An Adaptive Scene Compositor Model in MPEG-4 Player for Mobile Device

Hyunju Lee and Sangwook Kim

Computer Science Department, Kyungpook National University
1370 Sankyuk-dong Buk-gu, Daegu, 702-701, Korea
{hyunju, swkim}@cs.knu.ac.kr

Abstract. MPEG-4 supports dynamic scene composition through various interactions such as adding/removing/replacing objects and changing object's properties. MPEG-4 content can express rich meanings through various dynamic scene composition by object-based, and can be played to mobile or desktop environment. Therefore, the MPEG-4 player must effectively play according authoring purpose and meaning, and compose dynamic scene in any device. This paper proposes an adaptive scene compositor model. It is optimized for dynamic scene composition of MPEG-4 content in mobile device. This model supports exact analysis of the scene description, core information extraction for rendering through extraction rule table and object-based information management for object's reusability and flexibility.

1 Introduction

MPEG-4(Moving Picture Experts Group, ISO/IEC 14496)[1] is an international standard, comprehensive multimedia content standard that covers the coded representation and composition of media objects, interaction between objects, interaction between user and object, scene description, delivery and synchronization of media data. The MPEG-4 defines a multimedia system for interoperable communication of complex scenes containing various audiovisual objects.

The MPEG-4 player must play audiovisual scene that consists of various media objects, which change scene dynamically and also support various network environment. Other existing MPEG-4 players for mobile environment focus on transmitting and playing the limited number of multimedia data according to MPEG-4 standard. Therefore, it is insufficient for characteristic of MPEG-4 such as playback of various objects and playback of dynamic scene composition in any device environment.

In general, the MPEG-4 player's content playback is as follows. The MPEG-4 file formatter analyzes the header of an MPEG-4 file, which separates several media data that are included in a file, and delivers it to the system decoder. The system decoder reads media data in decoder buffer, and decodes according to decoding time. The audiovisual data is stored to composition buffer. The scene compositor analyzes the scene description, and arranges data of composition buffer on a screen. The player receives various user interactions.

K. Aizawa, Y. Nakamura, and S. Satoh (Eds.): PCM 2004, LNCS 3331, pp. 461–469, 2004.

For effective scene composition of dynamic MPEG-4 content in mobile device, we propose an adaptive scene compositor in MPEG-4 player. This scene compositor supports object information of scene description sufficiently. In order to improve content's playback efficiency in mobile device, the scene compositor provides module optimization from scene description's parsing to scene rendering. The scene compositor supports three core components: the content analyzer for analyzing/applying the scene description, the information manager for object-based management and the presenter for composition/rendering the scene.

This paper is organized as follows. Section 2 describes the scene compositor in the MPEG-4 player. Section 3 shows implementation result and evaluations. Finally, section 4 presents conclusions.

2 The Scene Compositor

The adaptive scene compositor module is composed of three major processing parts. These are the content analyzer for scene description's analyzing, the information manager for independent management of object information according to object's properties and the presenter for composition and rendering the scene. In order to expand module easily, this each module in scene compositor is managed independently.

Figure 1 shows the processing step of the scene compositor in the MPEG-4 player.

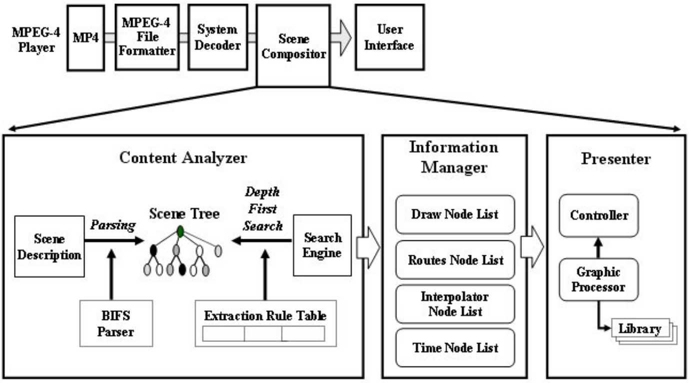


Fig. 1. The processing step of the scene compositor in the MPEG-4 player

After parsing the scene description, the scene tree is created. The search engine traverses a scene tree and extracts necessary information. To improve rendering efficiency, we redefined the information manager in order to object-based media management. We are going to explain in detail in next section.

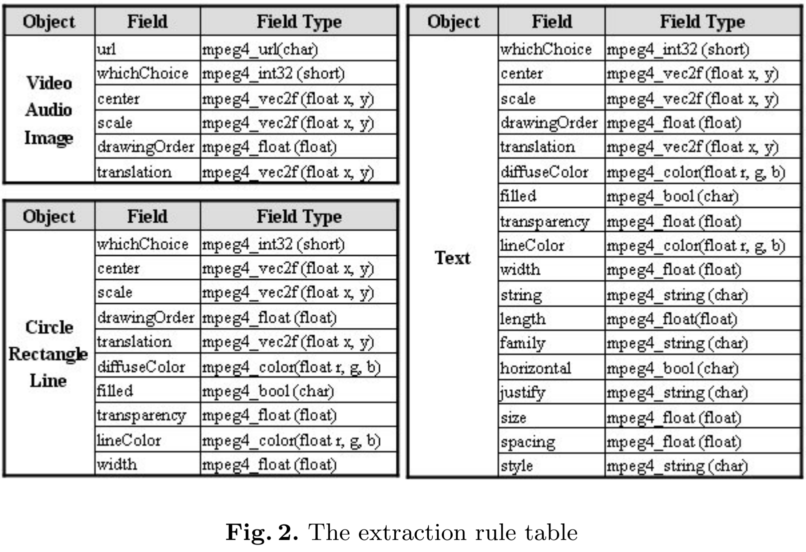
2.1 The Content Analyzer

The content analyzer analyzes exactly the scene description of MPEG-4 content, searches and extracts the core information of objects that need to rendering. It consists of the BIFS parser, the search engine and the extraction rule table.

The MPEG-4 describes the composition of a complex scene relies on the concept of both BIFS and OD. The BIFS(BInary Format for Scenes) describes the spatial-temporal composition of objects in a scene and provides this data to the presentation layer of the MPEG-4 terminal. The OD(Object Descriptor)[2]-[4] identifies and describes elementary streams and associates these streams with corresponding audiovisual scene data. It is used to connect BIFS and media stream resources. The MPEG-4 describes objects and their behavior in hierarchical models. The MPEG-4 uses the concept of a scene tree with object nodes. The scene tree can be obtained through BIFS parsing, and is a data structure used to hierarchically organize and manage the content of spatially oriented scene data. The multimedia information that is defined in scene tree can extract the necessary information through various search mechanism and can manage data from the user's point of view. To support sufficiently object's interactive information that is defined in tree, the MPEG-4 player needs to search the tree fast and exactly. It is important to MPEG-4 player's performance evaluation.

In this paper, we use a top-down method and DFS(Depth First Search) mechanism for object-based search. The search engine receives a tree's header, searches an object node and judges whether searched object node can draw. Now, current playback time and drawing order are considered. When the search engine searches an object, we extract object information according to extraction rules. The extracted object information is stored in the information manager.

Figure 2 shows the extraction rule table's contents.



Object	Field	Field Type
Video Audio Image	url	mpeg4_url(char)
	whichChoice	mpeg4_int32 (short)
	center	mpeg4_vec2f (float x, y)
	scale	mpeg4_vec2f (float x, y)
	drawingOrder	mpeg4_float (float)
	translation	mpeg4_vec2f (float x, y)

Object	Field	Field Type
Circle Rectangle Line	whichChoice	mpeg4_int32 (short)
	center	mpeg4_vec2f (float x, y)
	scale	mpeg4_vec2f (float x, y)
	drawingOrder	mpeg4_float (float)
	translation	mpeg4_vec2f (float x, y)
	diffuseColor	mpeg4_color(float r, g, b)
	filled	mpeg4_bool (char)
	transparency	mpeg4_float (float)
	lineColor	mpeg4_color(float r, g, b)
	width	mpeg4_float (float)

Object	Field	Field Type
Text	whichChoice	mpeg4_int32 (short)
	center	mpeg4_vec2f (float x, y)
	scale	mpeg4_vec2f (float x, y)
	drawingOrder	mpeg4_float (float)
	translation	mpeg4_vec2f (float x, y)
	diffuseColor	mpeg4_color(float r, g, b)
	filled	mpeg4_bool (char)
	transparency	mpeg4_float (float)
	lineColor	mpeg4_color(float r, g, b)
	width	mpeg4_float (float)
	string	mpeg4_string (char)
	length	mpeg4_float(float)
	family	mpeg4_string (char)
	horizontal	mpeg4_bool (char)
	justify	mpeg4_string (char)
	size	mpeg4_float (float)
	spacing	mpeg4_float (float)
	style	mpeg4_string (char)

Fig. 2. The extraction rule table

The extraction rule table is basis of search engine and information manager. In extraction rule table, object and object's main property field are defined. This is objects and object's properties that can do playback in mobile device, and is possible to extend according to playback environment. For example, if search engine searches a video object, the scene compositor extracts property information such as url, whichChoice, center, scale, drawingOrder, translation that is defined to extraction rule table. We can do optimization of search through core information in extraction rule table.

2.2 The Information Manager

The BIFS parser[5] creates interiorly the scene tree through scene description parsing. We applied the optimized search technique of scene tree to the MPEG-4 player. However, a tree is only possible hierarchic management of all object information that is included in MPEG-4 content. For reusability and flexibility of object-based, we propose the information manager that manages object information efficiently. It consists of five data structure and is designed adaptively for rendering.

The main characteristic of MPEG-4 is the object-based coding and representation of an audiovisual scene. For rendering object management, we create the DNL(Draw Node List). The DNL consists of head and sub. It is a structure that is added to the head and subs whenever an object is searched, searching is continued to last node of scene tree. The head contains object ID, the sub contains several property nodes. Whenever user event happens, the DNL is real-time updated. Several field's values of target object in the DNL are changed. The presenter again renders after recomposing the DNL.

The MPEG-4 BIFS's ROUTEs mechanism describes interactivity and behavior of objects. The event model of MPEG-4 uses ROUTEs to propagate events between scene elements. ROUTEs are comprised of connections that assign the value of one field to another field in the tree interiorly. It is information that processes dynamic user events that are defined in the MPEG-4 standard. ROUTEs combined with interpolators can cause animation in a scene. We create the RNL(Routes Node List) for interaction processing. The RNL consists of source object ID, event type, target object ID, the property field, and modification values. We extract interactive information after scene description's ROUTEs definition part parsing.

The MPEG-4 scenes can display a wide range of complicated dynamical behavior. The MPEG-4 provides stream animation to the scene with the BIFS-Animation tool. BIFS-Animation is a dynamic scene composition framework that enables optimal compression of the animation of all parameters of a scene. We compose two INLs (Interpolator Node List) for interpolator processing. If an object includes interpolator node when the search engine searches a scene tree, the search engine divides whether interpolator information is color or position, and store it in each List (CINL: Color Interpolator Node List, PINL: Position Interpolator Node List). The interpolator information consists of key and key-

Value. The INL consists of object ID, key(time value) and keyValue(changing value according to key).

The MPEG-4's another mechanism for dynamic scene composition is time-related interaction. In this paper, we create the TNL(Time Node List) for time-related interaction. An object needs playback start time and end time. The MPEG-4 player sets the system timer and processes the content's time information. The TNL consists of object ID, time value(start time or end time) and property value. The property value is activity or inactivity information of object. The time information is defined in the scene description's last part. The MPEG-4 can give playback time in object when content created. The MPEG-4 player preferentially processes time information than other user events. Each object of the first screen is played according to time information that is defined to the TNL, and receives other user events in addition.

Figure 3 shows five data structure of the information manager.

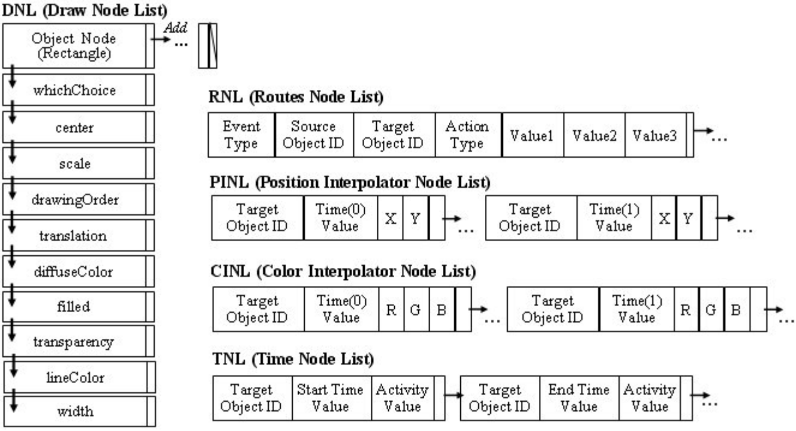


Fig. 3. Five data structure in the information manager

The information manager manages rendering and interaction processing by object-based, and reduces tree recomposition and frequent search of object information. Therefore, the new data structure reduces the load of system-side for additional scene update processing. Whenever scene is updated, it can offer high-level quality of scene to user through object's reusability.

2.3 The Presenter

In MPEG-4 Systems, scene rendering uses various frameworks. For scene rendering, the presenter consists of two components: the graphic processor and the controller.

The graphic processor consists of various graphic libraries to render various audiovisual object types. These graphic libraries consist of video/audio/image library for rendering, and support various data formats such as MP4v/H.264 video format, MPEG-4 AAC/G.723 audio format and JPEG/GIF/BMP image format. In order to expand easily scope of audiovisual data types by adding various kinds of graphic library, the library is independently managed from the presenter. Therefore, the presenter can support different audiovisual data types as well as various kinds of graphic libraries.

The controller connects media streams in scene compositor with media's graphic library in presenter. For example, if media object is mp4v format video object, the controller connects to video format library. If media object is JPEG format image object, the controller connects to JPEG format library. The controller is also independent module.

3 Implementation and Evaluation

The scene compositor supports Complete2D profile of the MPEG-4 standard. Additionally, the scene compositor can manage the MPEG-4 content that consists of various media objects. The scene compositor can be included in desktop's MPEG-4 player as well as mobile(PDA) device's MPEG-4 player. It is difficult that mobile device's MPEG-4 player plays various media objects at the same time due to resource limitations. Therefore, content's playback quality is inferior in case of desktop's player. This proposed scene compositor model is optimized to be applied on mobile-based player and desktop-based player. This scene compositor's development uses C/C++ in MS-Windows 2000/XP and Linux environment.

Figure 4 shows an execution example of MPEG-4 content in CLM MPEG-4 player. The CLM player contains our scene compositor model fully.

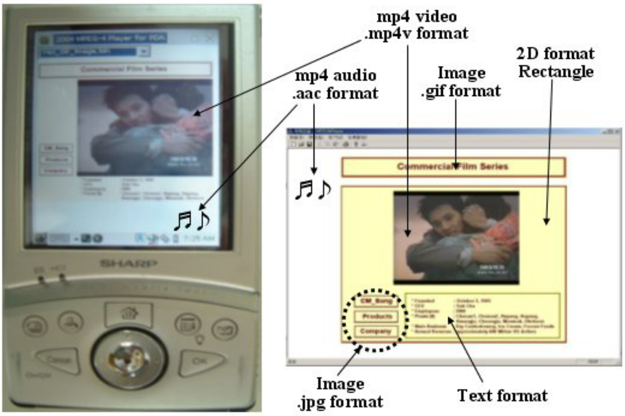


Fig. 4. An execution of MPEG-4 content in MPEG-4 player for PDA and desktop

In figure 4, an MPEG-4 content consists of an MPEG-4 video object(PDA: QCIF size 176x144 pixels, Desktop: CIF size 352x288 pixels), an MPEG-4 audio object(AAC mono, 64Kbps), several GIF/JPEG format images, and a geometric object. The image object receives click event. This MPEG-4 content is manufactured as commercial film and made from the MPEG-4 Authoring System[6]. We made two contents because playback environment is different. Object's sizes in contents are different but content's purpose and meaning are same.

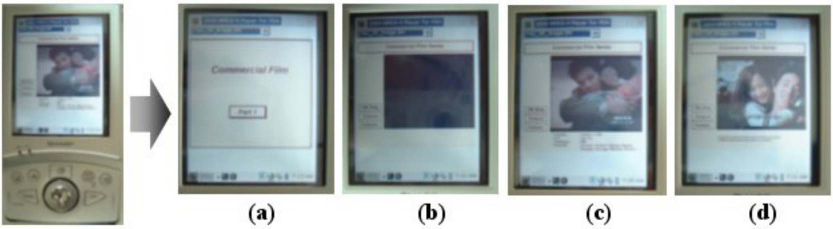


Fig. 5. An MPEG-4 content's playback example

In figure 5, the CLM player is PDA-based. The MPEG-4 content supports several user interactions.

The MPEG-4 scene's action scenario is as follows.

(a) The first scene is composed of two image objects, a front image object receives user's click event.

(b) If event happens, second scene runs. The second scene is composed of a title of image format, an MPEG-4 video object, three image icons and a rectangle object of background. All objects excepting video are defined time-related event. All objects are played after video playback 5 seconds.

(c) The video object is presented, the three icon images receive click event and presents a text. Video object is played during 4 minutes. While video object is played, user can give event in image object.

(d) If MPEG-4 video object ends, then the scene is ended.

To evaluate scene compositor's object information management and scene composition capability, we compared our scene compositor model with KNU player(ver 1.0) and IM1-2D player(ver 5.0).

The KNU player[7] and IM1-2D player are MPEG-4 content player of Windows-based. Specially, the IM1-2D player is an open architecture. Our scene compositor model is included in the CLM player. The CLM player is two versions that have different O/S and device environment, and included the proposed scene compositor in these two players. One is Linux-based for PDA, another is Windows-based for Desktop. Three players focused on the Systems part of the MPEG-4 standard and applied highly player's components modularization.

Table 1 shows the comparison of scene compositor's several capabilities in each player.

Table 1. The comparison of scene compositor's several capabilities in each player

Requirement	Scene Compositor in KNU player	Scene Compositor in IM1-2D player	Scene Compositor in CLM player
Audiovisual object information management	Supported (through various data structure and media libraries)	Supported (through the core class and rendering functions)	Supported (through various data structure and media libraries)
Spatial and temporal composition	2D composition of simple scene	2D composition of simple and complex scenes	2D composition of simple and complex scenes
Object-based Interactivity	Supported (limited: user interaction)	Supported (limited: user interaction and time-based in object)	Supported (limited: user interaction and time-based in object)
Independent management of module	Supported (limited)	Module dependency on the lower part in the player	Fully supported
Adaptability with other environment (platform independency)	Not supported (O/S dependency)	Not supported (O/S dependency)	Supported (Linux-based PDA and Windows-based Desktop)
Profile	2D Complete	2D Complete	2D Complete
Module extensity	Supported	Supported (complexity)	Supported (more ease through modularization
Module optimization for mobile environment	Not supported	Not supported	Fully supported

Several core comparison items are compositor's information management capabilities and adaptability with players of different environment(O/S, Device). The proposed scene compositor can improve player's performance through optimization, and play MPEG-4 content of same purpose to player of different environment through adaptability.

4 Conclusions

In order to compose a dynamic scene in MPEG-4 player for mobile device, research about scalability of the scene description and object-based data management are very important.

In this paper, we proposed the scene compositor model that is optimized for effective searching of scene composition information and management of the extracted information. The scene compositor consists of the content analyzer, the information manager and the presenter. For effective extracting of object and object's properties field value, we defined the extraction rule table. For extracted information management, we defined the information manager. The information manager improves reusability and flexibility by object-based that is MPEG-4 scene's characteristic. The presenter renders efficiently according to the information manager. It consists of the graphic processor for media objects of various formats and the controller for connection of the scene compositor and the presenter. This scene compositor model is designed to be adapted to any player regardless of O/S and device environment through module optimization.

In the future, to support sufficiently spatial-temporal relationships of content for mobile device, we will improve the scene compositor's composition capability, processing the various events and playback quality in MPEG-4 player that requires high QoS.

Acknowledgment. This work was supported by Korea Research Foundation Grant (KRF-2003-002-D00304).

References

1. R. Koenen, "MPEG-4 Overview," ISO/IEC JTC/SC29/WG11/N4668, March 2002, http://www.m4if.org/resources/Overview.pdf
2. A. Puri and T. Chen, Multimedia Systems, Standards, and Networks, Marcel Dekker, New York, USA, March 2000.
3. A.E. Walsh and M. Bourges-Sevenier, MPEG-4 Jump-Start, Prentice Hall, Upper Saddle River, NJ, USA, December 2001.
4. F. Bouilhaguet, J-C. Dugourd, S. Boughoufalah and C. Havet. "Interactive Broadcast Digital Television. The OpenTV Platform versus the MPEG-4 Standard Framework," IEEE International Symposium on Circuits and Systems, Vol. 3, pp. 626-629, Geneva, Switzeland, May 2000.
5. H. Lee and S. Kim, "An Effective Management Technique of MPEG-4 Scene Description," Proceedings of the 13th Joint Conference on Communications and Information, pp. I-A-2.1-4, Anmyeondo, Korea, April 2003.
6. K. Cha, H. Kim and S. Kim "The Design and Development of MPEG-4 Contents Authoring System," Journal of the Korea Information Science Society, vol. 7, No. 4, pp. 309-315, August 2001.
7. H. Lee, Y. Lim and S. Kim, "An MPEG-4 System supporting XMT Scene Description," Proceedings of the 14th Joint Conference on Communications and Information, pp. II-D-2.1-4, Chungmu, Korea, April 2004.