ABSTRACT

This paper proposes the Metadata Production Framework (MPF) as a common platform for generating content-based metadata. A lot of research on extracting useful information from audiovisual content has been conducted. Recently it has been necessary to integrate these research studies to get much higher semantic metadata. The aim of MPF is to provide an environment where we can easily make these types of processes, including multimodal integration. We have also released a Metadata Editor where a user can generate content-based metadata both manually and by using automatic metadata extraction modules based on MPF.

Categories and Subject Descriptors
H.4.0 [Information Systems Applications]: General

General Terms
Management, Design, Standardization.

Keywords
Metadata production, framework, MPEG-7, metadata editor, semantic metadata, multimodal integration, plug-in module.

1. INTRODUCTION

Various situations in recent years have greatly increased the importance of metadata. As for broadcasting, it is expected that the TV-Anytime [1] like service will be launched in Japan in the next few years. Content-based metadata plays a key role in providing user specific scenes of TV programs in this service. We broadcasters have the responsibility of generating accurate substantial metadata for our programs because the quality of the service depends on the quality of the metadata. Our target metadata is called ‘segment metadata’, which consists of mainly time information specifying the section of a video content, and the semantic information for the section. Currently, we broadcasters manually generate the metadata to maintain its completeness, but for a long time there has been sincere hope for an automation of these tasks. Recently multimodal processing has become popular for this purpose because using several media analyses can complement each other and increase accuracy. We have also tried the approach presented by M. Sano et al [2]. Although many analyses have been proposed so far, they haven’t put importance on the way to easily exchange their technologies and outputs. To achieve an environment where we can co-operate with one another, we need a certain framework. Therefore, we have defined MPF and implemented Metadata Editor as one of our references.

2. OVERVIEW OF MPF

The concept of MPF is illustrated in Figure 1. The constituents are a metadata server and various modules, each of which packages a specific process. The basic idea is that metadata manipulations, such as generation, revision, conversion, and utilization, are provided as modules, and we can combine them to generate desired metadata using the metadata server. MPF specifies only the grey parts, i.e. the metadata model and two interfaces.

We used the common metadata model MPEG-7 [3] in our research. It is so large and has so many complicated specifications that we had to make a compact-sized subset to describe the general information on the audiovisual content and segment metadata. We made it based on the Core Description Profile (CDP) in MPEG-7 Part 9.

The interfaces specified in MPF are for controlling a module from the outside (IF-1), and for handling the metadata with the metadata server (IF-2). Table 1 lists the main items of each of them.

Using the outlined specifications, we can develop various modules, packaging together certain types of technologies and exchange them, combining them without difficulty for various situations. The greatest merit of using a module is that one doesn’t need to and cannot look inside the module. It is important for industries to keep their own technologies hidden and leave the possibility for new business developments.

Copyright is held by the author/owner(s).

MM'06, October 23-7, 2006, Santa Barbara, California, USA.
ACM 1-59593-447-2/06/0010.

3. METADATA EDITOR

To put the MPF into shape, we have developed a metadata editor (Fig. 2). The metadata editor provides us with the functionality of both manual generation of segment metadata and automatic generation by using modules implemented with MPF interfaces. The metadata editor works on Microsoft .NET framework 1.1 and the modules are provided as Windows Dynamic Linking Library (DLL) files. Metadata Editor is available as a freeware with some sample modules [4]. There are also some source codes from sample modules that can help you to develop your own module. The following is an explanation of some of the functions of our Metadata Editor.

- A track and segment metadata
  A track is a kind of holder for segment metadata. Modules are assigned to the track as a unit. The track appears in the lower part of the editor as a horizontally long rectangle and segment metadata is represented as a small rectangle in the track. The horizontal direction of the track indicates a time line so you can easily grasp how much segment metadata is in which duration and the time line is scalable.

- Extract information separately for segment metadata
  Some of modules can input the information generated by other modules, i.e., cascading. You can extract a section firstly by a module, and then use another module to extract semantic meaning from the section and embed them to the section. The function allows you flexible ways to generate the metadata.

- Comparison
  Tracks and modules provide you with a good comparison method. If you have several modules generated by different companies for the same task, i.e., face-detection, you can create tracks and assign modules to tracks one by one, and you can easily check how each module outputs the segment metadata along the time line. Another comparison is to assign a module to several tracks with different parameters. In this case, you can easily grasp the parameters’ effects and decide on the best set of parameters.

- Interim segment metadata
  To generate complicated segment metadata, you may create several tracks for each module. Although it is sufficient enough to save only the final target metadata, you can also save all the interim tracks as they are outputted. By saving them, you can restart the process from any middle point. This makes it easier for other people who are interested in the process.

- Editing segment metadata
  You can edit the segment metadata in the track and tree-view window. When you click a segment metadata in a track, it changes color and its semantic information appears as a tree-view that represents a MPEG-7 format. You can directly change it. You can also watch the corresponding video section quickly for confirmation.

- Input/Output metadata as a MPEG-7 file
  You can save the generated data as a file and load it into the editor. The editor holds an MPEG-7 schema, so an automatic check if the data is conformant to MPEG-7 is done.

- Visualization of features as a graph
  Our metadata editor provides another type of track: a graph drawing tracks for research purposes. When we analyze things, we focus on some specific features and their visualization helps us to decide on the further process that might be necessary. If a module is made with a graphical interface, we can see how the features vary along the time line in a graph format.

You can find that various processes for metadata generation can be executed by assigning modules to the tracks by checking the functionality of the metadata editor.

4. CONCLUSIONS

We propose the Metadata Production Framework (MPF) as a common platform for generating segment metadata, whereby people involved with this task can co-operate with each other by exchanging their own technologies. In this framework, if only the module follows the specified interfaces, it is completely free for developers if they keep their technologies hidden like a “black box”, or give out all their specific implementation as an open source. We also developed a metadata editor that can be used as a common tool for MPF. It helps us to understand that MPF provides enough functionality as a platform and has enough flexibility and extensibility to make good use of up-to-date technology. We offer the MPF framework in the hope that it will contribute to the research and development of this field, and we would greatly welcome any feedback and discussion about the current version of MPF and any future directions that we might need to or should consider.

5. REFERENCES

[1] ETSI TS 102 822: “Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems”