

사용자 요구에 따른 능동적인 BIFS 시스템에 관한 연구

이상래*, 이영렬**, 박현욱***

*삼성전자

**세종대학교 컴퓨터공학부 인터넷학전공

***한국과학기술원 전자전산학과 전기및전자공학전공

e-mail : srlee73@samsung.com

A Study on the Active BIFS Generation System on User Interaction

SangRae Lee*, Yung Lyul Lee**, Hyun Wook Park**

*Samsung Electronics Co., Ltd

**Dept. of Internet Engineering, Sejong University

***Dept. of EE & CS, Korea Advanced Institute of Science and Technology

요 약

MPEG-4는 기존의 MPEG-1, MPEG-2와 달리 멀티미디어를 위한 객체 기반의 부호화에 관한 표준이다. MPEG-4의 장면은 여러 개의 객체로 이루어져 있기 때문에 각 객체들을 시간, 공간적으로 표현하기 위한 장면 기술의 방법으로 BIFS(Binary Format for Scene description)를 규정하고 있다. MPEG-4 단말기에서는 부호화된 BIFS 정보를 복호화하여 각 객체들을 화면에 구성하게 되며, 객체의 특성, 행동들을 정의하거나 변경할 수 있고 새로운 객체들을 포함하는 등의 사용자 상호 작용에 따른 기능을 수행하게 된다. 그러나, 사용자 상호 작용에 따른 장면 재구성은 초기 장면을 구성할 때 기술된 장면 정보를 벗어날 수 없는 제약을 받게 된다. 따라서 본 논문에서는 BIFS 수정 정보 발생기(BIFS modifier)를 가진 시스템 구조를 제안한다. 제안된 시스템 구조에서 BIFS 수정 발생기는 사용자의 상호 작용이 발생한 노드를 전달 받아 이에 상응하는 BIFS 정보를 기존의 BIFS 정보에서 찾아 후 이를 수정하여 BIFS 변경(BIFS update) 명령을 만들어내고 이에 따라 장면을 재구성하여 동적인 장면 구성을 가능하게 한다. 또한, 제안된 시스템 구조를 주요 응용 분야인 peer-to-peer(P2P)를 가정한 네트워크 기반의 게임에 직접 적용하여 보았다.

1. 서론

최근의 디지털 멀티미디어 응용 제품들은 기존의 MPEG-1, MPEG-2에서 요구되는 저장 매체나 전송을 위한 필수 조건인 높은 압축뿐만 아니라 여러 개의 객체를 직접 편집하거나 상호 작용을 할 수 있어 사용자가 요구하는 유연성 있는 장면 구성이 가능한 고급 기능을 필요로 하고 있다. 이런 필요 조건을 충족시키기 위해서 ISO/IEC MPEG(Moving Picture Experts Group)에서 1998년 10월 MPEG-4 표준을 발표하였고 1999년 말 새로운 도구가 추가된 MPEG-4 버전 2가

완성되었다.^[1]

MPEG-4는 객체 기반의 멀티미디어 부호화 표준으로 멀티미디어 응용 프로그램에서 요구되는 기능들을 제공한다. MPEG-4 객체는 영상, 음성 뿐만 아니라 합성 영상, 그래픽 등을 포함하고 있으며 사용자에게 보이는 장면 내 편집이 가능하다. MPEG-4 장면은 여러 객체로 이루어지며 이들간의 동기화를 위한 도구들이 필요하다. 또, 장면을 구성하고 합성하여 사용자의 단말에서 보여주기 위해서는 장면을 구성하는 객체들의 시공간적 위치 및 속성에 대한 기술 정보가 필요하게 된다. 이 정보를 장면 기술 정보라 한다.^{[1][3]}

MPEG-4는 BIFS(Binary for Scene Description)를 장면 기술 정보로 사용한다.^[3] BIFS는 필요한 정보의 양을 줄이기 위해서 이진화된 형식으로 영상, 음성 객체를 하나로 묶거나 객체의 시공간적 위치 및 속성 지정, 객체의 행동을 변경시키는 등의 기능 및 사용자의 상호작용을 지원한다. 하지만, 콘텐츠 제공자에 의해 제공된 초기 장면 정보에 정의된 상호 작용 외에 객체를 다루는 것은 불가능하다는 한계가 있다. 따라서, 사용자에 의한 능동적인 상호 작용을 가능하게 하기 위해서 MPEG-4 시스템 구조를 변경할 필요가 있다. 본 논문은 사용자의 능동적인 상호 작용이 가능하도록 수정된 MPEG-4 시스템 구조를 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 MPEG-4 시스템 표준을 분석하며, 3장에서는 사용자 상호 관점에서 MPEG-4 시스템 표준이 가지고 있는 한계를 분석하여 이를 개선한 MPEG-4 시스템 구성을 제안한다. 4장에서는 제안한 시스템 구조를 이용하여 응용 프로그램을 실험한 결과를 보이고 5장에서 본 논문의 결론을 맺고자 한다.

2. MPEG-4 System 표준

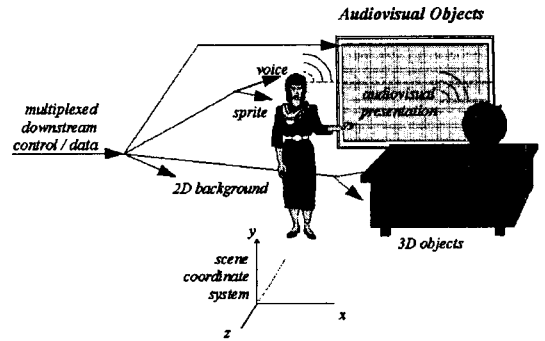
구형(rectangle) 영상만을 다루는 MPEG-1, MPEG-2와 달리 MPEG-4는 객체 기반의 부호화 방식이다.^{[1][2]} 즉, 장면 내의 영상, 음성 내용은 독립적인 객체(Object)로 인식되며 부호화된다. 객체로 불리는 것들은 임의 형상의 자연 영상, 컴퓨터 그래픽, 자연적인 또는 합성의 음성, 얼굴 및 몸의 애니메이션을 포함한다. MPEG-4에서 완전한 장면 구성을 위해서 다음과 같은 부가 정보를 필요하게 된다.

- 각 멀티미디어 객체를 시·공간 상에 어디에 위치하는가에 대한 정보
- 각 객체와 사용자 간의 상호 작용에 대한 정의
- 다양한 네트워크를 통한 전송을 위해서 모든 정보를 하나 이상의 스트림으로 다중화하는 방법
- 스트림간의 동기화를 위한 방법 등에 대한 정보

MPEG-4 시스템은 부호화하여 전송해야 할 3차원 공간 정보를 공간 내에 위치하는 사람과 건물 등의 각 객체별로 별도로 부호화함으로써 부호화 효율을 높이고 동시에 각 객체의 가공 및 편집을 가능하게 한다. MPEG-4에서 장면 구성을 위해 필요한 부가 정보를 BIFS(Binary Format for Scene)라 한다. <그림 1>은 MPEG-4의 한 장면을 보여주고 있다.

2.1 BIFS(Binary Format for Scene) 기술

MPEG-4에서는 구형 영상 뿐만 아니라 임의 형상의 객체를 부호화할 수 있다. 그리고, 사용자에게 제시되는 화면은 MPEG-2까지의 하나의 비디오(와 오디오)로 구성되는 장면에서 복수의 객체를 합성한 장면으로 변한다. 이 때문에 MPEG-4 시스템은 객체의 표시 방법과 특성을 지정하기 위한 장면 기술 언어로서 BIFS를 정의하고 있다. MPEG-4에서 사용자에게 보여지는 장면은 다양한 노드(node)들의 집합인 계층적 트리 구조로 이루어진다.



<그림 1> MPEG-4 장면 예

BIFS는 VRML을 바탕으로 하고 있다.^{[5][6]} VRML은 웹 상에서 3D 공간을 기술하기 위한 언어이며 좌표, 형상, 재질 등의 노드와 센서 노드, 루트(Route) 구조를 사용하여 사용자 상호작용으로 장면의 변경을 가능하게 한다. 그러나 VRML은 사용자가 미리 콘텐츠를 다운로드해야 하기 때문에 실시간으로 비디오 및 오디오를 취급하는 것이 불가능하다.

반면, BIFS를 이용할 경우 실시간 전송 및 구성이 가능하다는 것이다. 또, BIFS 정보가 기존의 멀티미디어 객체 정보와 다른 스트림으로 전송되기 때문에 사용자가 쉽게 객체를 옮기거나 특성을 바꾸는 등의 상호 작용을 할 수 있다.

MPEG-4는 BIFS 정보를 시간에 따라 전송 받아 차례로 구성하는 방식을 택한다. 초기 장면 정보를 전송 받아 장면을 구성하고 장면 내의 변화는 이후에 BIFS-Command 또는 BIFS-Anim 프로토콜 형태로 전송되는 정보를 이용하여 장면을 변경하는 방식이다.

2.2 MPEG-4에서 사용자 상호 작용과 한계

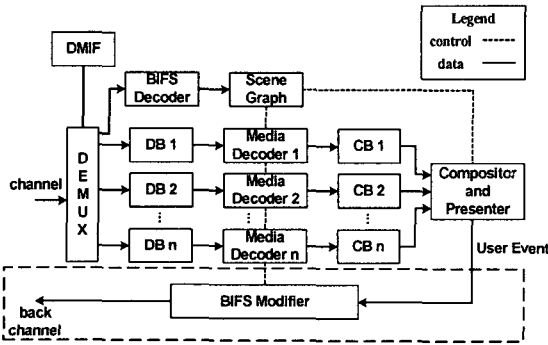
단말기에서 그려진 영상/음성 정보를 사용자가 상호 작용을 하기 위해서 MPEG-4는 사용자와 객체 간의 상호 작용을 지원한다. 이 상호 작용은 클라이언트 측면의 상호 작용과 서버 측면의 상호 작용으로 나눌 수 있다.

클라이언트 측면의 상호 작용은 주어진 콘텐츠를 사용자가 다루는 것을 포함한다. 이 상호 작용은 클라이언트 단말기 내에서 이루어지는 것이며 객체의 위치나 은닉, 글자 크기를 바꾸는 등 장면 기술 노드의 속성을 변화시키는 동작을 마우스 클릭이나 키보드 입력 이벤트를 통해서 할 수 있다. 클라이언트 측면에서 상호 작용은 장면 기술 정보에서 문법으로 정의되고 표준화되어야 한다. 따라서, MPEG-4의 상호 작용 메커니즘은 이벤트 소스, 목표, Route, 이벤트를 발생시킬 수 있는 특별한 노드인 센서의 연결된 형태로 이루어져 있다. 이벤트 소스와 목표는 장면 정보 노드의 일부분이며 특정 장면에서 연결된 형태이다. 이런 상호 작용적인 환경은 업스트림(upstream) 채널을 필요로 하지 않는다. 서버 측면의 상호 작용은 사용자의 작용에 의해서 일어나며 전송 측에서 일어나는 콘텐츠

의 조작을 포함한다. 이런 상호작용은 역방향 채널을 필요로 한다.

현재 MPEG-4는 서버 측면의 상호작용을 구체적으로 규정하지 않고 있다. 따라서, 사용자는 장면을 만든 저자의 설계에 따른 장면만을 볼 수 있으며 클라이언트 측면의 상호작용만 할 수 있을 뿐이다. 따라서 MPEG-4는 초기 장면 기술 정보에서 보이지 않은 보다 동적인 상호작용을 취급할 수 없다.

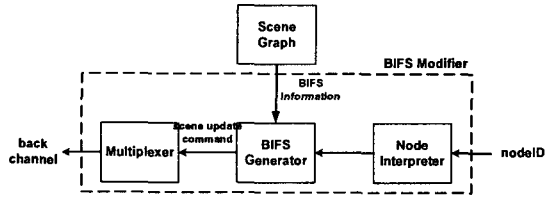
3. 제안한 시스템 구조



<그림 2> 제안한 시스템 구조

<그림 2>는 위에서 언급한 MPEG-4 사용자 상호작용이 가지는 제한을 개선하기 위해 제안한 전체 시스템 구조이다. 제안한 시스템 구조는 이벤트를 발생하는 센서노드와 목표 노드로의 이벤트 전달을 위한 ROUTE의 이벤트 구조를 사용한다. 제안한 시스템 구조는 사용자의 상호작용에 보다 능동적으로 변화할 수 있는 상호작용을 제공하며 초기 장면 정보의 양을 줄일 수 있다. 그림의 상단 부분은 기존의 MPEG-4의 시스템 구조이며 하단 부분은 능동적인 상호작용을 지원하기 위한 BIFS 수정기(BIFS Modifier)이다.

제안한 시스템의 동작 과정<그림 2>을 살펴보면 서버로부터 전송된 MPEG-4 스트림은 DAI(DMIF Application Interface)를 통해서 얻으며, DMIF^[7]에서 관리하고 있는 역다중화기(DeMux)에서 역다중화 된다. 역다중화된 스트림 중 BIFS 스트림은 BIFS 복호기에 전해지며 각각의 영상, 음성 미디어 스트림은 해당하는 복호기의 입력 버퍼(DB:Decoding Buffer)에 전해진 후 정해진 시간에 각 미디어 복호기에 의하여 복호된다. 복호된 결과는 합성 버퍼(CB:Composition Buffer)에 쌓이며 정해진 시간에 합성기에 전해진 후 합성되어 화면에 보이게 된다^[1]. 초기 장면 정보가 합성되어 보여진 후 제안한 시스템은 사용자 이벤트를 기다리게 된다. 사용자 이벤트가 발생하면 이벤트가 발생한 노드 번호가 BIFS 수정기에 입력된다. 노드 번호는 초기 장면 정보에서 DEF를 사용하여 부호화된 정수 값이며 미리 정해진다. 따라서, 센서 노드는 마우스 클릭 등 사용자 이벤트가 발생한 노드를 찾아 해당하는 ROUTE를 통해 BIFS 수정기에 노드 번호를 전송하게 된다. <그림 3>은 BIFS 수정기를 보이고 있다.



<그림 3> BIFS 수정기(BIFS modifier)

3.1. 노드 해석기(Node Interpreter)

초기 장면 기술 정보는 사용자의 단말에서 장면 갱신을 처리하기 위해서 서버와 클라이언트 사이에 미리 정해진 nodeID를 가지고 있다. 노드 해석기는 센서 노드에 의해 전달된 nodeID를 해석하여 미리 정해진 노드와 동일한지 검사를 한다. 만약 주어진 nodeID가 정해진 노드라면 노드 해석기는 BIFS 발생자에게 장면 갱신 정보를 보낼 것을 요청한다.

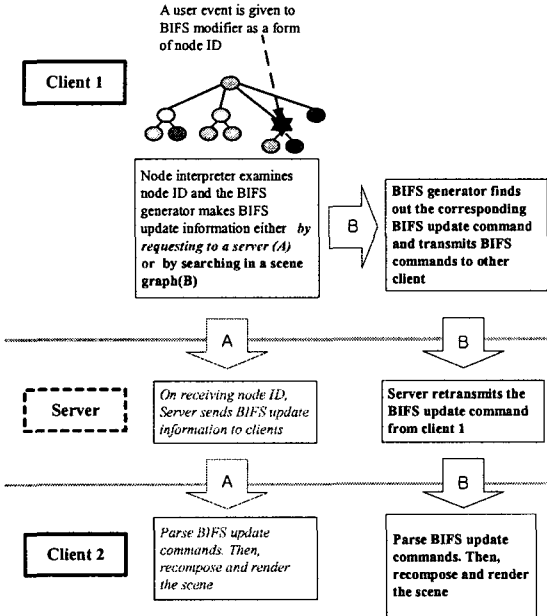
3.2. BIFS 발생기(BIFS Generator)

장면 기술 정보를 갱신하기 위해서 BIFS 발생기는 노드 해석기로부터 nodeID를 받아서 장면 트리 구조에서 해당하는 장면 기술 정보를 찾게 된다. 만약 해당되는 장면 기술 정보가 존재하면, BIFS 발생기는 장면 기술 정보를 수정해서 BIFS 갱신 명령을 만든다. 만약 해당되는 장면 기술 정보가 존재하지 않으면, BIFS 발생기는 서버에 원하는 장면 기술 정보를 요청하게 된다. 이 경우에 서버가 장면 기술 정보 요청을 처리할 수 있어야 한다. 장면 기술 정보는 BIFS 갱신 명령 내에 이진화되어 있고 갱신 명령은 삽입, 제거, 장면 내 노드 필드의 수정 등 세 가지로 분류할 수 있다. BIFS 갱신 명령은 다중화되어 서버에 전송된다.

4. 실험 결과

목표 어플리케이션의 요구 조건을 맞추기 위해서 시스템 프로파일 중 심플 2D 프로파일을 가정하였다^[4]. <그림 4>에 제안한 BIFS 수정기를 포함한 MPEG-4 시스템을 이용한 2개의 클라이언트간의 오델로 게임의 동작 과정을 보였다. <그림 4>에서 사용자가 마우스 클릭과 같은 이벤트를 발생시키면(예로서 node insert) 이벤트가 발생한 노드를 찾아 node ID를 얻고 node ID를 BIFS 수정기(BIFS modifier)에 전달한다. 노드 해석기(Node interpreter)는 node ID를 검사하여 BIFS 발생기(BIFS generator)에 전달한다. BIFS 발생기는 전체 장면 구조를 조사하여 BIFS 갱신 정보를 가지고 있는지 검색하여 만약 전체 장면 구조가 BIFS 갱신 정보를 가지고 있지 않으면 A로 진행하고, 만약 전체 장면 구조가 BIFS 갱신 정보를 가지고 있으면 B로 진행한다. A인 경우, BIFS 갱신 정보가 없으므로 서버에 BIFS 갱신 정보를 전송해줄 것을 요청한다. 이 경우 서버는 클라이언트 1으로부터 들어오는 요청(node ID)을 받아 BIFS 갱신 정보를 만든 후 클라이언트 2에 전송한다. B의 경우, BIFS 발생기는 장면 구조에서 찾은 BIFS 갱신 정보를 주어진 node ID를 이용

해서 수정하여 서버로 전송한다. 서버는 클라이언트 1 으로부터 전송된 BIFS를 클라이언트 2로 재전송해준다. 결국 클라이언트 2는 서버로부터 전송된 갱신 명령을 해석하여 장면을 갱신한다.



<그림 4> 제안한 시스템 구조의 동작 과정

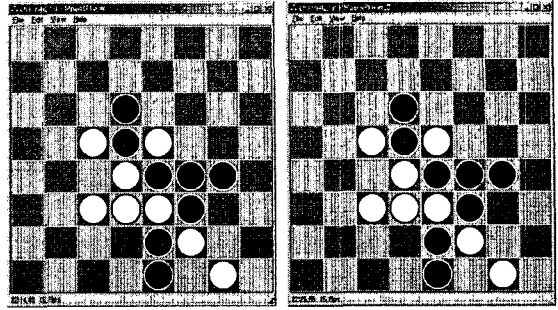
BIFS 정보의 크기를 비교하기 위해서 네트워크 게임을 기존의 MPEG-4 시스템 구조와 제안한 시스템 구조로 만들어 보았다. 기존의 MPEG-4 시스템 구조로 만든 초기 장면 정보는 모든 이벤트 정보와 이를 다루기 위한 노드들의 정보를 모두 담고 있다. 이런 방식에는 두 가지 단점이 존재한다. 첫째, 사용자 상호작용은 장면을 생성한 저자의 설계에 따라 제한을 받는다. 즉, 사용자는 장면 내 객체를 능동적으로 조작할 수 없다. 두 번째로 초기 장면 기술 정보가 BIFS 이벤트 모델을 사용해서 상대방의 장면 기술 정보를 그려야 하기 때문에 크기가 커진다.

반면, 제안한 시스템 구조인 경우 어떤 노드가 사용자 이벤트를 받을 것인가를 찾기 위한 센서 노드만이 초기 장면 정보에 포함된다. 따라서 두 시스템 구조에서 생성한 동일 장면에 대한 장면 기술 정보의 크기를 비교해 볼 때 필요한 BIFS 정보가 상당히 줄어들었음을 볼 수 있다.<표 1> 따라서, 저대역폭 네트워크 환경에서 보다 효율적으로 적용할 수 있을 것이다.

<그림 5>는 제안한 시스템 구조를 이용한 네트워크 게임을 실행한 결과이다. 마우스 클릭 이벤트를 통한 장면 갱신 명령을 양쪽에 전달되었음을 볼 수 있다.

	기존 시스템	제안한 시스템
Text	63.9 kbytes	38.7 kbytes
BIFS(*.bif)	5.5 kbytes	3.6 kbytes

<표 1> 기존 시스템과 제안한 시스템의 결과 비교



<그림 5> 응용 프로그램 실행 예

5. 결론

기존의 멀티미디어 전송을 위한 MPEG-1, MPEG-2 등은 2D, 3D 영상과 음성 등을 모두 포함하는 미디어 응용프로그램을 만들기 어렵다. 그러나, MPEG-4는 영상, 음성 등의 미디어를 생산, 전송, 사용자의 사용에 있어서 풍부한 기능을 제공한다. 특히, 객체 기반의 부호화 방식과 장면 기술 정보(BIFS)는 MPEG-4 장면 내의 미디어 콘텐츠를 사용자가 직접 다룰 수 있게 해 준다. 하지만, 현재의 MPEG-4 표준은 사용자 상호 작용 측면에서는 제한된 기능만을 제공하고 있다. 사용자는 콘텐츠 제공자의 의도에 따른 장면만을 보거나 상호 작용을 통해 다룰 수 있다.

본 논문에서는 보다 능동적인 상호작용을 사용자에게 제공하기 위한 MPEG-4 시스템을 제안하였다. 제안한 시스템 구조는 BIFS 수정기를 포함하고 있어 사용자 이벤트가 발생한 노드를 확인하고 원하는 BIFS 정보를 장면 구조에서 찾게 된다. BIFS 발생기는 장면 정보를 수정해서 BIFS 장면 갱신 명령을 만들어 역방향 채널로 보내게 된다. 제안한 시스템 구조를 네트워크 기반의 게임 어플리케이션에 응용하여 사용자가 요구하는 능동적인 상호작용이 가능하였다.

참고문헌

[1] ISO/IEC JTC1/SC29/WG11 N3850, "Information Technology - Coding of Audio-Visual Objects - Part 1: Systems," Oct. 2000.

[2] ISO/IEC JTC1/SC29/WG11 N3056, "Information Technology - Coding of Audio-Visual Objects - Part2: Visual Amendment 1: Visual Extensions," January. 2000.

[3] Julien Signes, Yuval Fisher and Alexandros Eleftheriadis, "MPEG-4's binary format for scene description," Signal Processing: Image Communication, vol. 15 (2000), pp. 321-345, 2000.

[4] R. Koenen, "Profiles and levels in MPEG-4: Approach and Overview," Signal Processing: Image Communication, vol. 15, pp. 463-478, 2000.

[5] ISO/IEC 14772-1, "The Virtual Reality Modeling Language, 1997," <http://www.vrml.org/Specifications/VRML97>.

[6] A. L. Ames, D. R. Nadeau, J. L. Moreland, 'The VRML 2.0 sourcebook', 2nd ed., John Wiley & Sons, Inc., 1997.

[7] ISO/IEC, ISO/IEC International Standard 14496-6, MPEG-4 Standard, Spring 1999.