

## MPEG-V 표준 동향 및 리치미디어 활용 연구

임성용\* · 이인재\*\* · 차지훈\*\*\*

### 1. 서 론

최근 출시되는 고정 및 이동 콘텐츠 소비단말들은 사용자 환경에 적합한 다양한 정보 소비 욕구를 충족하기 위하여 소형 카메라를 비롯한 터치스크린, GPS센서, 가속도센서, 접근센서, 조도센서, 진동자 등의 여러 가지 사용자 인터페이스 장치를 포함하고 있다. 이러한 사용자 인터페이스 장치들을 여러 가지 어플리케이션 소프트웨어 혹은 어플리케이션 콘텐츠에서 활용하기 위하여 각 단말 제조사는 개별 오픈 인터페이스를 제공하고 있다. 예를 들어, 인터넷 콘텐츠 분야의 사실 표준단체인 W3C(World Wide Web Consortium) 에서는 기본적인 사용자 인터페이스인 키보드, 마우스를 비롯하여, 진동 제어 API, GPS 정보 구조, 터치 이벤트 구조 등을 표준화 하고 있다[1].

한편, 동영상 압축 전문가 표준 그룹인 MPEG (Motion Picture Experts Group, ISO/IEC JTC1 SC29 WG11) 에서는 동영상을 비롯한 가상공간

콘텐츠를 실제 소비 환경과 상호 연동하기 위한 표준 데이터 구조를 “Media context and control (MPEG-V)” 라는 이름으로 2011년 7월 1<sup>st</sup> edition 표준을 발행하였다[2]. MPEG-V 표준에서는 현실 세계의 물리적인 현상을 감지하는 감지 정보 구조와, 반대로 현실 세계를 제어하기 위한 제어 정보 구조, 그리고 가상 공간 사이의 데이터 교환을 위한 가상 객체 정보 구조 등을 포함하고 있다.

하지만, 계속된 표준화 회의를 통해서 1<sup>st</sup> edition MPEG-V 표준 문서에 포함되지 않은 감지 장치와 제어 장치에 대한 표준화 필요성이 제기 되었으며, MPEG-V 표준에 따르는 XML 문서의 이진화 방법 필요성이 대두 되었다. 또한 MPEG-V 표준은 시스템 플랫폼에 독립적인 데이터 구조이기 때문에 다른 표준과의 연동 방법에 대한 고찰이 필요하다. 따라서 본 논문에서는 2011년 7월 발행된 2<sup>nd</sup> edition 의 CD(Committed Draft) 문서에서 추가된 내용과 MPEG-V 표준의 이진화 방법 그리고 현재 널리 알려진 리치미디어 플랫폼과의 연동방법을 설명한다.

본고는 다음과 같이 구성된다. 2장에서는 최근 진행 중인 MPEG-V 2<sup>nd</sup> edition 의 주요 변경 사항을 설명하고, 3장에서는 MPEG-V에서 스키마 형식으로 정의된 데이터 구조의 이진화 방법을

\* 교신저자(Corresponding Author) : 임성용, 주소 : 대전광역시 유성구 가정로 218 (305-700), 전화 : 042)860-1640, E-mail : seylim@etri.re.kr

\* 한국전자통신연구원 융합미디어연구팀

\*\* 한국전자통신연구원 융합미디어연구팀  
(E-mail : ninja@etri.re.kr)

\*\*\* 한국전자통신연구원 융합미디어연구팀  
(E-mail : jihun@etri.re.kr)

※ 본 연구는 한국전자통신연구원 정부 출연금 지원 사업의 일환으로 수행되었음.



```
<complexType
name="BodyTemperatureSensorType">
  <complexContent>
    <extension base="siv:TemperatureSensorType ">
      <attribute name="value" type="float"
use="required"/>
      <attribute name="location"
type="nonNegativeInteger" use="optional "/>
    </extension>
  </complexContent>
</complexType>
```

그림 2. 체온 감지 정보 구조

```
<complexType name="ElectrographSensorType">
  <complexContent>
    <extension base="siv :SensedInfoBaseType">
      <sequence>
        <element name="WaveValue"
type="mpeg7:FloatMatrixType" minOccurs="0"/>
      </sequence>
      <attribute name="unit"
type="mpegvct:unitType"
use="optional"/>
      <attribute name="waveformLabel"
type="mpeg7:termReferenceType" use="optional"/>
      <attribute name="maxAmplitude" type=" float"
use="optional"/>
    </extension>
  </complexContent>
</complexType>
```

그림 3. 전위 신호 감지 정보 구조

구조를 정의하였다. 신체의 여러 Channel에서 다수의 Sample을 감지하여 주기적인 전위 값을 표시할 수 있다. 이와 같은 전위 신호 감지 정보를 바탕으로 신체로부터 측정할 수 있는 널리 알려진 전위 신호, 즉 EEG (electroencephalogram), ECG (electrocardiogram), EMG (electromyogram), EOG (electrooculogram), GSR (galvanic skin response) 정보 구조를 정의하였다. 또한, 그림 4와 같이 미리 알려진 패턴의 전위신호를 표현하기 위하여 각 전위 신호를 위하여 분류 기호 (CS: Classification Scheme)를 별도로 정의하였다.

### 2.3 고급 사용자 인터페이스

```
<complexType name="EEGSensorType">
  <complexContent>
    <extension base="siv:ElectrographSensorType">
      <attribute name="wavePattern"
type="mpeg7:termReferenceType" use="optional"/>
    </extension>
  </complexContent>
</complexType>
<ClassificationScheme
uri="urn:mpeg:mpeg-v:01-CI-EEG_WaveformLabelCS-NS">
  <Term termID="EEG_FP1_F7">
    <Name xml:lang="en">EEG between FP1 and F7</Name>
    <Definition xml:lang="en">Describes the waveform between FP1 and F7 </Definition>
  </Term>
```

그림 4. EEG 감지 정보 구조 및 EEG 분류 기호 예제

MPEG-V는 현실 세계의 물리적인 감지 장치들을 지원하고 있기 때문에, 이러한 감지 정보를 개념적으로 활용한 사용자 인터페이스 부분이 필요하게 되었다. 예를 들어, Multi interaction point 감지 정보와 같은 경우에는 여러 사용자 인터페이스 장치를 활용하여 사용자의 선택 정보를 표현할 수 있도록 하였으며, 사용자 시선의 위치와 방향 정보를 표현하기 위하여 기존의 위치 감지 정보와 방향 감지 정보를 활용한 Gaze tracking 감지 정보를 기술하였다. 이러한 고급 사용자 인터페이스 정보는 기존의 마우스, 키보드가 아닌 멀티 터치 인터페이스, 시선 추적 장치, 혹은 제스처 인식 장치를 활용하여 가상공간의 콘텐츠를 제어할 수 있는 인터페이스를 제공하고 있다.

또한 위치 감지 정보와 관련하여 1<sup>st</sup> edition 에서는 특정 좌표계에 상대적인 위치 정보를 표현하던 것에 비하여, 2<sup>nd</sup> edition 에서는 최근 이동 단말에 포함되어 있는 GPS 를 이용한 절대적인 위치 감지 정보와, GPS 위치 정보를 활용한 절대적인 위치로의 이동 제어 명령 정보 구조도 추가하였다. 또한 해수면에 상대적인 고도 감지 정보 구조를 추가하여 GPS 정보와 함께 활용될 수 있도록

하였다.

### 3. MPEG-V 이진화 방법

기존의 MPEG-V 1<sup>st</sup> edition에서는 MPEG-V 표준에 따르는 모든 데이터는 XML 문서 형태로 존재하였다. 하지만 위치 감지 정보, 속도 감지 정보와 같은 일부 MPEG-V 데이터는 경우에 따라 초당 발생하는 횟수가 30회 이상일 수 있어 XML 문서를 작성하고, 송수신하고, 분석하기에는 송수신 대역폭 및 처리 부하측면에서 비효율적이다. 따라서, 상대적으로 효율적인 MPEG-V 데이터 이진화 방법의 필요성이 대두되었다.

다른 MPEG 표준의 이진화 방법은 표준의 별도 문서 혹은 부분에서 syntax 와 semantics를 설명하고 있었으나, MPEG-V 2<sup>nd</sup> edition의 이진화 방법 표준은 각 part별 표준 문서에 포함되어 있다. 다시 말하면, part2, part3, part4, part5, part6는 개별적인 데이터 스키마 구조를 가지고 있으므로, 해당 스키마 구조와 비교 가능하도록, 적합한 이진화 데이터 구조를 각 part에서 바로 설명하고 있다. 또한, 이진화 방법은 기존의 MPEG 7 visual descriptor에서 사용하는 방법을 따른다. 아래에서는 part 6의 최상위 요소 이진 데이터 구조와 part5의 감지 정보 및 제어 정보에 대한 이진화 데이터 구조에 대하여 설명한다.

#### 3.1 최상위 요소의 이진 데이터 구조

part6는 MPEG-V 전체 표준에서 공통되게 사용되는 최상위 요소와 공통 데이터 포맷의 스키마 구조가 설명되어 있다. 따라서 이진화 방법 측면에서는 이진 데이터의 최상위 구조 및 각 요소의 헤더 데이터 구조를 포함한다. 그림 5에서는 MPEG-V 최상위 요소의 스키마와 최상위 헤더의 이진 데이터 구조를 보여 준다.

스키마에서는 해당 문서의 해석에 필요한

```
<schema
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:mpeg7="urn:mpeg:mpeg7:schema:2004"
xmlns:mpegvct="urn:mpeg:mpeg-v:2010:01-CT-NS"
targetNamespace="urn:mpeg:mpeg-v:2010:01-CT-NS" version="ISO/IEC 23005-x"
id="MPEG-V-CT.xsd">
  <import
namespace="urn:mpeg:mpeg7:schema:2004"
schemaLocation="http://standards.iso.org/ittf/PubliclyAvailableStandards/MPEG-7_schema_files/mpeg7-v2.xsd"/>
```

| HeaderInfo{       | Number of bits | Mnemonic |
|-------------------|----------------|----------|
| Signature         | 40             | bslbf.   |
| Reserved          | 14             |          |
| ProfileIdentifier | 8              | uimsbf   |
| ElementIdentifier | 10             | bslbf    |
| }                 |                |          |

그림 5. MPEG-V 최상위 요소의 스키마와 이진 데이터 구조

namespace 그리고 참조 스키마를 포함하고 있으나, 실제 xml 문서에서는 특별하게 포함할 요소가 없다. 하지만 이진 데이터 구조에서는 먼저 끊임 없이 수신되는 이진 데이터 흐름에서 MPEG-V 이진 데이터가 시작됨을 알려주는 Signature와 특별하게 정의된 profile이 존재하는 경우에 필요한 ProfileIdentifier, 그리고 해당 이진 데이터가 포함하고 있는 실제 데이터 요소를 가리키는 ElementIdentifier를 포함하고 있다. MPEG-V 이진 데이터 수신부에서는 ElementIdentifier 코드에 따라서 해당 이진 데이터 구조에 따라 분석할 수 있다.

#### 3.2 감지 정보 및 제어 정보의 이진 데이터 구조

MPEG-V 스키마에서 모든 감지 정보와 제어 정보는 기본 타입을 바탕으로 이를 확장하고 있다. 마찬가지로 이진 데이터 구조에서도 기본 타입의 이진 데이터 구조는 다른 감지 정보와 제어 정보의 이진 데이터 구조에 포함되어 있다.

제어 정보의 기본 타입은 그림 6과 같이 제어 정보가 실행되기를 기대하는 시간 정보인 Time-

```
<complexType name="DeviceCommandBaseType"
abstract="true">
  <sequence>
    <element name="TimeStamp"
type="mpegvct:TimeStampType"/>
  </sequence>
  <attributeGroup
ref="iid:DeviceCmdBaseAttributes"/>
</complexType>
<attributeGroup name="DeviceCmdBaseAttributes">
  <attribute name="id" type="ID" use="optional"/>
  <attribute name="deviceIdRef" type="anyURI"
use="optional"/>
  <attribute name="activate" type="boolean"
use="optional" default="true"/>
</attributeGroup>
```

그림 6. 제어 정보 기본 타입 스키마

Stamp와 각 제어 정보를 구분하는 id, 해당 장치를 지정하는 deviceIdRef, 그리고 장치가 동작하거나 혹은 중지하라는 명령인 activate로 구성된다. 그림 7에서는 제어 정보 기본 타입의 이진 데이터 구조를 보여 준다. TimeStamp 와 DeviceCmdBaseAttributes 는 하위 이진 데이터 구조를 참조하고 TimeStamp를 표현하는 방식에 따라 절대적인 시간을 표시하는 AbsoluteTimeStamp, 별도의 Clock Tick 의 횟수를 표시하는 ClockTickTimeStamp, 그리고 Clock Tick 의 변화량으로 표시하는 ClockTickTimeDeltaStamp 방식을 선택할 수 있도록 한다. 또한, minOccurs 가 1 이상인 경우에는 고정된 크기를 사용하도록 하지만, optional 한 요소에 대해서는 Flag bit을 사용하여 존재하는 경우에만 bit를 할당하는 방식을 사용하였다.

같은 방법으로 감지 정보의 이진 데이터 구조에서도 기본 타입을 정의하고, optional 한 요소의 존재 유무를 위하여 Flag bit을 활용하는 방식은 다르지 않다.

### 3.3 Update 방식

감지 정보 혹은 제어 정보가 자주 발생하는 경

| DeviceCommandBaseType{          | Num. of bits | Mnemonic                    |
|---------------------------------|--------------|-----------------------------|
| TimeStamp                       |              | TimeStampType               |
| DeviceCmdBaseAttributes         |              | DeviceCmdBaseAttributesType |
| }                               |              |                             |
| TimeStampType{                  |              |                             |
| TimeStampSelect                 | 2            | bslbf                       |
| if(TimeStampSelect==1){         |              |                             |
| AbsoluteTimeStamp               |              | AbsoluteTimeStampType       |
| } else if (TimeStampSelect==2){ |              |                             |
| ClockTickTimeStamp              |              | ClockTickTimeStampType      |
| } else if (TimeStampSelect==3){ |              |                             |
| ClockTickTimeDeltaStamp         |              | ClockTickTimeDeltaStampType |
| }                               |              |                             |
| }                               |              |                             |
| DeviceCmdBaseAttributesType{    |              |                             |
| idFlag                          | 1            | bslbf                       |
| deviceIdRefFlag                 | 1            | bslbf                       |
| activateFlag                    | 1            | bslbf                       |
| If(idFlag) {                    |              |                             |
| id                              | ISO 10646    | UTF-8                       |
| }                               |              |                             |
| if(deviceIdRefFlag) {           |              |                             |
| deviceIdRef                     |              | UTF-8                       |
| }                               |              |                             |
| if(activateFlag) {              |              |                             |
| activate                        | 1            | bslbf                       |
| }                               |              |                             |
| }                               |              |                             |

그림 7. 제어 정보 기본 타입의 이진 데이터 구조

우에는 모든 정보가 변화하는 경우와 함께 일부 데이터만 변화하는 경우가 있다. 예를 들어 그림 8과 같이 객체의 움직임을 기술하는 Motion 감지 정보와 같은 경우에는 해당 시각에 속도는 일정하고 위치만 계속 변할 수 있다. 이를 위하여 MPEG-V 에서는 UpdateMode를 지원한다. MPEG-V 데이터의 생성부에서는 이전 데이터 내용을 알고

있기 때문에, 새로 생성되는 데이터가 일부 변하는 경우에는 UpdateMode를 사용하고, 모든 데이터를 새로 전달하고자 하는 경우에는 Normal

|                     |                |                        |
|---------------------|----------------|------------------------|
| MotionSensorType {  | Number of bits | Mnemonic               |
| UpdateMode          | 1              | bslbf                  |
| if(UpdateMode ==0){ |                |                        |
| MotionSensorNormal  |                | MotionSensorNormalType |
| }else{              |                |                        |
| MotionSensorUpdate  |                | MotionSensorUpdateType |
| }                   |                |                        |
| }                   |                |                        |

|                                |  |               |
|--------------------------------|--|---------------|
| MotionSensorUpdateType {       | Number of bits   | Mnemonic      |
| ...                            | ...  | ...           |
| ListUpdate                     | TimeStampFlag + IDFlag + ... + angularaccelerationFlag | bslbf         |
| ListItemNum = 0                |  |               |
| if (TimeStampFlag) {           |  |               |
| if (ListUpdate[ListItemNum]) { |  |               |
| TimeStamp                      |  | TimeStampType |
| }                              |  |               |
| ListItemNum++                  |  |               |
| }                              |  |               |
| ...                            | ...  | ...           |
| }                              |  |               |

그림 8. Motion 감지 정보의 UpdateMode

Type을 사용한다. UpdateType 에서는 각 Flag의 이진 조합인 ListItemNum 을 활용하여 실제 업데이트가 일어나는 요소를 미리 지정하고 수신부에서는 이를 실시간으로 확인할 수 있다.

#### 4. MPEG-V 데이터의 활용

MPEG-V 의 스키마 혹은 이진 데이터 구조는 특별한 시스템 플랫폼에 의존적이지 않은 형태로 기술되어 있다. 이는 여러 플랫폼에 적용 가능한

유연성을 제공하지만, 동시에 각 시스템 환경에 적합하도록 별도의 인터페이스를 제공해 주어야 한다.

LASeR 는 W3C의 SVG (Scalable Vector Graphics, [7]) 기술에 기반한 리치미디어 플랫폼 표준이다. LASeR 표준을 사용한 장면 정보는 사용자 인터페이스와의 상호 연동을 위하여 마우스, 키보드, 키입력을 위하여 W3C DOM Event 의 일부와 재생 및 화면 제어를 위해 LASeR에서 별도 정의한 LASeR Event를 함께 사용한다. W3C 의 Widget 은 Web 환경에서 사용하는 HTML 문서를 로컬 환경에서 소비하기 위하여 필요로 모든 파일을 하나의 패키지 형태로 제공하는 리치 미디어 플랫폼이다. Widget 은 W3C에서 정의한 DOM Event를 사용하여 사용자 인터페이스와 연동한다. 하지만 표준에서 정의하지 않는 새로운 사용자 인터페이스를 위해서는 기존의 Event를 재사용하거나, 혹은 표준을 벗어난 새로운 Event를 정의하여 사용할 수 있다. 본 절에서는 새로운 Event를 정의하여 MPEG-V 감지 정보 및 제어 정보를 전달하는 방법을 설명한다.

#### 4.1 새로운 외부 Event 방법

한가지 방법은 MPEG-V 데이터 구조를 새로운 인터페이스 형태로 변환하여 장면 정보에서 새로운 Event를 통하여 주고받는 방법이다. 그림 9에서와 같이 externalSensorEvent 는 기존의 LASeREvent를 상속하여 MPEG-V 감지 정보를 담을 수 있다. 장면 정보는 externalSensorEvent 의 Listener에 등록하여 MPEG-V 감지 정보를 받는다. 하지만 모든 MPEG-V 데이터를 새로운 Event 로 변환하기 위해서는 MPEG-V 데이터 구조를 알고 있는 변환 모듈을 필요로 한다.

## 4.2 새로운 데이터 전송 방법

상기 방법과 달리 MPEG-V 데이터의 세부 구조를 해석하지 않고, 전체 XML문서 혹은 이진 데이터를 주고받기 위한 방법을 그림 10에서 설명하고 있다. LAsER의 경우 sendDeviceCommand와 같은 새로운 element를 정의하거나, 새로운 Event를 정의하여 모든 XML 문서 혹은 이진 데이터를 전달하는 방법이다. 그림 10에서와 같이 XML 문서의 종류와 상관없이 deviceCommand 어트리뷰트의 값으로 XML 문서를 전달한다. 이 방법은 전달된 XML 문서 혹은 이진 데이터 구조를 다시 해석하는 모듈을 필요로 한다.

```
<lsr:sendDeviceCommand begin="rect_Red.click"
  xlink:href="#fdc1"
  deviceCommand="
    &lt;iidl:InteractionInfo&gt;
    &lt;iidl:DeviceCommandList&gt;
    &lt;iidl:DeviceCommand
      xsi:type="dcv:LightType"
      id="light1" deviceIdRef="fdc1"
      color="urn:mpeg:mpeg-v:01-SI-ColorCS-NS:red"
      intensity="5"/&gt;
    &lt;/iidl:DeviceCommandList&gt;
    &lt;/iidl:InteractionInfo&gt;
  &lt;/lsr:sendDeviceCommand>
```

그림 10. sendDeviceCommand 사용 예제

## 5. 결 론

본고에서는 최근 발행된 MPEG-V 1<sup>st</sup> edition과 새롭게 진행되고 있는 2<sup>nd</sup> edition의 변경사항인 바이오 센서, 고급 사용자 인터페이스 감지 정보를 설명하였다. 또한 2<sup>nd</sup> edition에 포함된 새로운 이진화 방법의 구조와 기본 타입 및 Update 방법을 설명하였다. 마지막으로 MPEG-V 데이터를 기존의 리치미디어 플랫폼에서 활용하기 위한 새로운 Event 모델을 제시하였다. MPEG-V

는 고급 사용자 인터페이스를 위한 데이터 표준인 MPEG-U[9], 그리고 다양한 리치미디어 플랫폼과 함께 가상공간 콘텐츠와 현실 공간 사용자를 연동하기 위한 중요 기술로 대두될 것으로 기대된다.

## 참 고 문 헌

- [1] W3C Device APIs Working Group, "Draft documents for Client-side APIs such as Calendar, Contacts, Camera, etc" retrieved from <http://www.w3.org/2009/dap/>
- [2] ISO/IEC 23005:2011, Information technology -- Media context and control --, 2011.
- [3] N12161, "Text of ISO/IEC CD 23005-5 2nd edition Data Formats for Interaction Devices", 2011.
- [4] N12162 "Text of ISO/IEC CD 23005-6 2nd edition Common Data Format," 2011.
- [5] ISO/IEC 14496-20:2008 "Information technology -- Coding of audio-visual objects -- Part 20: Lightweight Application Scene Representation (LAsER) and Simple Aggregation Format (SAF)" 2008.
- [6] W3C, "Widget Packaging and XML Configuration" 2011, retrieved from <http://www.w3.org/TR/widgets/>.
- [7] W3C, "Scalable Vector Graphics (SVG) Tiny 1.2 Specification" 2008, retrieved from <http://www.w3.org/TR/SVGTiny12/>.
- [8] W3C, "Document Object Model (DOM) Level 2 Events Specification" 2000, retrieved from <http://www.w3.org/TR/DOM-Level-2-Events/>.
- [9] N12155, "Text of ISO/IEC DIS 23007-2 Advanced user interaction interface" 2011.



임 성 용

- 1999년 한국과학기술원 전기및전자공학과(학사)
- 2001년 한국과학기술원 전기및전자공학과(석사)
- 2001년~현재 한국전자통신연구원 선임연구원
- 관심분야: 시스템 공학, 영상 처리



차 지 훈

- 1992년 명지대학교 전자계산공학과(학사).
- 1996년 Florida Institute of Technology 공학석사 (Computer Science).
- 2002년 Florida Institute of Technology 공학박사 (Computer Science).
- 2003년~현재 한국전자통신연구원 융합미디어연구팀장
- 2008년~현재 과학기술연합대학원대학교 부교수
- 관심분야: 멀티미디어 스트리밍, 영상처리, 컴퓨터비전



이 인 재

- 1999년 성균관대학교 전자공학과 (학사)
- 2001년 성균관대학교 전기전자및컴퓨터공학부 (석사)
- 2001년~현재 한국전자통신연구원 선임연구원
- 관심분야: 컴퓨터비전, 리치미디어, UI/UX