RULE-BASED SCENE EXTRACTION FROM VIDEO

Lei Chen and M. Tamer Özsu

Department of Computer Science
University of Waterloo
Waterloo, ON, N2L3G1
{l6chen, tozsu}@uwaterloo.ca

ABSTRACT

Instead of clustering video shots into scenes using low level image features, in this paper, we propose a rule-based model to extract simple dialog or action scenes. Through analyzing video editing rules and observing temporal appearance patterns of shots in dialog scenes of movies, we deduce a set of rules to recognize dialog or action scenes. Based on these rules, a finite state machine is designed to extract dialog or action scenes from videos automatically.

1. INTRODUCTION

Modeling of video requires identification and extraction of its components. Early video database systems segment video into *shots*, and extract key frames from each shot to represent it. Such systems have been criticized for two reasons: shots do not convey much semantics, and using key frames may ignore temporal characteristics of the video. Therefore, there have been several attempts [1, 2, 3] to cluster semantically related shots into *scenes*. However, current approaches only employ low-level image features, which may cause semantically unrelated shots to be clustered into one unit only because they may be "similar" in terms of their low-level image features. Furthermore, users may not be interested in the "general" scenes constructed in this way, but may focus on particular scenes. In particular, dialog and action scenes have special importance in video, since they constitute basic "sentences" of a movie that consists of three basic types of scenes [4]: dialogs without action, dialogs with action, and actions without dialog. Automatic extraction of dialog and action scenes from a video is an important topic for practical usage of video.

A given video clip may be (and commonly is) interpreted differently by different users. However, there is one viewpoint that is the most important: that of the video editor or director. From their viewpoint, a video is produced to express some concepts or stories that they want to communicate to the audience. The editing process follows certain rules that can be used in automatic extraction of scenes. In this paper, based on the video editing rules for dialog and action scenes, we propose a Finite State Machine (FSM) model to extract simple dialog or action scenes from movies.

2. DIALOG AND ACTION SCENE PATTERN ANALYSIS

Observation of a large number of video dialog and simple action clips reveals the existence of visual patterns, such as interleaving patterns of the appearance of the actors who are involved in a dialog. These visual patterns can form the basis for detecting dialog and action scenes. In order to extract these visual patterns, the video clips are analyzed from the point of view of how a dialog scene is produced. In this paper, we focus on the case where a dialog scene (DS) has at most two actors (a and b) in it. This assumption is made for two reasons. First, the rules for the positioning of actors and cameras are better understood and documented in movie literature. Second, the case of two actors is easier to explain and demonstrate. The extension of our work to more actors is explained at the end of the paper. DS is composed of a set of shots.

2.1. Patterns of Simple Dialog Scenes

In order to capture a dialog, two main factors must be considered:

- the spatial arrangement of the actors; and
- the placement of cameras to capture dialogs.

These two factors affect the appearance of actors in captured video shots. Through the analysis of actor arrangement and camera placement, we find that there are only three basic types of video shot patterns in a two person (call a and b) dialog scene:

- a shot in which only actor a's face is visible throughout the shot (Type A shot);
- a shot in which only actor b's face is visible throughout the shot (Type B shot); and

- a shot with both actors a and b, with both of their faces visible (Type C shot).

In addition to these, usually an *insert* or *cut-away* shot is introduced to depict something related to the dialog or not covered by those three types of shots. We use symbol $\#$ to represent it. These constitute *video type set* $V = \{A, B, C, \#\}$.

2.2. Patterns of Simple Action Scenes

The rules governing the actor arrangement and camera placement in simple action scenes (e.g., one-on-one fighting), are the same as those for producing simple dialog scenes. This is true even though, in an action scene, actors move rapidly and cameras follow the actors. Therefore, video shots in a simple action scene can also be classified into four types: A,B, C and $\#$ as defined above.

3. RULE-BASED EXTRACTION OF DIALOG OR ACTION SCENES

After a set of video shots are obtained from cameras that are used to film dialogs, the issue becomes how these shots can be used to construct a dialog scene to express a conversation. This is a challenging question for a video editor. However, there are some basic techniques that a video editor typically follows in constructing dialog scenes [4, 5].

3.1. Editing Techniques to Construct a Dialog

Editing a dialog scene consists of two steps:

1. **Setting up the dialog scene.** In the first step, video editors set up the dialog scenario. The preference is for a scene that either consists of shots involving both actors (type C scene) or consists of shots that show alternating actors (i.e, either AB or BA), because these give the audience an early impression of who are involved in the dialog. During this setting up process, the basic building blocks of dialog scenes are constructed. We call these basic building blocks as *elementary dialog scenes*. An elementary dialog scene includes a set of video shots, and can itself be a dialog scene or be expanded to a longer dialog scene. The set of elementary dialog scenes are determined empirically, based on the analysis of editing rules used to establish dialog scenes and observations of dialog scenes of five movies [1]. As a result, we have identified eighteen types of elementary dialog scenes as depicted in Table 1 along with statistics about their occurrence frequency in the five movies under consideration.

[1] 1. "Conair", 1998; 2. "Life is Beautiful", 2000; 3. "First Knight" 1998; 4. "Deconstruction", 1990; 5. "What dreams may come" 1998.

elementary dialog scenes	appearance percentage
ABAB or BABA	41.21%
CAB or CBA	21.21%
C or C#C	19.39%
ABC or BAC	6.06%
CAC or CBC	3.63%
ABAC or BABC	2.42%
ACC or BCC	2.42%
ACA or BCB	2.42%
ACB or BCA	1.21%

Table 1. Statistical data on elementary dialog scenes

2. **Expanding the dialog scene.** In the second step, each elementary dialog scene can be expanded by appending three types of shots. During this editing process, the basic rule that an editor uses is to give a contrast impression to the audience. For example, if the ending shot of one scene is an A type shot, usually a B type shot is appended to expand the scene. Similarly, the editor can append a C type shot as a re-establishing shot from time to time to remind the audience of the whole scenario surrounding the dialog scene. Table 2 lists expansion rules.

type of end shot in the scene	types of shots that may follow
A	B or C
B	A or C
C	A or B or C

Table 2. Possible types of shots to be appended

3.2. Video Shot String

We introduce the concept of a *video shot string* (VSS) to represent the temporal presentation sequence of different types of shots in a video. A VSS is a string which is composed of symbols from V. Each symbol in VSS represents a shot in a video. The ordering of symbols in the string is from left to right, which represents the shot presentation sequence.

Based on the analysis of the above-discussed two editing steps, we define a VSS of a dialog scene as a string whose prefix is one of the elementary dialog scenes that can be expanded by the rules given in Table 2. The starting elementary dialog scene classifies a VSS as well. Consequently, there are eighteen types of VSSs corresponding to those types of dialog scenes. It is easy to prove that these are regular languages over set V. We do not give a complete proof due to lack of space, but the following is the proof of one of these cases, namely the VSS whose prefix is $ABAB$. Proof of other cases are similar. $\{A\}$, $\{B\}$, $\{C\}$ are regular languages over V. $\{ABAB\}$ is a product of regular langauges $\{A\}$ and $\{B\}$: $\{ABAB\} =$

$\{A\} \bullet \{B\} \bullet \{A\} \bullet \{B\}$, so $\{ABAB\}$ is a regular language over V, too. VSS that starts with $ABAB$ includes string $ABAB$ and all the strings which are expanded from $ABAB$ by appending A, B or C using the rules in Table 2. Appending a shot to a scene is a concatenation operation ($\bullet$). Therefore, by definition of a regular language [6], a VSS of a dialog scene that starts with $ABAB$ is a regular language over V.

By taking the union of the eighteen types of VSSs, we again obtain a regular language over set V. Therefore, VSSs that are used to represent the temporal appearance patterns of video shots in dialog scenes are regular languages over set V.

3.3. Finite State Machine to Extract $VSSs$ of Dialog Scenes

Since VSSs of dialog scenes are regular languages, the next issue will be how to automatically extract the VSSs which correspond to dialog scenes from VSSs of the whole video. In other words, how to extract specified regular languages from VSSs? In this paper, we propose a finite state machine (FSM) model to extract dialog scenes from videos. Note that we are not using the FSM to determine whether a language is a regular language over V, but constructively to extract those parts of a VSS that form regular languages with certain properties. In our proposed FSM, a VSS is used as an input to the FSM. A state is used to represent the status after a number of shots have been processed. An edge between states will determine an allowable transition from current state to another state under a labeled condition. The label of the arc is a symbol which is used to represent a type of shot. A sub-string of the VSS will be extracted by the FSM if and only if there exists a path from initial state to one of final states. The symbols on the path correspond to sequence of the shots in that sub-string of VSS. Figure 1 shows the transition diagram of our proposed FSM which is used to extract VSSs of simple dialog scenes between two actors.

3.4. Extract Action Scenes

Since a video editor follows similar rules that are used to construct dialog scenes to compose simple action scenes, temporal appearance patterns of video shots in simple action scenes are similar to those of dialog scenes. The FSM model discussed above is also suitable for extracting simple action scenes (one-on-one fighting). However, in order to give the audience a different feeling between action scenes and dialog scenes, several other techniques are used to enhance visual effects. These techniques involve manipulating the length of a shot and combining static and moving cameras, etc. Among these, the length of a shot is an important factor to express visual effects of an action scene. In our approach, the average length of shots in a scene will be used to differentiate between a dialog scene and an action scene.

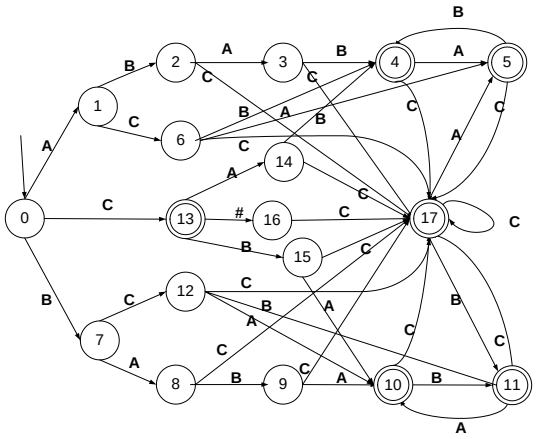

Fig. 1. A FSM extracts VSSs of dialog scenes between actor a and actor b

4. EXPERIMENT RESULTS

In this section, we present the results of some extraction experiments that were conducted using our FSM model. Three different movies are used in our experiment (Table 3), which are first segmented into shots and appearances of actors are manually marked.

Our focus is on retrieval precision and recall, which are defined identical to their use in the information retrieval literature. Precision measures the proportion of correctly recognized scenes, while recall measures the proportion of scenes that are recognized.

Tables 4 and 5 show the result of extracting simple dialog scenes and simple action (one-on-one fighting) scenes respectively.

Table 4 shows that our model can achieve high precision and recall in extracting dialog scenes from movies. There is an interesting fact in Table 4 that the FSM model achieves better results on the movie "Patch Adams" compared to the other two movies. As shown in Table 5, and as our analysis showed, "Patch Adams" is completely composed of dialog scenes without any action scenes. This ensures that we do not falsely detect a dialog scene as an action scene or vice versa. In the movie "Crouching Tiger and Hidden Dragon", the precision is low, because several action scenes are mixed with dialog scenes. This is an artifact of the fact that we use the same FSM (which implements the same set of rules) to detect both types of scenes, and in some action scenes in this movie, long shots are used to show the action effects, which leads our model that uses the average shot length to differentiate dialog and action scenes to misclassify these action scenes as dialog scenes.

After we extract simple dialogs from movies, we can easily retrieve dialog scenes involving three or more actors. This can be achieved by finding pairs of dialog scenes with

a common actor and overlapping durations. Table 6 shows the performance of this approach in detecting multi-actor dialog scenes in the three movies under consideration.

movie title	genre	year	duration (min)	No. shots
Gladiator	Action	2000	154	1363
Crouching Tiger and Hidden Dragon	Action	2000	120	1575
Patch Adams	Comedy	1998	120	1131

Table 3. The experiment data

movie title	No. detected dialogs	precision (%)	recall (%)
Gladiator	95	89.47	96.60
Crouching Tiger and Hidden Dragon	154	80.52	90.51
Patch Adams	195	91.79	97.28

Table 4. Dialog scenes extracted by the FSM

movie title	No. detected actions	precision (%)	recall (%)
Gladiator	25	84	84
Crouching Tiger and Hidden dragon	64	76.56	81.6

Table 5. Action scenes extracted by the FSM

movie title	No. group conversion	No. missed
Gladiator	6	0
Crouching Tiger Hidden Dragon	6	2
Patch Adams	8	0

Table 6. The detected group conversion scenes

5. CONCLUSION AND FUTURE WORK

In this paper, based on the analysis of video editing techniques, a set of rules on the temporal appearance patterns of shots are deduced. An FSM is designed based on these rules to extract simple dialog or action scenes. The experimental results show that our model can efficiently extract dialog and action scenes from movies, and with simple dialog scenes extracted from movies, group conversions can be easily detected. Our FSM model is a rule based model, it will be very suitable for online query processing. As we know, audio is an important feature for video analysis. Our future work will focus on integrating audio classification algorithms into our model to achieve higher accuracy and to extract more types of semantic scenes.

There is limited related work in the area. Yoshitiaka et al. [7] propose an algorithm to extract scene semantics(conversation, tension rising, and action) based on a grammar of the film. However, in their approach, only the repetition of similar shots $(A - B - A' - B')$ is employed to detect conversion scenes. Also Lienhart et al. [8] develop a technique to extract dialog scenes with the aid of an face detection algorithm. However, they only extract dialog scenes which show shot/reverse shot patterns. Compared to both of these approaches, our method has following advantages:

- We can, in addition to shot/reverse shot dialogs, detect single shot dialogs, dialogs with insertions and cuts, and dialogs with shot/reverse shots and recovering shots.
- Our model is rule based, which is very suitable for on-line content based query processing.
- Our model is based on the editors' point of views, rather than relying on individual users' interpretation of the video, which may cause semantic heterogeneity problems as a result of different interpretations.

6. REFERENCES

[1] M. M. Yeung and B. L. Yeo, "Time-constrained clustering for segmentation of video into story units," in *Proceedings of 13th International Conference on Pattern Recognition*, 1996, pp. 375–380.

[2] Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structure beyond the shots," in *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, 1992, pp. 237–240.

[3] A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automatically segmenting movies into logical story units," in *Proceedings of International Conference on Visual Information Systems*, 1999, pp. 229–236.

[4] D. Arijon, *Grammar of the Film Language*, Focal Press, 1976.

[5] S. D. Katz, *Film Directing Shot by Shot Visualizing From Concept to Screen*, Michael Wiese Productions, 1991.

[6] J. L. Hein, *Theory of Computation: An introduction*, Jones and Bartlett Publishers, 1996.

[7] A. Yoshitaka, T. Ishii, and A. Hirakawa, "Content-based retrieval of video data by the grammar of film," in *Proceedings of IEEE Symposium on Visual Languages*, 1997, pp. 310–317.

[8] R. Lienhart, S. Pfeiffer, and W. Effelsberg, "Scene determination based on video and audio features," in *Proceedings of International Conference on Visual Information Systems*, 1999, pp. 685–690.