

장면 검색을 지원하는 XML 기반 멀티미디어 검색 시스템

정미라*, 황부현
전남대학교 전산학과

e-mail:{mrjoun, hwang}@sunny.chonnam.ac.kr

XML Based Multimedia Retrieval System supporting Scene Search

Mi-Ra Joung*, Bu-Hyun Hwang

*Dept of Computer Science, Chonnam National University

요약

오디오 비디오 데이터의 활용이 증가함에 따라 멀티미디어 데이터의 내용에 대해 표현하려는 연구와 함께 멀티미디어 데이터의 내용이나 메타데이터를 저장하고, 검색하고, 조작하는 연구의 필요성이 증가하였다. 멀티미디어 데이터의 표현은 사용자가 원하는 내용만을 쉽게 검색하고, 접근할 수 있도록 표현되고 저장되어야 한다. 그러나 기존의 멀티미디어 검색 시스템들은 특정 객체에 중점을 두고 색상, 위치, 모양 등의 정보를 가지고 유사 객체를 찾는 방식을 취하고 있으므로 특정 사건이나 구체적인 인물 정보나 에피소드의 정보를 검색하고자 할 때는 키워드에 의한 검색을 해야하므로 불필요한 정보가 다량으로 검색되며 여러 번의 검색이 이루어져야 하는 단점이 있다. 또한 일반 사용자들은 주로 특정 장면에서 특정 객체의 특징이나 행동, 장소, 사건 등의 정보에 대해 관심을 갖고, 이에 따른 질의를 하는 경향이 있다. 따라서 본 논문에서는 "장면"이라는 계층 구조에 중점을 두고 멀티미디어 데이터의 내용 정보와 구조 정보를 표현 및 저장을 하며, 사용자는 특정 사건이나 객체들의 특징 정보를 가지고 장면이나 전체 구조를 검색할 수 있는 시스템을 설계하고 구현한다. 멀티미디어 데이터의 표현 및 저장 검색의 모든 과정은 데이터의 재사용성과 접근 용이성을 위해 XML을 기반으로 하여 처리된다. 이렇게 XML로 표현된 데이터는 사용자들에게 구조 정보나 내용 정보에 있어서 다양한 검색 결과를 제공할 수 있는 장점이 있다.

1. 서론

최근 멀티미디어 콘텐츠(content)의 급속한 확산으로 인해 멀티미디어 콘텐츠를 효과적으로 표현하고 이를 통한 효과적인 검색 방법에 대한 연구가 증가하고 있다. 최근 MPEG-7 그룹에서도 new MPEG-7 표준인 멀티미디어 콘텐츠 표현에 대한 인터페이스의 표준화를 공식화하고 있다. 이 멀티미디어 표현 기술은 콘텐츠를 쉽게 찾고, 검색하고, 접근하고, 여과하고, 관리할 수 있게 해주며, 멀티미디어 데이터베이스와 비디오 브라우저에서 저장과 검색에 대해 효율적인 응용을 가능하게 하므로 그 중요성이 강조되고 있는 실정이다[1][2]. 따라서 멀티미디어 표현과 이를 이용한 검색 시스템에 관한 많은 연구들이 이루어지고 있으며[3][4][5], XML을 사용하여

멀티미디어 데이터의 내용을 표현하고 검색하려는 연구도 활발히 이루어지고 있다[1][7]. 그러나 기존의 멀티미디어 검색 시스템들은 특정 객체에 중점을 두고 색상이나 위치 모양 등의 정보를 가지고 유사 객체를 찾는 방식을 취하고 있으므로, 특정 사건이나 구체적인 인물 정보나 에피소드의 정보를 검색하고자 할 때는 키워드에 의한 검색을 해야하므로 불필요한 정보가 다량으로 검색되며 여러 번의 검색이 이루어져야 하는 단점이 있다.

본 논문에서는 멀티미디어 데이터를 구조와 내용에 따른 질의 및 검색이 가능하도록 XML를 이용하여 표현하고, 사용자에게 메타 정보, 구조 정보, 내용 정보에 따라 다양하게 장면/시퀀스/비디오를 검색하고, 접근할 수 있는 시스템을 제안한다. 이 시스템에서는 표현, 저장, 검색 등 모든 내부 처리가 XML을 통해 이루어지며, 이는 데이터의 재 사용성

*) 본 연구는 정보통신부, 디지털 영상물 콘텐츠 제작도구 개발 과제 사업에 의해 지원되었음.

과 부분 적인 데이터에 대해 유연한 접근이 가능하고, 사용자에게 다양한 형태의 검색결과가 제공된다. 사용자는 이 시스템을 통해 인물이나 객체의 행동, 특징 등의 시각 정보나 사건, 사건이 일어난 장소, 날짜 등의 내용 정보나 제목, 감독, 주연 등의 메타 정보를 통해 원하는 멀티미디어 데이터를 구조별로 접근할 수 있다.

2. 멀티미디어 데이터 표현

멀티미디어 데이터는 일반적으로 계층 구조를 포함하고 있다. 예를 들면, 하나의 카메라를 가지고 끊지 않고 찍은 일련의 프레임(frame)이 샷(shot)이다. 하나 이상의 샷이 모이면 장면(scene)을 이루는데, 장면은 한 객체와 객체의 관심사에 중점을 두고 있다. 이 장면들이 모이면 세그먼트(segment, sequence라고도 함)를 이루는데, 세그먼트는 영화속에서 분리될 수 있는 이야기들이다[3]. 이렇게 복잡하고 구조적으로 이루어진 멀티미디어 데이터를 표현할 경우, 사용자가 원하는 정보를 빠르고 정확하게 검색하고 접근할 수 있도록 표현하여야 한다. 이를 위해서는 우선 사용자의 요구 사항에 대한 정보를 파악해야 한다. 비디오 데이터를 브라우징(browsing)하거나 검색할 때 사용자는 대부분 “누가(who)”, “무엇을(what)”, “어디서(where)”에 관심을 갖고 있다. 즉, 질의와 시각 정보에 대한 표현을 시각 정보 기술자인 *people/object*, *action/event*, *location*과 같은 표현을 사용해 작성할 수 있다[4]. 또 [3]에서는 사용자와의 면접을 통해 샘플 질의를 작성하고 이 질의를 통해 1)서지 정보(title, abstract, subject keywords, genre, director, cast등)와 2)구조 정보(segments, scenes, shots, transitions, sound effects등)와 3)내용 정보(keyframe sequences, scripts, objects and actors in scenes등)의 세 가지 형태의 메타 데이터 인덱스를 정의하고 있다. 제안한 멀티미디어 표현 및 검색 시스템은 웹 상에서 정보의 표현과 교환의 표준으로 사용하는 XML을 기반으로 하고 있으며, 이러한 사용자 질의 경향을 반영하여 장면(Scene)을 기본 단위로 하고 상위 계층인 시퀀스(Sequence)까지의 계층 구조를 사용하며, 장면 내에서 인물들의 특징이나 행동, 사건, 장소 등의 내용 정보를 character, date, place, event, action 등의 기술자를 사용하여 표현한다. 이를 기반으로 정의된 DTD(Document Type Definition)는 <그림 1>과 같다.

```

<!ELEMENT video(meta,sequence*)>
<!ATTLIST video src CDATA #REQUIRED>
<!ATTLIST video start_offset CDATA #REQUIRED>
<!ATTLIST video end_offset CDATA #REQUIRED>

<!ELEMENT meta(title, date, director, cast, producer, production,
file_format, genre, awards)>
<!ATTLIST meta duration CDATA #REQUIRED>
<!ELEMENT title(#PCDATA)>
<!ELEMENT date(#PCDATA)>
<!ELEMENT director(#PCDATA)>
<!ELEMENT cast(#PCDATA)>
<!ELEMENT producer(#PCDATA)>
<!ELEMENT production(#PCDATA)>
<!ELEMENT file_format(#PCDATA)>
<!ELEMENT genre(#PCDATA)>
<!ELEMENT awards*(award_name, award_title, target_name)>
<!ELEMENT award_name(#PCDATA)>
<!ELEMENT award_title(#PCDATA)>
<!ELEMENT target_name(#PCDATA)>

<!ELEMENT sequence(se_title,scene*)>
<!ATTLIST sequence id CDATA #REQUIRED>
<!ATTLIST sequence seq_start CDATA #REQUIRED>
<!ATTLIST sequence seq_end CDATA #REQUIRED>

<!ELEMENT se_title(#PCDATA)>

<!ELEMENT scene(sc_title,character*, e_date?, e_place?, event?,
music*)>
<!ATTLIST scene id CDATA #REQUIRED>
<!ATTLIST scene sc_start CDATA #REQUIRED>
<!ATTLIST scene sc_end CDATA #REQUIRED>

<!ELEMENT sc_title(#PCDATA)>

<!ELEMENT character(character_type, character_sex, real_name?,
role_name?, clothes?, property*, action*)>
<!ELEMENT character_type(#PCDATA)>
<!ELEMENT character_sex(#PCDATA)>
<!ELEMENT real_name(#PCDATA)>
<!ELEMENT role_name(#PCDATA)>
<!ELEMENT clothes(clothes_type*, clothes_color*)>
<!ELEMENT clothes_type(#PCDATA)>
<!ELEMENT clothes_color(#PCDATA)>
<!ELEMENT property(#PCDATA)>
<!ELEMENT action(#PCDATA)>

<!ELEMENT e_date(#PCDATA)>
<!ELEMENT e_place(#PCDATA)>
<!ELEMENT event(#PCDATA)>

<!ELEMENT music(m_title, m_composer, m_singer)>
<!ELEMENT m_title(#PCDATA)>
<!ELEMENT m_composer(#PCDATA)>
<!ELEMENT m_singer(#PCDATA)>
    
```

<그림 1> 시스템에서 정의된 DTD

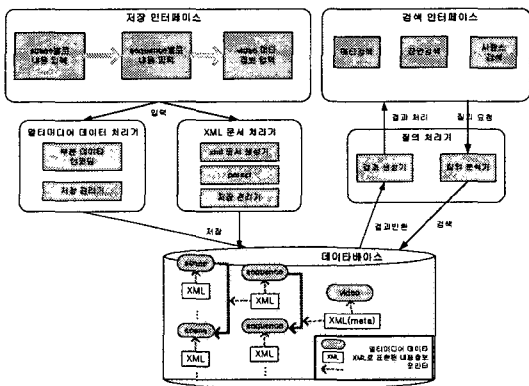
3. 장면에 의한 멀티미디어 데이터 검색 시스템

기존에 연구된 멀티미디어 표현 및 검색 시스템들은 사용자의 질의에 따르는 다양한 인덱스(Index)와 풍부한 표현을 제공해 주기는 하지만, 검색의 결과가 프레임(Keyframe)사용 단위로 이루어지고 있으며, 이를 위해 샷의 자동 검출이나 각 샷에서의 키 프레임임을 찾는 등의 복잡한 작업들이 포함되어져 있다. 그러나 웹 상에서 단순히 몇 가지 정보를 가지고 특정 장면만을 검색하고자 하는 경우는 효율성이 떨어지게 된다. 따라서 본 논문에서는 장면을 기본 단위로 하여 특정 장면에서의 사건 정보나, 장소, 객체(인물)의 동작이나 특정 정보를 XML을 사용하여 표현하며, 이렇게 표현된 내용 정보를 통한 장면 검색 및 구조 검색을 지원하는 멀티미디어 검색 시스템을 제안한다.

3.1 시스템 전체 구조

시스템은 전체적으로 저장 모듈과 사용자 모듈의 두 가지 모듈로 구성되어 있다. 저장 모듈은 시스템의 관리자 측면으로 멀티미디어 데이터와 그에 대한 정보를 저장하는 부분이다. 먼저 저장을 위한 사용자 인터페이스는 멀티미디어 데이터 파일을 관리자가 직접 보면서 내용에 대한 편집을 할 수 있도록 구성되어 있다<그림 4>. 내용 편집 시에 다른 내용 정보들과 함께 각 장면/시퀀스/비디오의 시작과 끝의 위치를 관리자가 직접 파악하여 입력한다. 작성된 내용을 처리기로 넘기면 처리기에서는 장면/시퀀스/비디오의 시작과 끝 포인터 정보를 보고 부분(part) 데이터를 인코딩하여 데이터베이스에 저장하며, 해당 내용 정보를 받아서 XML 문서를 만든 후 엘리먼트 별로 테이블에 저장한다. 데이터의 저장 구조와 저장 형태에 대해서는 3.3절에서 자세히 설명한다.

시스템의 사용자 측면은 사용자가 원하는 멀티미디어를 원하는 장면, 시퀀스, 비디오로 검색하여 볼 수 있는 부분이다. 검색 인터페이스에서 사건, 장소, 인물의 특징이나 행동 등의 내용 정보나 제목, 제작일, 감독, 주연, 제작, 파일 형태, 장르 등의 메타 정보를 통해 질의를 하면 원하는 장면/시퀀스/비디오와 해당 정보가 디스플레이 된다. 또한 원하는 장면 리스트 뿐 만 아니라 그 장면에 대한 전체 구조 정보까지 볼 수 있도록 되어 있다. 시스템 전체 구조는 <그림 2>와 같다.



<그림 2> 시스템 전체 구조

3.2 저장 구조

제안된 시스템의 하부 저장 구조는 기존의 고성능의 관계형 데이터베이스를 이용하고, XML 스키마로는 DTD(Document Type Definition)를 사용한다. DTD는 3.1에서의 표현구조를 가지고 정의되어 있

다. DTD에서 관계형 테이블로 사상하는 방법으로는 1)DTD에서 각 element를 하나의 테이블로 사상하고, element의 attribute를 테이블의 필드로 두는 방법과 2)element의 많은 자손들을 가능한 한 테이블의 필드로 뭉치어서 단편화 문제를 해결하는 방법, 3)DTD 그래프에서 in-degree가 없는 element만을 하나의 테이블로 사상하고, 그 테이블의 필드는 element의 자손들로 뭉치어서 element와 테이블이 1:1로 사상되는 결점을 보완하는 방법 등이 있다[6]. 본 시스템에서는 질의 형태가 구조적으로 이루어지며 구조가 데이터에 따라 동적인 구조를 가지므로 DTD에 독립적인 스키마 구조를 사용하여 저장한다. 4개의 테이블로 구성되어 있으며 설계된 테이블은 <그림 3>과 같다. 각 테이블에서 doc_id나 el_id 등은 유일성을 보장하기 위해 시퀀스로 생성하였고, 구조정보 저장을 위해 strc_id라는 계승 엘리먼트 ID를 사용한다. <그림 2>의 저장 인터페이스에서 해당 장면의 내용 정보를 입력하여 처리기로 넘기면, 처리기에서는 해당 장면의 시작과 끝 포인터 정보를 파악하여 인코딩 한 후 장면 테이블에 저장하게 된다. 또 이 장면에 대한 내용 정보는 엘리먼트와 속성 테이블에 엘리먼트와 속성별로 분리되어 저장된다. 또한 메타 정보를 이용하여 전체 비디오를 요청할 경우를 위하여, 문서테이블에 파일 전체에 대한 소스파일과 해당 내용 정보를 저장하고 있다.

doc_id	Number(7)
dtid_id	Number(7)
Source	Blob
Content	Clob

el_id	Number(7)
doc_id	Number(7)
el_name	varchar2(100)
el content	varchar2(200)
strc_id	varchar2(200)

atr_id	Number(7)
doc_id	Number(7)
atr_name	varchar2(100)
atr_value	varchar2(100)
el_id	Number(7)

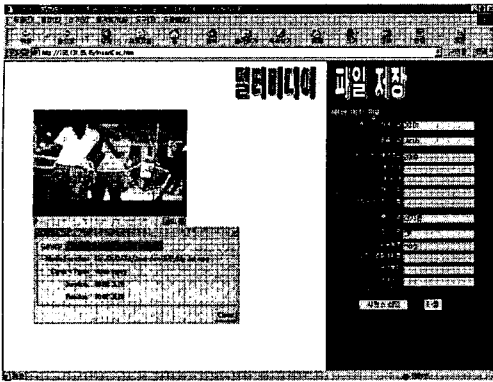
doc_id	Number(7)
el_id	Number(7)
sc_id	Number(7)
sc_sour	Varchar2(100)
sc_name	varchar2(100)

<그림 3> 설계된 데이터베이스 스키마

3.3 사용자 인터페이스

관리자는 <그림 4>와 같이 자신이 저장하기 원하는 멀티미디어 데이터를 선택하여 왼쪽 프레임에 로드(load)시킨 후, 데이터를 보면서 오른쪽 프레임에 메타 정보, 장면 정보, 시퀀스 정보 등을 차례차례 입력하면 된다. 입력을 마치고 데이터베이스에 등록하면 서버의 처리기에서 이 정보들을 받아 3.1절에

설명한 적절한 처리 과정을 거친 후 데이터 베이스에 저장된다.



<그림 4> 저장을 위한 사용자 인터페이스

사용자는 시스템에서 정의된 DTD(그림1)를 적용한 HTML 폼 형태의 사용자 인터페이스를 통해 특정 장면/시퀀스/비디오에 대한 검색을 하게 된다. 검색 결과 리스트에서 사용자가 원하는 장면을 선택하게 되면 해당 장면과 함께, 해당 비디오의 전체 구조 정보나 내용 정보 등을 볼 수 있으며, 관리자 모드로 들어가면 수정이 가능하도록 되어있다.

4. 결론

본 논문에서는 기존의 멀티미디어 검색 시스템에서 검색하기 힘들었던 인물의 특징, 사건, 특정 사건이 일어난 장소, 특정 배경 음악에 관한 정보를 가지고 장면 단위의 검색이 가능한 시스템을 설계하였다. 또한 이 시스템은 XML로 표현하고 저장하고 처리하였기 때문에 데이터를 재 사용하거나 검색 시에 유연하게 구조 접근이 가능하며, 검색된 장면에 대한 전체 구조 정보도 볼 수 있는 등의 다양한 검색 결과를 제공한다.

향후에는 다양하고 편리한 방식으로의 저장을 시도하고, 더욱 효율적인 저장과 검색을 위한 연구가 필요하며, 다양한 질의 방법을 통한 더욱 효율적인 검색의 고안이 필요하다.

참고 문헌

[1] Chung-Sheng Li, Rakesh Mohan and John R. Smith, "Multimedia Content Description In The InfoPyramid," IEEE ICASSP, Vol.6, pp.3789-3792, 1998

[2] Ajay Divakaran, Hirofumi Nishikawa, Kohtaro Asai, "A Description Scheme For Video Based On Feature Extraction In The Compressed Domain," IEEE, pp. 278-279, 2000

[3] Lawrence A. Rowe, John S. Boreczky, and Charles A. Eads, "Indexes for User Access to Large Video Databases," IS&T/SPIE Symp. on Elec. Imaging Sci. & Tech., San Jose, CA, February 1994

[4] A. Müfit Ferman, A. Murat Tekalp, and Rajiv Mehrotra, "Effective Content Representation for Video," IEEE, pp.521-525, 1998

[5] Zhigang Chen, See-Mong Tan, Aamod Sane, Dong Xie, "Video and Audio: Organization and Retrieval in the WWW," 1996

[6] 박은경, 정채영, 김현주, 배종민, "XML DTD로부터 관계형 테이블로의 사상구조 설계," 한국정보과학회, 2001 춘계 학술발표논문집, 제 28회, 2001

[7] 김연희, 신관섭, 김병곤, 이재호, 임해철, "내용기반 검색을 지원하는 멀티미디어 XML 데이터베이스 시스템," 한국 정보과학회, 2001 춘계 학술발표논문집, 제 28회, 2001