2004－DD－44　(1)
2004／5／28

XML-based Information Format and Access Designed for Digital Motion Picture Creation

SHEN Jinhong　Seiya MIYAZAKI　Terumasa AOKI　Hiroshi YASUDA

Abstract　We are developing a software tool that can interpret the textual screenplay into digital movie with effects of computer animation, real images and their simple composition. To efficiently represent various data of different functional modules and conveniently communicate among these modules, Extensible Markup Language XML is employed to address these demands involving three classes of data coding: screenplay formatting, motion picture making description, and video contents description for indexing and retrieval because XML is suitable for data saving of specific programs, multimedia presentation, data presentation on the Internet, and so on.

Keyword　Digital Movie, XML, Ontology, Content-based Retrieval, MPEG-7, Knowledge-based

1. Introduction

A low-cost easy-to-use moviemaker system has good entertainment and education marketplace. The advanced broadband network techniques already enable us to access multimedia streams (such as through the World Wide Web) and display them (e.g. digital videos, most of them are compressed) on personal computer. A great challenge on Web communication lies in giving such an environment that enables any people to make his presentation and deliver it uncomplicatedly. After analyzing the feasibility of realization, we think it is reasonable to automatically interpret a verbal screenplay into a relevant motion picture with visual effects like real image, 3D animation, or their composition, where real images are extracted from digital video (movie, animation, TV programs, etc.) library. This software system for automated moviemaking we are implementing is named DMP (Digital Movie Producer) [1].

The aim of DMP project is to develop such a tool that provides an integrated environment for dealing with the complete process of creating digital film. It comprises of different technical modules such as natural language understanding, animation creation, video retrieval, composition and edition. One of the most important problems in building these modules concerns the representation of various data of functional modules and the interoperability between them. We chose Extensible Markup Language XML to format the data of screenplay, describe motion picture contents in XML ontology of digital filmmaking, and indexing video features and contents in MPEG4/7 XML for video retrieval. This choice is dependent on taking account of XML's very wide application areas such as data saving of specific programs, multimedia presentation, data presentation on the Internet, and so on.

This paper organized by the following writings. Section two introduces the production pipeline of DMP which acts as the background of this paper. The third section analyzes the requirements of data processing in this system and puts forward relevant methods to realize them. The next part demonstrates the coding designs in detail,

School of Engineering, The University of Tokyo
4-6-1 Komaba, Mekuro-ku, Tokyo, 153-8904 Japan
e-mail:
{j-shen, seiya, aoki, yasuda}@mpeg.rcast.u-tokyo.ac.jp

followed by the summaries of conclusions and future work in the last section five.

2. Motion Picture Creation

A film can create a five-dimensional world of sight, sound, touch, taste and smell in the two-dimensional medium of film of sight and sound. In the large body of knowledge that surrounds learning styles, humans learn 83% through sight, 11% through hearing. Compared with pure text and static image, motion picture is the sort of multimedia full of vast information, enabling us to absorb information the most interestingly, conveniently and effectively.

2.1 Background

Nowadays there are three classes of digital movies classified by their different ways of generation:

(1) Digitally-stored conventional film,
(2) Digital video (DV),
(3) Computer Graphics (CG) movie.

The traditional film production by the first way is very expensive and time-consuming so that it is not a rational approach for personal usage. Movies generated by the ways (2) and (3) mentioned above are considered "real" digital ones, but it is impossible to produce personal movie readily within short time when we come up with an idea for movie, because a variety of limits exist in these approaches. For example, the process of generating dynamic 3D Computer Graphics using tools (such as Alias *Maya* and Autodesk *3D StudioMax*) or the post-production of DV using video editors (like Adobe *Premiere* and Apple *iMove*) is quite troublesome. The technical orientation is to develop an easy-to-learn and easy-to-use desktop software tool by which a general person can make his own digital visual contents and deliver it easily [2].

2.2 DMP System Structure

DMP aims to interpret a verbal screenplay into a relevant motion picture automatically with various visual effects like real image, 3D animation, or their composition, where real images are extracted from digital video library (figure 1).

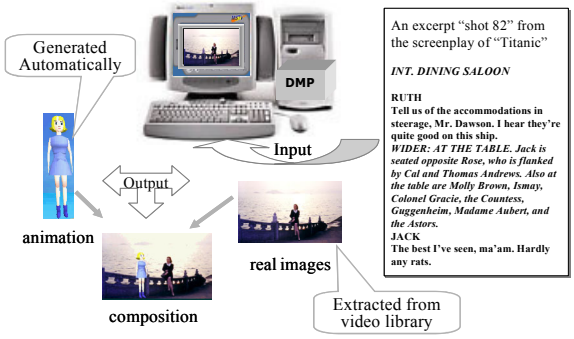

Figure 1. DMP (Digital Movie Producer)

Supposing the existence of a library that stores 3D models and actions mentioned in the script, it is possible to combine objects and actions according to the screenplay and to choose optimal placement for the camera automatically. Reusing digital movie involves the whole processing for video retrieval that contains content analysis and feature extraction, content modeling, indexing and querying [3].

2.3 Production Pipeline

To be more accessible to non-programmer users, verbal screenplay is proposed as input form. Users should not need to own knowledge of mathematics, computer 3D techniques, and artists to understand a complex software package. That indicates that DMP should be able to transform user's imagination into motion picture with high degree autonomy.

By using Artificial Intelligent approach, DMP creates digital motion picture and decide the temporal order of video clips automatically. A virtual film director (figure 2) is responsible for the visual aspect of screenplay dependent on knowledge of plot structure in KB. He gives commands for the dramatic structure, pace, and directional flow elements of the sounds and visual images to visualize the event. Composition, the location

of characters, lighting styles, depth of field and camera angle are all determinant factors in the formulation of the visual information. Movie player assembles the resultant plan created by inference engine into images. Virtual camera records the frames that are to be played as a still or a sequence of images. The intelligent module Virtual Director is embedded as a subsystem in the integrated system environment to realize the automation.

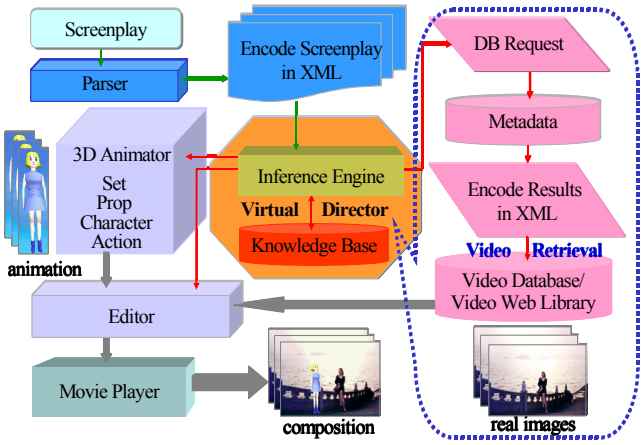

Figure 2. System Architecture Diagram

We utilize Japanese NHK's TVML player to render our digital movie, which can show animation or live-action film. It supports movie files (AVI, QuickTime, SGI), audio files (WAV, AIFF), TIFF still pictures, and also supports OpenInventor and VRML 1.0 for the modeling data format of computer-generated characters, sets and props.

The pipeline drawn in figure 3 shows the data flow beginning from screenplay input to the generated movie output, covering all main data conversion step by step.

3. Classes of Data Processing

Diverse media types and formats will be processed by DMP, mainly listed as the follows:

- ✓ textual file (screenplay input),
- ✓ digital video document (from video library),
- ✓ 3D models (pre-prepared for animation),
- ✓ animation (computer generated),
- ✓ movie contents (created by DMP)

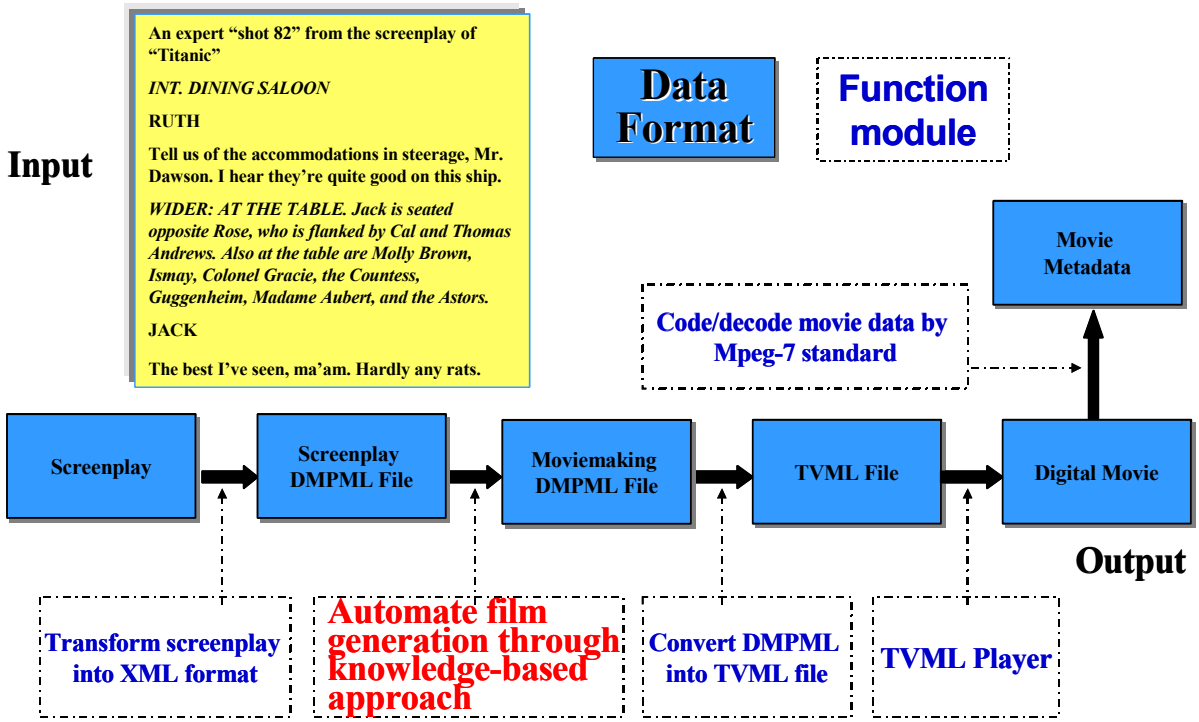

Figure 3. DMP System Data Flow Diagram

Representing and storing the information contained in above documents require us to seek a flexible, scalable, and reusable way based on compatible structure [3], involving three aspects of data structuralization:

1. Formatting the data of screenplay;
2. Representing motion picture making in ontology of digital filmmaking;
3. Describing and storing video features and contents for video indexing and retrieval.

We using XML language and its data representation mechanisms as a glue structure because it can address the above demands for its suitability in application areas of data saving of specific programs, multimedia presentation, and data presentation on the Internet. The next section will explain these three kinds of DMP XML standards respectively.

4. DMPML (DMP Markup Language)

4.1 DMPML-S (DMPML-Screenplay)

XML allows the definition of structures to format information of screenplay since XML is a language suited for describing structured information and its properties.

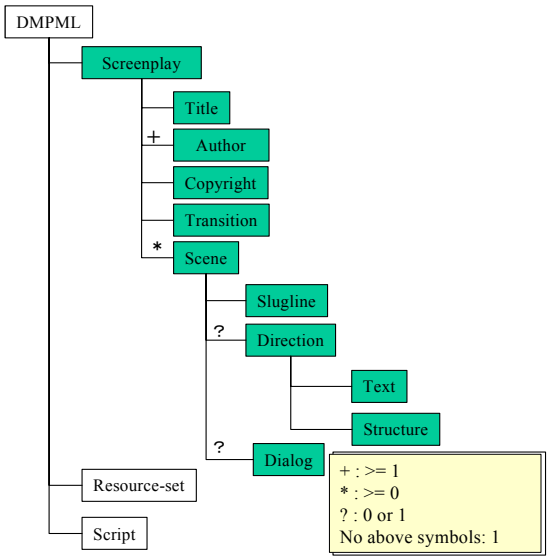

Figure 4. DMPML-Screenplay Tree

For example, an excerpt of screenplay "Daytime in park, Jack takes the goods" may be expressed in DMPML-S (figure 4) as the follow sentences:

```
<Screenplay>
  <Title>title</Title>
  <Author> The University of Tokyo Yasuda
        Lab. </Author>
  <Copyright> The University of Tokyo Yasuda
        Lab. 2003 </Copyright>
  <Transition type="FADE IN:" />
  <Scene>
   <Slugline place="EXT." timeOfDay="DAY">
    <BasicLocation name="park"/>
   </Slugline>
   <Direction>
    <Text> Jack takes the goods.</Text>
    <Structure>
     <Who name=" Daughter " />
     <WhatObject name=" the good " />
     <WhatAction name=" take " />
    </Structure>
   </Direction>
   <Dialogue name=" Daughter ">I got it!
         </Dialogue>
  </Scene>
  <Transition type="FADE OUT:" />
 </Screenplay>
```

Tags	Properties	Special Codes
<Slugline>	place	"DAY", "NIGHT", "DAWN", "DUSK", "CONTINUOUS", "MORNING", "AFTERNOON", "EVENING", "SUNRISE", "SUNSET", "LATER", "MOMENTS LATER", "SAMETIME"
<Slugline>	timeOfDay	"INT.", "EXT.", "I/E."
<Transition>	type	"FADE IN:" "FADE OUT:", "CUT TO:", "CUT BACK TO:", "DESOLVE TO:", "MIX TO:", "LIGHTS UP:", "WIPE TO:", "ZOOM IN:", "ZOOM OUT: "

Figure 5. Special Codes

Some tags such as <Slugline> still contain sub-tags (e.g <BasicLocation>) or properties (e.g. timeOfDay). Some data of element must be written in special codes (figure 5).

4.2 DMPML-M (DMPML-Movie)

Virtual Director A film is made up of shots arranged in sequence. We will deal with the filmmaking techniques that could be utilized in cyberspace applications. The virtual director establishes a point of view on the action that helps to determine the selection of shots and camerawork through rule-based planning, timing out every shot and important camera move. He first makes high-level shooting plan such as "track one's face" for each event based on his directorial expertise, then gives commands about shot types and shot sequence, at last calculates the parameters of camera position, orientation, and movement to satisfy the these commands. If there are suitable video clips in video database or video web library, the required clips will be extracted from the database/library, otherwise, 3D animation will be created based on cinematic knowledge. See the usual steps in linear animation generation (figure 6).

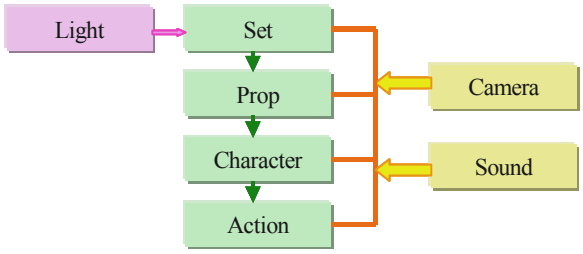

Figure 6. Control Flow of Moviemaking

Computer-generated animation Sets, props, and characters with action abilities have been pre-made and stored in a database as candidates since it is impossible to create accurate primitive objects automatically without modeling them in advance. If necessary, composition of animation and real image will be made. Composition, the location of characters, lighting styles, depth of field and camera angle are all determinant factors in the formulation of the visual information. Camera works and sound can be set up at any stage.

Ontology Motion picture contents are represented in XML ontology of digital filmmaking which had been designed when building filmmaking knowledge base. For human beings, information that we encounter is understood through our own internal data structures, which are our own implicit organizations of knowledge retrieved from the world we inhabit. With regards to information processing in machines, automatic movie generation system also needs an ontology that possesses sufficient semantics for making movie from script. Knowledge is without meaning unless it is contextualized. Ontology can be seen as a conceptual map where the links between individual pieces of knowledge are delineated. A precise manner is needed to encode the large body of cinematic knowledge into knowledge base for computer to manipulate.

DMXML-M Figure 7 outlines the DMPML-Movie design. Within <Script> type, those subtypes of <Camera>, <Light>, <Character>, <Prop>, <PlayMovie>, etc contain information of coordinates and directions for action. An example description in DMPML-M is showed in figure 8.

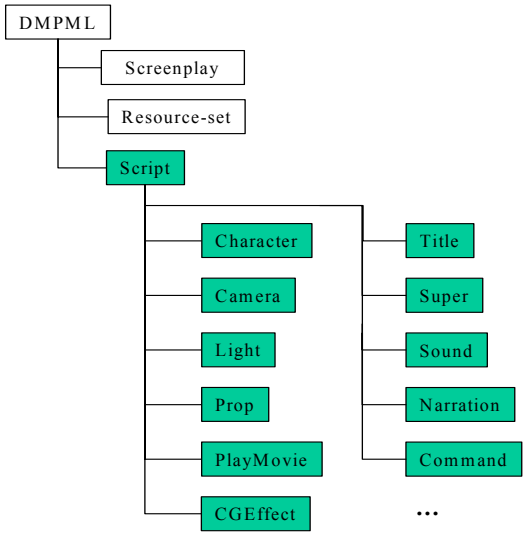

Figure 7. DMPML-Movie Tree

```
<SCRIPT programTitle="title">
 <SCENE name="park" setName="park" >
  <CHANGE-VOICE name="david" voicetype="e_man" />
  <POSITION-CHARACTER name="david" d="0.0" z="2.0" posture="standing" y="0.0" x="0.0" />
  <SET-LIGHT-MODEL-FLAT name="light1 " r="1.0" g="1.0" b="1.0" x="1.0" y="0.2" z="1.0" />
  <SET-LIGHT-MODEL-FLAT name="light2 " r="1.0" g="1.0" b="1.0" x="-1.0" y="0.2" z="1.0" />
  <SET-LIGHT-MODEL-AMBIENT name="light_ambient" r="0.5" g="0.5" b="0.5" />
  <CAMERA-MOVEMENT name="Acam" x="0.0" y="1.5" z="9.0" pan="0.0" tilt="0.0" roll="0.0" vangle="50.0" transition="immediate" />
  <SKIPSCRIPT switch="off" />
  <TALK name="david" text="hello!" />
  <Walk name=”david” x=”0.7” sync=”par” />
  <Look name=”david” what=”Acam” />
  <WAIT time="5.0" />
  <SKIPSCRIPT switch="on" />
 </SCENE>
</SCRIPT>
```

Figure 8. DMPML-M Example Description

4.3 DMPML-C (DMPML-Contents)

4.3.1 DMPML-C

In figure 9, those dark squares indicate the main contents should be extracted from video in order to reuse the video.

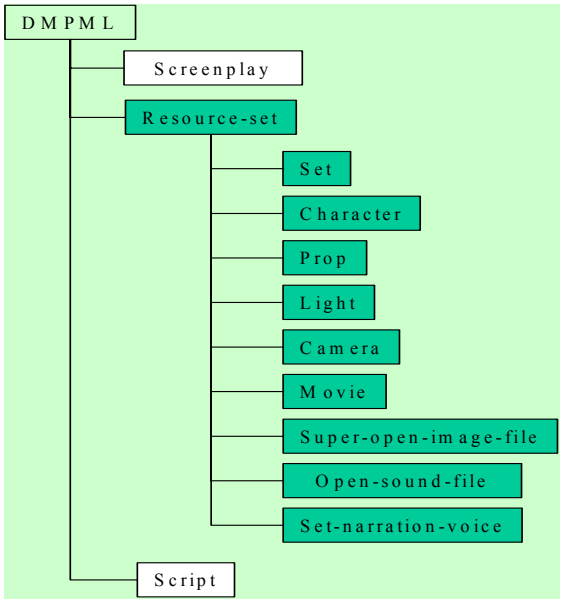

Figure 9. DMPML-Contents Tree

Under the type of <Resource-set>, those subtypes of background <Set>, character <Character>, object <Prop>, lighting information <Light>, camerawork <Camera>, moving picture <Movie>, static image <Super-open-image-file>, music and effects <Open-sound-file>, dialogue or talk <Set-narration-voice> will be utilized to describe the features and contents of videos in

```
<RESOURCE-SET>
 <Character name="Father" cid="F011/123456789ABD"
            type="urn:u-tokyo:dmp:cs:v0.5:Object:3DModel"
            href="http://foo.tv/Father.jar">
  <Feature type="Format" value="TVML Character" />
  <Feature type="Type" value="Human" />
  <Feature type="Gender" value="Male" />
  <Feature type="Age" value="Middle" />
  <Feature type="Voice:Style" value="Deep" />
  <Feature type="Voice:Language" value="English" />
  <Feature type="Hair:Style" value="Casual" />
  <Feature type="Hair:Color" value="Black" />
  <Feature type="Hair:Length" value="Short" />
  <Feature type="Skin:Color" value="Yellow" />
  <Feature type="Eye:Color" value="Brown" />
  <Feature type="Glasses:Style" value="Two Point" />
  <Feature type="Clothes:Shirts:Style" value="Open Neck" />
  <Feature type="Clothes:Shirts:Sleeve" value="Short" />
  <Feature type="Clothes:Shirts:Color" value="Striped Blue" />
  <Feature type="Clothes:Trouser:Style" value="Jeans" />
  <Feature type="Clothes:Trouser:Color" value="Blue" />
  <Feature type="Clothes:Trouser:Length" value="Long" />
  <Feature type="Action" value="Walk" />
  <Feature type="Action" value="Talk" />
 </Character>
</RESOURCE-SET>
```

Figure 10. DMPML-C Example Description

library. An example DMPML-M description of character element and its features like age hair clothes is showed in figure 10.

But based on current technology, not all of the information can be extracted and annotated automatically [4, 5]. From the point of view of data analysis, video surrogates can be classed under the headings *raw video features* (e.g. file size), *physical features* (spatio-temporal distribution of pixels: e.g. color) and *semantic features* (high-level concept: e.g. object). Varied indexing schemes have been put forward for different video

retrieval goals. Usually the whole processing for retrieval are divided into content analysis and feature extraction, content modeling, indexing and querying.

4.3.2 DMPVR

DMPVR (DMP Virtual Retrieval), a subsystem of DMP, focuses on design multi-modal video indexing. Giving an overview of DMPVR in one sentence, a suitable multi-category video modeling and multi-modal query mechanism with multi-modal video indexing were constructed based on MPEG-7 from the perspective of film director.

- *Multi-modal query mechanism* Visual content may be conveyed in both narrative (language) and image.

Multimodal Query	Retrieval Items
Query by example	Visual features
Query by text (Keywords and free-text)	Cinematic structure Semantic content (of annotated video)
Query by standard query language	Semantic content (of un-annotated video)

- *Multi-category video modeling* By taking advantage of ontology as mentioned in the above section, it is possible to facilitate conceptual search.
- *Multi-modal video indexing* Systems that combine visual features, sound, text as well as structured descriptions can get powerful retrieval. We will use textual information (such as closed captions) whenever available for video indexing.

MPEG-7 XML In DMP the data model may use metadata of MPEG-7 in order to provide more effective and efficient video retrieval. MPEG-7 standard has been used to encode video data by DMP for MPEG-7 is mainly intended for content identification purposes while other coding formats such as MPEG-2, 4 are mainly intended for content reproduction purposes.

MPEG-7 (DSs, Ds, DDL based on XML) standardizes the information exchange of descriptive information [6,7]. We use its low-level and high-level descriptive metadata for video data modeling and retrieval. But only MPEG-7 is not completely suitable enough to serve as a multimedia data model, for its aim was not taking into different purposes. In DMP, XML tags are supported by our DMPML.

Video Segmentation Shot change detection may be realized by *direct pixel comparison* or a more robust method *histogram comparison*. A shot change is detected if a significant percentage of pixels differ or if the histograms of two consequent frames differ significantly. (Camera operations such as zooming, tilting, and panning will make it difficult to detect shot changes.) After the video is divided into different shots by using one or more of the above techniques, the shots are classified based on the models (e.g. weather forecast, news).

Automated Annotation To detect and track *objects*, a typical strategy is to initially segment regions based on color and texture information. After the initial segmentation, regions with similar motion vectors can be merged subject to certain constraints such as adjacency. Human *faces* can be detected by using human skin color and DCT transform coefficients in MPEG and broad shape information. It is possible to recognize certain facial expressions and gestures using models of face or hand movements. Particular movements such as entering/exiting a scene and positioning objects using motion vectors are able to detect.

Query and transaction models of video database systems differ from those of the traditional database systems. With the advancement of techniques on computer vision and multimedia database, video retrieval systems developed from *traditional text-based* video indexing annotated manually

(using keyword, attribute, free-text to present high-level concept), *content-based* video indexing exploiting the technique of signal processing (focusing mainly on extracted low-level visual features: color, shape, texture, motion), to current *semantics-based* video indexing by semantic annotation exploiting the techniques of Artificial Intelligence (high-level semantic features: object, event; and higher-level semantic features: emotion). But it is still not easy to be annotated automatically, only realized in some domains such as sports (basketball) and dance (ballet).

5. Conclusion and Future Work

Our research question mentioned in this paper is on how to organize these data for effective and efficient query applied for the use in DMP system. To efficiently represent various data of different functional modules and conveniently communicate among these modules, Extensible Markup Language XML is employed to address these demands involving three classes of data coding: screenplay formatting, motion picture making description, and video contents description for indexing and retrieval where the video indexing subsystem is operated based on MPEG-7 to take advantage of its metadata for the effective retrieval of video data.

Beside dialogue, other audio modalities such as music and sounds have not been added into experiment now. Another work needed to do is that the examples we used for demo are still simple. Along with growth of the complexity and calculating quantity, more problems will appear. Next step, the effective and flexibility of XML structure design will be examined and refined by expected multi-modal moviemaking process resulted from using more complex rule-based reasoning.

Reference

[1] SHEN Jinhong, Seiya MIYAZAKI, Terumasa AOKI, Hiroshi YASUDA, “Filmmaking Production System with Rule-based Reasoning”, Image and Vision Computing New Zealand (IVCNZ 2003), Palmerston North, New Zealand, Nov. 26-28, 2003

[2] G. Ahanger and T.D.C. Little, "Automatic Digital Video Production Concepts," Handbook on Internet and Multimedia Systems and Applications, CRC Press, Boca Raton, FL., December 1998.

[3] [Ricoh, 02] Ricoh: “Ricoh MovieTool Home”, June 2002, http://www.ricoh.co.jp/src/multimedia/MovieTool/

[4] H.J. Zhang, John Y. A. Wang, and Yucel Altunbasak. “Content-based video retrieval and compression: A unified solution”, In Proc. IEEE Int. Conf. on Image Proc., 1997.

[5] Salwa, Video Annotation: the role of specialist text. PhD Dissertation, Dept. of Computing, University of Surrey, 1999

[6] Smith, Manjunath, Day, ICCE 2001 MPEG-7 Tutorial Session, 6/17/2001

[7] MPEG 7 Main Page http://www.darmstadt.gmd.de/mobile/MPEG7/