# Automatic Generating Detail-on-Demand Hypervideo Using MPEG-7 and SMIL

Tina T. Zhou
School of Design, Communication & IT
The University of Newcastle
NSW 2308, AUSTRALIA
61 2 6125 8180
Tina.Zhou@cs.anu.edu.au

Tom Gedeon
Department of Computer Science
The Australian National University
Canberra 0200, AUSTRALIA
61 2 6125 1052
Tom.Gedeon@cs.anu.edu.au

Jess S. Jin
School of Design, Communication & IT
The University of Newcastle
NSW 2308, AUSTRALIA
61 2 4921 7912
Jesse.Jin@newcastle.edu.au

## ABSTRACT
Detail-on-demand hypervideo will provide a powerful mechanism to allow viewers to see additional information of video segments through hyperlinks. A large number of tools are devoted to the identification of selectable video objects and the synchronization mechanisms for linking additional information to selectable video objects. We focus here on the automatic generation of additional information and the integration of the additional information to its corresponding selectable video object. We demonstrate a method using the Multimedia Content Description Interface defined in MPEG-7 and the Synchronized Multimedia Integration Language (SMIL) to automatically generate detail-on-demand hypervideos.

## Categories and Subject Descriptors
I.7.2 [**Document Preparation**]: Hypermedia, multi/mixed media, languages and systems.

## General Terms
Design, Experimentation

## Keywords
detail-on-demand hypervideo authoring, MPEG-7, and SMIL.

## 1. INTRODUCTION
Detail-on-demand hypervideo is defined as a video where viewers who require additional information about a given section of video are able to view this information through hyperlinks. Authoring detail-on-demand hypervideos relies on three basic elements:

- Digital video contents
- Definition of selectable video objects
- Integration of additional information


*MM'05*, November 6–11, 2005, Singapore.


Digital video contents are available in various video formats. However, it is often difficult to identify a selectable video object and to find a powerful hyperlink and synchronization mechanism to link it to the corresponding additional information. For this purpose, a large number of tools have been reported in the literature. Less attention has been devoted to automatic generation of additional information and integration of the additional information with the corresponding selectable video object.

MPEG-7 [1], as an ISO/IEC standard developed by MPEG (Moving Picture Experts Group), offers a comprehensive set of audiovisual Description Tools in the form of Descriptors (Ds) and Description Schemes (DSs) to create descriptions of video content. The descriptions cover three aspects of video content: structural aspects, contextual aspects and conceptual aspects. We find the structural aspects descriptions can be used to define selectable video objects, and the conceptual and contextual aspects descriptions can be used to generate the additional information of video structural objects. Therefore, we propose a system for automatic authoring detail-on-demand hypervideos based on MPEG-7 schema. We choose Synchronized Multimedia Integration Language (SMIL) [2] as rendering language for multimedia presentation to deliver the final detail-on-demand hypervideos to viewers.

The rest of the paper is organized as follows: Section 2 briefly introduces the two essential technologies of our system: MPEG-7 and SMIL. Along with the description of them, the basic ideas and design considerations are also presented in this section. Section 3 presents the architecture of our system. Sections 4 and 5 describe two aspects of our system in details. Section 6 summarizes our work and gives an outlook of future work.

## 2. MPEG-7 and SMIL
In MPEG-7, five types of DSs are defined according to their functionalities [3]: 1) the creation and production DS, 2) the media DS, 3) the usage DS, 4) the structural DS and 5) the conceptual DS. However, for our purposes, we group the first three DSs as contextual DSs since they address primarily information related to the management of the content. The structural DS is used to describe physical and logical structure of audio/visual (AV) content. The conceptual DS is used to describe the conceptual notions of AV content. Currently this kind of DS is still under development and often is related to *TextAnnotation* DS.

For each MPEG-7 document, we could form a document tree. According to three types of MPEG-7 DSs, we classify the document tree nodes into three categories as well. They are: 1) structural nodes, 2) contextual nodes, and 3) conceptual nodes. The relationships among nodes are represented by arches of tree. Figure 1 shows an example of MPEG-7 document trees.

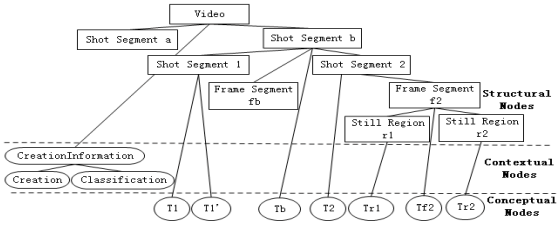


**Figure 1: Example of MPEG-7 document tree**

The basic idea of our system is to use the structural nodes as selectable video objects and use the semantic information contained in the contextual nodes and the conceptual nodes as the additional information of video contents.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
       xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 Mpeg7-2001.xsd">
 <Description xsi:type="ContentEntityType">
  <MultimediaContent xsi:type="VideoType">
   <Video>
    <TemporalDecomposition>
     <VideoSegment>
      <TextAnnotation>                                   Additional Info.
       <FreeTextAnnotation>http://www.google.com</FreeTextAnnotation>
      </TextAnnotation>          Shot Beginning
      <MediaTime>
      <MediaTimePoint>T00:00:00:0F25</MediaTimePoint>
     <MediaIncrDuration mediaTimeUnit="PT1N25F"> 100</MediaIncrDuration>
      </MediaTime>                                       Shot Duration
      <TemporalDecomposition>
       <VideoSegment>
        <MediaTime>
         <MediaTimePoint>T00:00:03:5F25</MediaTimePoint>
        </MediaTime>
       </VideoSegment>
      </TemporalDecomposition>
     </VideoSegment>
          . . .
    </TemporalDecomposition>
   </Video>
  </MultimediaContent>
 </Description> </Mpeg7>
```

**Figure 2: Example of MPEG-7**

In MPEG-7, it is expected that most descriptions corresponding to low-level features (e.g., color, texture, motion, etc.) will be instantiated by automatic analysis tools whereas human interaction will be used for semantic descriptions (i.e., information contained in the contextual DSs and the conceptual DSs). Many current MPEG-7 video annotation tools (e.g., IBM VideoAnnEx annotation tool [4], Ricoh MovieTool [5], Olivier Steiger [6], etc.), are based on this principle. They automatically create the video structural DSs, and leave the contextual DSs and the conceptual DSs for users to add later on. In order to enhance flexibility, our system allows users to choose any of these tools to assist the creation of MPEG-7 descriptions, in which the desired additional information could be associated with the specified video object.

Figure 2 shows a simple MPEG-7 example description created by IBM VideoAnnEx [3]. VideoAnnEx automatically segments an input MPEG video stream into shots (specified by *VideoSegment* DS), and uses *MediaTimePoint* D and *MediaIncrDuration* D to specify the automatically-generated time information of each shot or key frame. Human interaction, subsequently, is used to create semantic information of shots or spatial regions of key frames. These semantic descriptions are described by *FreeTextAnnotation* D, and could act as the additional information that is exactly what a detail-on-demand hypervideo needs.

Having the selectable objects and additional information specified by MPEG-7, we use Synchronized Multimedia Integration Language (SMIL) to create detail-on-demand hypervideo presentation since it is well supported by many well developed media tools (e.g., AMBULANT/X [7], Internet Explorer [8], RealOne Platform [9], etc.). Its rendering constructors satisfy our requirements of detail-on-demand hypvervideos, where only one link is available at any given time and the additional information is displayed in a window different from the display window of video content.

In our system, *anchor* element of SMIL is used to define the selectable video shot object and the *href* attribute of *anchor* element specifies the location of shot's additional information. An example of a SMIL document for rendering the video along with linking opportunities is shown in Figure 3. It is also the corresponding transformation result of the MPEG-7 example shown in Figure 2. The value of *end* attribute of *anchor* element is calculated based on the values of *MediaTimePoint* D and *MediaIncrDuration* D of MEPG-7.

```
<SMIL>.                                  Shot Beginning
          . . .
<video region="region_video"     src="face_dance.mpg">          Shot End
   <anchor begin="00:00:00" end="00:00:04"    show="new"
           href="http://www.google.com" id="vAnchor0" />
          . . .                                  Additional Info.
</video>
</SMIL>
```

**Figure 3: Example of SMIL document**

## 3. SYSTEM ARCHITECTURE

As illustrated in Figure 4, the design framework for our detail-on-demand hypervideo authoring system is a three-tier architecture of server, middleware and client.

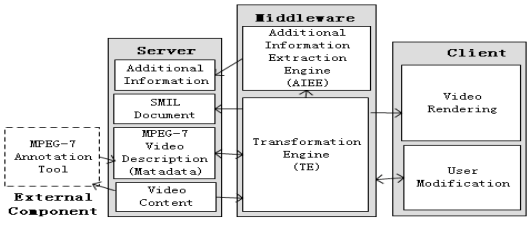


**Figure 4: System Architecture**

The server in our system is a file system which stores digital video contents, MPEG-7 descriptions, additional video information, and generated SMIL documents. Digital video contents are original video streams which form the basis of detail-on-demand hypervideos. The MPEG-7 descriptions are generated with the assistance of the current available annotation tools that we have

discussed in Section 2. The additional video information, such as video frame images and html files, are extracted from MPEG-7 descriptions and stored in the server with a consistency to the video structure. The SMIL documents are the detail-on-demand hypervideo presentations resulting from the transformation process of MPEG-7 descriptions. They are stored in the server for later retrieval and display.

The middleware consists of a transformation engine (TE) and an additional information extraction engine (AIEE). The TE analyses the MPEG-7 description and creates the document tree to represent all the information specified in MPEG-7 descriptions. To solve the problem of gaps and overlaps among structural nodes of a tree, and to ensure there is no conceptual information missing for a particular segment of video, the TE applies a simplification process (described in Section 4) on the document tree. The AIEE extracts additional video information from the simplified document tree and stores the information in the server. More detail about AIEE is given in Section 5. After the additional information has been created, the TE constructs a SMIL document based on the structural nodes of the simplified document tree and their additional information. The middleware also handles the modification of MPEG-7 descriptions or SMIL presentations.

**Figure 5: MPEG-7 to SMIL converter**


**Figure 6: An example detail-on-demand hypervideo playing in RealPlayer**

In the client end of our system, users can make a modification on the original MPEG-7 descriptions or the transformed results – SMIL documents to customize the detail-on-demand hypervideo presentations. The interface for modification is shown in Figure 5. The display client could be any SMIL player as we mentioned in Section 2. Figure 6 shows an example detail-on-demand hypervideo playing in RealPlayer. In this example, the digital video content is arranged on the left media player window, while the additional information related to the clicked video shot object is presented on the right web browser.

## 4. DOCUMENT TREE SIMPLIFICATION

In our definition of detail-on-demand hypervideo, only one linkage opportunity to its addition information is available for an identical video segment at a given time. However, in MPEG-7, structural DSs are arranged in a hierarchical tree structure where a structural DS may be subdivided into sub-segments and there may be gaps or overlaps among the sub-segments. To avoid the association of duplicated linkage opportunities and the lack of linkage opportunities to the selectable video segment, we need to simplify the complex document tree to solve the problem of gaps and overlaps.

We identify that the *VideoSegment* DS with a *Duration* D is the description of a shot segment. We retrieve all the shot segment nodes. We check if the shot segment node has a parent node, also of the shot segment type. If it has not, we simply keep it at its original position. Otherwise, we process it.

We define four types of segment temporal decomposition relationships according to the natural characteristics of video. They are: 1) segment decomposition without gap or overlap; 2) segment decomposition with gaps; 3) segment decomposition with overlap; and 4) segment decomposition with absence. In order to display as much as possible related information on the video shot, we develop a strategy to handle these four types of segment temporal decompositions. The strategy is visually illustrated in Figure 7. Also, we give a detailed explanation as follows:

1. *Segment decomposition without gap or overlap*. With this type of segment decomposition, we simply copy all the conceptual nodes of the parent node to all its children, and then delete the parent node.

2. *Segment decomposition with gaps*. If there is a gap between two segments, we create a new segment to cover the gap. So, the parent node now has one more child node. We copy the parent node's conceptual nodes to all its child nodes, and then delete the parent node.

3. *Segment decomposition with overlap*. If there is an overlap between two child segments, we deduct the overlapped area from the two child segments and create a new child segment to cover the overlapped area. We copy both old child nodes' conceptual nodes to the new child node. We also copy the parent node's conceptual nodes to all its child nodes regardless whether they are new or old, and then we delete the parent node.

4. *Segment decomposition with absence*. If part of a parent segment is not covered by the child nodes, we create a new child segment for this part. We then copy all the conceptual nodes of the parent node to all its child nodes including the new one, and finally we delete the parent node.

In the spatial domain, for example, the situation where a still region of a frame is subdivided into sub-regions, we need to simplify the tree as well. We apply a similar strategy as we applied to the temporal domain to make sure no information is missing during the simplification process.

However, a frame segment (*VideoSegment* DS with only a *TimePoint* D is the description of frame segment.), along with its still regions are treated as information on the corresponding shot in our system. So, in the simplification process of the document tree, a frame and its still regions have double characters: one character is the structural node which needs to be simplified;

another character is the conceptual node which needs to be passed on to video shot segments' child shot nodes.

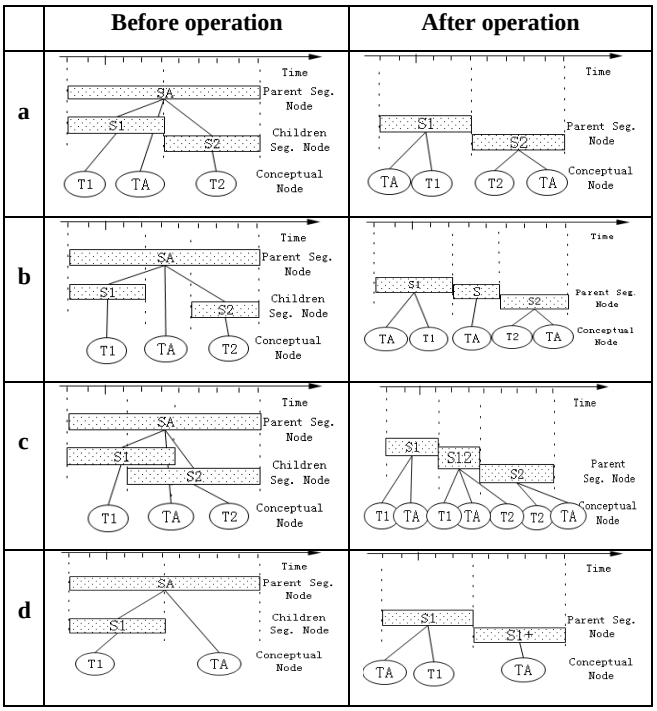


**Figure 7: Examples of Segment Decomposition trees and their simplified versions: a) Segment decomposition without gap or overlap; b) Segment decomposition with gaps; c) Segment decomposition with overlap; d) Segment decomposition with absence.**

After performing a recursive simplification process, no matter how complicated the document tree of MPEG-7, we will simplify it into a tree where there is only one layer of temporal or spatial decomposition. For example, the document tree shown in Figure 1 becomes the tree shown in Figure 8 after the simplification process.

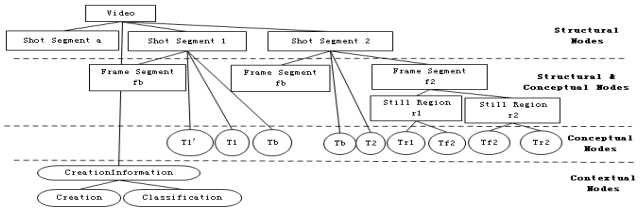


**Figure 8: Simplified tree of Figure 1.**

# 5. ADDITIONAL INFORMATION EXTRACTION ENGINE (AIEE)

AIEE is designed to store the additional information for each selectable video object. It stores information in the form of a HTML file in which the information could be organized nicely.

The AIEE stores all the information of contextual nodes in one file since this information is related to the whole video. We design a visible button on the left-bottom corner of the video display window for the user's convenience. If the user wants to see the contextual information of the playing video, he/she could click on the button at any given time.

After the transform engine has done the simplification process on the document tree, the AIEE checks every structural node (SN) in the simplified document tree and stores information in the conceptual nodes (CNs) that the SN links to. If there is a structural and conceptual node (SCN) (in our scenario defined in Figure 8, the SCN is a frame) linked to the SN, the AIEE also stores the SCN. If the SCN is decomposed into still regions or it has corresponding CNs, the SCN becomes a selectable object. If there is no decomposition of SCN, the AIEE stores corresponding information of its CNs, otherwise, the AIEE stores the conceptual information of each region as well, and hence, the user could click on the region of the video frame to get further information.

However, not all the information in CNs needs to be stored in the server. If there is only one CN related to SN, the AIEE checks if its content is a URL. If it is, the AIEE simply ignores it since the URL could be directly linked when user activates the linking opportunity. Otherwise, the AIEE creates HTML files to store the information.

# 6. CONCLUSION

In this paper we provide a possible use of a content description technique, i.e. to use MPEG-7 automatically generating additional information of video content, and hence, detail-on-demand hypervideos with the assistance of SMIL can be automatically generated and viewed. We have described the general system architecture. Experiments demonstrate that features commonly used in general purpose content-based video information retrieval systems can be used for authoring detail-on-demand hypervideos.

In the future, we would like to integrate the generation of additional personalized information into our system. We would also like to explore a new and powerful technique to support the identification of moving objects so that the *MovingRegion* DS of MPEG-7 could act as selectable object in detail-on-demand hypervideo.

# 7. REFERENCES


[1] Martinez, J. M. *MPEG-7 Overview (version 9)*, ISO/IEC JTC1/SC29/WG11N5525, 2003.

[2] Hoschka, P. Bugaj, S. Bulterman, D. and et al. *Synchronized Multimedia Integration Language - W3C, Working Draft 2-February-98*, W3C, 1998

[3] ISO/IEC TC JTC1/SC 29/WG 11, *Information Technology – Multimedia Content Description Interface – Part 5: Multimedia Description Schemes*, 2000

[4] URL: http://www.research.ibm.com/VideoAnnEx/index.html

[5] URL: http://www.ricoh.co.jp/src/multimedia/MovieTool/

[6] URL: http://ltswww.epfl.ch/~steiger/software.shtml

[7] URL: http://www.cwi.nl/projects/Ambulant/distPlayer.html

[8] URL: http://www.microsoft.com/windows/ie/default.mspx

[9] URL: http://www.real.com