The Family Video Archive: An annotation and browsing environment for home movies

Gregory D. Abowd
College of Computing & GVU Center
Georgia Institute of Technology
801 Atlantic Drive
Atlanta, GA USA 30332-0280
+1-404-894-7512
abowd@cc.gatech.edu

Matthias Gauger & Andreas Lachenmann
Computer Science Department
University of Stuttgart
Universitaetsstr. 38
D-70569 Stuttgart, Germany
{Matthias.Gauger, Andreas.Lachenmann}@studi.informatik.uni-stuttgart.de

ABSTRACT
We present the Family Video Archive as a tool to give consumers the ability to annotate and browse large collections of informal family movies. The informal nature of home movies makes it difficult to use fully-automated techniques for scene detection and annotation. Our system explores the symbiosis between automated and manual techniques for annotation. We also explore the use of a zooming interaction paradigm for browsing and filtering large collections of video scenes.

Keywords
Video annotation, Zooming user interfaces, home movies.

1. INTRODUCTION
For many decades, families have recorded their history using film or video. With today’s advanced home video camera technology it is easy and convenient to record huge amounts of video data. There are even services to convert older formats of film into more modern digital formats. Commercial video editing products also allow for the creation of home videos that greatly increase the quality and presentation of home movies, increasing the value of home movies as family memorabilia. Decreasing storage costs and improved video compression techniques make it practical to consider storing decades of family movies on a computer.

As easy as it is to record and edit family movies, archiving and retrieving them is not quite as effortless. Unlike a stack of photos, a collection of home movies is not easily browsed. It is a wonderful gift to share with a loved one a compilation of recorded events of his or her life, but it is often a very arduous and time-consuming task to assemble the individually recorded snippets in order to prepare a cohesive presentation. Our goal in this work is to provide a means for archiving and annotating a large collection of home movies, turning the browsing and assembling of relevant subsets of the archive into a simple and enjoyable task. The Family Video Archive system presented in this paper supports the management, annotation, and browsing of a large collection of digitized home movies.

There is much work in the area of video annotation and browsing. What makes this work significant is its focus on the informal nature of home movies. This has several implications for the research contribution of our work.

• Home movies record informal activities and are usually of relatively low production quality, containing little, if any, relevant metadata concerning the content of the video. As a result, we have focussed on interaction techniques that provide full manual annotation capabilities but also leverage off automation. This symbiosis between fully manual and fully automated annotation is an important research topic.

• Home movies are meaningful to a community of family, friends and relatives. While complete strangers might enjoy browsing through the collection of videos, the real meaning of the videos is idiosyncratic to the family. A strictly defined and universally applied annotation schema would not be appropriate across different families. In our solution, we allow a single user to develop a complete hierarchy of metadata to associate to video scenes. The metadata is stored internally using XML and is strongly influenced by the MPEG-7 standard [10]. We explore the power and flexibility of this annotation hierarchy.

• We are exploring the use of a zooming user interface paradigm for browsing and searching a large database of video scenes. In this work, we are showing how a zooming interface can be used to provide semantic organization of related video scenes and provide an intuitive method to search through a large set of scenes and filter out the ones of interest.

Overview
We begin in Section 2 with a discussion of some relevant related research that influenced our own efforts. The basic idea behind the Family Video Archive is to support browsing and searching through a large collection of annotated scene’s of a family’s history. The system has two main parts: an annotation interface and a searching interface. In Section 3, we describe the annotation interface for organizing collections of videos, segmenting video into meaningful scenes and assigning descriptive metadata to scenes. The key to annotation is a user-defined metadata “tag” hierarchy, as well as techniques for more free-form and automated annotation. In Section 4, we describe the zoomable browsing interface for filtering through a large collection of videos. This semantic browsing is driven by the user-defined tag
hierarchy. In Section 5, we give some details on the implementation of the Family Video Archive system. In Section 6, we explore some of the relevant design issues that influenced the evolution of the Family Video Archive. In Section 7, we conclude with some interesting directions for this work.

2. RELATED WORK

The proliferation of digital artefacts, such as photographs and videos, has motivated many researchers and companies to address the general problem of collecting, organizing and browsing memories. Much work has been done for digital photographs, including automated methods for organizing based on feature analysis of the images [11], visualization strategies [1, 6] and annotation techniques similar to the drag-and-drop annotation we will present here [9]. Zooming user interfaces, or ZUIs, have been popular in the HCI and Information Visualization communities since the late 1980’s. Bedersen’s PhotMesa system demonstrated a compelling use of a ZUI for organizing and browsing a large collection of digital still photographs, and that work has influenced our own attempts to browse video scenes [1].

For video, automated techniques abound for being able to do keyframe extraction and scene boundary detection, and many of these assume high-quality, structured video and/or textual metadata such as a time-synchronized transcript [5, 14, 15]. Archives of television broadcasts are scripted, higher quality productions, often synchronized with closed-caption text. The better quality video and metadata make it possible to use automated techniques to process and analyze the video for information like scene boundaries and topic. Some very useful research (e.g. CMU’s Informedia Project [15]) and commercial (e.g., the VideoLogger from Virage [14]) tools show the promise of tools to browse and search these digital video libraries.

Video annotation environments have been developed for detailed analysis of audiovisual data. Researchers in linguistics and film studies perform fine-grained analysis of audiovisual data and need tools that allow for precise alignment of audio, video and descriptive text. Tools such as Anvil [8] and IBM’s VideoAnnEx [4] support this kind of analysis. While our work is similar in spirit to these systems, the focus on home movies, emphasizes simplified browsing over hours of content and selection of video scenes over fine-grained analysis of any one scene.

We are not directly interested in the problem of video editing. Ultimately, the task of filtering through a video collection to find the precious and relevant scenes is a precursor to authoring some distributable artefact, like a DVD. Several research (e.g., CMU’s SILVER [2], FX-PAL’s Hitchcock system [3]) and commercial (e.g., Apple Computer’s iMovie, iDVD and FinalCut Pro or Adobe’s Premier or Encore) systems address the problem of editing from a collection of video segments into a single movie. The SILVER system is explicit in its use of both a zooming metaphor and also metadata to simplify editing but does not address the annotation problem to assign metadata. Some more recent research suggests interesting interaction techniques for handling digital video [12]. We hope that a system like the Family Video Archive will work in conjunction with these efforts to support nonlinear digital video editing.

Much of the motivation for supporting the collection of family memories stem from our own prior work on the Living Memory Box [13] In this project, we have conducted focus group studies to identify why and how families want to collect and organize all forms of family memorabilia, including physical and digital artefacts. What emerged most strongly in those studies was that the notion of shared but personalized annotation is very important. We built the Family Video Archive from the perspective of a single individual, and the flexibility of the tag hierarchy makes it inherently a personalized annotation system. However, it is clear that the real value of the system is what it provides to a community, like a family. We must further investigate how to support the collaborative nature of annotation while preserving the personalization.

3. THE ANNTOATION INTERFACE

Figure 1 is a screenshot of the annotation interface for the Family Video Archive. The system is written using the Java Media Framework and currently supports the Microsoft Windows Media (AVI) format. Video files can be imported into the system by dragging them into the video files browser panel in the lower left-hand corner of the interface. A video file is viewed in the video playback panel, providing the usual VCR-like controls. A timeline indicates where user-defined scene boundaries occur. We will later discuss how scene detection is aided through a semi-automated process. Once a scene has been defined, it can be manipulated in the system separate from the video file in which it resides. For example, a collection of scenes on a common theme (e.g. family reunions) can be collected into an album and placed in the album browser panel in the top left corner of Figure 1.

This interface mainly supports the annotation of the currently viewed scene. The upper middle panel of Figure one shows all metadata associated with the current scene. There are three kinds of annotations possible — date, freeform text, and a metadata tag.

We can associate a time to a scene. Though many camcorders do record timestamps, most capture software does not access that information, and older forms of film, such as 8mm, popular from the 1950s until the early 1980s, do not contain timing information. We therefore wanted to provide a quick and flexible way to add time information. After several iterations in the design, we created a pop-up date widget (not shown in Figure 1) to allow the user to optionally set the year, season, month and day in one relatively smooth interaction. Because most scenes in a video file will be near in recording time to the previous scene, we also provided a button to assign the date of the previous scene.

Earlier prototypes of the Family Video Archive were shown to potential users. In group settings, most users preferred to talk aloud (often simultaneously) about the contents of a video scene. One person would write down the information associated to the scene. In this way, the verbal annotation with scribing could occur in near real-time. Motivated by this observation, we created a free-form text window into which arbitrary text is typed. The first line of the text window becomes the title for the scene. As we will see later, this free-form text is used to suggest other, more structured annotations to add to the scene.
Motivated by annotation schemes like MPEG-7 [10], we wanted to provide a way to assign structured tags to a given scene. Tags are the most important concept for adding metadata information to video scenes. A tag is a piece of information describing one part of the content of a scene. Tags drive the visualization interface described later. Once created, tags are arranged in a hierarchy, shown on the right of Figure 1, and can be dragged over a video scene to assign it to that scene. Currently assigned tags are shown in the panel just above the video in Figure 1. By default, a tag is associated to the full extent of a video scene, but the user can adjust that extent manually. Because of their importance in the overall system functionality, we will describe the tag hierarchy more completely.

3.1 The tag hierarchy
The user has total freedom in creating the tag hierarchy. No special meaning is assigned by the system to any tag or portion of the tag hierarchy. The interpretation is entirely user-driven. In our initial example archive, 3 top-level categories were defined — “Who”, “Where”, and “Event”— and these were further subdivided into relevant subcategories (e.g., “Relatives,” “My Family,” and “Friends” for the “Who” category). There is no restriction to the depth and breadth of the tag hierarchy. Starting with only one top-level category the user can add more and more structure by subdividing categories into more specialized subcategories.

In our use of the system, the tag hierarchy has grown and undergone major modifications in structure, and we expect this would happen with any prolonged use of the Family Video Archive. The system supports this evolution of the tag hierarchy by allowing any modification of the hierarchy at any time during annotation. The only restriction is that a tag cannot be removed from the hierarchy if it is currently used as an annotation for any video scene.

Any classification scheme can suffer from the problem of not having disjoint categories. In developing the Family Video Archive’s tag hierarchy, we ran into this problem. We created a subcategory of “Where” containing homes where many family scenes were shot. We also created a subcategory of important locations, and subdivided that by state and city. A home should be categorized into its relevant city. To solve this cross-categorization problem, we created tag aliases that could be placed anywhere in the hierarchy and pointed to any other tag in the hierarchy. Now, when a scene is labelled with a tag from the home category, it is implicitly tagged with a subcategory of the appropriate city. Each video scene that is tagged with the referenced category is internally also marked as tagged with any alias associated with that category, and vice versa.

Figure 1: The annotation interface. The lead author is pictured on the far right in the video scene. He is a little older now.
It is very easy to create a very large number of tags, and manipulating the tag hierarchy becomes a serious challenge. One simplification we provided was the ability to parameterize a tag with another category. For example, it is common to film a birthday event. Should the tag associated to the scene simply be “birthday” or should it be “Gregory’s birthday”? The latter is more descriptive, but it requires a different tag for every individual’s birthday. Instead, when the birthday tag is created, it declares another tag category as a parameter (see Figure 2). Whenever an instance of the tag is assigned to a video scene, the user is prompted to provide the appropriate tag parameter. This results in a new tag being created (e.g. “Birthday Gregory”) as well as an alias tag under the parameter category (e.g. “Gregory”).

3.2 Automating annotation

As the tag hierarchy grows, it becomes increasingly difficult to navigate. Frustration mounts as a video scene is playing and the user struggles to locate the tag representing the distant relative in the scene. While we might hope for a future in which video analysis works well enough to automatically assign metadata to video, we are nowhere close to being able to do that today on the quality of typical home videos. However, simple yet clever automated techniques can speed up the annotation process.

The “Who” category resembles a family tree. In many family video scenes, whole families will be present. To facilitate this, we defined a modified drag event (holding the control key down while dragging a tag icon) that tags the current scene with all tags one level below the dragged tag. If all members of a family are underneath one category (e.g. “My Family”), then they can all be assigned to one scene in one simple action. When dealing with older videos, it is possible that not all members of a family were born during the current scene. A tag can be assigned valid begin and end dates, and we constrain the system so that only valid tags can be assigned to any scene. For tags in the “Who” category that represent people, we use the begin date to represent the birth of that individual and the end date for the death. Other tag categories would use the valid begin/end dates to represent something else (e.g. when a house was occupied).

As we described above, it is desirable to allow the ease of free-form annotation. We run a matching algorithm over the free-form text and provide a ranked list of existing tags that might be associated to the current scene. This window (shown in the middle of the left-hand side in Figure 1) will contain far fewer tags than the tag window, and if our matching algorithm works well, its suggestions near the top of the window are most likely appropriate to assign to the scene. We also include tags from the previous scene, because it is often the case that there is similarity between consecutive scenes. In a similar way, the free-form text is used to suggest new tags that can be added to the existing tag hierarchy (see the bottom right panel in Figure 1). Leveraging off the free-form text in this way simplifies some of the annotation task.

3.3 Defining scene boundaries

The first step in annotating a video file is to define the scene boundaries which divide the videos into meaningful segments. This scene detection is ultimately a very user-centered decision. In our initial experience with the system, this task took a great amount of time. As a result, we wanted to use some automated techniques to speed the process up without sacrificing any user control. Using a straightforward frame-by-frame color histogram comparison algorithm, we created an automated scene detection service, shown in Figure 3. The user sets a threshold value with a slider and that determines the sensitivity of the scene detection algorithm. Once satisfied with the threshold setting, the user manually adds/removes/adjusts the scene boundaries. Dismissing this dialog box returns the user to the annotation interface, and the appropriate metadata can be assigned to the scenes. This scene detection service represents a good compromise between techniques that are fully automated, and error-prone, and those that are fully manual, and time-consuming.

In the long run, we expect the problem of defining scene boundaries by hand to disappear. Current digital video cameras...
deliver reliable information about shot boundaries based on the times when camera recording is turned on and off. This information is exploited in Apple’s iMovie software. The user will then only have to delete some of the scene boundaries to combine successive shots that belong to the same event and should be tagged together.

4. THE SEARCH INTERFACE
Inspired by Bedersen’s PhotoMesa, a smoothly zooming interface on a large repository of digital still photographs [1], we wanted to explore how zooming could help to browse and search for related video scenes. Video scenes are represented by single keyframes. Currently the keyframes are arbitrarily chosen as the middle frame in the video scene or can be set explicitly by the user in the annotation interface. The search interface is shown in Figure 4.

4.1 Semantic zooming and filtering
An important characteristic of the design of the search interface is that it allows searching by browsing. The video scene keyframes give the user an overview of the result of his current search that is dynamically updated. The Family Video Archive demonstrates a semantic zooming capability, meaning that the effect of zooming in and out is determined by the meaning of the underlying data structure being browsed. In the zoomable scene browser panel, the visualization at every level of zooming is defined by the tag hierarchy. At the very top (fully zoomed out view), all video scenes are accessible and are divided into the number of first-level branches in the tag hierarchy. The space allocated for each region is proportional to the number of scenes containing at least one tag from that branch of the hierarchy. Note that a scene can appear in more than one region because a scene can have any number of tags assigned to it. Hovering over any video scene reveals a pop-up window of the video scene. The video can be played within this pop-up window and the metadata can be browsed, but not changed.

A left mouse click within any region zooms (with a relatively smooth animation) into that portion of the tag hierarchy, dividing and proportioning the screen according to the number of branches in the tag hierarchy. For example, in Figure 4, we have zoomed three times into the “Event” branch of the hierarchy (first down the Event branch, then the Holidays branch and then the “Christmas” branch.). A right mouse click zooms back up the tag hierarchy. The trail of the zooming is shown visually in the Query Window. Large icons in the Query Window (the “Christmas” icon in Figure 4) represent current filters on the set of video scenes. Small icons represent the zooming history. Clicking on the “Refine search” button moves the visualization back up to the top of the hierarchy (zoomed out), keeping the current filtering. Clicking on the “Reset search” button clears all filters and zooms back up to the top of the hierarchy.

Besides zooming in and out, the search interface contains several other search mechanisms based on the hierarchy of tags that help the user in retrieving the desired result set. The system analyzes the scenes of the current result set and calculates a set of tags that are most common to these scenes. These most common tags are then displayed in the search interface and can be added to the search criteria with a simple drag and drop operation to the query window. The intent of the most common tags is to provide inferences to the user that wouldn’t become obvious otherwise.

A category tree instance showing the tag hierarchy helps the user keep an overview of the tag hierarchy. It also allows dragging tag elements to the query window, adding them to the set of search criteria. This is particularly useful for quickly adding known
search criteria before starting to browse the remaining video scenes in the scene browser.

If the user recognizes that one scene fits his current search especially well, he can drag the keyframe of the scene from the scene browser to the query window and add its tags to the search criteria so that only scenes similar to this scene are retained in the search result. The user can also search on the free-form text field by typing into the textbox to the right of the zooming history path.

4.2 Temporal zooming and filtering
Temporal filtering can be performed using the zoomable timeline filter at the bottom of the interface. This timeline shows a barchart of the distribution of currently filtered video scenes. This information is supposed to give the user a good impression of the temporal context of his current search. Hopefully, the user can occasionally infer useful information, for example by finding out that all scenes he was looking for took place in a special time period.

Like the scene browser, the timeline is displayed as a zoomable interface. From the timeline view showing the distribution of video scenes over years, the user can zoom into a view showing the distribution of scenes over months and with a further zooming step to a view visualizing the distribution of video scenes over days.

The red bar underneath the timeline is a temporal filter. It highlights the time intervals that are shown in the zoomable scene browser panel. This filter bar can be split into arbitrary segments and can be simply set to filter on regular temporal intervals. For example, the user can select a three month interval in a zoomed in view of the timeline and then request that that interval be repeated over every year.

4.3 Creating albums
Once the user is satisfied with his search result, he can store it in an album which then contains the scenes of the result set in chronological order. Albums are stored permanently in a data structure displayed in a separate window of the annotation interface (see Figure 1). Once an album is created, the user can reorganize or delete scenes within the album. The whole album can be played back as consecutive scenes, regardless of the video files in which they reside. In that sense, the album behaves like a single video file. Indeed, the system allows the user to export the whole album as a single AVI file to be played outside of the system. This provides a simple way to share the results of a search with others not owning the Family Video Archive. There is even an option to export the album as separate files to be imported into a third-party video editing system.

5. IMPLEMENTATION
The current Family Video Archive system has been written entirely in Java using the Java Media Framework. The zooming user interface is implemented using the open-source Piccolo toolkit developed at the University of Maryland (available at http://www.cs.umd.edu/hcil/jazz/). The system has been developed over a period of six months and has undergone much iteration in that time period as well as extensive use by a now expert user with input from his family.

The initial case study used to develop the tag hierarchy and test the zooming interface consists of two separate home movie archives. The first archive is a collection of 8mm family movies covering the time period of 1955–1984, split into 283 video files and covering roughly 23 hours of playing time. The second archive, covering the period of 2000–2002 is a collection of miniDV camcorder movies, split into 25 video files totalling two hours in length. The 308 video files have been further subdivided into 912 scenes, 577 of which have been annotated. At the time of writing, more than half of the 8mm and miniDV archives have yet to be divided into scenes and annotated, so we expect to have between 1500–2000 scenes when annotation is complete.

The particular family that is the subject of this initial case study is unusually large. It covers four generations and over 100 relatives. Currently, 371 tag categories have been created and 95 aliases. The entire archive, including XML-encoded metadata and AVI files compressed using the Windows Media codec, requires a little over 18 GBytes of storage.

6. DESIGN ISSUES
While video annotation environments have been around for awhile, the application to informal, unstructured home movies and the use of a zooming visualization are novel. It is important to reflect on some of the design issues, both to determine if they were appropriate for this application and if they have any merit in other annotation activities.

6.1 Tagging efficiency
The driving motivation for the annotation interface is to maximize the accuracy and completeness of annotation while minimizing user effort. Though we have not instrumented the system to achieve real quantitative results, our experience indicates that it takes roughly two times real time to annotate a scene. There is great variability in this estimate; longer scenes typically can be annotated in less time than the extent of the scene, while shorter scenes require a pause of the playback. We do not have any benchmark to compare against, but this annotation speed seems reasonable.

Several features of the system facilitate rapid annotation. First, the semi-automatic scene detection can save considerable time, but varies with the quality of the video recording, as one would expect. Leveraging off of the free-form text to suggest existing tags and new tags clearly saves time, and expert use tends to make extensive use of this feature for quickly accessing existing tags. We also introduced the ability to assign a tag to multiple scenes. Many times, in viewing some scenes a new piece of information is observed (perhaps a new person is identified or the location is clarified) and this information applies to several of the preceding scenes. Rather than having to load each of those scenes and add the new annotation, the collection of scenes can be highlighted in the video files or albums panel and the appropriate tagged can be applied to all of them. A similar feature to remove tags from a set of scenes would be desirable, but is not implemented.

Despite this speed-up for annotation, there is still plenty of room for improvement. With the current level of accuracy of the annotation, the whole annotation process should take less than real-time. Furthermore, it should be possible to perform more accurate annotation more simply. Not every person in a scene is accounted for through annotation, though the Where and Event information is usually fairly complete. Though the system supports the assignment of a tag to any arbitrary subset of a scene,
Once the “Need Input” category has been created and used to annotate some scenes, it cannot be removed from the visualization without removing all annotations. In this situation, it makes sense to allow for selective removal (and reintroduction) of parts of the tag hierarchy from the visualization. This could be done as a property of the tag itself, but would require a significant redesign of the visualization.

7. CONCLUSIONS
We have presented The Family Video Archive as a tool to annotate and search archives of home movies. Annotation consists of temporal, structured and free-form metadata, with emphasis on interaction techniques that improve the efficiency and accuracy of the annotation process. We exploit a zooming metaphor and use a natural means of browsing a hierarchical categorization of video scenes to filter down to a set of interest to the user. While the initial motivation for the Family Video Archive was to enhance the ability to share and reminisce over treasured family memories, our system can be extended and improved in a number of ways.

The value of an annotated video archive is clear from our own experience and that of others. While the interface provided by our current system does accelerate the annotation process, it is complex and time-consuming. Such an expert interface will only likely be appealing to video enthusiasts. We are currently working on ways to improve the initial usability of the system and to decrease the amount of effort required to do annotation. One important idea is to maximize the amount of annotation that can happen at the time of recording. Already, camcorders encode the time of a recording. It is reasonable to assume we will be able to mark a recording with location information and possibly even information about who is nearby. We are investigating ways to complement the capture of content with the capture of relevant context.

The search interface makes it relatively easy to isolate scenes of an individual in particular locations over interesting intervals of time. As such, it shows some promise as a means of tracking developmental trends of children, such as language and social interaction skills. With increased density of the recorded archive, it might also be possible to use the recorded record for more near-term applications, such as support for retracing one’s steps in an attempt to recover a misplaced item.

Finally, though the design of the Family Video Archive has been driven by a real application case study, we have not spent enough time understanding the needs of the wider population of potential users of such a system. Deeper consideration of the needs of both casual and devoted family movie archivists should be studied to inform the further evolution of this system.

8. ACKNOWLEDGEMENTS
Mr. Gauger and Mr. Lachenmann thank the Fulbright Foundation for its support of their visit to Georgia Tech, where this work was completed. The authors acknowledge the work of Rudinei Goularte (University of Sao Paolo at Sao Carlos) and Niels van Hese (University of Delft), other visiting students to Georgia Tech who worked on earlier versions of the Family Video Archive. This work was supported in part by the National Science Foundation (Grant ITR-0121661) and by the Aware Home Research Initiative at Georgia Tech. Dr. Abowd acknowledges his father, the late Richard G. Abowd, Jr., who created the 1955-84 film collection that was a subject of this research. More
information on the Family Video Archive project, including downloadable executable and user manual, is available at http://swiki.cc.gatech.edu:8080/ahome/97.

9. REFERENCES


