

지상파 DMB 대화형 서비스를 위한 MPEG-4 바이너리 컨버터 설계 및 구현

이인재*, 김기완**, 유원혁*, 김중규**, 김규현*

*한국전자통신연구원 방송미디어연구그룹

**성균관대학교 정보통신공학부

ninia@etri.re.kr

Design and Implementation of MPEG-4 binary Converter for Interactive T-DMB Service

Injae Lee*, Kiwan Kim**, Wonhyuk Yu*, Joongkyu Kim** and Kyuheon Kim*

*Broadcasting Media Technology Department, ETRI

**School of Information and Communication Engineering, SKKU

요약

방송과 통신의 융합, 방송서비스의 다양화와 더불어 방송의 디지털화는 기존의 아날로그 방식에 비해 고화질 및 고음질의 방송 서비스 제공 및 타 매체와의 호환성 그리고 다양한 부가 서비스의 제공 등에 있어서 많은 이점을 제공할 수 있다. 또한, 디지털방송 방식은 기존 아날로그 방식의 이동수신 문제점을 해결하고자 디지털 오디오 방송(DAB: Digital Audio Broadcasting)에 멀티미디어 서비스를 제공하는 디지털 멀티미디어 방송(DMB: Digital Multimedia Broadcasting)을 제안하였다. 지상파 DMB 방송은 유럽의 Eureka-147 방식의 DAB를 기반으로 하며 디지털 라디오방송과 함께 오디오 외에도 다양한 멀티미디어를 전송하고, 콘텐츠에 대화형 기능을 포함하는 서비스 형태를 목표로 하고 있다. 현재 지상파 DMB 서비스에서는 기본적인 AV 서비스를 위해 비디오는 AVC, 오디오는 BSAC에 대해 규정하고 있다. 또한 AV 규격 외에 대화형 서비스를 제공하기 위한 보조 데이터로서 장면기술 데이터와 그래픽 데이터를 지원하고 있다. 따라서 본 논문에서는 지상파 DMB를 이용한 대화형 서비스를 제공하기 위해 요구되는 초기 객체 정보, 객체 정보, 장면 구성 정보를 바이너리 형태로 생성해 주는 MPEG-4 바이너리 컨버터를 설계하고 구현하였다.

1. 서론

최근 방송환경의 급격한 변화와 더불어 휴대폰, PDA, 노트북 컴퓨터 등 휴대용 기기가 일반화됨에 따라 이동 중에도 선명한 화질의 TV를 즐기고자 하는 시청자의 요구가 급격히 증가하고 있다. 하지만 기존 지상파 이동 아날로그 TV의 경우, 이동 중 화면 떨림, 잡음 등이 발생함으로써 고선명, 고음질의 수신 상태를 제공하지 못하고 있다. 또한 지상파 디지털 TV의 경우, 일반 가정의 시청자에게는 고화질의 선명한 TV 서비스를 제공할 수 있으나, 이동환경에서는 전달하여야 할 정보량에 비해 사용 가능한 대역폭이 부족하여 만족할 만한 이동 수신 성능을 얻기는 어렵다. 따라서 국내에서는 이동 시 지상파 A/V를 시청하기 위한 대안으로 지상파 DMB가 제안되었다.

지상파 DMB는 현재 아날로그 TV 방송에 이용되고 있는 초단파 대역을 통하여 이동 중에도 선명한 화질을 수신할 수 있는 일종의 디지털 TV 서비스로 규정할 수 있

으며, 지상파 DMB 서비스는 고품질의 오디오, 영상, 데이터 서비스를 언제 어디서나 고정 및 이동, 휴대 단말로 받아 볼 수 있는 이동멀티미디어 방송 서비스라 할 수 있다. 지상파 DMB 방송은 유럽의 Eureka-147 방식의 DAB(Digital Audio Broadcasting)를 기반으로 하며 디지털 라디오방송과 함께 오디오 외에도 다양한 멀티미디어를 전송하고, 콘텐츠에 대화형 기능을 포함하는 서비스 형태를 목표로 하고 있다.

현재 국내 지상파 DMB 비디오 서비스를 위한 비디오 코덱은 MPEG-4 part 10 (AVC: Advanced Video Coding)이 선정되었으며[1], 오디오 코덱으로는 MPEG-4의 BSAC(Bit Sliced Arithmetic Coding)이 선정되었다[2]. 또한 AV 규격 외에 대화형 서비스를 제공하기 위한 보조 데이터로 장면기술 데이터와 그래픽 데이터에 대해 규정하고 있다. 따라서 본 논문에서는 지상파 DMB를 이용한 대화형 서비스를 제공하기 위해 초기 객체 정보(IOC: Initial Object Descriptor), 객체 정보(OD: Object Descriptor), 장면 구성 정보(SD: Scene Description) 등

의 보조 데이터들을 바이너리 형태로 생성해 주고[3], 생성된 IOD, OD, BIFS(Binary Format for Scene) 및 미디어 스트림들을 하나의 MP4 형태로 생성시켜 주는[4] MPEG-4 바이너리 컨버터를 설계하고 구현하였다.

본 논문의 2장에서는 지상파 DMB 서비스의 비디오 코덱과 오디오 코덱 및 MPEG-4 시스템을 따르는 보조 데이터 기술 방법에 대해 설명하였으며, AVC를 이용한 MP4 file 생성 방법에 대해 기술하였다[5]. 3장에서는 상기 보조 데이터들을 생성하기 위한 MPEG-4 바이너리 컨버터의 구조에 대해 제안하였다. 4장에서는 본 논문의 결론을 맺는다.

2. 지상파 DMB 데이터 생성을 위한 바이너리 변환 기술

2.1. AVC

지상파 이동멀티미디어 방송에 채택된 비디오 부호화 규격은 AVC(Advanced Video Coding)로, 정식 명칭은 ITU-T Rec. H.264 | MPEG-4(ISO/IEC 14496) Part 10 AVC 이다. AVC가 기존 비디오 부호화 방식과 크게 다른 점은 2계층 구조를 가진다는 것이다. 즉, 디지털 통신망과 다양한 전송 프로토콜이 존재하는 전송 환경에 잘 적용될 수 있게 하기 위해 AVC는 동영상 스트림을 크게 VCL(Video Coding Layer)와 NAL(Network Abstraction Layer)로 계층화한다. 여기서 VCL는 비디오 데이터의 압축을 담당하고, NAL은 압축된 데이터를 전송망에 적합한 형태로 변형하는 기능을 담당한다. JVT에 의한 동영상 스트림의 정보는 NAL 유닛에 담겨 전송된다. NAL 유닛은 크게 세 종류의 데이터를 포함할 수 있는데, 이는 다음과 같다.

- 슬라이스 데이터 : 독립적으로 복호화가 가능한 슬라이스 데이터
- 파라미터셋 : 복호기의 설정 및 제어와 관련된 정보로 주로 Sequence와 Picture 헤더정보
- SEI(Supplemental Enhancement Information) : 복호된 데이터의 화면 표현과 관련된 시간 정보 및 부가 정보

여기서 슬라이스 데이터와 파라미터셋은 VCL에서 소비되어 영상을 복호하는데 활용되고, SEI 데이터는 대부분 복호된 영상을 활용하는 application 계층에서 소비되거나 극히 일부분의 데이터는 복호 과정에 소비될 수 있도록 VCL 계층에 전달된다.

AVC 관련 ES(Elementary Stream)는 그림 1(a)와 같이 AVC 비디오 스트림만이 존재하도록 1개의 ES로 구성하거나, 그림 1(b)와 같이 순수한 AVC 비디오 스트림과 파라미터셋 스트림 2개의 ES로 구성할 수 있다.

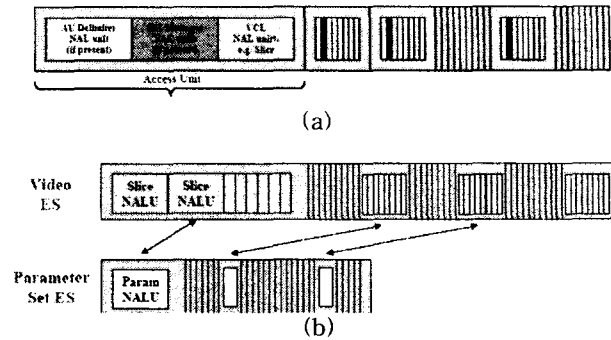


그림 1. AVC ES의 구조

이때 그림 1(a)와 같이 AVC 비디오 스트림만 존재할 경우 파라미터셋 정보는 별도의 ES로 가지 않고 OD 안에 기술되어 전송될 수 있다. 이때, OD에는 decoderSpecificInfo에 다음과 같이 AVC ES에 대한 프로파일 및 파라미터셋 정보 등을 담고 있는 AVCDecoderConfigurationRecord를 기술해 준다.

```
aligned(8) class AVCDecoderConfigurationRecord {
    unsigned int(8) configurationVersion = 1;
    unsigned int(8) AVCProfileIndication;
    unsigned int(8) profile_compatibility;
    unsigned int(8) AVCLevelIndication;
    bit(6) reserved = '111111'b;
    unsigned int(2) lengthSizeMinusOne;
    bit(3) reserved = '111'b;
    unsigned int(5) numOfSequenceParameterSets;
    for (i=0; i< numOfSequenceParameterSets; i++) {
        unsigned int(16) sequenceParameterSetLength;
        bit(8*sequenceParameterSetLength)
            sequenceParameterSetNALUnit;
    }
    unsigned int(8) numOfPictureParameterSets;
    for (i=0; i< numOfPictureParameterSets; i++) {
        unsigned int(16) pictureParameterSetLength;
        bit(8*pictureParameterSetLength)
            pictureParameterSetNALUnit;
    }
}
```

그밖에 AVC 비디오 스트림을 사용하기 위해서는 streamType은 0x04(Visual Stream)로 설정해 주고 objectTypeIndication은 해당 ES가 AVC 비디오 스트림일 경우 0x21로, 해당 ES가 파라미터셋 스트림일 경우 0x22로 처리해 준다.

2.2. BSAC

아날로그 FM 라디오 방송을 대체할 지상파 DMB 방송에서 오디오 신호에 대한 최소한의 요구사항은 현재의 아날로그 FM 방송보다는 우수하고, CD와 유사한 음질을

제공할 수 있어야 한다는 것이다. 또한 비디오 신호와 같이 전송되는 환경에서 최소한의 대역폭을 같은 부호화 알고리즘을 선택하여야 한다. 현재 국제 표준으로 사용되는 오디오 압축 알고리즘 가운데 상기의 요구사항에 부합되는 것은 MPEG-4 AAC, BSAC, TwinVQ, aacPlus 등이 있다. 따라서 국내 지상파 표준에서는 적절한 음질을 제공하면서도 상대적으로 유리한 로열티 정책을 가지고 있는 MPEG-4 BSAC (Bit Sliced Arithmetic Coding) 을 지상파 DMB 서비스를 위한 오디오 코덱으로 채택하였다.

MPEG-4 BSAC은 1999년도에 제정된 MPEG-4 오디오 표준 기술로서, ISO/IEC 14496-3 규격 중 ObjectID 22인

Syntax	No. of bits	Mnemonic
AudioSpecificConfig ()		
{		
audioObjectType;	5	bslbf
samplingFrequencyIndex;	4	bslbf
if (samplingFrequencyIndex==0xf)		
samplingFrequency;	24	ulmsb1f
channelConfiguration;	4	bslbf
if (audioObjectType == 1 audioObjectType == 2		
audioObjectType == 3 audioObjectType == 4		
audioObjectType == 6 audioObjectType == 7)		
GASpecificConfig();		
if (audioObjectType == 8)		
CelpSpecificConfig();		
if (audioObjectType == 9)		
HvxcSpecificConfig();		
if (audioObjectType == 12)		
TTSSpecificConfig();		
if (audioObjectType == 13 audioObjectType == 14		
audioObjectType == 15 audioObjectType == 16)		
StructuredAudioSpecificConfig();		
/* the following Objects are Amendment 1 Objects */		
if (audioObjectType == 17 audioObjectType == 19		
audioObjectType == 20 audioObjectType == 21		
audioObjectType == 22 audioObjectType == 23)		
GASpecificConfig();		
if (audioObjectType == 24)		
ErrorResilientCelpSpecificConfig();		
if (audioObjectType == 25)		
ErrorResilientHvxcSpecificConfig();		
if (audioObjectType == 26 audioObjectType == 27)		
ParametricSpecificConfig();		
if (audioObjectType == 17 audioObjectType == 19		
audioObjectType == 20 audioObjectType == 21		
audioObjectType == 22 audioObjectType == 23		
audioObjectType == 24 audioObjectType == 25		
audioObjectType == 26 audioObjectType == 27) {		
epConfig;	2	bslbf
if (epConfig == 2 epConfig == 3) {		
ErrorProtectionSpecificConfig();		
}		
if (epConfig == 3) {		
directMapping;	1	bslbf
if (! directMapping) {		
/* tbd */		
}		
}		
}		
}		

Syntax	No. of bits	Mnemonic
GASpecificConfig (samplingFrequencyIndex,		
channelConfiguration,		
audioObjectType)		
{		
frameLengthFlag;	1	bslbf
dependsOnCoreCoder;	1	bslbf
if (dependsOnCoreCoder) {		
coreCoderDelay;	14	ulmsb1f
}		
extensionFlag;	1	bslbf
if (! channelConfiguration) {		
program_config_element 0;		
} if ((audioObjectType == 6) (audioObjectType == 20)) {		
layerNr;	3	ulmsb1f
} if (extensionFlag) {		
if (audioObjectType == 22) {		
numOfSubFrame;	5	bslbf
layer_length;	11	bslbf
} if (audioObjectType == 17 audioObjectType == 18		
audioObjectType == 19 audioObjectType == 20		
audioObjectType == 21 audioObjectType == 23) {		
aacSectionDataResilienceFlag;	1	bslbf
aacScalefactorDataResilienceFlag;	1	bslbf
aacSpectralDataResilienceFlag;	1	bslbf
}		
extensionFlag3;	1	bslbf
if (extensionFlag3) {		
/* tbd in version 3 */		
}		
}		

그림 2. BSAC 관련 AudioSpecificConfig 기술자의 syntax

ER-BSAC 규격에 다음과 같은 제한사항을 준 기술이다.

- AudioSpecificConfig()에서
 - EpConfig: 0
 - FrameLengthFlag: 0
 - DependsOnCoreCoder: 0
- Bsc_header()에서
 - SBA_mode: 0
- General_header()에서
 - LTP_data_present: 0

MPEG-4 BSAC ES를 처리하기 위해서는 OD의 decoderSpecificInfo에 AudioSpecificConfig를 이용하여 BSAC ES에 대한 정보를 기술해 준다. MPEG-4 오디오는 objectTypeIndication을 0x40으로 사용한다. 일례로 AAC와 BSAC은 둘다 MPEG-4 오디오이므로 동일한 objectTypeIndication값을 가지게 된다. 따라서 AudioSpecificConfig의 audioObjectType을 통해 해당 object가 AAC인지 BSAC인지 구분할 수 있다.

BSAC 오디오 스트림에 대해 기술하기 위해서는 streamType은 0x05(Audio Stream)로 설정해 주고 objectTypeIndication은 0x40으로, audioObjectType은 0x16(22)으로 처리해 준다.

2.3. File Format

지상파 DMB 대화형 서비스를 지원하기 위해서는 기본적으로 ISO/IEC MPEG-4 Systems의 Core2D@Level1을 따르는 장면 구성 정보가 필요하고, 그밖에 대화형 서비스를 위한 보조데이터로서 초기 객체 정보(IOD), 객체 정보(OD) 스트림도 필요하다. 이때 생성되는 OD 스트림은 2.1절과 2.2절에 기술된 것처럼 단말에서 AVC와 BSAC을 디코딩하기 위해 필요한 정보들도 포함하고 있다.

또한 대화형 서비스를 위한 지상파 DMB 데이터로서 AVC, BSAC 외에 초기 객체 정보(IOD), 객체 정보(OD),

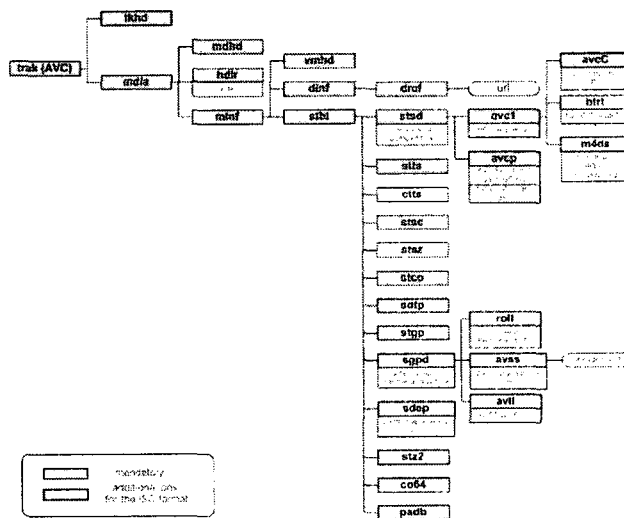


그림 3. AVC track의 구성도

장면 구성 정보(BIFS) 스트림 및 부가 미디어 스트림 (JPEG, etc) 등 다양한 다수의 ES들이 존재할 수 있으며, 경우에 따라 이러한 DMB 데이터들을 ES별로 관리하는 것은 번거로울 수 있다. 따라서 지상파 DMB 데이터의 보다 효율적인 관리를 위해 IOD, OD, BIFS 및 미디어 스트림들을 하나의 MP4 형태로 생성시켜줄 수 있다. 이때 생성되는 MP4 파일은 ISO/IEC 14496-12에 규정된 ISO base media file format과 14496-15에 규정된 AVC file format을 기반으로 하며, 이때 AVC를 위한 track의 구조는 그림 3과 같다.

AVC를 위한 track을 구성할 때 avcC, avcP box에 AVC ES 혹은 파라미터셋 ES와 관련된 OD의 AVCDecoderConfigurationRecord 정보가 저장된다. 그리고 sdtg, stgp, sdep, sgpd, roll, acss, avll는 AVC 비디오에서 layer, sub-sequence 기능을 사용할 경우 활용될 수 있는 box들이다.

3. 지상파 DMB 대화형 서비스를 위한 MPEG-4 바이너리 컨버터

MPEG-4에서는 콘텐츠 저작언어로 MPEG-4의 객체 기술과 장면 기술을 콘텐츠 저작자가 잘 알아볼 수 있도록 표현하기 위한 텍스트 형식(XMT)과 스트리밍이 가능한 바이너리 형식(MP4)의 파일 포맷을 제공하고 있다. 그러므로, 콘텐츠의 효율적인 저작, 관리 및 스트리밍을 위해서는 두 포맷간의 상호 변환을 수행할 수 있는 컨버터가 필요하다.

텍스트 형식으로 저장된 MPEG-4 콘텐츠는 장면 기술과 객체 기술만을 포함하고 있고, 미디어 객체 자체는 포함하고 있지 않다. 그러므로, 텍스트 파일로부터 MP4 파일을 만들기 위해서는 먼저, 텍스트 파일에 기술되어 있는 장면 기술을 이진 형식의 BIFS 스트림으로, 객체 기술을 이진 형식의 OD 스트림으로 부호화한 뒤, 부호화된 장면 기술 스트림과 객체 기술 스트림을 부호화된 미디어 스트림과 함께 MP4 파일 규격에 규정되어 있는 box 구조에 맞도록 구성한다.

그림 4는 MPEG-4 바이너리 컨버터의 구조를 나타낸 것이다. 그림 4에서 보여지는 것처럼 MPEG-4 바이너리 컨버터는 객체 부호화기(object encoder)와 MP4 부호화기(MP4 encoder)로 구성되어 있으며, 객체 부호화기는 객체 기술자 부호화기, 장면 부호화기 및 미디어 객체 부호화기로 구성된다. 여기서 객체 기술자 부호화기와 장면 부호화기를 통해 생성되는 OD, BIFS 스트림은 그대로 지상파 DMB 데이터로서 활용될 수 있다. 그리고, 미디어 객체 부호화기는 텍스트 파일에 포함되어 있는 미디어 객체가 부호화 되지 않은 데이터인 경우 객체 기술자에 기술되어 있는 부호화에 필요한 정보를 이용하여 적절히 부호화를 행한다. 그리고, MP4 부호화기에서는 MP4 규격에 따라 각 객체 부호화기에서 부호화된 객체 스트림들은 'mdat' box에 저장하고, 각 객체 스트림에

대한 메타 데이터들을 'moov' box에 저장한다.

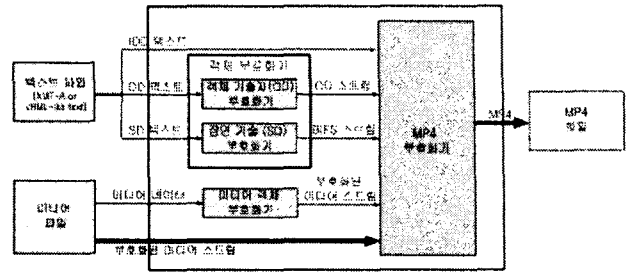


그림 4. MPEG-4 바이너리 컨버터의 구조도

4. 결론

본 논문에서는 지상파 DMB 대화형 서비스를 위한 MPEG-4 바이너리 컨버터의 개발에 대하여 기술하였다. MPEG-4 바이너리 컨버터는 MPEG-4의 저작 언어로 사용되는 텍스트 파일(XMT 혹은 vrm1-like한 text)을 MPEG-4에서 제공하는 이진 형식의 BIFS, OD 혹은 MP4로 상호 변환을 행하는 기능을 가진다. 개발한 MPEG-4 바이너리 컨버터는 본 연구팀에서 개발하고 있는 지상파 DMB 대화형 서비스를 위한 MPEG-4 Systems 기반 콘텐츠 저작 SW 시스템에 내장할 예정이며, XMT-A 혹은 vrm1-like하게 기술된 텍스트 파일을 입력으로 하여 OD, BIFS, MP4 파일로의 변환을 수행한 결과 개발된 컨버터가 효율적으로 동작함을 확인할 수 있었다.

제안한 지상파 DMB 대화형 서비스를 위한 MPEG-4 바이너리 컨버터는 지상파 DMB 기술을 이용하여 언제 어디서나 다양한 콘텐츠를 향유할 수 있는 이동 멀티미디어 방송 서비스 뿐 아니라, 방송콘텐츠의 단순 시청을 벗어나 시청자 대화성을 비롯한 다양한 형태의 부가서비스를 포함하는 대화형 멀티미디어 콘텐츠 생성을 위해 효율적으로 활용될 수 있다.

참고문헌

- [1] ISO/IEC JTC1/SC29/WG11 N5555, "MPEG-4 Part 10: Advanced Video Coding FIDS", March 2003.
- [2] ISO/IEC JTC1/SC29/WG11 N3995, 14496-3, "Information Technology - Generic Coding of Audio-Visual Objects, Part 3: Audio", March 2001.
- [3] ISO/IEC JTC1/SC29/WG11 N4848, 14496-1 : MPEG-4 systems, "Information technology, Part1: Systems, ISO/IEC 14496-1:2002", March 2002.
- [4] ISO/IEC JTC1/SC29/WG11 N5298, 14496-14 : MP4 File Format, "Text of ISO/IEC FDIS 14496-14," April 2003.
- [5] ISO/IEC JTC1/SC29/WG11 N5780, 14496-15 : AVC File Format, "ISO/IEC FDIS 14496-15," August 2003.