A fast interviews searching scheme in news sequences

Alberto Albiol[1], Luis Torres[2], and Edward J. Delp[3]

[1] Politechnic University of Valencia,
Crta Nazaret-Oliva S/N, 46730 Grao de Ganda, Spain,
alalbiol@dcom.upv.es,
WWW home page: http://ttt.gan.upv.es/ alalbiol
[2] Politechnic University of Catalonia,
Barcelona, Spain
[3] Purdue University,
West Lafayette, USA

Abstract. In this paper we address the problem of seeking anchor person shots in news sequences. This can be useful since usually this kind of scenes contain important and reusable information such as interviews. The proposed technique is based on our a priori knowledge of the editing techniques used in news sequences.

1 Introduction

The amount of digital video has undergone an explosive growth in the last years. One of the problems when dealing with digital video is the huge amount of data to be analyzed. This problem can be minimized if we are able to select relevant portions of the video where more powerful analysis tools can be applied.

In broadcast news, around 85% of the time consists of a television announcer or an off-voice of a journalist reading a report, while only the remaining 15% consists of anchor person shots of interviewed people. We are interested in locate these shots where more sophisticated tools such as speaker or face recognition can be applied. However, in order to efficiently apply these techniques a tool for seeking interviewed people is needed. In this paper we address this problem. Our approach takes advantage of our a priori knowledge of the editing techniques used in news sequeces. In section 2 the main elements that build a news sequence are described, this will give us the key idea to the presented approach. Sections 3 to 5 present the analysis tools that will be used. Some results are provided in section 6.

2 Elements of a news sequence

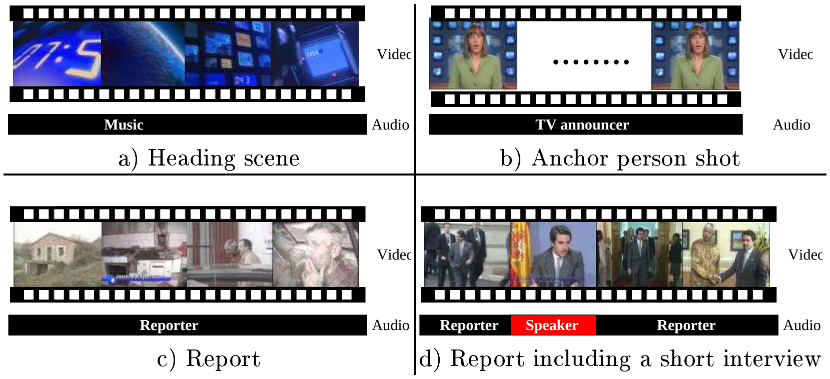
News sequences are composed usually of the following elements:

1. Headings.

2. Live journalist speeches. Usually the images are either a close-up of the television announcer or topic related scenes.
3. Prerecorded videos, usually containing the journalist off-voice while some related images are displayed. Also, short anchor person scenes of interviewed people are usually inserted in the videos.

Figure 1 illustrates the previous scenes types. In this paper we are interested in locating interviewed people as in 1.d. The editing procedure for this type of scenes can be summarized as follows:

1. The reporter records his/her voice but no image is recorded yet.
2. If an interview is going to be inserted, then its audio and video are inserted together, creating a simultaneous audio and video cut.
3. If the reporter needs to continue with his/her speech or more interviews need to be inserted, the two first steps are repeated.
4. Finally, images are added for the reporter voice periods, usually several shots are inserted for each audio segment.



a) Heading scene

b) Anchor person shot

c) Report

d) Report including a short interview

Fig. 1. News sequence elements

The consequence of this editing procedure is that interviews can be easily detected studying the matching of audio and video cuts. Sections 3 and 4 will describe the algorithms to detect audio and video cuts, while in section 5 a procedure to combine those results will be proposed.

3 Audio segmentation

The goal of speaker segmentation is to locate all the boundaries between speakers in the audio signal. Some speaker segmentation systems are based on silence detection [1]. These systems relies on the assumption that utterances of different people are separated by significant silences. However reliable systems would require cooperative speakers which is not the case for broadcast news. Other segmentation approaches are based on speaker turn detection. These systems aim to segment the audio data into homogeneous segments containing one speaker only. Usually a two-step procedure is used, where the audio data is first segmented in an attempt to locate acoustic changes. Most of these acoustic changes will correspond to speaker turns. The second step is used then to validate or discard these possible turns. The segmentation procedures can be classified into three different groups: phone decoding[2, 3] , distance-based segmentation [4, 5], hypothesis testing [6]. In this paper we will use the DISTBIC algorithm [5] to partition the audio data. DISTBIC is also a two-step segmentation technique. In the first step distance between adjacent windows is obtained every 100ms. This result in a distance signal $d(t)$, see Figure 2. In our implementation we use the symmetrical kullback-Leibler [4] distance. The significant peaks of $d(t)$ are considered as turn candidates. In the second step the turn candidates are validated using the ΔBIC criteria [7]. To that end, the acoustic vectors of adjacent segments are modeled separately using Gaussian models. The model of the union of the acoustic vectors of both segments is also computed and then the ΔBIC criteria is used to check is the likelihood of the union is greater than the likelihood of both segments individually. In the case that the likelihood of the union is greater then the turn point is discarded. Otherwise the turn point is validated.

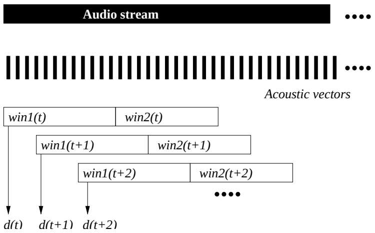


Fig. 2. Sliding windows

4 Video segmentation

Several techniques have been proposed in the literature for detection of cuts in video sequences [8–10]. Most of them rely on the similarity of consecutive frames. A popular similarity measurement is the *mean absolute frame difference (MAFD)*:

$$MAFD(n) = \frac{1}{IJ} \sum_{i=1}^{I} \sum_{j=1}^{J} |f_n(i,j) - f_{n-1}(i,j)| \qquad (1)$$

where I and J are the horizontal and vertical dimensions of the frames, n is the frame index, and (i, j) are the spatial coordinates.

Other popular similarity measures include the displaced frame difference (DFD), which reduces the contribution of camera and objects motion, at the expense of a greater computational load. In this work, techniques that need motion estimation are avoided because of the higher computational requirements. Also low resolution images obtained from the DC coefficients of the MPEG compressed video stream will be used to compute the MAFD measurement. This has the advantage that does not require full decompression of the video in order to find the cuts [8].

The causes of dissimilarity between consecutive frames include:

- Actual scene transitions
- Motion of objects
- Camera motion
- Luminance variations (flicker)

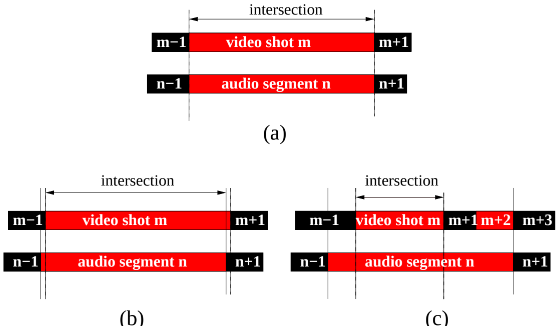
In standard (good condition), the last contribution is normally negligible (except for special situations such as the presence of flashlights). Motion of objects and camera normally occur during more than one transition which produces wide pulses in the MAFD signal. On the other hand, an abrupt scene transition produces a peak of width one in MAFD. This difference can be exploited to distinguish motion and cuts in video. Basic morphological operations, such as openings and closings, can be applied for this purpose. The proposed algorithm for cut detection can be summarized as follows:

- Obtain MAFD(n)
- Compute the residue of the morphological opening of MAFD(n)
- Threshold the residue to locate the cuts.

We are well aware that more sophisticated video cut detection algorithms exist. However, this simple algorithm provides very good results, since other transitions effects such as wipes or fades are not usually used in news reports. Moreover, the interviews do not usually show a high shot activity (usually the scene is an anchor person), therefore the false alarm rate within these intervals is nearly zero.

5 Audio and video correspondence

Once the audio and video segments are located the objective is to find the correspondence between then. Figure 3.a shows the ideal situation that we are trying to find, i.e. the audio and video segments overlap. However, for real sequences the borders of audio and video segments do not overlap, as shown in figure 3.b. This is due mainly because silence periods are usually located in the audio segment borders creating a small inaccuracy. Figure 3.c shows an example of the typical situation for report segments, where a long audio segment coexists with short video segments. Given an audio segment in the time interval $[t_{min1}, t_{max1}]$ and



(a)

(b)

(c)

Fig. 3. (a) Audio and video borders math exactly. (b) Audio and video borders almost match (c) Audio segment contains several video shots

a video segment defined in the interval $[t_{min2}, t_{max2}]$. The intersection interval is defined as:

$$[t_{min\cap}, t_{max\cap}] = [\max(t_{min1}, t_{min2}), \min(t_{max1}, t_{max2})] \tag{2}$$

then if $(t_{max\cap} - t_{min\cap}) > 0$ for a pair of audio and video segments, we define the overlap degree as:

$$overlap = \min\left\{\frac{(t_{max\cap} - t_{min\cap})}{(t_{max1} - t_{min1})}, \frac{(t_{max\cap} - t_{min\cap})}{(t_{max2} - t_{min2})}\right\} \tag{3}$$

If *overlap* > 0.9 then the audio and video segments are said to match and a new index entry is created.

6 Results and conclusions

The previous algorithms have been tested on several 30 minutes news sequences. The results have been evaluated with the following parameters: Detection Rate (DR):

$$DR = 100 \times \frac{\text{number of detected interviews}}{\text{number of actual interviews}} \tag{4}$$

False alarm rate (FAR):

$$FAR = 100 \times \frac{\text{number of false alarms}}{\text{number of actual interviews} + \text{number of false alarms}} \tag{5}$$

and Selected Time (ST)

$$ST = \frac{\text{total duration of the selected shots}}{\text{Sequence duration}} \tag{6}$$

Table 1 presents some results. It can be seen how the algorithm allows to discard a large portion of the sequence from consideration with minimal processing.

DR	FAR	ST
94 %	41 %	31 %

Table 1. Results

Almost all false detected shots correspond to anchor person shots where the speaker is a reporter.

We have presented a novel fast algorithm to detect interviews without needing to analyze in detail the whole sequence. Once the segments of interest are located more sophisticated analysis tools can be used such as: speaker or face recognition. These analysis tools can be used independently or they can also be combined to obtain more reliable results.

References

1. M. Nishida and Y. Ariki, "Speaker indexing for news, articles, debates and drama in broadcasted tv programs," in *IEEE International Conference on Multimedia, Computing and Systems*, 1999, pp. 466–471.
2. T. Hain, S. E. Johnson, A. Tuerk, P. C. Woodland, and S. J. Young, "Segment generation and clustering in the htk broadcast news transcription system," in *Proceedings of DARPA Broadcast News Transcription Understanding Workshop*, Landsdowne, VA, 1998, pp. 133–137.
3. D. Liu and F. Kubala, "Fast speaker change detection for broadcast news transcription and indexing," in *Proceedings ESCA Eurospeech'99*, Budapest, Hungary, 1999, vol. 3, pp. 1031–1034.
4. M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proceedings of the DARPA speech recognition workshop*, Chantilly, Virginia, February 1997, pp. 97–99.
5. P. Delacourt and C. J. Wellekens, "Distbic: A speaker-based segmentation for audio indexing," *Speech communication*, vol. 32, no. 1-2, pp. 111–127, September 2000.
6. S. Wegmann, P. Zhan, and L. Gillick, "Progress in broadcast news transcription at dragon systems," in *Proceedings of International Conference on Acoustics Speech and Signal Processing*, Phoenix, AZ, 1999, pp. 33–36.
7. S. S. Chen and P. S. Gopalakrishnan, "Speaker environment and channel change detection and clustering via de bayesian information criterion," in *DARPA Speech Recognition Workshop*, 1998.
8. J. S. Boreczky and L. A. Rowe, "Comparison of video shot boundary detection techniques," in *Proceedings SPIE Conference on Visual Communications and Image Processing*, 1996.
9. A. Albiol, V. Naranjo, and J. Angulo, "Low complexity cut detection in the presence of flicker," in *Proceedings 2000 International Conference on Image Processing*, Vancouver, Canada, October 2000.
10. B. Liu and B. Yeo, "Rapid scene analysis on compressed video," *IEEE Transactions on Circuits and Systems*, vol. 5, no. 6, pp. 533–544, September 1995.