DETECTION OF SLIDE TRANSITION FOR TOPIC INDEXING

Chong-Wah Ngo
Department of Computer Science,
City University of Hong Kong,
cwngo@cs.cityu.edu.hk

Ting-Chuen Pong
Department of Computer Science
Hong Kong University of
Science & Technology
tcpong@cs.ust.hk

Thomas S. Huang
Beckman Institute
University of Illinois at Urbana Champaign
huang@ifp.uiuc.edu

ABSTRACT

This paper presents an automatic and novel approach in detecting the transitions of slides for video sequences of technical lectures. Our approach adopts a foreground vs background segmentation algorithm to separate a presenter from the projected electronic slides. Once a background template is generated, text captions are detected and analyzed. The segmented caption regions as well as background templates together provide salient visual cues to decide whether a slide is flipped and replaced. The partitioning of videos according to slide changes not only structure the content of video according to topics, but also facilitate the synchronization of video, audio and electronic slides for effective indexing, browsing and retrieval.

1. INTRODUCTION

Research on the structural analysis of video content has been actively conducted since the last decade [8]. Typical efforts include the decomposition of videos into shots and the clustering of similar shots into scenes. The developed techniques are mostly applied to general video sequences. In this paper, we constrain the domain of analysis to the video sequences of lecture presentation. More specifically, our aim is to index automatically the content of videos captured during lectures according to the topics of discussion. The structured videos together with the electronic slides prepared by lecturers are ultimately presented on web such that students from distance can browse and retrieve the video content according to the topics of interest.

Major research issues in this area are the detection of slide transition, the detection of text region in viewgraph, recognition of characters and words, tracking of pointers and animation, speech recognition and the synchronization of videos, audio and presentation slides. Related works include [1, 2, 3, 4, 5, 6]. Existing systems include Classroom 2000 [1] and Interactive Virtual Classroom [2].

In this paper, we focus on the automatic detection of slide transitions by both background and caption cues. We present a method of locating background and detecting text captions in an image volume. A new slide is detected and indexed whenever there is a change of figure or caption from the previous slide. The system set up is as follows. A camera is mounted in the lecture hall so as to capture the presenter and the projected electronic slides. The position of camera is fixed and it stays stationary throughout a lecture. A presenter can move freely in front of the projected screen and use pointers to explain or highlight important concepts. The captured videos are digitized and encoded in MPEG format.

Unlike shot transitions, slide transitions do not show significant color changes in most cases. Since most presenters tend to apply same design to all electronic slides in one presentation, the color content of adjacent slides could be very similar. Traditional shot boundaries detectors [7, 8], on one hand, may miss such transitions, on the other hand, can easily cause false alarms due to the motion of a presenter. While most approaches adopt audio cues to detect slide transitions, we propose in this paper the use of visual cues. Better results are expected if both audio and visual cues are integrated. Other approach that based on visual cues include [4, 6]. In [4], motion information is utilized to detect slide changes, while in [6] text layouts are used for detection. The former approach does not take into account the text information in slides, while the latter approach can only deal with slides that contain only text captions, in addition, could not handle cases like the occlusion of projected slides as a presenter moves. Our approach, in contrast, takes into account the background vs foreground information, figures and caption regions in slides when detecting transitions.

In our approach, to discount the effect of motion due to presenters, foreground (presenter) and background (lecture hall and electronic slide) scenes are segmented. To be effective, an image volume formed by hundreds of video frames is analyzed as a whole at each iteration. To be efficient, compressed data is extracted directly for processing. The segmented background is then utilized for detecting texts and identifying caption changes in slides. Since most figures occupy a relatively large portion in slides, the change of background information is a good indicator for the change of figures. In our algorithm, whenever there are changes due

to background and captions between two image volumes, video frames inside the volumes will be further investigated to detect the exact slide transitions.

2. OVERVIEW OF THE FRAMEWORK

A video is first temporally partitioned into divisions of fixed interval. Each division contains a set of image frames. For convenience, we refer a division as a time frame. Two adjacent time frames are always partially overlapped. Figure 1 shows the overview of our framework. The proposed algorithm works directly on time frames. Initially, a set of templates are computed from a time frame for background and foreground segmentation. The resulting background template is used as a mask to detect caption and to compute energy due to background change. A text mask is also generated for the computation of energy due to caption change. Both background and caption energies are utilized to decide if a time frame contains slide transitions. Once a transition is suspected, the caption and background similarity among the image frames within a time frame are compared to detect the exact slide transition.

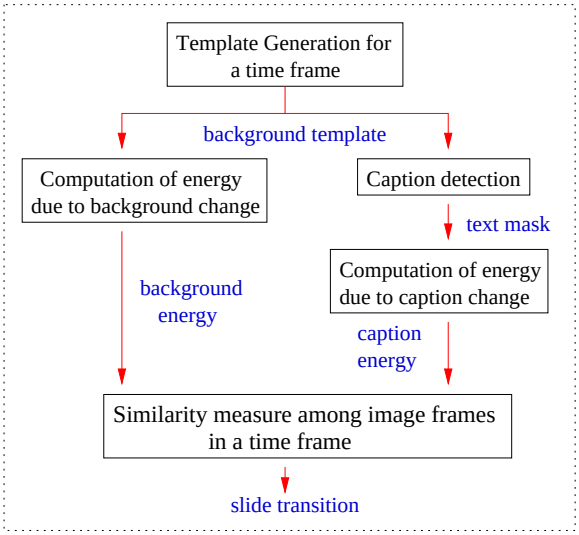

Figure 1: Framework.

Our approach operates directly in the MPEG compressed domain. Instead of original size image frames, DC image sequence extracted from DC coefficients are used for template generation and energy computation. Besides DC coefficients, AC coefficients are also utilized directly for caption detection. For the ease of understanding, the following notations are used in the remaining paper:

- The k^{th} time frame of a video is denoted as $\mathbf{T}_k$. The total number of image frames in a time frame is $\#\mathbf{T}_k$. The t^{th} DC image frame of $\mathbf{T}_k$ is denoted as $f_k(t)$. The pixel value of $f_k(t)$ at location (i, j) is written as $f_k(i, j, t)$. The size of a DC image frame is assumed as $M \times N$.
- The energy computed by background change detector is denoted as $\mathbf{E}_b$, while the energy computed by caption change detector is denoted as $\mathbf{E}_c$.
- The DCT coefficients are denoted as ρ_{uv}, where ρ_{00} denotes DC coefficient while ρ_{uv} denotes AC coefficients for $u, v \neq 0$. The coefficients of a DCT block at $f_k(i, j, t)$ is simply indexed as $\rho_{uv}(i, j, t)$.

3. TEMPLATE GENERATION

The mean μ_k and deviation σ_k templates at time frame $\mathbf{T}_k$ are computed as

$$\mu_k(i,j) = \frac{1}{\#\mathbf{T}_k} \sum_{t \in \mathbf{T}_k} f_k(i,j,t) \quad (1)$$

$$\sigma_k(i,j) = \frac{1}{\#\mathbf{T}_k} \sqrt{\sum_{t \in \mathbf{T}_k} \{f_k(i,j,t) - \mu_k(i,j)\}^2} \quad (2)$$

A background template which contains only binary values is then generated as

$$b_k(i,j) = \begin{cases} 0 & \sigma_k(i,j) > 2 \times \Upsilon_k \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

where $\Upsilon_k = \frac{1}{N} \sum_i \sum_j \sigma_k(i,j)$ is the mean standard deviation of time frame $\mathbf{T}_k$ and N is the size of an image. The background template will be used as a mask for computing $\mathbf{E}_b$ and $\mathbf{E}_c$. The energy $\mathbf{E}_b$ due to background change is

$$\mathbf{E}_b = \frac{1}{\#b_{k-1}} \sum_i \sum_j b_{k-1}(i,j) \times |\mu_k(i,j) - \mu_{k-1}(i,j)| \quad (4)$$

where $\#b_k = \sum_i \sum_j b_k(i,j)$ is a normalizing term.

4. CAPTION DETECTION

Texture-based approach is adopted to detect captions since text regions are generally composed of a unique texture pattern [9]. This pattern is due to the horizontal intensity variations caused by the characters within a text line and the vertical intensity variations caused by the spacing between text lines. We utilize the AC coefficients ρ_{uv} of MPEG videos to capture directly the intensity variations. At each time frame $\mathbf{T}_k$, the potential text regions are characterized by the horizontal EH_k and vertical EV_k text energies where

$$EH_k(i,j) = \frac{b_k(i,j)}{\#\mathbf{T}_k} \times \sum_{t \in \mathbf{T}_k} \sum_{v=2}^{6} |\rho_{0v}(i,j,t)| \quad (5)$$

$$EV_k(i,j) = \frac{b_k(i,j)}{\#\mathbf{T}_k} \times \sum_{t \in \mathbf{T}_k} \sum_{u=1}^{6} |\rho_{u0}(i,j,t)| \quad (6)$$

Initially, a DCT block indexed by (i, j) is detected as a potential candidate if $EH_k(i, j)$ is greater than 1.45 times the average horizontal text energy of all the corresponding DCT coefficients in $\mathbf{T}_k$. The detected text regions are then refined by two morphological operators: a *closing* operator followed by an *opening* operator. Falsely detected regions and isolated noises are also pruned at this stage. Subsequently, contour tracking and connectivity analysis are implemented to segment text regions. Finally, three criterions are used to assess the validity of a text region: 1) the ratio of width over height is at least 1.5; 2) the size must cover at least 10 DCT blocks; 3) the total EV_k of a text region is at least 80. Figure 2 shows the segmented background and detected caption regions.

Figure 2: (left) Original image frame; (middle) segmented background; (right) detected captions.

Denote c_k as the text mask for $\mathbf{T}_k$, the caption energy $\mathbf{E}_c$ due to caption change is computed as

$$\mathbf{E}_c = \frac{1}{\#c_{k-1}} \sum_i \sum_j c_{k-1}(i,j) \times |\mu_k(i,j) - \mu_{k-1}(i,j)| \quad (7)$$

where $\#c_k = \sum_i \sum_j c_k(i,j)$, $c_k(i,j) = 1$ if the corresponding location belongs to a text region and $c_k(i,j) = 0$ otherwise.

5. DETECTION OF SLIDE TRANSITION

The algorithm for locating the exact slide transitions is given in Figure 3. For each time frame $\mathbf{T}_k = \{f_k(0), f_k(1), \ldots, f_k(m)\}$, energies due to background and caption changes are computed respectively. If either energy exceeds a pre-defined threshold, the exact transition will be determined by measuring similarity among the image frames within $\mathbf{T}_k$. The detector which possesses a greater energy will always be used to conduct similarity measure.

Denote Sim_c and Sim_b as the caption and background similarities respectively between two image frames $f_k(t)$ and $f_k(t+1)$, we have

$$Sim_c = \frac{\sum_i \sum_j c_{k-1}(i,j) \times |f_k(i,j,t) - f_k(i,j,t-1)|}{\sum_i \sum_j c_{k-1}(i,j) \times |f_k(i,j,t+1) - f_k(i,j,t)|}$$

$$Sim_b = \frac{\sum_i \sum_j b_{k-1}(i,j) \times |f_k(i,j,t) - f_k(i,j,t-1)|}{\sum_i \sum_j b_{k-1}(i,j) \times |f_k(i,j,t+1) - f_k(i,j,t)|}$$

1. Generate new templates for time frame $\mathbf{T}_k$.
2. Compute $\mathbf{E}_b$ as shown in Eqn(4).
3. Detect captions and generate text mask.
4. Compute $\mathbf{E}_c$ as shown in Eqn(7).
5. If either $\mathbf{E}_b$ or $\mathbf{E}_c$ is greater than a threshold
   (a) Locate the exact transition by similarity measure among the image frames in $\mathbf{T}_k$.
   (b) Re-compute the templates for $\mathbf{T}_k$.
6. Repeat Step 1 for $\mathbf{T}_{k+1}$.

Figure 3: Slide transition detection algorithm.

In principle, the similarity value will be low if the value of denominator is large. The numerator is a weighting factor such that only local minimum will be detected as a slide transition. A frame $f_k(t+1)$ is determined as the beginning of a new slide presentation if $Sim_c < 0.4$ or $Sim_b < 0.4$. Notice that the masks c_{k-1} and b_{k-1} are employed instead of c_k and b_k. The current c_k and b_k are not reliable since there are computed from the image frames of $\mathbf{T}_k$ which may contain slide transitions. New templates include μ_k, σ_k, b_k and c_k will be re-computed from frames $\{f_k(t+1), f_k(1), \ldots, f_k(m)\}$ after a transition is detected.

6. EXPERIMENT

We conduct experiments on two lecture videos. Each lecture lasts for approximately 75 minutes. All slides are presented through PowerPoint. Similar design template and color scheme are applied for all slides in one presentation. The main difference between slides are figures and captions changes. The flipping time of most slides involve only two frames. We compare and contrast the proposed method with two other approaches: frame difference [7] and color histogram difference [8]. All the tested approaches operate directly in the YCbCr color space of MPEG domain. In the implementation, a slide transition is detected if the value of frame difference or color histogram difference is a local maximal. For the proposed approach, each time frame is composed of 120 DC image frames. Two adjacent time frames are overlapped by 60 images.

To evaluate the performance, we need to count the numbers of actual transitions N_T, falsely inserted transitions N_I, falsely deleted transitions N_D and correctly detected transitions N_C. The following performance measures are employed

$$\begin{aligned} \mathrm{Recall} &= \frac{N_C}{N_C + N_D} \qquad \mathrm{Precision} = \frac{N_C}{N_C + N_I} \\ \mathrm{Accuracy} &= \frac{N_T - (N_D + N_I)}{N_T} = \frac{N_C - N_I}{N_T} \\ \mathrm{ErrorRate} &= \frac{N_D + N_I}{N_T + N_I} = \frac{N_D + N_I}{N_C + N_D + N_I} \end{aligned}$$

Table 1: Slide transition detection on lecture-A of $129,010$ frames.

	N_c	N_I	N_D	Accr.	Error R.	Prec.	Recall
Proposed Approach	66	1	8	0.88	0.12	0.99	0.89
Frame Diff.	46	6	28	0.54	0.43	0.88	0.62
Color Histo.	13	3	61	0.14	0.83	0.81	0.18

Table 2: Slide transition detection on lecture-B of $122,011$ frames.

	N_c	N_I	N_D	Accr.	Error R.	Prec.	Recall
Proposed Approach	36	4	6	0.76	0.22	0.90	0.86
Frame Diff.	12	4	30	0.19	0.74	0.75	0.29
Color Histo.	4	2	38	0.05	0.91	0.67	0.10

The values of R*ecall*, P*recision* and E*rror Rate* are in the range of $[0, 1]$. Low recall values indicate frequent occurrence of false deletions, while low precision value indicates the frequent occurrence of false alarms. E*rror Rate* puts more penalty to false deletion than false insertion, meanwhile A*ccuracy* has negative value if $N_C < N_I$.

Tables 1 and 2 show the performance comparison of the three tested approaches based on various evaluation methods. As indicated in the tables, our proposed method significantly outperforms two other approaches. Both color histogram and frame difference approaches suffer from low recall. The former approach fails since the color of text captions and the design template of slides are similar. The latter approach, on the other hand, fails because it is equally sensitive to the difference between slides and the motion of foreground objects. As a result, local maximal may not be found when slides are flipped. Our proposed approach, in contrast, focuses only on caption and background changes. In most cases, when there are changes in figures, $\mathbf{E}_b$ will possess large value. Similarly, $\mathbf{E}_c$ will show large value whenever there is a change of discussion topics. In the experiments, false insertions are caused by the sudden change of illumination. False deletions are mainly due to the low contrast between slides as well as the low resolution of video quality. Few transitions are not detected because the main title of adjacent slides is same. In the current implementation, the proposed approach can process approximately 70 frames per second on a Pentium-IV platform.

7. CONCLUSION

We have presented a novel approach to detect slide transitions by utilizing background and caption cues. Experimental results indicate that the approach is robust and capable of operating in real time. Our future works include the recognition and integration of speech and caption information to synchronize audio, videos and electronic slides for more effective indexing.

Acknowledgement

This work is supported in part by RGC Grants HKUST6072/97E and DAG01/02.EG16, and SSRI grant SSRI99/00.EG11.

8. REFERENCES

[1] G. Abowd *et. al.* , "Teaching and Learning as Multimedia Authoring: The Classroom 2000 Project," *ACM Multimedia*, pp. 187-198, 2000.

[2] S. G. Deshpande & J.-N. Hwang, "A Real-time Interactive Virtual Classroom Multimedia Distance Learning System," *IEEE Trans on Multimedia*, vol. 3, no. 4, pp. 432-444, Dec 2001.

[3] L. He *et. al.* , "Auto-Summarization of Audio-Video Presentations," *ACM Multimedia*, pp. 489-498, 1999.

[4] S. X. Ju *et. al.* , "Summarization of Videotaped Presentations: Automatic Analysis of Motion and Gesture," *IEEE Trans on CSVT*, vol. 8, no. 5, pp. 686-696, 1998.

[5] T. F. S. -Mahmood, "Indexing for topics in videos using foils,", *CVPR*, pp. 312-319, 2000.

[6] S. Mukhopadhyay & B. Smith, "Passive Capture and Structuring of Lectures,", *ACM Multimedia*, pp. 477-487, 1999.

[7] B. L. Yeo & B. Liu, "Rapid Scene Analysis on Compressed Video," *IEEE Trans. on CSVT*, vol. 5, no. 6, pp. 533-44, 1995

[8] H. J. Zhang, A. Kankanhalli & S. W. Smoliar, "Automatic Partitioning of full-motion video," *ACM Multimedia System*, Vol. 1, No. 1, pp. 10-28, 1993.

[9] Y. Zhong *et. al.* , "Automatic Caption Localization in Compressed Video," *IEEE Trans. on PAMI*, vol. 22, no. 4, pp. 385-392, 2000.