

## MPEG-4 BIFS를 이용한 멀티미디어 서비스 구현

이희훈<sup>o</sup> 이경주 남진아 이보경 정의훈

한국산업기술대학교 컴퓨터공학과

{ hiuni<sup>o</sup>, jinwool97, abulbjina, leebk, ehjeong }@kpu.ac.kr

### Implementation of multi media service using MPEG-4 BIFS

Heehoon Lee<sup>o</sup>, Kyungju Lee, Jina Nam, Bokyoung Lee, Euihoon Jung  
Korea Polytechnic University

#### 요 약

MPEG-4 BIFS는 장면서술언어로서 본 논문에서는 MPEG-4 영상을 이용한 양방향 멀티미디어 서비스를 구현하는데 그 목적이 있으며, 장면서술은 MPEG-4 장면을 구성하는 객체들의 시공간적 구성관계를 기술하는 것이다. 또한, 본 논문에서는 MPEG-4에 정의된 다양한 데이터 포맷들로 영상 신호와 대응되는 실시간 데이터정보를 Display 하여 사용자 입력을 받아 그 정보를 사용자에게 보여 줌으로써 양방향 서비스를 사용자에게 제공할 수 있도록 하였으며, 양방향데이터서비스가 지원되는 플레이어 개발하였다.

#### 1. 서 론

MPEG-4는 기존의 MPEG-2포맷과는 달리 적은 전송율의 인터넷 대역에서 사용할 수 있도록 영상의 압축률이 높으나 화면의 품질은 MPEG-2포맷의 화면보다는 화질이 떨어진다. 현재 지상파 DMB는 MPEG-4 표준을 따르지만 영상의 경우에는 H.264기술[1]을 사용하여 화질이 떨어지는 단점을 보완하였다.

MPEG-4의 영상품질은 작은 화면의 모바일기와 같은 곳에 사용되도록 맞춰졌으며, 기존의 영상과 오디오만이 아닌 데이터 즉 BIFS(Binary Format for Scenes)를 포함하여 영상 신호와 대응되는 실시간 데이터정보를 디스플레이 할 수 있으며 네트워크 연동을 할 경우 사용자 입력을 받아서 그 결과를 처리하여 서버에서 사용자 입력의 처리를 통계하여 영상에 반영 할 수 있는 서비스를 제공할 수 있다.

MPEG-4 파일 형식 중 장면서술 정보를 저장하는 BIFS(Binary Format for Scenes)형식은 장면서술을 이진 부호화한 것이다. MPEG-4는 VRML과 2D 객체, FBA, 그리고 3D 메쉬 부호화를 포함시키기 위하여 VRML을 확장하였다. 이를 확장하여 정의하고 있는 MPEG-4 장면서술이 XMT(eXtensible MPEG-4 Texture Format)이다. XMT로 저장된 MPEG-4 콘텐츠는 XML에 기반을 두고 있기 때문에 멀티미디어 콘텐츠를 인터넷상에서도 활용할 뿐만 아니라 기존 언어의 확장성 및 이식성을 용이하게 하는 등의 특징을 가질 수 있다[2].

현재 MPEG-4 BIFS를 재생기 중 GPAC Osmo4 플러그인이 있는데, 이 재생기는 MPEG-4 시스템 표준에 기반한 멀티미디어 프레임워크 이다[3]. GPAC은 많은 멀티미디어 포맷들을 지원하는데, 단순한 오디오비주얼 컨테이너(avi, mov, mpg)에서부터 복잡한 프리젠테이션

포맷들(MPEG-4 Systems, VRML/X3D, SVG)이르기까지 지원한다. 또한 XMT, BT로 구성된 파일을 MP4BOX라는 파싱프로그램을 사용하여 MP4파일로 만들 수 있으며 이러한 기능을 사용하여 본 논문에서는 XMT와 BT로 구성된 파일을 사용하여 멀티미디어 서비스 구현을 한다.

#### 2. 관련기술

##### 2.1 MPEG-4 파일 구조

아래의 그림 1은 MPEG-4 파일의 개략적인 파일 포맷의 구조를 보여준다.

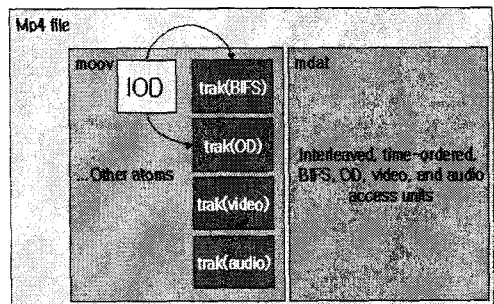


그림 1 MPEG-4 파일 구조

그림에서 MPEG-4 파일의 구조는 크게 프레젠테이션과 관련되어 Meta-data 정보를 제공하는 moov와 실제 Media에 대한 데이터를 제공하는 mdat로 나뉘어진다. MPEG-4 포맷을 구성하는 단위는 Object로 여기서 Object는 moov와 mdat를 구성하는 최소 단위의 의미로 해석하면 된다. moov Object(Atom)는 앞에서 설명한 것 같이 프레젠테이션을 위한 모든 Meta-data 정보를 가지고 있어서, MPEG-4 표준을 위하여 IOO, OD, BIFS, 미디어 데이터에 대한 Meta-data 정보를 제공해 주게 된다. 이 Meta-data 정보를 통해서 실질적인 Media 데이

터(mdat)에 대한 접근과 해석을 할 수 있게 된다[4].

2.2 MPEG-4 재생기 구조

MPEG-4 재생기 시스템에서 디멀티플렉서에 의해 다역상(Demuxing)이 수행된다. 디역상된 결과로 생성되는 장면서술 정보와 각 미디어 데이터들은 디코딩과 컴포지션 과정을 거친다[5]. MPEG-4 재생기는 장면서술 정보에 따라 각 미디어 렌더링 모듈에 의해 사용자 인터페이스 상에서 재생이 되는 구조이다. 재생 중에 발생하는 사용자 이벤트처리는 사용자 이벤트 매니저모듈에 의해 관리 및 처리된다. 본 논문에서 사용된 재생기의 구조는 그림 2와 같다.

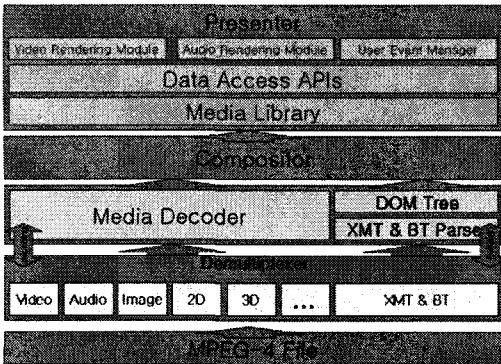


그림 2 재생기 구조

3. 시스템 구현

3.1 BIFS(Binary Format for Scenes)

MPEG-4 장면을 기술하게 되는 노드는 다양한 노드들의 집합으로 표현될 수 있는데, 노드는 MPEG-4에서 규격화된 비디오와 오디오를 지원하기 위한 노드, 2D 장면을 나타내기 위한 노드, 3차원 장면 재생을 위한 3D 노드 등으로 구성될 수 있다. 장면서술은 또한 2D 장면의 일부에 3D 장면을 포함시켜 복잡한 장면을 기술할 수 있을 뿐만 아니라, 확장VRML형식의 노드 기술에서는 사용자 조작과 제어 작업을 직접 함으로써 장면을 자유롭게 갱신 될 수 있다.

아래 그림 3은 확장VRML인 BT언어를 사용하여 기술한 예이다. 본 논문에서는 3D 객체의 노드 기술을 사용하여 객체 렌더링에 필요한 노드 정의뿐만 아니라, 3D 노드가 가진 특성이 기술되는 속성노드와 노드의 값이 기술된다. 그리고, 각 노드는 그룹노드와 자식노드로 직접 지정 가능하며, 또한 MPEG-4 각 객체는 하나의 노드를 생성하고, 생성된 노드에는 여러 특성들에 해당하는 속성노드를 생성한다.

그 노드는 객체노드의 자식노드로 불게 되고 그러한 객체노드들은 그룹을 형성하여 그룹노드의 자식노드가 되는 구조로 여러 단계의 계층구조를 가질 수 있다. 각 객체노드에는 요소(Element)들을 가지고 있으며, 요소들은 다시 자신들의 특성(Attribute)을 가지며, 이러한 요소와 특성들은 여러 단계의 계층 구조를 가질 수 있는 형태를 가지고 있다. 그러한 노드는 하나나 그 이상의 자식노드를 가질 수 있으며, 노드들을 그룹으로 묶어서 해당노드

의 위치나 속성을 지정 할 수 있다.

시스템에서 정의되는 노드의 단계는 group, children, transform, shape 순으로 포함되며, Sphere, Cone, Cylinder, Box등의 형상 속성이 있으며, 각 노드는 노드의 특성에 맞게 geometry, position, color, texture 등과 같은 속성들을 가지게 된다. 예를 들어 Cone노드인 경우 Shape노드를 부모노드로 가지며, bottomRadius, height, side, bottom 같은 속성노드를 가진다. 노드들로 구성된 객체의 장면서술은 텍스트 기반의 특징으로 사용자 이벤트 같은 추가적인 장면 제어로 노드의 확장이 가능하다[6]. 또한 이러한 신 서술 방법은 기존의 BIFS에 의한 장면서술 방법을 포함하는 것이기도 하다.

```

# switch start
Group { children [ Transform { translation 0 -2 0 children [ DEF MySwitch
whichChoice 0
choice { Shape { appearance Appearance {
material Material ()
texture ImageTexture {url "bl:
}
geometry IndexedFaceSet {
coord Coordinate {
point [-3 -3 -1, 3 -3
]
}
coordIndex [0, 1, 2, 3]
}
}
}Shape
}
Anchor {
url "http://www.terious.co.kr/"
children [
#-0
Shape {
appearance Appearance {
material Material ()
texture ImageTexture {url "ahnp.jpg"}
geometry IndexedFaceSet {
coord Coordinate {point [-1 -1
]
coordIndex [0, 1, 2, 3]
}
}
}
}
Anchor {
url "http://kr.sports.yahoo.com/jspark/"
children[
Shape { #-1
appearance Appearance {
material Material ()
texture ImageTexture {url "pap.jpg"}
}
geometry IndexedFaceSet {
coord Coordinate {
point [-1 -2 0, 1 -2 0, 1 0 0,
]
}
}
}
}
    
```

그림 3 BT파일

아래의 그림 4는 MPEG-4의 장면서술 언어인 XMT로 기술된 예제 파일로써 그림 3과 같은 내용의 장면을 가지는 파일이다. 표현형식은 XML에 기반하였으며 MPEG-4 시스템의 BIFS를 기술하기 위한 언어이다. 그림 4에서 위부분은 버전과 인코딩이 어떠한 것인지 나타내며 그 아래 부분부터는 해당파일에 대한 정보를 나타내고 있다. 이와같은 부분은 BT파일도 비슷하나 다만 표현형식이 XMT와 같은 경우에는 XML형식을 취하여서 화면의 표현에 대한 설정과 해상도 정보, 객체들에 대한 정보가 있으며, BT파일과는 달리 구문시작과 끝이 표기되어 있어 사용자로 하여금 알아보기 편하게 각 노드의 시작과 끝이 구분되어 표시되어 있다. 이 경우 복잡하게 기술되어 있는 프로그램을 한눈에 알아 볼 수 있어 잘못된 부분을 쉽게 찾을 수 있다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<XMT xmlns="urn:mpeg:mpeg4:data:schema:2002" xmlns:xsi="http://www.w3.org/2001/
<Header>
<InitialObjectDescriptor objectDescriptorID="J1" binaryID="1" >
<ProfileLevel profileLevelIndication="1" graphicsProfileLevelIndication="1" />
<Descr>
<ES_Descr ES_ID="eel" binaryID="1" >
<DecConfigDescr>
<DecoderConfigDescriptor objectTypeIndication="1" streamType="3" bufferSize
<DecSpecificInfo>
<BIFSDef ncdIDbits="5" routeIDbits="5" >
<comandStream>
</BIFSContig>
<DecSpecificInfo>
<DecoderConfigDescriptor>
</DecConfigDescr>
<SContigDescr>
<CustomUseAccessUnitStartFlag="true" useAccessUnitEndFlag="true" timeStac
</Custom>
</SContigDescr>
</DecConfigDescr>
</ES_Descrictor>
</ES_Descr>
</Descr>
</InitialObjectDescriptor>
</Header>
<Body>
<Place>
<Scene>
<Group>
<children>
< transform DEF="NO" translation="0 3 0">
<children>
<Shape>
<appearance>
```

그림 4 XMT파일

### 3.2 인터넷 연결

확장된 VRML로 작성시 노드를 인터넷 사이트로 연결 시키려면 노드를 Anchor의 단계 안에 url 속성을 지정하여 해당사이트의 주소를 입력하면 된다. 이것은 노드 자체를 링크로 작용 할 수 있게 하며, vrmlscript를 사용하여 switch문을 사용해서 여러 노드를 링크를 걸어 사용자 재생기의 화면에서 효율적으로 노드에 접근 할 수 있도록 할 수 있다.

아래 그림 5는 노드의 인터넷 링크와 vrmlscript를 사용하여 switch문을 사용한 예이다. switch문은 switch그룹 내의 정의된 각 노드를 순서대로 번호가 지정되어 그 순서에 따라서 vrmlscript에서 번호로 대응되어 if, else문으로서 분기하여 선택되어 진다. ROUTE를 사용하여 앞에서 지정된 touchTime과 switchTime, MyScript를 지정하여 실행 할 수 있다.

```
Anchor {
  url "http://www.terious.co.kr/"
  children [
  Shape {
    appearance Appearance {
      material Material {
        texture ImageTexture (url "ahnp.jpg") )
        geometry IndexedFaceSet {
          coord Coordinate (point [-1 -2 0, 1 -2
          coordIndex [0, 1, 2, 3]
        })
      }
    }
  }
}

Anchor {
  url "http://kr.sports.yahoo.com/jspark/"
  children[

DEF MyScript4 Script {
  eventIn SFTime switchTime
  eventOut SFInt32 whichChoice
  url "vrmlscript:
  function switchTime() {
    if (whichChoice == 0)
      whichChoice = 0;
    else whichChoice = 0; }
}

ROUTE TS4.touchTime TO MyScript4.switchTime
ROUTE MyScript4.whichChoice TO MySwitch1.whichChoice

ROUTE TS5.touchTime TO MyScript5.switchTime
ROUTE MyScript5.whichChoice TO MySwitch1.whichChoice
ROUTE TS6.touchTime TO MyScript6.switchTime
ROUTE MyScript6.whichChoice TO MySwitch1.whichChoice
```

그림 5 vrmlscript 기술

아래 그림 6은 VRML을 기반으로 만들어진 BIFS데이터를 재생기를 사용하여 재생한 화면이다. 그림 6에서 플레이어의 재생된 화면의 좌측 하단 버튼을 클릭하게 될 때 선수들의 그림이 나오며 해당 선수를 클릭하게 되면 하단에 선수들에 대한 간략한 설명이 나오며, 설명 그림을 클릭하게 되면 인터넷에 연결되어 해당선수의 홈페이지나 관련된 정보 사이트로 연결되게 된다. 보여진 메뉴들의 화면을 다시 없애려면 좌측하단 메뉴버튼을 클릭하거나 노드외의 다른 영역을 클릭하여 열려진 노드를 안 보이게 하도록 할 수 있다.

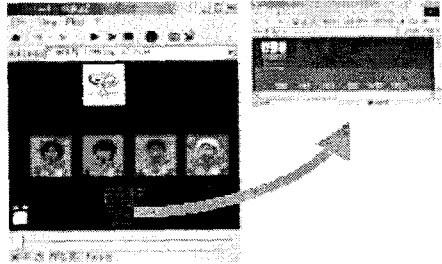


그림 6 결과 화면

### 3.3 그래픽 모듈

그래픽 라이브러리는 사용자 인터페이스 상에서 MPEG-4에 정의된 각 객체들의 재생을 위한 것으로 비디오, 오디오, 이미지, 2D, 3D 등의 다양한 미디어 객체를 데이터 액세스 APIs를 통해 호출하며, 기존의 MPEG-4 재생기에서 지원하지 못했던 MPEG-4에서 정의된 객체들의 다양한 데이터 포맷을 지원한다. 또한 이 라이브러리는 재생 시스템에서 독립적으로 관리되어질 수 있고, 필요시 호출될 수 있다.

## 4. 결론

본 논문에서는 BIFS 기반의 MPEG-4 장면 정보를 기술하는 확장VRML인 BT와 XMT를 이용하여 노드를 기술하고 해당되는 디코더로 MPEG-4 파일의 속성을 추출하여 해당 그래픽 라이브러리와 장면 정보를 화면에 재생하여 미리 기술된 사용자의 마우스 이벤트를 감지하여 원하는 아이콘 즉 노드를 클릭 이벤트가 발생하였을 때 사용자가 원하는 정보를 가진 다른 노드를 보여주거나 해당 사이트의 주소를 가진 인터넷 익스플로러로 연결하여 보다 자세한 정보를 볼 수 있는 멀티미디어 서비스를 구현한다.

### 4.1 향후 연구과제

MPEG-4 시스템의 데이터형식인 BIFS는 인터넷을 연결하게 될 때 해당 노드에 링크를 걸어 연결하게 되는데, 이러한 단방향 서비스를 네트워크상에서 BIFS 정보를 받아 화면에 반영되어 질수 있는 모듈개발이 있으며, 영상에서 움직이는 것들에 대해서 객체단위로 분리하여 그 객체에 대한 정보나 하이퍼링크를 통해 정보를 얻을 수 있는 모듈개발이 앞으로의 연구과제이다.

#### 4.2 향후 발전 방향

MPEG-4는 많은 기술적 장점에도 불구하고 아직까지 데이터 포맷은 컨텐츠로서 활용되고 있지 않고 있다. MPEG-4의 영상 포맷은 압축률이 높으나 그만큼 화질의 손실이 많아서 일반적인 큰 화면에는 화질이 떨어지겠지만 핸드폰이나 PDA와 같은 작은 액정화면에 활용되어질 경우에는 영상 압축률이 높기 때문에 용량에 대한 부담을 덜 수 있다. 또한 이러한 문제는 지상파 DMB의 영상 포맷인 H.264/AVC포맷[7]으로 활용할 수 있으나 이것을 활용하고 사용될 컨텐츠의 개발이 필요하다.

#### 참고 문헌

- [1] Iain E.G Richardson "H.264 and MPEG - 4 (차세대 영상압축기술)" pp184, 2004.
- [2] M. Kim, S. Wood and L.T. Cheek, "Extensible MPEG-4 textual format(XMT)," in Proc. ACM multimedia workshops., Los Angeles, California, United States, 2000, pp.71-74
- [3] GPAC "gpac.sourceforge.net".
- [4] 박기현 "코덱의 세계로의 초대(이론과 실무의 조화)" pp361, 2006.
- [5] W3C recommendation(Ver1.0), Document Object Model(DOM) Level2 Core Specification, <http://www.w3.org/TR/DOM-Level-2-Core/>.
- [6] Fernando Pereira, Touradj Ebrahimi, Leonardo Chiariglione "The MPEG-4 Book : Prentice Hall Imsc Press Multimedia Series (Hardcover)" 2002.
- [7] 카도노 신야 "H.264/AVC 비디오 압축 표준" pp30, 2005.