Towards a Unified System for Digital Film Production

Jake Seigel¹, Sam Fisher², and Stephen Brooks¹

¹ Dalhousie University Halifax, Nova Scotia

² NSCAD University Halifax, Nova Scotia

seigel@cs.dal.ca, sfisher@nscad.ca, sbrooks@cs.dal.ca

Abstract. We present a system which aids content management during the pre- and post-processing aspects of digital film production. Editors can synchronize their footage with their imported scripts to help in the organization of the video "bin". In addition to this, we provide a zoomable interface to facilitate the process and help organize pre-production media such as images, and video clips.

Keywords: zoomable, video editing, video analysis.

1 Introduction

Video editing typically involves sifting through large collections of raw footage stored in folders called "bins". These video files are usually logged, which means the filenames are tagged with the scene, shot, and take numbers. However, this provides only limited information and the editor must still sort through large numbers of clips, maintain a list of what parts of the script they cover, and ensure that there is a consistent contrast ratio between cuts. Therefore, one of the problems we attempt to address is the lack of organization and assistance during video editing.

We present a system which allows users to import film scripts written in Final Draft® for the purpose of synchronizing text to the spoken dialogue within the video clips. Final Draft® has been selected as it is considered industry standard software. The results of aligning the script to the clip bin are displayed in a timeline visualization. In addition to this, the system we present can also be used to organize pre-production media. Collections of media such as images, and video files can be imported and organized within a unified zoomable interface.

The following section introduces work related to our system. Then we describe the details of our system in the next three sections. The first of these sections contains an explanation of the user interface elements implemented. The second describes the speech recognition aspect and the processes related to improving those results. The third section details the visualization we employ for managing the post-processing film content. Next we present a case study to illustrate the usage of our software using video files from a real student film. Lastly, we discuss future directions for our system.

Fig. 1. A general view of our system. An imported Final Draft® script in a channel (*left*). A media channel (*center*). Several loose media panels (*top right*).

2 Related Work

2.1 Zoomable Interfaces

At a coarse-level, the user interface of our system has similarities to existing zoomable user interfaces. We will mention only the earliest examples, or those most related. In particular, Pad and Pad++ are early prototypes of this concept which demonstrated the effectiveness of zoomable interfaces [1] [2]. Two additional toolkits have since succeeded Pad++, namely Jazz and Piccolo. Jazz is a graphics toolkit for development in Java [3], while Piccolo offers zoomable features [4].

iMapping is an example of a zoomable application designed for general information structuring [5]. The author compares the iMaps created in the system as a whiteboard where information is added similar to post-it notes which can be nested within each other, and the system provides adequate navigation tools for viewing.

One last related zooming technique we have implemented is Igarashi and Hinckley’s speed-dependent automatic zooming technique for large documents [6] [7]. This causes the view to smoothly zoom out while the user is scrolling a document rapidly. The purpose of which is to allow the user to quickly navigate the workspace without becoming disoriented as the perceptual scrolling speed stays constant.

2.2 Script Authoring Suites

There are three pre-production authoring systems that will be discussed here. The first is Celtx (Greyfirst Corporation) [8], the second is Final Draft® (Final Draft Inc.) [9], and the third is Story (Adobe®) [10]. These systems all allow writers and planners to create scripts and in the case of Celtx, shooting schedules and 2D layouts.

Celtx is a free software system which consists of tools that aid in the creation of screenplays and related media. In addition to a typical text editor for screenplay writing, this software provides several pre-visualization tools and a scheduling system for managing film shoots among other features.

Final Draft® is considered to be the industry standard for screenwriting and provides authors with a set of tools for formatting standards. This software provides the user with no significant pre- or post-production visualization features, but we have chosen to support this screenplay authoring format for our software since the Final Draft® files are XML-based.

Adobe® Story is one program out of a series used for film production. As with the other systems, it is used for the creation of screenplays. Importing a Story script into Adobe® Premiere allows the script to be synchronized to video files using speech recognition. Additionally, the speech synchronization feature allows editors to create rough cuts of the film. But, a limitation of this system is that the production crew must only use Adobe products during production to make use of all of the offered features. Unfortunately, it may not be feasible or desirable to use Adobe® OnLocation during filming or Adobe® Premiere for editing as the filming studio may be invested in alternate systems for these tasks. Furthermore, while Adobe® products provide tools for film editing and metadata embedding, they do not offer any options for a comparative analysis of each clip.

3 System Overview

This section will provide an overview of the system and explain the user interface elements we have implemented in our software. The focus here is on elements related to pre-production and production, with further discussion of post-production editing revisited in sections 4 through 6.

Our system attempts to address the lack of organization of media during the pre-production process as well as to aid in the selection of video content during editing. We present a zoomable workspace to aid the use of our provided tools. The workspace in our system allows the user to import several media types to facilitate an integrated view from pre- to post-production. These media types include images, video files, and text files.

Fig. 2. Three media panels. An image panel (*left*), a video panel (*center*), and a floor plan panel (*right*).

The content in our system is represented as widget panels which can be moved or scaled, similar in some aspects to BumpTop [11]. Figure 2 illustrates examples of several media widgets in our system. Image and video panels are represented in our system simply by static thumbnail images (*left* and *center* respectively). Video panels can be invoked, which will play the video file using an embedded version of the VLC media player.

Scripts imported from Final Draft® are added to their own channel by default as shown in figure 3. As an additional navigational tool in our system, we have implemented speed-dependent scrolling [7]. This technique will automatically zoom

out the view window when the user scrolls rapidly, which helps to ensure that the scrolling speeds appears more perceptually consistent to the viewer.

4 Speech Recognition for Post-production Editing

Our speech recognition program produces a list of lines dialogue which were matched from the clip, the time that the line occurred, and a confidence value for the match. We use a series of filters to help remove false matches based on low confidence and false duplicates before applying a clustering algorithm. This algorithm groups together matches that are adjacent in the actual script. We then take these clusters and use them for two processes; first to remove further false matches, and second, to interpolate occurrence times for any dialogue lines which were not detected by the speech recognition engine.

In an example using every clip covering the first scene in the film (close to 20 minutes of footage), the average accuracy of predicted lines of dialogue was 1.4 seconds with a standard deviation of 1.02 seconds. The variation is dependent on the length of the lines and the density of dialogue in each part of the clips. In most instances, the larger predicted time windows were due to longer time periods of no dialogue within the video itself.

5 Post-production Visualization

The timeline visualization allows the user to view the alignment of spoken dialogue in each video clip to the imported script for that scene. This is represented by a timeline for each clip in the bin, displayed in a vertical channel as shown in figure 3. Each timeline is populated by boxes representing the lines of dialogue from the script. The position of each box represents the time position within the clip, while the width of the each box corresponds to the approximate length of time of the spoken dialogue.

Fig. 3. A line from the script is selected (*left*). Boxes on the timelines are highlighted to show the location of the selected script sentence (*left*).

Sections of the script can be selected which will highlight the segments of the timelines where they occur in the scene. Figure 3 illustrates that after the user selects the dialogue text, the predicted occurrence time window for each clip in the bin is displayed on the series of clip timelines. This allows us to show which clips the dialog was successfully matched in, and where the line is likely to have occurred.

6 Case Study

Here we present a short case study that provides an example usage of our system. Our data set is comprised of raw movie footage from a digital film production at a university-film school. To demonstrate our system, we will focus only on clips covering the first scene of the film.

The process begins by dragging the Final Draft® script for the film onto the window of our system along with several of the clips in our scene one bin. This adds the list of clips to a channel and the system begins processing the content. The speech recognition system analyzes each file and returns a list of dialogue matches and their occurrence times. Once the analysis is complete, each entry is listed in the timeline visualization. Not all of the video clips span the same parts of the script, and some are cut short due to on-set issues with the actors.

For example, if the editor wishes to select a subset of the videos containing a key line of dialogue; it is selected from the script channel. The videos which the dialogue has been recognized in are displayed and the position in time is highlighted in the timelines for each clip.

7 Conclusion

In summary, we present a software system for aiding in the organization of content for video editing as well as offering a pre-production media organization solution. Our zoomable interface provides access to the features of the system and scales well with large script files as well as large amounts of clips in the bin. For organizational clarity and modularity, we also provide vertical channel widgets to gather media into organized groups. Furthermore, due to the potential size of scripts and video data, speed-dependent scrolling plays a key role in the navigation of the script and the general workspace.

Our speech recognition system and error removal processes are useful for synchronizing the script with the video files. During times when the software does not always match spoken dialogue to the correct part of the script, our match clustering technique is able to reasonably fill in the gaps through positional interpolation. The timeline visualization also aids in illustrating the script alignment to the clips in the bin.

Lastly, we have presented a short case study demonstrating the use of our system on real video data. However, no formal user testing has been carried out to date, though we plan to gather systematic user data in further student productions in the near future. Additionally, we intend to expand our visualization to include a multivariate comparison tool.

References

- [1] Perlin, K., Fox, D.: Pad: an alternative approach to the computer interface. In: *Proceedings of Computer Graphics and Interactive Techniques*, pp. 57–64. ACM, New York (1993)
- [2] Wardrip-Fruin, N., et al.: A zooming sketchpad, a multiscale narrative: Pad++, PadDraw, Gray Matters. In: *ACM SIGGRAPH 1997 Visual Proceedings: The art and Interdisciplinary Programs of SIGGRAPH 1997*, p. 141. ACM, New York (1997)
- [3] Bederson, B., Meyer, J., Good, L.: Jazz: An extensible zoomable user interface graphics toolkit in Java. In: *Proceedings of User Interface Software and Technology*, pp. 171–180. ACM, New York (2000)
- [4] Bederson, B., Grosjean, J., Meyer, J.: Toolkit Design for Interactive Structured Graphics. *IEEE Transactions on Software Engineering* 30(8), 535–546 (2004)
- [5] Haller, H., Abecker, A.: iMapping: a zooming user interface approach for personal and semantic knowledge management. In: *Proceedings of Hypertext and Hypermedia*, pp. 119–128. ACM, New York (2010)
- [6] Cockburn, A., Savage, J.: Comparing speed-dependent automatic zooming with traditional scroll, pan and zoom methods. In: *People and Computers XVII: British Computer Society Conference on Human Computer Interaction*. University of Canterbury, Bath (2003)
- [7] Igarashi, T., Hinckley, K.: Speed-dependent automatic zooming for browsing large documents. In: *Symposium on User Interface Software and Technology*, pp. 139–148. ACM, San Diego (2000)
- [8] Greyfirst Corporation, Celtx, <http://www.celtx.com>
- [9] Final Draft, Inc., Final Draft, <http://www.finaldraft.com/>
- [10] Adobe, Story, <http://www.adobe.com/products/story>
- [11] Agarawala, A., Balakrishnan, R.: Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. In: *SIGCHI*, pp. 1283–1292. ACM, New York (2006)