의 비디오 데이터베이스 시스템들은 대부분 간단한 간격을 기반으로한 관계와 연산을 지원하는 모델을 이용하였다. 본 논문에서는 시간 지원 교양비디오 데이터베이스 시스템을 구축하기 위한 비디오 모델을 제안하였다. 비디오 모델은 객체 지향 개념을 기반으로 한 모델로서 비디오의 논리적인 스키마, 객체의 속성과 연산 관계, 그리고 상속과 주석을 이용한 메타데이터 설계를 통하여 비디오 데이터에 대한 전체적인 모델 구조를 제시하였다. 또한 점 시간과 시간 간격을 정의하여 시간의 개념을 객체 지향 기반 모델에 부여함으로써 시간 변화에 따른 비디오 정보를 보다 효율적으로 활용할 수 있도록 하였다. 본 논문에서 제시한 시간지원 비디오 모델링은 전체적인 프로토타입의 구현과 비디오 데이터의 정형화를 통한 스키마 정의와 비디오 데이터에 시간 개념을 적용하여 비디오 데이터베이스에서 객체, 관계, 연산 등에 대한 유연성을 제공하였고, 다양한 비디오 분야의 모델링 기반을 마련하는데 기여하였다.

참고문헌

- [1] Ait-Braham, B. Theodoulidis and G. Karvelis, "Conceptual Modeling and Manipulation of Temporal databases," Rproc. of the Entity-Relationship Conf., 1994.
- [2] Tansel, Clifford, Gadia, Jajodia, Segev and Snodgrass, "Temporal Databases," The Benjamin/Cummings Publishing Company, Inc, 1993.
- [3] Wu, G. T. J. and U. Dayal, "A Uniform Model for Temporal Object-Oriented Databases," Proc. of the Int'l Conf. Data Engineering, pp. 584-593, 1992.
- [4] M. Koprulu, N. K. Cicekli and A. Yazici, "Spatio-temporal querying in video databases," information Sciences, Volume 160, Issues 1-4, pp. 131-152, 2004.
- [5] E. Oomoto and K. Tanaka, "Video database Systems-Recent Trends in research and Development Activities," In Handbook of Multimedia Information Management, Chap. 8, pp. 405-448, Prentice Hall, 1997.
- [6] R. Hjelsvold, R. Midtstraum and O. Sandst, "Searching and Browsing a Shared Video Database," Multimedia Database Systems, Design and Implementation Strategies, Chap. 4, Kluwer Academic Pub. 1996.