

MCML 기반 모션캡처 데이터 저장 및 퍼지 기반 모션 검색 기법

Storing and Retrieving Motion Capture Data based on Motion Capture Markup Language and Fuzzy Search

+정현숙* · 이성주*

Hyun-Sook Chung and Sung-Joo Lee

* 조선대학교 컴퓨터공학부

요 약

모션 캡처 기술은 현실감 있는 캐릭터 동작을 얻기 위해 많이 사용되고 있지만, 모션 캡처 데이터의 상이한 포맷들로 인하여 효율적인 모션 데이터의 저장과 검색이 어려운 문제점을 가지고 있다. 본 논문에서는 상이한 형식의 모션 캡처 데이터를 통합하고 효과적으로 저장 및 검색하기 위한 프레임워크를 제안한다. 상이한 모션 캡처 데이터 포맷들을 통합하기 위한 XML 기반의 표준 포맷을 MCML(Motion Capture Markup Language)로 정의하고 있으며 서로 다른 포맷의 모션 캡처 데이터 파일을 하나의 단일화된 MCML 파일로 변환하여 관계형 데이터베이스 또는 XML 데이터베이스에 저장함으로써 동일 데이터의 중복 저장 및 공유 문제를 해결한다. 모션캡처 데이터의 검색은 퍼지 문자열 검색(Fuzzy string searching) 기법에 의한 유사어 검색으로 특정 키워드를 포함하는 MCML 문서들을 찾거나 특정 위치의 시작 프레임에서 일련의 프레임들을 선택적으로 추출할 수 있는 모션클립(motion clips) 검색이 가능하도록 하였다.

키워드 : 모션 캡처 파일 포맷, 마크업 언어, 퍼지 문자열 검색, 모션 데이터베이스

Abstract

Motion capture technology is widely used for manufacturing animation since it produces high quality character motion similar to the actual motion of the human body. However, motion capture has a significant weakness due to the lack of an industry wide standard for archiving and retrieving motion capture data. In this paper, we propose a framework to integrate, store and retrieve heterogeneous motion capture data files effectively. We define a standard format for integrating different motion capture file formats. Our standard format is called MCML (Motion Capture Markup Language). It is a markup language based on XML (eXtensible Markup Language). The purpose of MCML is not only to facilitate the conversion or integration of different formats, but also to allow for greater reusability of motion capture data, through the construction of a motion database storing the MCML documents. We propose a fuzzy string searching method to retrieve certain MCML documents including strings approximately matched with keywords. The method can be used to retrieve desired series of frames included in MCML documents not entire MCML documents.

Key Words : Motion Capture file format, Markup Language, Fuzzy string search, Motion database

1. 서 론

모션 캡처 기법은 실세계의 캐릭터 움직임을 특수 카메라로 포착한 다음 이를 컴퓨터상의 가상 캐릭터에 적용하는 동작 생성 기법중 하나로서, 다른 동작 생성 기법과는 다르게 실제 세계의 캐릭터의 동작을 그대로 재현할 수 있다는 데 큰 이점이 있다. 그러나 모션 캡처 기술은 결정적인 단점을 갖고 있는데 현실감 있는 캐릭터 모션을 얻기 위해서는

기술력 있는 고수준의 인력, 고가의 장비, 많은 시간과 노력의 투자가 있어야 한다는 점이다. 또한 어렵게 생성한 고품질의 동작 데이터는 특정 캐릭터 및 상황에 맞춰진 고정화된 데이터로써 다른 목적으로 편집 또는 수정하여 재사용하기가 쉽지 않다는 것이다. 이러한 문제점을 극복하기 위해 많은 연구자들이 모션 합성 기술을 연구해오고 있다[6,7].

하지만 여러 동작들을 조합하여 새로운 동작을 생성하기 위해서는 모션 캡처 데이터가 가지는 다음의 몇 가지 문제를 먼저 해결해야 한다. 첫째, 모션캡처 데이터 포맷의 다양성 문제이다. 모션캡처 장비에 따라 서로 다른 구조의 데이터를 생성함으로써 다른 장비에 의해 캡처된 데이터의 통합, 교환, 공유 등을 어렵게 한다. 둘째, 이미 캡처된 데이터는 파일시스템에 의해 저장 및 검색되므로 접근성 및 데이터 중복의 문제가 발생한다. 현재 서비스되고 있는 모션 캡

접수일자 : 2006년 11월 17일

완료일자 : 2007년 3월 21일

감사의글 : 이 논문은 조선대학교 교내연구비의 일부지원에 의한 논문입니다.

+교신저자

처 데이터 라이브러리를 보면 동일한 동작 데이터를 가지는 서로 다른 포맷의 파일들이 제공되고 있다[4]. 셋째, 모션 데이터 파일의 검색이 키워드 검색과 디렉토리 검색에 의존하고 있으며 키워드와의 정확한 매칭에 의한 검색만 지원한다. 넷째, 캡처된 데이터 파일에서 일련의 프레임들로 구성된 부분적인 동작 데이터를 검색하고 추출하기가 어렵다. 모션 캡처 데이터 라이브러리에서는 파일단위로 제공되며 파일의 동작 데이터에 대한 설명이 부족하다. 또한 하나의 파일에 들어있는 동작을 부분적으로 검색할 수 있는 기능이 없으므로 여러 파일들에서 원하는 동작 프레임들만 선택해서 추출하기가 어렵다.

본 논문에서는 이러한 문제를 해결하기 위해 먼저, 다양한 포맷의 모션 캡처 데이터를 통합하기 위한 XML(eXtensible Markup Language)[15] 기반의 표준 마크업 언어인 MCML(Motion Capture Markup Language)[6]을 정의하였다. 서로 다른 포맷의 모션 캡처 데이터 파일을 하나의 단일화된 MCML 파일로 변환하여 저장함으로써 데이터의 중복성을 해결하였으며 XML 기반의 데이터 구조이므로 엘리먼트 수준의 검색이 가능함으로써 캡처된 데이터의 부분적 동작 검색을 위한 기반을 마련하였다. 또한 모션 캡처 데이터 라이브러리의 공유, 보안 및 접근성 문제는 관계형 또는 XML 데이터베이스에 MCML 파일을 분해하여 저장함으로써 해결할 수 있다. 본 논문에서는 관계형 데이터베이스에 저장하기 위해 MCML의 엘리먼트를 테이블과 컬럼으로 매핑하는 논리 스키마를 설계하였다. 모션캡처 데이터의 검색은 퍼지 문자열 검색(Fuzzy string searching) 기법에 의한 유사어 검색으로 특정 키워드를 포함하는 MCML 문서들을 찾거나 특정 위치의 시작 프레임에서 일련의 프레임들을 선택적으로 추출할 수 있는 모션클립(motion clips) 검색이 가능하도록 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문의 관련 연구에 대해 소개하고, 3장에서는 MCML의 구조와 내용에 대해 자세히 설명한다. 4장에서는 MCML 파일 생성, 저장 및 검색을 위한 시스템 구성을 소개하고 5장에서는 실험을 통하여 구현된 시스템의 결과를 보인다. 마지막 6장에서는 본 논문의 결론 및 향후 연구방향을 기술한다.

2. 관련 연구

우선, 캐릭터 애니메이션, 가상현실 그리고 다른 분야에서 데이터를 표현하기 위한 방법으로 XML 기반의 새로운 마크업 언어 개발과 관련된 기존 연구를 개략적으로 요약할 수 있다. Morales[5]는 본 논문의 MCML과 유사한 XML 기반의 모션 캡처 데이터 저장 방법을 고안했다. 이 방법에서 모션 캡처 데이터는 XML 데이터 형식으로 변환되어 저장됨으로서 애니메이터들이 웹 기반 환경 같은 상호 협력적인 환경에서 공동의 데이터에 접근 가능하게 하였다. 본 논문의 연구와 비교해 볼 때, 이 접근법은 세그먼트 단위의 간단한 형식으로 저장된 모션 캡처 데이터만을 다룰 수 있으며 캐릭터 구조체(skeleton)를 형성하는 계층적 구조는 고려되지 않았다. 더구나 본 논문의 MCML 언어와 같은 모션 캡처 데이터 통합을 위한 표준 마크업 언어의 구현을 제시하지 않았고 단지 XML을 사용한 데이터 변환의 가능성만을 언급했을 뿐이다.

VRML(Virtual Reality Modeling Language)[16]는 웹 상에서 3차원 환경을 구현하기 위해 제작된 언어이고

H-ANIM[17]은 아바타의 3차원 모델링에 사용되는 구조를 설명하는 것으로 Web3D 협회에 의해 설립된 표준 규정이다. H-ANIM에서 휴머노이드(humanoid)의 명세화는 VRML 2.0에서 사용된 휴머노이드를 나타내는 표준 방법을 따르고 있다. H-ANIM의 휴머노이드 노드의 구조는 모션 캡처 데이터의 구조와 유사하다. 이는 조인트(Joint), 세그먼트(Segment), 사이트(Site), 뷰포인트(Viewpoint) 노드들로서 인체 형태의 특징을 나타내는 골격 및 결합 구조를 정의하는 전반적인 역할을 하기 때문이다. H-ANIM은 온라인상의 가상 환경에서 인간 캐릭터를 표현하는데 있어서 적합한 언어이긴 하지만 H-ANIM이 가지고 있는 너무 많은 부가적인 구문들로 인해 표준적인 모션 캡처 데이터 포맷으로 사용하기에는 너무 복잡한 언어이다.

모션 캡처 데이터의 프레임 단위가 아닌 연속된 프레임의 집합인 세그먼트를 분리하고 제약조건을 만족하는 세그먼트들을 합성하여 하나의 동작을 완성하는 연구로서 [6,7,11,12]이 있다. 이들 연구는 모션 데이터의 의미 정보를 활용하기 보다는 예제(example) 동작과 유사한 동작을 제약조건 기반으로 분석하여 탐색하는 방법이다. 부분적 모션 클립을 검색하는 것과 관련된 연구로서 비디오 데이터에서 특정 구간을 검색하기 위해 색감, 색상, 텍스처와 같은 저수준의 정보 또는 자유 형식의 텍스트 설명과 같은 고수준의 정보를 기반으로 하는 비디오 데이터 검색이 있다[9,14]. 이 비디오 데이터 검색의 자유 형식 텍스트 기술 방법은 비표준적이고 여러 비디오 데이터 사이에 공유가 어려운 점이 있으나 온톨로지에 기반하여 의미 정보를 부여할 경우 여러 모션 데이터의 동작에 대한 표준적인 의미 정의와 다양한 연관관계를 정의할 수 있으며 이러한 의미 정보는 모든 모션 데이터들 사이에 공유가 가능해진다.

3. MCML 문서구조 설계

초기 설계에 비해 현재의 MCML은 아스키(ASCII) 형식의 ASF(Acclaim Skeleton File)/AMC(Acclaim Motion Capture data)[1], BVH(Biovision Hierarchical data)[2], HTR(Hierarchical Translation-Rotation)[12], 뿐만 아니라 바이너리 형식의 C3D[3] 모션 캡처 데이터 포맷들을 단일 형식으로 통합하기 위한 태그들을 정의하고 있다. ASF/AMC, BVH, HTR 및 C3D 형식은 가장 보편적이며 산업 표준 모션 캡처 데이터 형식이며 대부분의 모션 소프트웨어에서 지원하고 있다. 또한 현재 유료 또는 무료로 구할 수 있는 모션 캡처 데이터 파일의 대부분이 이 네 가지 파일 형식으로 제공되고 있다. 그러므로 MCML은 이 네 형식을 통합할 수 있는 구조를 가진다.

MCML의 문서 구조에서 루트 엘리먼트인 `<mcml>`은 그 하위에 `<meta>`, `<header>`, `<skeleton>`, `<motion>` 엘리먼트들을 가진다.

3.1 META 엘리먼트

`meta` 엘리먼트는 8개 하위 엘리먼트를 가지고 있으며 MCML 문서 파일의 메타데이터를 기술한다. `meta` 엘리먼트의 구조는 그림 1에서 보이고 있다.

`title` 엘리먼트는 MCML 문서 파일에 주어진 제목이며 `creator` 엘리먼트는 모션 캡처 데이터를 생성한 사람이나 기관을 말한다. `subject` 엘리먼트는 모션 캡처 데이터 파일의 캡처된 동작에 대한 주제(예: 발레, 춤 등)이며 `de-`

scription 엘리먼트는 동작에 대한 설명을 기술하는 곳이다. *format* 엘리먼트는 ASF/AMC, BVH, HTR 등의 모션 캡처 데이터 포맷을 나타내며 *duration* 엘리먼트는 전체 프레임의 실행 시간을 나타낸다. *category* 엘리먼트는 동작의 카테고리(예: 스포츠, 이동, 인간의 상호작용 등)를 기술하는 곳이다. *category*, *subject*, *title*은 모션 캡처 데이터 파일을 카테고리별로 구성 및 검색하기 위해 사용된다.

```
<! ELEMENT meta (title, creator, subject, description,
date, format, duration, category)
<! ELEMENT title (#PCDATA)>
<! ELEMENT creator (#PCDATA)>
<! ELEMENT subject (#PCDATA)>
<! ELEMENT description (#PCDATA)>
<! ELEMENT date (#PCDATA)>
<! ELEMENT format (#PCDATA)>
<! ELEMENT duration (#PCDATA)>
<! ELEMENT category (#PCDATA)>
```

그림 1. MCML 메타데이터의 구조

3.2 HEADER 엘리먼트

header 엘리먼트는 15개의 하위 엘리먼트로 구성되어 있으며 그림 2에 그 구조를 보이고 있다. *meta* 엘리먼트와 달리 *header* 엘리먼트는 MCML 파일의 데이터 특성을 기술하기 위해 사용된다.

3.3 SKELETON 엘리먼트

skeleton 엘리먼트는 캐릭터의 계층적 구조를 정의한다. 그림 3에 *skeleton* 엘리먼트의 논리적인 구조를 보이고 있다.

```
<! ELEMENT header (filetype, datatype?, filename,
version?, skeleton_name?, units?, num_segments?,
num_frames, dataframe_rate, euler_rotation_order?,
calibration_unit?, rotation_unit?,
global_axis_of_gravity?,bone_length_axis?,scale_factor?)
```

그림 2. MCML header 엘리먼트의 구조

```
<! ELEMENT skeleton (root, bone+)>
<! ELEMENT root EMPTY>
<! ATTLIST root order CDATA #REQUIRED
axis CDATA #REQUIRED
position CDATA #REQUIRED
orientation CDATA #REQUIRED>
<! ELEMENT bone (name, direction?, length?, position?,
axis?, order?, dof?, limits?,bodymass?, cofmass?, offset?,
channels?, bone+)>
<! ATTLIST bone id ID #IMPLIED>
```

그림 3. MCML skeleton 엘리먼트의 구조

root 엘리먼트는 캐릭터 계층 구조의 부모 엘리먼트로서 *axis*와 *order*속성은 초기 오프셋과 루트 노드 변환의 수행 순서를 나타낸다. *position* 속성은 루트 변환을 나타내며 *orientation* 속성은 을 정의한다. *skeleton* 엘리먼트는 하나 또는 그 이상의 *bone* 엘리먼트를 갖고 있으며 *bone* 엘리먼트는 캐릭터의 계층 구조를 정의하기 위해 그 하위에 순환

적으로 *bone* 엘리먼트를 가진다. *name* 엘리먼트의 값으로는 표 1에 정의한 MCML 관절체 이름을 기술하여야 한다.

3.4 MOTION 엘리먼트

motion 엘리먼트는 하나 이상의 *frame* 엘리먼트로 구성 되어 있으며 *frame* 엘리먼트는 각 관절체의 프레임을 정의 한다. *motion* 엘리먼트의 논리적인 구조는 그림 4에 나타나 있다.

frametime 엘리먼트가 각 프레임당 실행 시간인 반면 *frames* 엘리먼트는 프레임의 개수를 말한다. *frame* 엘리먼트는 *skeleton* 엘리먼트에서 정의된 각 관절체의 동작을 나타내기 위해 하나 또는 그 이상의 *frame_bone* 엘리먼트를 가지며, 하나의 *frame_bone* 엘리먼트는 한 프레임에서의 관절체의 전이 및 회전 값을 가진다.

```
<! ELEMENT motion (frames, frametime, frame,
motion_name*)>
<! ELEMENT frames (#PCDATA)>
<! ELEMENT frametime (#PCDATA)>
<! ELEMENT frame (frame_name, frame_bone+)>
<! ATTLIST frame id ID #IMPLIED>
<! ELEMENT frame_bone EMPTY>
<! ATTLIST frame_bone name CDATA #REQUIRED
Tx CDATA #IMPLIED Ty CDATA #IMPLIED
Tz CDATA #IMPLIED
Rx CDATA #REQUIRED Ry CDATA
#REQUIRED Rz CDATA #REQUIRED>
```

그림 4. MCML motion 엘리먼트의 구조

4. 시스템 구성

모션 캡처 데이터 통합 및 구현된 시스템은 그림 5와 같이 모션 캡처 데이터 파일을 MCML 파일로 변환 및 저장하는 MCML 변환 관리 모듈과 MCML 데이터베이스로부터 퍼지 검색으로 MCML 파일을 찾거나 질의어를 이용한 부분 모션 클립들을 검색하는 검색 모듈로 나누어진다. MCML 변환 관리 모듈의 주요 기능은 모션 캡처 데이터 파일과 MCML 파일 사이의 상호 변환 기능, MCML 파일의 편집 기능, 모션 데이터의 플레이 기능, MCML 파일의 관계형 데이터베이스로의 저장 기능 등이며 검색 모듈은 색인 관리, 퍼지 검색, 질의어 처리 등의 기능을 가진다.

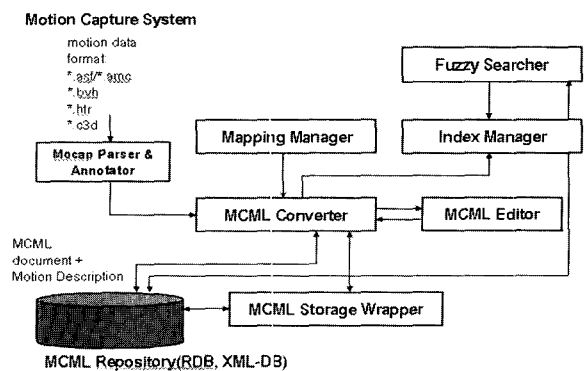


그림 5. MCML 관리를 위한 시스템구조

표 1. 모션캡처 데이터 형식의 상이한 관절체명과 단일화한 MCML의 관절체명

	MCML	ASF/AMC	BVH(1)	HTR	C3D
1	root	h_root	root	undefined	undefined
2	right_head	h_head(head)	head	head	RFHD
3	left_head	undefined	undefined	undefined	LFHD
4	neck1	h_neck1(upperneck)	neck	undefined	undefined
5	neck2	h_neck2	undefined	undefined	undefined
6	left_shoulder	h_left_shoulder(lclavicle)	leftcollar	undefined	LSHO
7	left_up_arm	h_left_up_arm(lhumerus)	leftuparm	lupperarm	LUPA
8	left_low_arm	h_left_low_arm(lradius)	leftlowarm	llowarm	LFRM
9	left_wrist	(lwrist)	undefined	undefined	LWRA
10	left_hand	h_left_hand(lhand)	lefthand	lhand	LFIN
:	:	:	:	:	:
50	right_toe_two	h_right_toe_two	undefined	undefined	undefined
51	right_toe_three	h_right_toe_three	undefined	undefined	undefined
52	right_toe_four	h_right_toe_four	undefined	undefined	undefined
53	right_toe_five	h_right_toe_five	undefined	undefined	undefined

4.1 MCML 변환 및 저장

ASF/AMC, BVH 또는 HTR 포맷의 모션 캡처 데이터 파일이 입력으로 들어오면 가장 먼저 MoCap 구문분석기(MoCap Syntax Analyzer)가 구문 분석을 시작한다. 이 구문분석기는 파일의 내용을 탐색하면서 유효한 토큰들을 추출하고 토큰테이블에 이 토큰과 값들을 저장한다. 토큰테이블에는 헤더 토큰, 스켈레톤 토큰, 그리고 모션 토큰들이 저장된다. 구문분석기에 의해 토큰테이블에 토큰들이 저장되면 MCML 변환기는 MCML 파일 생성에 이 토큰테이블과 매핑테이블을 이용한다. 매핑테이블은 태그 매핑테이블과 관절체 매핑테이블로 구성된다. 관절체 매핑테이블은 MCML 관절체 명과 다양한 모션 캡처 포맷의 관절체 명 사이의 매핑관계를 저장하는 테이블이고 태그 매핑테이블은 MCML 태그 집합과 다양한 모션캡처 포맷의 키워드들 사이의 매핑관계를 저장하는 테이블로서 헤더 데이터 태그 집합, 캐릭터 구조체 태그집합, 그리고 모션 데이터 태그집합을 내용으로 가진다. 이들 두 테이블은 MCML 변환기가 변환 과정에서 참조하는 사전으로 제공된다.

4.2 퍼지 문자열 검색

퍼지 문자열 검색은 주어진 키워드와 정확하게 일치하지 않더라도 유사성에 기반하여 검색 결과를 생성하는 기법이다. MCML 문서의 헤더에는 제목, 주제, 설명이 있으며 각각 모션캡처 데이터의 모션 제목, 주제, 그리고 모션에 대한 설명을 가진다. 이 문자열들에서 입력된 키워드와 유사한 단어를 가지는지 문자열 검색을 함으로써 특정 모션을 포함하고 있는 MCML 파일들을 검색할 수 있다.

예를 들어, 설명에 “walking, jumping, turn left and kick”의 문자열을 가질 경우 입력된 키워드 “walk”에 대해 n-그램(n-gram) 문자열 매칭 기법에 따라 walk와 walking 사이에 유사성을 발견하게 된다. 모션캡처 데이터의 제목, 주제, 설명은 미리 결정된 통제어(controlled vocabulary)가 아닌 자유롭게 기술될 수 있기 때문에 입력된 키워드와 정확하게 일치하는 검사로서는 원하는 모션 데이터

를 검색하기 어렵다. 따라서 단어 사이의 유사값 산출에 의해 매치되는 모션 데이터를 검색하는 퍼지 문자열 검색 기법이 효과적이다.

퍼지 문자열 검색을 위해 MCML 변환시에 제목, 주제, 설명에 들어있는 단어들에 대한 표제어를 형태소 분석기를 통해 산출한다. 예를 들어, walking, walks, walked 등의 표제어는 walk가 되며 이 표제어를 MCML 파일 ID와 함께 역색인에 저장하는 것이다. 그림 6은 역색인 구조를 보이고 있다. 키워드가 입력되면 형태소 분석기를 통하여 표제어를 산출하고 이를 포함하고 있는 MCML 파일의 ID들을 역색인에서 검색한다. 검색된 ID 값들을 조건절로 가지는 SQL문을 작성하고 모션 데이터베이스로부터 ID 값을 가지는 MCML 파일들을 검색한다.

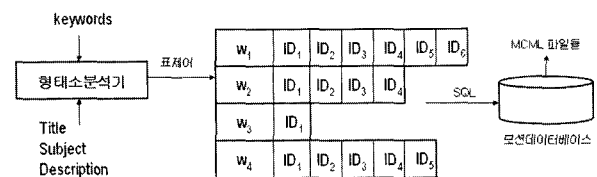


그림 6. 모션 표제어 역색인 구조

5. 구현 결과

실험 데이터로 본 논문에서는 카네기 멜론 대학의 그래픽스 연구실에서 제공하는 모션 캡처 데이터 라이브러리의 파일들을 사용하였다. 이 파일들은 무료로 제공되고 있으며 댄스, 감정표현, 상호작용, 기본 동작 등의 카테고리별로 ASF/AMC, BVH, HTR, C3D 포맷의 파일들을 다운로드할 수 있도록 한다. 다운로드받은 모션 캡처 데이터 파일은 MCML 변환기를 이용하여 MCML 파일을 생성하고 아래 그림 7의 관계형 데이터베이스 스키마에 저장하거나 그림 8의 엑셀론(eXcelon) DXE Manager에 저장한다.

MCML 데이터베이스에서 모션 데이터를 검색하는 방법은 데이터베이스에 직접 접근하여 SQL 질의어나 XPath 질의어로 검색하는 방법과 퍼지 문자열 검색을 이용한 유사어 검색 방법이 있다. 퍼지 문자열 검색을 위해 MCML 문서의 제목, 주제, 설명에 대해 역색인을 구축하였으며 이 색인을 이용하여 키워드 검색을 지원하였다. 그림 9는 키워드 검색에 의해 모션 데이터베이스로부터 검색된 댄스 동작들을 보인다.

검색된 동작 토픽을 선택시 해당 토픽의 상세 내용을 보이는 JSP 페이지를 실행한다. 상세 페이지에는 동작명, 동작 설명, 한글명, 상위 동작명, 연결된 모션 클립의 주소 등을 보인다. 모션 클립의 주소를 클릭하면 모션 데이터베이스에서 MCML 파일의 해당 프레임 구간을 검색하여 XML 구문으로 보이고 모션 뷰어를 이용하여 동작을 플레이한다.

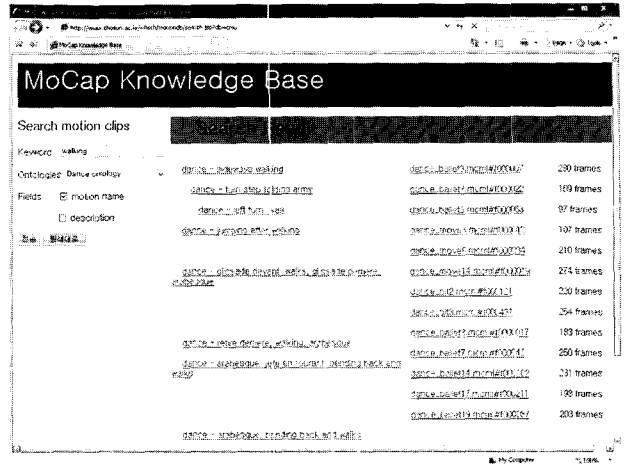


그림 9. walking 키워드 검색 결과

6. 결론

본 논문에서는 다양한 모션 캡처 데이터 포맷을 통합하기 위한 XML 기반의 마크업 언어를 설계하였고 이를 효과적으로 저장하고 퍼지 문자열 검색에 의해 유사 검색이 가능한 프레임워크를 제안하였다. 기존의 모션 캡처 데이터 라이브러리는 모션 캡처 데이터를 파일 단위로 저장함으로써 데이터의 중복, 공유의 어려움, 보안 및 관리의 어려움 등이 있다. 그러나 MCML로 변환되어 저장할 경우 상용 데이터베이스를 활용함으로써 데이터의 공유, 트랜잭션 처리, 보안 등 여러 가지 문제를 해결할 수 있다. 또한 파일 전체를 검색하고 다운로드하는 대신 엘리먼트 수준의 작은 의미 단위로 검색할 수 있으며 캐릭터 구조체와 모션을 분리하여 검색할 수도 있다.

키워드 검색에서는 퍼지 문자열 검색 기법을 적용하여 문자열 사이의 유사성 매칭을 수행함으로써 자유롭게 기술된 모션 제목, 주제, 설명 등으로부터 정확한 검색을 유도할 수 있다. 모션 주제는 댄스, 운동, 게임 등 캡처된 모션 데이터가 어떤 분야의 동작을 표현한 것인지를 가리킨다. 이 주제를 구체화함으로써 모션 온톨로지를 생성할 수 있다.

모션 온톨로지는 모션 데이터베이스 상위에 의미 기반 검색을 위한 지식층을 제공함으로써 유사한 동작을 가지는 모션 클립들을 쉽게 검색할 수 있도록 한다. 향후 연구 과제인 모션 클립의 자동 추출 및 온톨로지로서의 자동 분류가 가능할 경우 본 논문의 모션 캡처 데이터베이스와 온톨로지는 게임 및 영화 제작, 스토리텔링 시스템, 캐릭터 애니메이션 등 모션 캡처 데이터를 활용하는 모든 분야에 지식 저장소로서 활용될 수 있으리라 본다.

참고 문헌

- [1] Acclaim, *ASF/AMC File Specifications page*, <http://www.darwin3d.com/gamedev/acclaim.zip>.
- [2] Biovision, *BVH Specifications page*, <http://www.biovision.com/bvh.htm>.
- [3] C3D Specification, <http://www.c3d.org>.
- [4] CMU Motion Database, <http://mocap.cs.cmu.edu>.

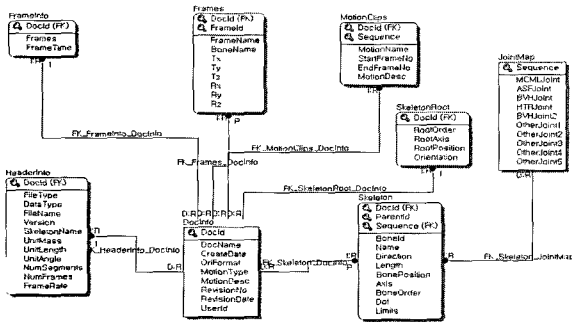


그림 7. MCML 파일을 관계형 데이터베이스에 저장하기 위한 스키마

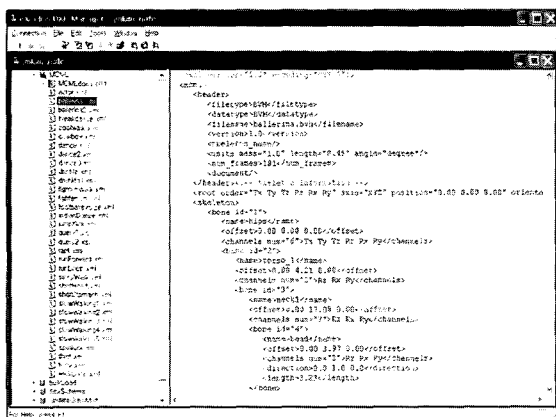


그림 8. eXcelon DXE Manager에서 저장된 MCML 파일을 검색한 화면

- [5] C.R. Morales, *Development of an XML Web Based Motion Capture Data Warehousing and Translation System for Collaborative Animation Projects*, In Proceedings of the 9-th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2001.
- [6] E. Hsu, S. Gentry and J. Popovic, *Example-Based Control of Human Motion*, Eurographics/ACM SIGGRAPH Symposium on Computer Animation, 2004.
- [7] L.M. Tanco and A. Hilton, *Realistic Synthesis of Novel Human Movements from a Database of Motion*, In Proceedings of the IEEE Workshop on Human Motion HUMO 2000.
- [8] M. Hacid, C. Declair and J. Kouloumdjian, *A Database Approach for Modeling and Querying Video Data*, IEEE Transaction on Knowledge and Data Engineering, Vol. 12, No. 5, pp.729-750, 2000.
- [9] M. Meredith and S. Maddock, *Motion capture file formats explained*, Department of Computer Science Technical Report CS-01-11.
- [10] O. Arikan and D.A. Forsyth, *Interactive Motion Generation from Examples*, In Proceedings of ACM SIGGRAPH 2002.
- [11] O. Arikan, D.A. Forsyth, and J.F. O'Brien, *Motion Synthesis from Annotations*, In Proceedings of ACM SIGGRAPH 2003.
- [12] S. Rosenthal, B. Bodenheimer, C. Rose, and J. Pella, *The process of motion capture: Dealing with the data*, In Proceedings of the 8th Eurographics Workshop on Animation and Simulation 1997.
- [13] S.P. Chao, C.Y. Chiu, S.N. Yang and T.G. Lin, *Tai Chi synthesizer: a motion synthesis framework based on key-postures and motion instructions*, Comp. Anim. Virtual Worlds 2004, vol. 15, pp.259-268, 2004.
- [14] W3C, Extensible Markup Language (XML) 1.0, <http://www.w3c.org/TR/1998/RECxml-19980210>, 1998.
- [15] Web3D Consortium, VRML International Standard, http://www.web3d.org/technicalinfo/specifications/ISO_IEC_14772-All/index.html.
- [16] Web3D Consortium, H-ANIM 2001 Specification, <http://www.h-anim.or>

저 자 소 개



정현숙(Hyun-Sook Chung)

1995년 : 대구가톨릭대학교 전산통계학과 이학석사
 2003년 : 연세대학교 컴퓨터과학과 공학박사
 2006년~ 현재 : 조선대학교 컴퓨터공학부 전임강사

관심분야 : 게임공학, 웹컨텐츠제작, 모션그래픽스, 시멘틱웹
 E-mail : hsch@chosun.ac.kr



이성주(Sung-Joo Lee)

1992년 : 광운대학교 전자계산학 이학석사
 1998년 : 대구가톨릭대학교 전자계산학 이학박사
 1988년~1990년: 조선대학교 전자계산소 소장
 1995년~1997년: 조선대학교 산업대학장
 1981년~ 현재 : 조선대학교 컴퓨터공학부 교수

관심분야 : 소프트웨어공학, 프로그래밍언어, 러프집합
 E-mail : sjlee@chosun.ac.kr