VIDEO FRAME IDENTIFICATION FOR LEARNING MEDIA CONTENT UNDERSTANDING

Ying Li and Chitra Dorai

IBM T.J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598

E-mail:{yingli,dorai}@us.ibm.com

ABSTRACT

This paper presents our latest work on identifying frame content types for understanding learning media content. In particular, we categorize frames into six classes namely, *slide*, *web-page*, *instructor*, *audience*, *picture-in-picture* and *miscellaneous*, which make up salient narrative modes in learning videos. Various image and video analysis approaches are explored to achieve this task. Preliminary experiments carried out on three recorded seminars have yielded encouraging results. The identification of fine-grained visual content types can assist us in content understanding, access, browsing and searching of generic learning videos.

1. INTRODUCTION

Online learning or Web-based e-learning is rapidly emerging as a viable means for offering customized and self-paced education to students. Many universities and industrial organizations have started offering online education and training programs as an alternative to traditional classroom-based education. As a result, the amount of instructional videos available on corporate intranets and the Internet is dramatically increasing. This paper describes our ongoing efforts on analyzing these learning media content to facilitate automatic content structuralization and annotation for various e-learning applications.

Specifically, this work attempts to understand learning videos such as recorded seminars by identifying the following six visual frame content types: *slide*, *web-page*, *instructor*, *audience*, *picture-in-picture* and *miscellaneous*. As the name implies, the first three categories contain a close-up view of a slide, a web-page and the instructor, respectively. The audience frames, in contrast, refer to those containing a long shot of the meeting room. We call a frame that has an embedded sub-image as a picture-in-picture, which occurs, for instance, when the instructor launches a demo. Note that the frame that has a small inset instructor picture is not considered of this type. Finally, the miscellaneous category accommodates all other currently unconsidered frame types. For illustration purpose, Fig. 1 shows examples of the first five types.

(a) (b) (c) (d) (e)

Fig. 1. Five frame content types: (a) slide, (b) web-page, (c) instructor, (d) audience, and (e) picture-in-picture.

Video content analysis has been studied for decades, yet very few research efforts have reported on the identification of frame content in learning context. The only work that we found was from Haubold and Kender [1], where a decision tree was applied to classify keyframes into six categories including board, class, computer, illustration, podium and sheet. Nevertheless, too many heuristics were implied in that work which makes it impractical to apply this approach to generic learning video content.

This paper proposes a robust hierarchical frame identification scheme for understanding generic learning videos. It consists of the following five modules to process a digital video stream: (1) *homogeneous video segmentation* which partitions the video into cascaded homogeneous segments; (2) *picture-in-picture (PiP) segment identification* which recognizes a PiP frame by exploiting its unique characteristics in terms of content variation and physical locality; (3) *agglomerative segment clustering* which groups segments with similar dominant colors into clusters; (4) *instructor segment identification* which recognizes the instructor frames based on face analysis; and (5) *audience/slide/web-page segment identification* which recognizes each of these three frame types based on line profile analysis. Each module is detailed below.

2. HOMOGENEOUS VIDEO SEGMENTATION

This module aims at partitioning a video into homogeneous segments where each segment contains frames of the same content type. We achieve this goal by developing an approach which is sensitive enough to distinguish frames of distinct content types while tolerating changes within frames of the same type. It proceeds in the following two steps.

2.1. Frame Content Change Detection

A *peak-based* histogram comparison algorithm is designed to detect the content change between neighboring frames at this step. Compared to the popularly used L1/L2-norm or histogram intersection based approaches, the proposed one is able to better tolerate content changes caused by digitization noise, jerky camera motion and illumination changes. It proceeds as follows.

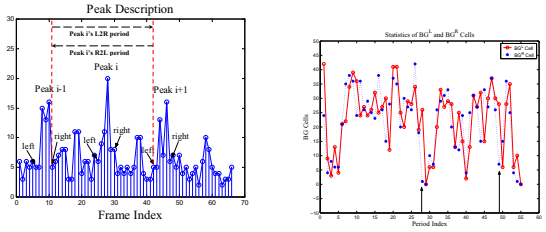
1. Given a frame, we first partition it into a $MD \times ND$ grid, compute an intensity histogram for each cell and identify its peaks. Then, we associate each peak with its component bins (*i.e.,* those bins that contribute to the peak), and term it as a *peak category*.

2. For each frame pair, we compare their corresponding cells by examining their peak categories (PC). Specifically, if major component bins are shared by any peak category of both cells, we claim that they have same content. For simplicity, we call the cell

0-7803-9332-5/05/$20.00 ©2005 IEEE

whose content remains the same as *background* (BG) cell; otherwise, the *foreground* (FG) cell.

3. We quantify the content difference between two frames by summing up the number of their FG cells. The frame content change boundaries are then identified from the video's FG cell distribution based on its detected peak locations. To ease the rest of discussion, for each peak P_i that refers to a video content change point, we denote its left and right flanks by P^i_{left} and P^i_{right} (in unit of frame). We have also defined two processing periods, *L2R period* and *R2L period*, where L2R expands from P^{i-1}_{right} to P^{i+1}_{left}, and R2L in the opposite direction as shown in Fig. 2 (a).



(a) (b)

Fig. 2. (a) Descriptions of a peak, and (b) statistics of BG^L and BG^R cells for one test video.

2.2. Homogeneous Segment Boundary Detection

Based on the frame content change boundaries, we now identify the homogenous segment boundary using the following *bi-directional BG cell tracing* approach.

1. Given peak P_i, we compute a temporal cell histogram CH^L over its L2R period (denoted by $(P^{i-1}_{right}, P^i_{middle}, P^{i+1}_{left})$), where CH^L_j indicates the number of consecutive frames for which cell j's content has remained unchanged since P^{i-1}_{right}. Similarly, we compute CH^R over P_i's R2L period except that we direct the search backwards.

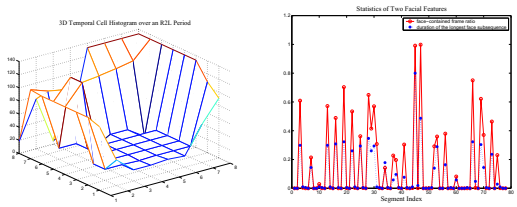
2. Examine if CH^L_j exceeds the duration of $(P^i_{right} - P^{i-1}_{right})$. If yes, it is a BG cell since it remains unchanged regardless of the content transition across the peak. Evidently, the fewer the BG cells, the more likely that P_i refers to a content change between frames of distinct types. Denoting the number of BG cells identified over a peak's L2R and R2L periods by BG^L and BG^R, we plot their distributions in Fig. 2 (b) for one test video. We see that for most of the time, the two values are consistent with each other, *i.e.,* when one is small, so is the other. Two exceptions are observed though (indicated by the black arrows), with both being caused by a camera zooming operation.

3. Identify the peaks that possess both small BG^L and BG^R values, and mark them as the *homogeneous* segment boundaries.

3. PICTURE-IN-PICTURE SEGMENT IDENTIFICATION

This module identifies the segments that contain picture-in-picture (PiP) frames by exploiting the following two facts: 1) a PiP segment normally presents a much larger content variation since the content of the sub-image keeps changing over time; 2) the aforementioned content change is confined to a local image area (*i.e.,* the sub-picture area).

To measure the content variation of a segment, we first identify the number of FG cells for every frame pair, then use their variance as the indicator. Intuitively, the larger the variance, the higher the content variation. In addition, we examine if all FG cells are located within a restricted image area. Fig. 3 (a) plots the CH^R cell histogram calculated over a PiP segment, which clearly shows that its major content change only occurs around the image center. We apply a feature thresholding scheme to identify all PiP segments.



(a) (b)

Fig. 3. (a) The CH^R histogram of a PiP segment, and (b) statistics of two facial features of a test video.

4. AGGLOMERATIVE SEGMENT CLUSTERING

Observing that frames containing instructor, audience, slide and web-page usually have distinct dominant colors, we group segments with similar color tones into clusters in this module.

We extract a frame's dominant color by applying the hierarchical color clustering scheme proposed by Wan and Kuo [2]. Specifically, it applies an octree color quantizer to cluster all image colors into natural groups. Each tree node, in this case, defines a color subspace, and is described by two quantities: the normalized number of pixels that pass through the node (pass-number p) and the average color c (in Luv space) of these pixels. It has also considered various clustering resolution levels, and accordingly, designates the image's dominant colors as the average of the first three nodes that have the largest pass numbers at the coarsest resolution.

In this context, we first represent a frame by its three dominant colors together with their respective pass numbers, and define a segment's dominant colors to be the average of its component frames. The similarity between two segments is then measured by taking both dominant colors and their pass numbers into account. Finally, we apply the agglomerative clustering approach (with group average method) to perform the segment clustering, which stops once the minimum inter-cluster distance exceeds a pre-defined threshold. By intentionally using a small threshold, we are able to identify tightly clustered segments while avoiding false positives at the same time.

5. INSTRUCTOR SEGMENT IDENTIFICATION

This module identifies the instructor segment by exploiting two facial features: face-contained frame ratio fr, and duration of the longest face sub-sequence fd. Specifically, for a segment S, fr indicates the percentage of frames that contain a face, while fd indicates the duration of its longest subsequence within which every frame is face-contained. Intuitively, an instructor segment shall have a higher fr ratio than a segment containing slides or web-pages. The adoption of fd feature, however, comes from the observation that, when a face is truly detected in a frame, it should

remain detectable in multiple subsequent frames due to the content continuity. A face may be falsely identified in one frame, yet the possibility of making the same mistake in a row will be much lower. Fig. 3 (b) plots the distribution of two facial features (normalized) for a test video, which supports the observation very well that if both features reach their local maxima over a segment, it will be a truly instructor segment.

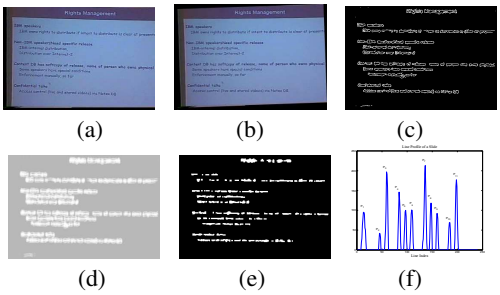
We identify instructor segments by empirically thresholding their facial features. In addition, we check the percentage of detected instructor segments for every cluster C. If it exceeds 0.5, then we designate C as an instructor cluster and consequently, set all of its segments to the instructor type.

6. IMAGE LINE PROFILE ANALYSIS

This module extracts various image and text features from video frames which are subsequently applied to distinguish audience, slides and web-page frames. It proceeds in the following five steps.

1: Image smoothing, sharpening and unwarping. We apply this step to improve the original image quality which may have been reduced in the course of video projection and digitization, and to unwarp the image in case of foreshortened video frames.

An edge-strength guided smoothing approach is first applied to smooth the image which preserve edges at the same time [3]. Then, we apply image sharpening to enhance object edges. Finally, we perform image unwarping [4] to correct foreshortened video frames if necessary (this is currently manually determined). Fig. 4 (b) shows the unwarped version of the slide in (a).



(a) (b) (c) (d) (e) (f)

Fig. 4. (a) A slide, (b) the unwarped slide, (b) the Canny edge map, (c) the neighborhood edge map, (d) the text map, and (e) the line profile.

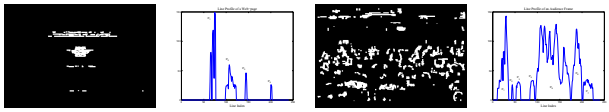
2: Canny edge detection. Given the smoothed, sharpened and unwarped image I, this step performs a Canny edge detection to obtain its edge map [3]. Fig. 4 (c) shows the slide's edge map where all horizontal and vertical straight lines have been removed.

3: Text map extraction. This step locates the image's possible text regions by exploiting the following two facts: 1) a text character has strong vertical and horizontal edges; and 2) text characters are always grouped in words, sentences and paragraphs.

Specifically, we first derive a *neighborhood edge map* (NEM) from I's Canny edge map by setting pixel p's value to the average of its neighboring area. One example is shown in Fig. 4 (d), which clearly shows that pixels residing in a tightly clustered text regions are much brighter than those on line edges. Then, we binarize the NEM map using a fixed threshold (currently set to 8), perform a morphological open operation to fill holes within text lines, and remove image lines that contain extremely fewer number of pixels. Fig. 4 (e) shows the text map that we finally obtained for the slide in (a).

4: Title line detection. This step identifies the image's title line which: 1) is the first text line in the image; 2) is well separated from other texts; and 3) has all of its texts tightly connected together. A line profile analysis is performed to achieve this goal. Specifically, the *line profile* reveals the overall text distribution in an image [5], which is currently obtained by summing up all text pixels in each image row. The line profile for the slide example is shown in Fig. 4 (f).

To detect the title line, we first identify all distinct peaks from the line profile. In the example of Fig. 4, totally 11 peaks are detected, each of which, as we could easily tell by comparing (e) and (f), refers to one text line. The stop valley between two peaks, in this case, corresponds to the space between lines. Then, starting from the first peak, we locate the line pointed by the peak tip, and examine the average distance between every of its two adjacent text pixels. In this context, a large distance will indicate that the line has sparsely distributed texts which decreases its possibility of being a title line. Consequently, we ignore the current peak and continue with the next.



(a) (b) (c) (d)

Fig. 5. (a) Text map and (b) line profile, of the web-page in Fig. 1 (b); (c) text map and (d) line profile, of the audience frame in Fig. 1 (d).

In the slide example, the line pointed by the first peak makes a perfect title line, so does in case of the web-page (Fig. 1 (b)) whose text map and line profile are shown in Fig. 5 (a) and (b), respectively. In contrast, for the audience frame in Fig. 1 (d), no qualified title lines could be found as evident from its text map and line profile in Fig. 5 (c) and (d). These results conform to the ground truth very well.

5: Feature extraction. Based on the above analysis, we extract the following five features from every video frame: coordinates of the rectangle that bounds all text pixels (COR), the number of horizontal/vertical lines that are longer than half of the image width/height (HL, VL), the distance from the top frame edge to the title line (TD), and the width of the title line (WT). Especially, feature COR reveals the texts' physical distributions, while HL and VL examine the existence of long straight lines in displayed content. Finally, feature TD indicates the title position and WT embodies the text size.

7. AUDIENCE, SLIDE AND WEB-PAGE SEGMENT IDENTIFICATION

We now distinguish the audience, slide, and web-page segments by making use of the following three observations: 1) the audience frame usually has scattered text pixels without a title line; 2) multiple horizontal lines corresponding to the address and title bars are usually detected in web-pages which is not true for the audience frame; and 3) slides of the same presentation should have title lines at similar physical locations with similar font sizes, which is not likely with web-pages.

A two-step process is performed to fulfill this task. First, we derive five segment-level features from its component frames based on a voting scheme: the text distribution pattern, the number of horizontal and vertical lines (HL and VL), the distance from the title line to the top frame edge (TD) and the title width (WT). Note that a quantization scheme has been applied to tolerate minor measurement variations for the last two features. Then, we derive five cluster-level features from its component segments (excluding the outliers): the text distribution pattern, the averages of HL and VL, and the variation of TD and WT.

Now, if a cluster has text spread without vertical, horizontal or title lines, we set its segment label to the audience type. Otherwise, we examine the variations of TD and WT. If both of them are smaller than pre-defined thresholds, it shall contain slides; otherwise, it contains web-pages if the average number of horizontal lines exceeds a threshold. Finally, the miscellaneous class accommodates all unclassified clusters, leaving room for further future classification. Note that thresholds applied in this decision process are all empirically determined.

8. EXPERIMENTAL RESULTS

Preliminary experiments were carried out on three learning videos to validate the proposed approach. All three videos were taken from a collection of pre-recorded e-seminars at IBM Research with an average duration of 90 minutes. Precision and recall rates were evaluated for the detection of homogeneous segments, as well as the identification of segments containing the five content types. Table 1 lists the ground truth for each test video where column 1 specifies the total number of homogeneous segments while the rest indicates the number of segments of each type.

Table 1. Grountruth of the three test videos.

	H. Seg	PiP	Inst.	Aud.	Slide	Webpage
Video 1	16	3	5	0	3	4
Video 2	78	0	26	22	18	9
Video 3	67	0	18	21	21	0

The homogeneous video segmentation performed well on all three videos except that in Video 1, two neighboring segments, which contain web-pages and instructor respectively, were mistakenly returned as one due to the extremely slow content transition. It also fails to separate two slide segments from their neighboring web-pages in Video 2. On average, we achieved 100% precision and 97.1% recall rates.

The picture-in-picture frame type was only observed in Video 1, which was successfully recognized. In addition, we achieved 93% precision (100% recall) rate on detecting instructor segments with only five false alarms in total. Specifically, three of them were caused by human faces in web-pages and the host's face, while the other two resulted from false inclusion of two audience segments in the instructor clusters. This imperfect clustering also led to three false negatives in identifying the audience segments, which resulted in in 97% precision.

In the case of slide segments, perfect identification was achieved for the first two videos with one false alarm in Video 3 due to the false inclusion of an audience segment in slide clusters. Finally, all four web-page segments were correctly identified in Video 1, whereas two web-page segments in Video 2 were mistaken as instructor type due to the existence of human faces, thus leading to 100% and 89% precision and recall rates. No web-pages were found in Video 3.

Fig. 6 shows the temporal distributions of the five frame content types observed in test videos. Each vertical bar indicates one segment whose width is proportional to its duration and whose color indicates its content type. Specifically, we use red, green, blue, cyan and black to represent slide, web-page, instructor, audience and picture-in-picture segments, respectively (note that most audience segments are too short to be visible in the figure). From the figure we see that, compared to Video 1, the other two are better edited where much shorter segments nicely alternate between different content types following a pre-determined pattern. We also observe that *slide* and *instructor* are the two main content types in these test videos, which should be true for most presentation-assisted learning videos.

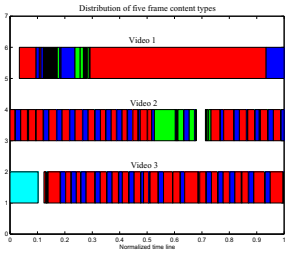


Fig. 6. Distributions of the five frame content types in test videos.

9. CONCLUSION AND FUTURE WORK

A hierarchical frame content identification scheme is presented in this paper which recognizes five major visual content types in learning videos. Compared to reported work, the features we exploited to achieve this task are more general and could thus be applied to videos in various learning domains. Many applications such as mobile e-learning and e-learning content management could be developed based on the proposed work. Currently, we are exploring more effective features to distinguish slides from web-pages, as well as considering other possible content types such as whiteboard. We will also conduct extensive experiments to validate the performance of the proposed system.

10. REFERENCES

[1] A. Haubold and J. Kender, "Analysis and interface for instructional video," *ICME'03*, 2003.

[2] X. Wan and C. C. Kuo, "A new approach to image retrieval with multiresolution color clustering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 6, pp. 705–712, 1998.

[3] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision, Second Edition*, Brooks/Cole, 2002.

[4] S. Mukhopadhyay and B. Smith, "Passive capture and structuring of lectures," *ACM Multimedia'99*, 1999.

[5] B. Erol, J. Hull, and D. Lee, "Linking multimedia presentations with their symbolic source documents: Algorithm and applications," *ACM Multimedia*, pp. 498–507, 2003.