INTERACTIVE MULTI-VIEW VISUAL CONTENTS AUTHORING SYSTEM

*Injae Lee, Myungseok Ki, Seyoon Jeong, *Kyuheon Kim*

Broadcasting Media Department, Electronics and Telecommunications Research Institute (ETRI)
* Department of Electronics Engineering, Kyunghee University
ninja@etri.re.kr

ABSTRACT

This paper describes issues and consideration on authoring of interactive multi-view visual content based on MPEG-4. The issues include types of multi-view visual content; functionalities for user-interaction; scene composition for rendering; and multi-view visual content file format. The MPEG-4 standard, which aims to provide an object based audiovisual coding tool, has been developed to address the emerging needs from communications, interactive broadcasting as well as from mixed service models resulting from technological convergence. Due to the feature of object based coding, the use of MPEG-4 can resolve the format diversity problem of multi-view visual contents while providing high interactivity to users. Throughout this paper, we will present which issues need to be determined and how currently available tools can be effectively utilized for interactive multi-view visual content creation.

1. INTRODUCTION

Fast growth of network bandwidth and computing power has brought huge changes on multimedia services. Users are demanding more various types of content (e.g., text, graphics, audio, and video); higher resolution and quality; and especially more interactivity. Conventional A/V content has been created mainly targeting on uni-directional services rather than bi-directional - i.e., users enjoy content passively with little interaction. Multi-view visual content (MVC) is gaining interests due to its high interactivity feature. MVC is geometrically modified and synthesized visual data. Compared to conventional 2D images, multi-view images can provide more interactivity to users because users can freely control their views. Panoramic visual content (PVC), which is generated by stitching and blending more than 2 images so as to provide wider FOV (field-of-view), and object visual content (OVC), which allows scene change of a object in concentric manner, are the typical examples. Several MVC relevant products[1][2] are available in the market. However, exchange or sharing of the content generated from different products are impractical because each product has a proprietary format. The paper is motivated from this difficulty. MPEG-4 includes several extensions compared to MPEG-1 and -2 especially in terms of object-based interactivity. In addition to advanced A/V compression methods (MPEG-4 Part2, Part3 and Part 10), MPEG-4 Part1, which is named "Systems", contains additional tools such as binary/textual file format, multiplexing, synchronization scheme, scene description, etc. The use of MPEG-4 can resolve the format diversity problem as well as provide high interactivity to users. Once the scene of MVC is composed by MPEG-4, the content can be realized in the forms of textual or binary files with various events (graphic animation, time event, etc) which other tools don't support. And it can be transmitted through channels.

We have been developing interactive MVC authoring tool based on MPEG-4. Authoring implies creation of content which can be manipulated and distributed as a file. From the experience, this paper presents key issues on MVC authoring and recommendations for the issues. This paper is organized as follow: In section 2, issues for MVC authoring are presented. In section 3, brief descriptions on MPEG-4 Systems and BIFS nodes for scene composition are presented. In section 4, the architecture of developing authoring tool is presented, and we conclude in section 5.

2. ISSUES FOR DESIGNING AUTHORING TOOL

In this section, we present issues and consideration on designing the developing authoring tool.

2.1. Content type

There exist various types of MVC such as stereoscopic images, hologram, multiple viewpoint image, and panorama. But, from rendering perspective, they can be classified into 2 categories: rendering by 3D display devices vs. rendering by 2D display with viewpoint control. Stereoscopic data need special display devices to give depth impression. In order to provide multiple viewpoints at the same time, lenticular type or integral photographic type display devices are necessary. For panoramic data, large cylindrical or hemi-spherical screen can be used for full support of FOV (field of view). It is highly expected that such 3D display devices would be deployed more widely in near future, so users can enjoy more realistic 3D impression using them. However, at least currently, it is hard for general users to have such special 3D display devices due to the cost or other technical problems.

Because of such barriers, we decided to focus on content which can be rendered by conventional 2D display with viewpoint control and defined two MVC types as the main target content for authoring tool design: panoramic visual content (PVC) and object visual content (OVC). Fig. 1 shows examples of the two content types. Fig. 1-(a) shows an example of PVC and Fig. 1-(b) shows an example of OVC which provides different viewpoint of single object.

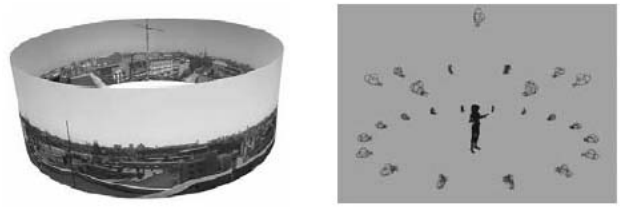


Figure 1. Examples of cylindrical image and multi-view capturing[3].

2.2. User interactivity

PVC and OVC can provide user interactivity allowing users to change viewpoints by themselves. Thus the existing MVC tools only support the interaction relevant to the viewpoint change. But such intrinsic interaction is not novel anymore. In order to increase immersive impression of users by PVC and OVC, it is necessary that authoring tool should support object based interactivity for creating content. The media object based interactivity implies that media object (e.g. video clip, image, graphics image, or text) is embedded into PVC or OVC, and such media objects are layered on top of background visual data with insertion, deletion or modification capability. For instance, the media object based interaction can be realized by creating a panoramic image with a button which controls the play of sound. Table 1 show media types for PVC/OVC and media objects which are determined to support in our authoring tool design.

2.3. Content format

From authoring MVC while providing aforementioned interactivity with the media object, there are two more issues that need to be determined: scene composition and binary file format. Scene composition is about locating and defining action of media object, and binary file format allows creation of tangible binary file. The file format must be able to incorporate the information of scene composition and encode whole data in indexed and compressed form.

Among feasible solutions, we chose MPEG-4 Systems specification as the best solution. MPEG-4 is an open standard with a number of tools. First of all, the use of the MPEG-4 standard can resolve the format diversity problem while providing high interactivity to users. Once the scene of 3D visual content is composed by MPEG-4, the content can be created in the forms of textual or binary files, and can be transmitted through channels using other MPEG-4 tools in straightforward manner. In the following section, we present brief description on MPEG-4 Systems and we also show which part of the specification is selected for our purpose.

3. MPEG-4 BIFS FOR MVC

MPEG-4 defines a toolbox of advanced algorithms for audio and visual information. The part 1 of MPEG-4, named `Systems' addresses the issue of composing data streams. It provides a standard file format, a general event model, interfaces to terminal/networks in the form of Java API, a multiplexing tool with timing information, etc. The MPEG-4 Systems also addresses

Table 1. Media object types for object-based interaction

	PVC	OVC
Input data	● Panoramic still image ● Panoramic video ● Panoramic multiple patches	● Object still images ● Object video
Supporting media data	● Video clip (rectangular, arbitrary shape) ● Geometric graphic data (rectangle, circle, box, cylinder, sphere) ● Still image ● Text ● Audio	
Supporting user controls & events	● Viewpoint navigation ● Web link ● Mouse event (object insertion /deletion) ● Time event ● Audio/video (play/stop/pause) ● Animation (rotation, translation, color, size)	
Output data	● Textual format (XMT-A) ● Binary format (MP4)	

the relationship between the audio-visual components that constitute a scene. BIFS(Binary Format for Scene) describes the spatio-temporal arrangements of the objects in the scene. Each node of the hierarchy is a media object and each node has attributes for behavior control and a local coordinate system for handling the media object in space and time. MPEG defined several profiles regarding BIFS nodes conceiving required functions for various applications[4]. However, MVC relevant service was not considered with enough care, so it was not able to choose one appropriate profile among pre-defined ones. Thus required features for each content type are investigated and proper BIFS nodes for the features are presented.

Panoramic image generally has 3D rendering plane such as cylinder or sphere. This implies the BIFS nodes for 3D spatial/temporal arrangement of A/V objects and 3D texture mapping should be incorporated in scene description. Unless a whole panoramic image is displayed on a large scale screen, users should be able to choose a certain area which generally has lower resolution than that of panorama. In some cases, a panoramic image can be composed of several small-sized images.

Table 2. Proposed list of nodes for PVC scene composition

Function	Nodes
3D Scene arrangement	Group, Transform
Object interaction	InputSensor, TouchSensor, TimeSensor, VisibilitySensor, Conditional, ROUTE
Navigation	Viewpoint, NavigationInfo
Video synchronization	MediaControl, MediaSensor, Script
Web link	Anchor
Media representation	Sound, AudioSource, ImageTexture, MovieTexture

(a) Scene graph nodes

Function	Nodes
3D geometric object modeling	Shape, Appearance, Material, IndexedFaceSet, Coordinate
Texture mapping	TextureCoordinate
3-D geometric object and text	Box, Cylinder, Sphere, Text, fontStyle

(b) Graphics nodes

Table 3. Proposed list of nodes for OVC scene description

Function	Nodes
2D Scene arrangement	Group, Transform2D
Object interaction	InputSensor, TimeSensor, Valuator, PlaneSenser2D, Conditional, ROUTE
Scene choice	TouchSensor, Switch
Video synchronization	MediaControl, MediaSensor, Script
Web link	Anchor
Media representation	Sound2D, AudioSource, Bitmap, ImageTexture, MovieTexture

(a) Scene graph nodes

Function	Nodes
2D geometric object and text	Background2D, Apprearance, Shape, Material2D, Rectangle, Circle, Text, fontStyle

(b) Graphics nodes

Therefore, nodes for rendering arbitrary located region and nodes for mapping multiple images/videos into one panoramic scene should be supported basically. In addition to the basic nodes for panoramic image rendering, nodes for object-based interaction and timing event handling are also required. Table 2 shows the proposed set of nodes and its functions for scene description of PVC. The nodes in Scene Graph provide 3D media arrangement of image and video, viewpoint controllability of users, and 3D navigation. The pre-defined nodes in Graphics allows 3D object modeling and 3D texture mapping. Considering the proposed nodes from the MPEG-4 profiling aspect, the Complete profile of Graphics and Scene Graph profile dimensions include all the nodes in the Table 2. However, if the PVC decoder is designed with the Complete profile profile, it is waste of resource because some nodes in this profile would be barely required for PVC. If the demand for PVC becomes higher, defining a new profile in BIFS should be managed additionally.

Regarding the scene description for OVC, a viewpoint control panel, which is a set of image or graphic objects, may be placed on the front of content to receive user-commands. Whereas the PVC required 3D rendering space, the view-change of OVC can be realized without 3D nodes – i.e., scenes from different views are arranged on 2D space and they are switched for view change. Hence, the basic scene change for OVC can be realized using the nodes for 2D media arranging, scripting nodes for switching, and media controlling nodes for temporal synchronization among scenes from different views. Table. 3 shows the list of selected nodes and their functions. From the perspective of MPEG-4 profiles, the Simple 2D+Text or Core2D profiles for Graphics, and the Complete 2D for Scene Graph are the smallest pre-defined profiles that includes all the required nodes. As the case of PVC, however, the Complete2D contains nodes which are unlikely used, so additional study for profiling MVC may be also needed in the future.

4. ARCHITECTURE OF MVC AUTHORING SYSTEM

We presented briefly what functionalities were targeted for user interaction, and how they can be realized by means of MPEG-4 Systems. Now we describe how the authoring tool is constructed for PVC/OVC creation with object based interactivity.

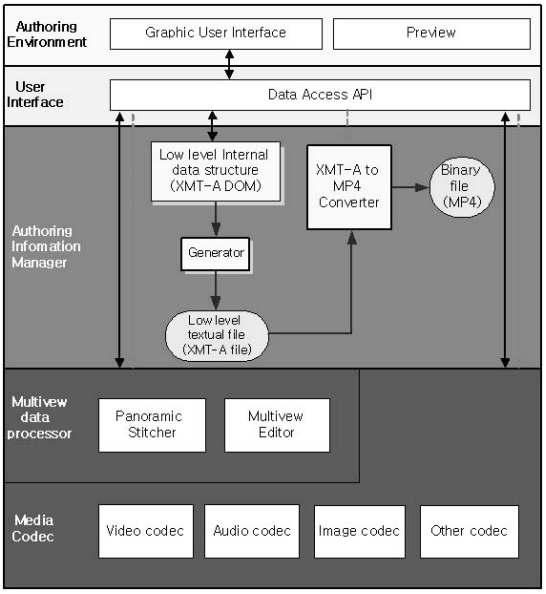
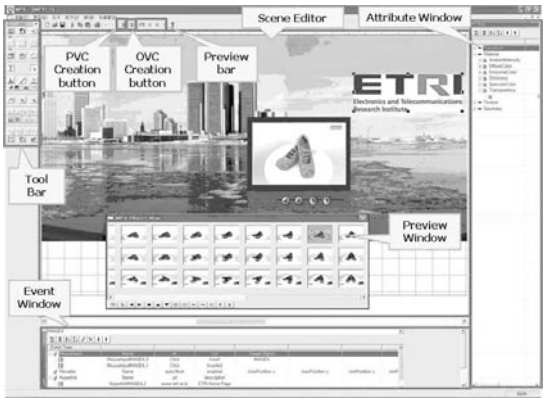
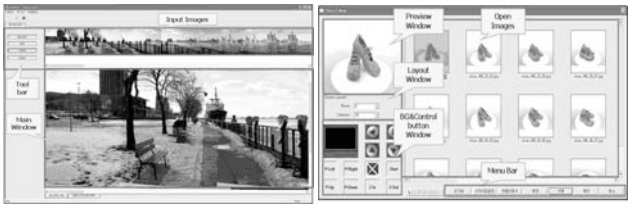


Figure 2. Architecture of MVC authoring system.



(a)



(b) (c)

Figure 3. The presented MVC authoring system: (a) authoring system for interactive MVC, (b) panoramic stitching tool, (d) OVC editor.

The architecture of the authoring tool is shown in Fig.2. It consists of the following modules; *Authoring Environment, User Interface, Authoring Information Manager, Multi-view Data Processor, and Media Codec*.

The *Authoring Environment* module provides convenient ways to combine individual image or video components for easy creation of PVC/OVC. This module also provides means for composing a scene with various media objects. Fig. 3 shows the GUI of the presented authoring tool. It has 'tool bar' for embedding media objects; 'scene editor' for locating and resizing objects; 'event window' for defining events (insert, delete, animation and timing); 'attribute window' for attribute of objects (e.g. color, rotation, position); and 'preview window' for previewing the authoring result.

The *User Interface* generates internal data structure based on scene composition information from *Authoring Environment* module using Data access API. And the *Authoring Information Manager* module, which is core unit of authoring tool, generates XMT-A based on internal data information from *User Interface* module, and also creates MP4 file from XMT-A. Since the XMT is described by a textual syntax, an author can easily exchange their contents with other creators. Also, authoring tools using the XMT will conveniently edit audiovisual objects, and add interactive events between objects in content or between objects and users for interactive broadcasting programs. However, XMT has large amount of data. Thus, for streaming the MPEG-4 contents, the XMT should be converted to the MPEG-4 binary file (MP4) using XMT-A to MP4 converter.

The *Multiview Data Processor* module provides the pre-processing to generate panoramic data or object visual data. Fig. 3-(b) and Fig. 3-(c) are the screen shots from PVC and OVC generation, respectively.

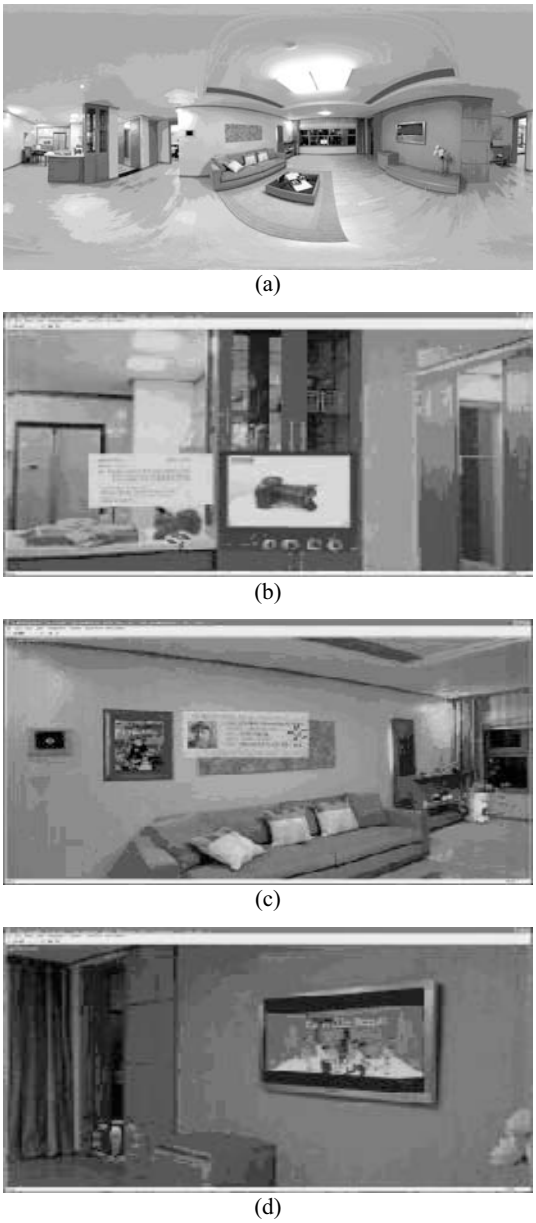
Finally, the *Media Codec* module contains various types of codecs for image, video, and audio. It decodes media data and transfer the decoded media data to *User Interface* module for editing and previewing.

5. CONCLUSIONS

In this paper, issues and MPEG-4 Systems based solutions for authoring MVC were presented. We defined two types of MVC (PVC and OVC) and also defined the level of interactivity for this content. For scene description and file creation, we adapted tools from MPEG-4 Systems. There exist several proprietary formats for MVC creation in the market. However, a standardized open format having the means for storage and transmission is needed for easy-sharing and enlarging relevant markets. We analyzed that MPEG-4 Systems (ISO/IEC 14496-1) should be the best choice and studied efficient use of MPEG-4 tools. We presented the suitable BIFS nodes for PVC and OVC, and suggested that a new profile is needed not to waste hardware and software resources.

Fig. 4 shows screen shots of contents which is created using the proposed authoring tool. Fig.4-(a) is a panoramic image for background, and Fig. 4-(b) is from an OVC-type content which provides different views of a camera. A control panel for view change is placed on the bottom of scene, and one image object is embedded for the showing the details of the camera. The shots of Fig.4-(c), (d) show the content when media objects (images, one geometry and one video clip) are appeared. Those objects are originally invisible, and become visible by clicking a certain button on the content.

As the demand for interactive content grows, the multi-view visual content is gaining high attention as an eligible candidate. The way of effective creation of the content needs to be further investigated covering whole relevant areas. We believe that this paper should be useful information for further development.



(a) (b) (c) (d)

Figure 4. Screen shots of interactive MVC created by the MPEG-4 based MVC authoring system.

6. REFERENCES

[1] Apple QuickTime VR Authoring Studio, http://www.apple.com/quicktime/qtvr

[2] Photovista Virtual Tour, http://iseemedia.com

[3] ISO/IEC JTC1/SC29/WG11 N5878, Report on Multi-viewAV Exploration, 2003

[4] ISO/IEC JTC1/SC29/WG11 N4848, Information technology Part1:Systems, March 2002.