Summarization of Visual Content in Instructional Videos

Chekuri Choudary, *Student Member, IEEE*, and Tiecheng Liu, *Member, IEEE*

***Abstract*—In instructional videos of chalk board presentations, the visual content refers to the text and figures written on the boards. Existing methods on video summarization are not effective for this video domain because they are mainly based on low-level image features such as color and edges. In this work, we present a novel approach to summarizing the visual content in instructional videos using middle-level features. We first develop a robust algorithm to extract content text and figures from instructional videos by statistical modelling and clustering. This algorithm addresses the image noise, nonuniformity of the board regions, camera movements, occlusions, and other challenges in the instructional videos that are recorded in real classrooms. Using the extracted text and figures as the middle level features, we retrieve a set of key frames that contain most of the visual content. We further reduce content redundancy and build a mosaicked summary image by matching extracted content based on K-th Hausdorff distance and connected component decomposition. Performance evaluation on four full-length instructional videos shows that our algorithm is highly effective in summarizing instructional video content.**

***Index Terms*—E-learning, instructional video analysis, key frame selection.**

I. Introduction

SUMMARIZATION of instructional videos is of great importance for e-learning and distance learning [1]. Unlike professionally edited videos like news, sports, and documentary videos, the instructional videos recorded in real classrooms are highly redundant in visual content. Therefore, it is imperative to develop effective content-summarization techniques for this category of videos. In this paper, we refer to the **visual content** in instructional videos as the handwritten text, formulae, and drawings on board. The summary of the visual content is useful in three aspects. First, it provides an overview of the written content in a lecture. Second, the extracted text and figures in the visual summary can be used for handwritten recognition, indexing, and retrieval. Third, in the application of video streaming over low bandwidth networks, the video quality can be enhanced by using the visual summary as a substitute for original video frames.

In a real teaching environment, instructors may use a variety of narrative forms, such as electronic slides (e.g., PowerPoint), handwritten slides, whiteboard, and chalkboard presentations. Nevertheless, the presentation using chalkboards remains one of the widely adopted narrative forms, especially in math-intensive courses. In this paper, we focus on the instructional videos of board presentation and present our methods for summarizing the content written on the chalkboard. The work presented in this paper can be used as a preprocessing step to facilitate the research on handwritten recognition, video event detection and structuring [2], [3], and indexing and retrieval of instructional videos [4].

In the previous works [5]–[10], the video key frames that contain significant visual content are retrieved as video summaries, to provide an overview of digital videos [5]. These key frame selection techniques generally measure the content similarity among video frames based on low-level image features (e.g., colors, edges, shapes). Particularly, image color histogram is widely used for detecting content change in the past works [5], [8]–[10]. Some other features are also used, for example, the wavelet decomposition coefficients [6] and MPEG coefficients [7].

In [11], [12], motion-based features are used for key frame selection. In [11], motion activity is considered as a measure of summarizability, and more key frames are retrieved from the video segment of high motion activities. In [12], video motion is measured and key frames are retrieved at local minima of the motions to avoid the frames with motion blur. However, in instructional videos, motion analysis does not provide much clues regarding the content significance of video frames, because the motion-based features do not reflect the written content on boards.

To retrieve the key frames that represent the written content on boards, the selected image features should reflect the lecture content. Unfortunately, low level features are not sufficient for representing the visual content in instructional videos. Two video frames can have similar text content on the board but drastically different image histograms and edge maps. The progressive development of content on the board also makes the low-level features ineffective for inter-frame matching. In this paper, the written content on the boards is used as the middle-level feature for content summarization. We not only retrieve the key frames that contain significant content, but also match and mosaic the key frames to reduce content redundancy and to build compact visual summaries of instructional videos.

A. Challenges in Content Analysis of Instructional Videos

The first step to summarize instructional video content is to extract the handwritten text and figures on the boards. However, this task is not trivial and is different from text detection in other video domains, in the following three aspects.

First, most existing works [13]–[16] on video text detection are developed mainly for optical characters that have fixed fonts and clear text lines. They are suitable to handle well-defined text

Manuscript received July 3, 2006; revised June 14, 2007.

C. Choudary and T. Liu are with the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, 29208 USA (e-mail: choudary@cse.sc.edu; tiecheng@cse.sc.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2007.906602

such as the enclosed captions in videos, but not the handwritten characters and figures that vary significantly in size and edge density. In [13], [14], the bounding boxes of text lines are first detected. However, in instructional videos, there are no clear text lines in the handwritten figures and drawings. Purely edge based methods [15], [16] are not quite suitable to detect chalk pixels on the boards, for three reasons. 1) The handwritten characters, drawings, and figures on the boards do not show uniform edge density. 2) Unlike the overlaid video captions, the chalk pixels have weak contrast with the board background. 3) Because of chalk dust and the irregularity of the board, the board regions contain many noisy edges.

Second, the text and figures on the boards cannot be extracted by a trivial thresholding operation. In real classrooms, the board is not always "black": It may be dark green, dark blue, or any other dark color; in some light conditions, the board may even show light gray color. The luminance and the color of the board regions vary among video frames due to the change of light conditions. Within a frame, the pixels in the board regions have significant variation in luminance due to the wear of the board and light reflection. During a class session, instructors may erase some chalk content, and the chalk dust left by erasures causes the change of luminance values, making content extraction more difficult.

Third, in instructional videos, camera movements and occlusions are very common and unavoidable. The board content and the classroom activities are usually recorded **nonintrusively** by a camera (or cameras) mounted at the back of the classroom. Due to the limitations of camera resolution and the dimensions of the board, the cameras usually can only capture a portion of the board. Constrained by the nature of board presentations, instructors unavoidably occlude some content while they write on the board. Although there is extensive research on text detection from printed, scanned, or camera-recorded documents [17], [18], the effects of occlusions and the noise in instructional videos still cannot be effectively handled.

B. Other Related Works

Previous research has investigated video summarization in different presentation formats, for example, the transparency-slides [19] and the handwritten-slides [20], [21]. But the summarization methods for the slide videos are not applicable to board presentation videos because they do not address the issues of camera movements and excessive occlusions. Analysis of PowerPoint presentation videos was given significant attention in the past several years [22]–[28]. However, the content in PowerPoint slides is still more close to that in printed documents than the handwritten content on boards, and the visual content can also be accessed via the PowerPoint files.

In [29], Onishi *et al.* presented the research on detecting blackboard content based on background subtraction. A similar approach was employed in [30] for analyzing the handwritten content on white board. However, Onishi's work is restrictive because it needs the registration of a clear board background with no chalk content and occlusions. In the classrooms that have multiple board panels, the switching of board panels makes the image registration technique [29] degrade in performance. In [31], Heng and Tian developed a system for content enhancement of instructional videos. However, the content is detected based on the thresholding of pixel intensity, which is prone to false content pixels because of chalk erasure and luminance variation in board regions.

In [32], a framework was developed for real-time streaming of instructional videos. The content analysis technique adopted in this framework is based on edge detection. In [33], Mittal *et al.* provided a content-based compression technique for chalkboard videos. Wallick *et al.* [34] presented an approach for grouping the content text on chalkboards into homogeneous regions. In [35], Yokoi *et al.* developed a technique for detecting the events in chalkboard videos and generating a time shrunk video for quick content view. Wang *et al.* [36] developed a method for editing the instructional videos based on posture, gesture and text analysis. In all the above-mentioned works, the instructional video content is only coarsely estimated, but not accurately extracted and summarized.

There are several methods and systems developed for content analysis of whiteboard videos [37]–[40]. However, the technical challenges in the summarization of chalkboard videos are different from those in whiteboard videos, for the following two main reasons. First, the accumulating chalk dust and wear of the board causes significant nonuniformity in the chalkboard regions. The chalk pixels are "merged" with the board background, and the chalkboard background pixels do not have high and distinctive luminance values as in the case of white board videos [37], [38]. Second, there are more camera movements and occlusions in chalkboard videos than in whiteboard videos, because the chalkboards usually have larger areas and in some cases, have multiple board panels. In the usual classroom settings, an instructor unavoidably occludes a part of the chalkboard regions. While in the regular conference meeting settings, a speaker only occludes the whiteboard for a short time period when he is writing on the whiteboards.

In the existing works [41]–[46], audio analysis and speech recognition were exploited for summarization and retrieval of instructional videos. These techniques did not consider the written or printed visual information in instructional videos. In [47], Erol and Li provided an in-depth survey of the state-of-the-art research on e-lecture and e-meetings.

C. Our Approach

The proposed approach to the summarization of instructional videos consists of three steps: content analysis, content-sensitive temporal sampling, and content matching and mosaicking. 1) In the first step, we extract the handwritten content on the chalkboard with high accuracy. 2) In the second step, we quantify the visual content in the video frames and then develop an algorithm to select the content-rich frames as key frames. 3) In the third step, we develop matching techniques to remove redundancy among the key frames. Finally, the nonredundant frames are stitched together to provide an abstraction of the visual content in the video.

The three processing steps are presented in Sections II–IV, separately. Section V shows the experimental results, and Section VI gives a short conclusion of the work and future directions.

D. Originality and Contributions

In this work, we present a summarization method specially developed for chalkboard presentation videos. The contributions of this work are in the following four aspects.

1) The summarization method is developed with the consideration of the nature of classroom videos. It takes into consideration several factors such as light reflection and nonuniformity of the chalkboard, chalk dust caused by erasures, multiple board panels, occlusions by the instructors, camera movements, and image and video noise. It integrates some existing techniques and adapts them to this application.
2) The proposed summarization approach is based on middle level features. The handwritten content on the chalkboard is used for content matching in the instructional video frames.
3) A content-sensitive temporal sampling technique is presented for extracting the key frames from instructional videos with the consideration of excessive occlusions and camera movements. The proposed matching and mosaicking technique is able to handle content redundancy despite camera movements.
4) This paper presents a new performance-evaluation method for summarization algorithms on instructional videos. The evaluation method is based on the content elements, which are introduced as the basic units of instructional video content.

II. Content Analysis of Instructional Videos

In this section, we briefly present the method we developed for extracting the content from chalkboard video frames. The reader is referred to [48] for complete details of the content extraction method.

As instructional videos are highly redundant in visual content, it is not necessary to process each frame. Our experiments indicate that at a fixed sampling rate of one frame per 150 frames (5 s in MPEG coded videos), the resulting frames still contain all the text and figure content. So in our experiments, we use this fixed sampling rate to retrieve a subset of images for content analysis.

The pixels in the board regions vary in color and luminance, both spatially across the board background and temporally during a video session. In our work, we first apply the mean-shift segmentation developed by Comaniciu *et al.* [49] to partition instructional video frames into over-segmented regions. Then we refine the segmentation using a probabilistic model and locate the board regions.

We segment the sampled frames using the Mean-Shift method [49] and find the largest regions. Then we estimate the distribution parameters for the pixels in the largest regions. Denote $\bar{a}, \bar{b}, \sigma_a, \sigma_b$ as the averages and the standard deviations of the a and b color components of all the pixels in the largest regions. For the luminance component, we estimate the range $[L_{\text{low}}, L_{\text{high}}]$ that covers 95% of the largest region pixels. For each image pixel $\mathbf{x}$, we compute its probability of being a board region pixel. This probability $Pr(\mathbf{x})$ is modelled as

$$Pr(\mathbf{x}) = Pr^{(L)}(\mathbf{x})Pr^{(a)}(\mathbf{x})Pr^{(b)}(\mathbf{x})$$

where $Pr^{(L)}(\mathbf{x}) = 1$ if the luminance of $\mathbf{x}$ is in $[L_{\text{low}}, L_{\text{high}}]$; otherwise, $Pr^{(L)}(\mathbf{x}) = 0$. $Pr^{(a)}(\mathbf{x})$ and $Pr^{(b)}(\mathbf{x})$ are modelled as gaussian probabilities with means $\{\bar{a}, \bar{b}\}$ and standard deviations $\{\sigma_a, \sigma_b\}$.

Suppose a video frame is partitioned into multiple regions using the Mean-shift segmentation method [49]. The probability of a region R_i being a board region is calculated as

$$Pr(R_i) = c \sum_{\mathbf{x} \in R_i} \frac{Pr(\mathbf{x})}{|R_i|}$$

where c is a normalization factor that scales $Pr(R_i)$ to the range of $[0, 1]$. $c = 2\pi\sigma_a\sigma_b$. $|R_i|$ measures the total number of pixels in the region R_i. We empirically choose the threshold as 0.3. If $Pr(R_i) > 0.3$, we consider R_i as a content region and merge it with the other content regions; otherwise, we classify it as an irrelevant region. All the board regions are merged together to form the board background:

$$R = \bigcup \{R_i | Pr(R_i) > 0.3\}.$$

The above probabilistic model treats the luminance component (L) and color components (a and b) separately. By allowing a large margin for the luminance variation and modelling color components as Gaussian, this model can largely handle the difficulties of nonuniform board color, light condition changes, and the chalk erasures. The extraction of text and figures in the board background is based on top-hat morphological filtering followed by adaptive thresholding.

The details of adaptive thresholding technique are as follows. We divide the morphologically processed board background into blocks of 16×16 pixels. For each block, we compute a threshold to extract content chalk pixels. Let μ be the number of edge pixels calculated using the Sobel operator in a block. The binarization threshold is selected as $(1.2\ \mu)$-th largest luminance value of the pixels in that block. All the pixels with luminance value above the threshold are extracted as chalk content pixels. The content text and figures are further thinned to have one pixel width.

Fig. 1 shows the sample results of board background separation and content extraction in instructional videos. The thinned images are used for content matching and summarization of visual content in the video.

III. Content-Sensitive Temporal Sampling of Instructional Videos

In instructional videos, the content text and figures are progressively developed. Hence, the visual content in an hour-long video can be summarized by matching the extracted content and retrieving a few tens of frames that are disjoint in content. However, matching the extracted content among successive frames throughout the video is computationally intensive. As the content in the videos is highly redundant, it is not necessary to process all the frames.

In this section, we present a content-sensitive sampling method to extract the key frames that contain most of the content in the instructional videos. This step also reduces the computational expense for further matching. First, using the number of chalk pixels in each frame as a heuristic measure of

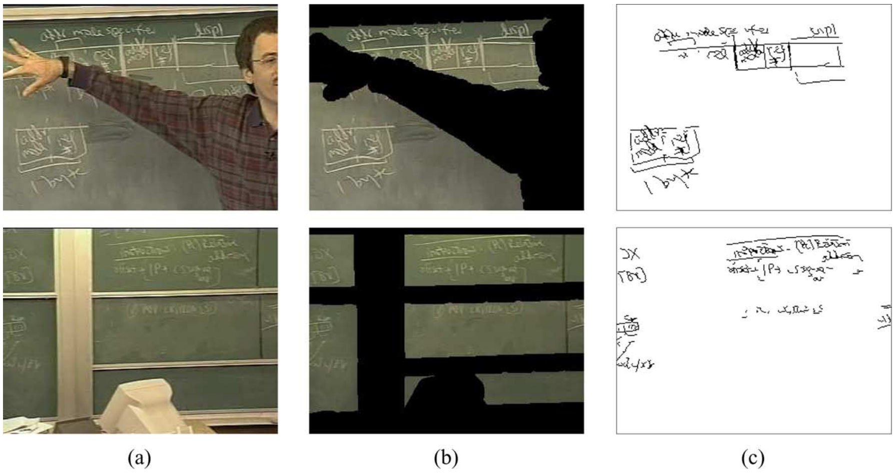


Fig. 1. Sample results of board background separation and content extraction in instructional videos. (a) original frames; (b) the separated board regions and irrelevant regions (marked in black); (c) content extraction result. As shown in the figure, the board regions and content are accurately extracted.

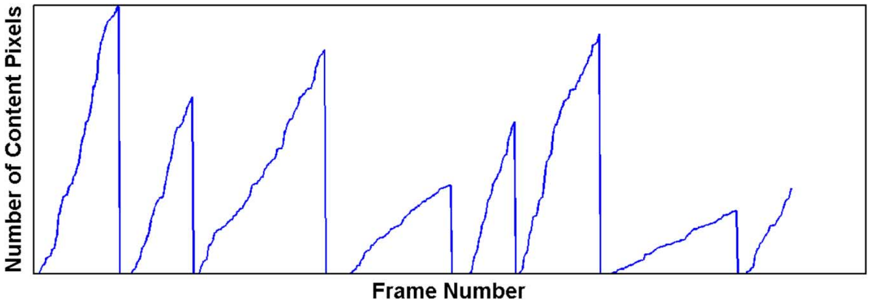


Fig. 2. Illustration of a hypothetical content fluctuation curve. The number of content pixels are monotonically increasing within a teaching topic and drops to zero at the end of the topic.

video content, we define a content fluctuation curve that reflects the variation of actual content in the video. We discuss the nature of chalkboard presentations and then present a shifting window technique to find the optimal temporal positions of the frames that contain most of the content in the video.

A. Content Fluctuation Curve

After extracting the content from instructional video frames, the number of chalk pixels in each frame can be used as a measure of the content in the frame. The change of content pixel numbers across the frame sequence reflects the fluctuation of video content. We refer to the plot of content pixel numbers against time as the *content fluctuation curve* of the instructional videos.

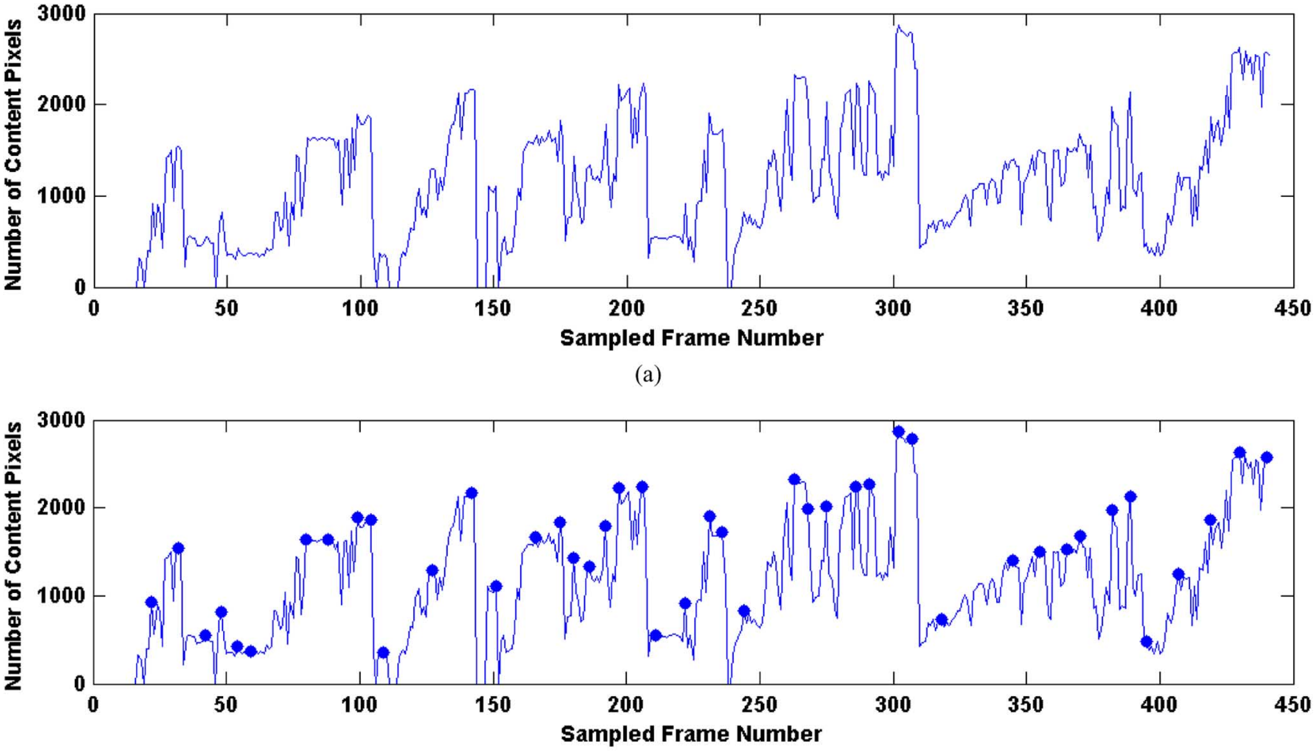
The content fluctuation curve in a hypothetical situation is shown in Fig. 2. During a chalkboard presentation, an instructor usually starts writing on a blank region of the board. Then, the instructor continues writing on board while discussing the course content. So in the content fluctuation curve, the "content level" starts with a low value of content pixels at the beginning of the presentation, then it increases monotonically because of additional content on board, finally it drops to low value when the instructor moves to another blank board region or slides another panel to start a new topic.

However, in real classroom presentations, the number of chalk pixels are not monotonically increasing within the discussion of a topic. The instructors may unavoidably occlude a portion of the content on the board. The content levels may drop because of the camera movements. The content fluctuation curve of an instructional video captured in real classroom is shown in Fig. 3(a). Note that, the curve contains a lot of dips in content levels because of occlusions and camera movements. The peaks in the curve correspond to the frames that contain most of the visual content.

Considering the nature of the classroom presentations and the characteristics of real content fluctuation curves, we develop a shifting window algorithm to locate the local maxima in the curve. The corresponding frames are retrieved as key frames.

B. Shifting Window Algorithm for Content-Sensitive Temporal Sampling

Let $f_1, f_2, \ldots, f_n$ be the n processed frames in an instructional video. Let $\{g(m), m = 1, 2, \ldots, n\}$ be the content fluctuation curve of the video. $g(m)$ represents the number of content pixels in f_m. Obviously, $g(m) \geq 0$. The algorithm for locating local maxima in the content fluctuation curve is shown in Table I. In this algorithm, we introduce a temporal window of k frames. γ records the previous local maximum of content pixels in the temporal window. At first, we align the window with the beginning of the curve, i.e., the data in the window is $\{g(1), g(2), \ldots, g(k)\}$. We find the video frame that has the largest number of content pixels in the window. Suppose $g(t) = \max\{g(1), g(2), \ldots, g(k)\}$, we assign $g(t)$ to the variable γ and shift the window to contain $\{g(t), g(t+1), \ldots, g(t+k-1)\}$.



(a)

(b)

Fig. 3. Content sensitive sampling technique extracts the frames that contain most of the visual content in the instructional video as key frames. (a) Content fluctuation curve of a real instructional video obtained after content extraction; (b) the positions of the key frames (shown as solid circles) obtained by our shifting window technique.

TABLE I
SHIFTING WINDOW ALGORITHM FOR RETRIEVING THE KEY FRAMES

```
Algorithm: retrieving candidate key frames
$i \leftarrow 1$
$\gamma \leftarrow 0$
    find $t$ such that $g(t) = max\{g(i), g(i+1),$
        $\cdots, g(i+k-1)\}$;
    if $g(t) \leq \gamma$
        output $f_t$ as a candidate key frame;
        $\gamma \leftarrow 0$;
        $i \leftarrow i + k$;
    else
        $\gamma \leftarrow g(t)$;
        $i \leftarrow t$;
```

If $g(t)$ is still the maximum value in this window, we consider $g(t)$ as a local maximum point and register f_t as a candidate key frame; otherwise, we update γ and shift the window to another new position. This process continues until the end of the video.

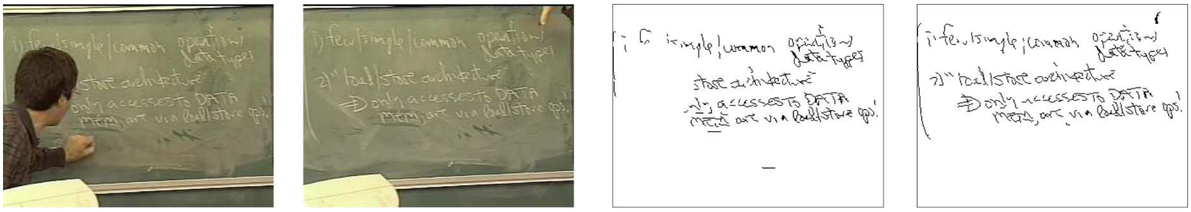
The motivation for the shifting window technique is as follows. The number of content pixels are not monotonically increasing within the presentation of a teaching topic. During the development of the content on the board, the content pixels are mostly increasing, but they may also drop because of the occlusions of the content area and camera movements. In the process above, we experimentally set the window size k to be 5. As the video is sampled at the rate of one frame per 150 frames, the window size is about 25 s in MPEG coded videos. This window size is large enough to handle most of the dips caused by occlusions and camera movements, yet small enough to capture all teaching topics. Since the handwritten content on the board is developed in a progressive manner, by comparing the current maximum with the previous one, we only keep the frames that have significantly large amount of content pixels in each teaching topic.

Fig. 3(b) shows the positions of the key frames for an instructional video. All the positions identified are peaks in the content fluctuation curve. The dips that are smaller than the temporal window size are properly handled. These key frames contain most of the visual content in the video and are fewer in number. Thus they are more suitable for content matching.

IV. MATCHING AND MOSAICKING OF INSTRUCTIONAL VIDEO CONTENT

The sampling method in the previous section is based on the heuristic measure of the number of chalk pixels in video frames. It is introduced mainly to retrieve the frames containing most of the visual content as key frames and to reduce the computational expense for further matching. The spatial locations of the chalk pixels in the frames are not considered in the process. Since the key frames obtained could still be redundant in content, we develop techniques to remove redundancy by matching the extracted content in the key frames. Thus we generate a compact set of key frames that are highly disjoint in content.

There exist two types of content redundancy in the key frames. Case (1): the content of one key frame is completely contained in the previous or the next key frame. In this case, it is required to identify and remove the redundancy. Case (2): there is an overlap of content between two key frames. This usually happens when the cameras cannot capture all the


1) few/simple/common operations/data types
2)" load/store architecture
⇒ only accesses to DATA MEM, are via load/store ops!

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

Fig. 4. K-th Hausdorff distance based content matching is able to detect redundant frames in spite of occlusion, camera jitter, and small inaccuracies in content extraction. (a) and (b) are two successive key frames of a video; (c) and (d) are the extracted content of (a) and (b). Using the K-th Hausdorff distance based matching, frame (a) is identified as a redundant key frame because its content is totally contained in frame (b).

content on the board. In the latter case, we need to identify the overlapping content and mosaic the content together to form a **mosaicked frame** that is more informative and appealing for summarization purposes.

Note that, in both the cases of content redundancy mentioned above, the spatial variations of the content are involved. In Case (1), though the content of one key frame is contained in the other frame, there could be small spatial variations because of camera jitter or significant spatial variations caused by camera panning. In Case (2), significant spatial variations are involved because of camera panning.

We develop a two-step approach with increasing computations to match the content in the key frames. Our matching technique is able to effectively handle the spatial variations caused by camera jitter as well as camera panning. It is also robust to the noise in the videos. In the first step, we match the extracted content between adjacent key frames using K-th Hausdorff distance. Some of the redundant key frames whose content are subsets of other key frames are identified and removed. This step effectively handles the camera jitter. In the second step, we handle the camera panning by basing our matching on connected components. This matching technique is effective in two ways. First, the key frames with the same content but significant spatial variations are identified and the redundant one is removed. Second, the key frames with overlapping content are identified and mosaicked together.

A. *Hausdorff-Distance-Based Matching*

Let f_i and f_j be the sets of extracted chalk pixels in two adjacent key frames. We define the directed distance from f_i to f_j as

$$d(f_i, f_j) = K^{th} \max_{\mathbf{x}\in f_i} \min_{\mathbf{y}\in f_j} \|\mathbf{x}-\mathbf{y}\|.$$

The definition above is based on the K-th Hausdorff distance proposed in [50]. $\|\mathbf{x}-\mathbf{y}\|$ measures the distance between two pixels $\mathbf{x}$ and $\mathbf{y}$. The computation of the directed distance between two key frames is as follows. For each content pixel $\mathbf{x} \in f_i$, we compute its nearest distance to the set f_j, i.e., $\min_{\mathbf{y}\in f_j} \|\mathbf{x}-\mathbf{y}\|$. Then we rank all the distances and find the K-th largest one. We experimentally set the K-th to be the distance that is larger than 80% of all the distances. We choose a threshold $\delta = 4$ (for frames of size 320 × 240). If $d(f_i, f_j) \le \delta$, we consider f_i as mostly contained in f_j.

We make a sequential scan of all the key frames. For two adjacent key frames, if the content of one is a subset of the other, we remove it. We continue this process until the content distances between all the key frames are greater than δ and no more key frames can be removed.

The reasons for content matching using the K-th Hausdorff distance are as follows. First, it identifies the redundant key frames in spite of occlusion. For one key frame to be redundant, its content pixels are within a distance of δ from those in the other key frame. Second, it can handle small location variations of the content pixels caused by camera jitter. Third, it is robust to small amounts of noise (false alarms) in the content extraction process. By choosing the K-th as 80%, we are able to surpass the noise and match the content.

Fig. 4 shows the redundant key frame detected using the Hausdorff-distance-based matching. The content in Fig. 4(a) is partly occluded by the instructor. There is a small variation in spatial locations of the content pixels in two key frames because of camera jitter. Further, there are a few false alarms in the extracted content. The Hausdorff-distance-based matching is able to detect and remove such redundancy while preserving the visual content in the video.

B. *Connected-Component-Based Matching*

The content matching using the K-th Hausdorff distance detects the redundant key frames without considering significant spatial variations of instructional content. However, some key frames have the same content that appears in different spatial locations. In some other key frames, there is a partial overlap of the content. As significant spatial variations are involved in these two cases, it is required to search for a translation to align the key frames for comparison and mosaicking.

In the cases of content overlap, mosaicking the actual key frames is computationally expensive. Moreover, because of chalk dust, light condition changes and the instructor's movements, the viewing experience may degrade if pixel-level mosaicking is employed. So, in our summarization approach, instead of mosaicking the key frames, we mosaic the extracted content in the key frames.

In previous research [13], [14], the bounding boxes of text lines were first detected and then tracked in subsequent frames. The tracking is based on matching the features of the bounding boxes such as projection profiles and the sum of squared differences. Using video text lines or words as the units for matching is not suitable for instructional videos of chalk board presentations, for the following reasons. First, instructors may use a lot of formulae, figures, and tables apart from text. Second, the text on boards is handwritten, and hence the bounding boxes of text lines or words are not always clearly defined and separable.

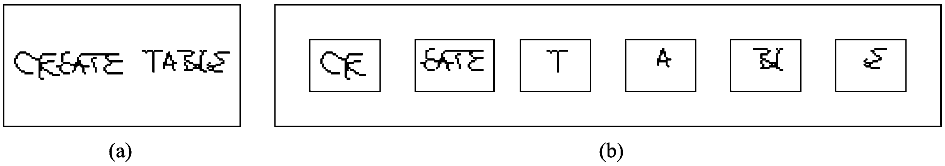

(a) (b)

Fig. 5. Decomposition of instructional video content into connected components. (a) The text extracted from an instructional video frame; (b) the connected components that constitute the text in (a). The connected components are used as the basic units for matching content text and figures in instructional videos.

Here we propose an efficient and robust technique for instructional video content matching using "connected components" as the basic units.

In the content extraction step, we extract the chalk pixels and thin the strokes to one pixel width. A connected component is a set of chalk pixels that are spatially connected in the thinned binary images. Fig. 5 shows an example of decomposing instructional video content into connected components. Using the connected components as units for matching the content is advantageous because, unlike text lines or words, they can be viewed as the basic units for any type of instructional video content (e.g., text, formulas, figures, and tables). Also, decomposing a frame of chalk pixels into connected components is computationally efficient.

The connected-component-based matching consists of two steps. First, we search for similar connected components in two adjacent key frames and record the related translations as candidate translations. In the second step, we align the two key frames with each candidate translation and measure the number of corresponding pixels in the overlapping region. Based on the percentage of matching pixels, we determine whether one key frame is redundant and whether there is a content overlap.

1) Finding the Matching Pairs Among Connected Components: Let f_1 and f_2 be the sets of the chalk pixels in two adjacent key frames. These key frames are the ones retained after removing the redundant frames using Hausdorff-distance-based matching. Suppose f_1 is decomposed into connected components $\phi_i, i = 1, \ldots, m$ and f_2 is decomposed into connected components $\psi_j, j = 1, \ldots, n$. Obviously, $f_1 = \cup_i\{\phi_i\}$ and $f_2 = \cup_j\{\psi_j\}$. For each connected component ϕ_i and ψ_j, we compute the number of chalk pixels (denoted as $|\phi_i|$, $|\psi_j|$) and their geometric centers (denoted as $C(\phi_i)$, $C(\psi_j)$).

For each pair of connected components (ϕ_i, ψ_j), we calculate the correspondence between the components ϕ_i and ψ_j, as follows.

1) If $||\phi_i| - |\psi_j|| > \sigma$, where σ is an experimental threshold of 25, there is no matching between ϕ_i and ψ_j. This is based on the assumption that two corresponding connected components should have similar number of chalk pixels if only Euclidean transformations are involved.

2) If $||\phi_i| - |\psi_j|| \leq \sigma$, we check the correspondence between the pixels in the components based on Hausdorff-distance-based matching. First, the two components are aligned so that their geometric centers coincide. Define $T(\phi_i, \psi_j)$ as the translation that transforms $C(\phi_i)$ to $C(\psi_j)$. Let ϕ_i^T be the transformed set of the connected component ϕ_i. We compute the directed distances between the two sets ϕ_i^T and ψ_j, i.e., $d(\phi_i^T, \psi_j)$ and $d(\psi_j, \phi_i^T)$, where $d(\cdot)$ is the directed K-th Hausdorff distance as defined in Section IV-A. If both distances are less than a threshold of 4 pixels, we consider that the two connected components are similar and record the related translation $T(\phi_i, \psi_j)$ as a candidate translation; otherwise, we consider that (ϕ_i, ψ_j) is not a matching pair of connected components.

Using the above process, the number of connected component pairs that need to be checked for Hausdorff-distance-based matching is at most mn. In reality, most of the pairs do not pass the first test of similar number of chalk pixels. Only a small percentage of connected component pairs need to be checked with Hausdorff-distance-based matching. Therefore, the number of the matching pairs of connected components is significantly less than mn.

2) Checking Content Redundancy by Analyzing the Candidate Translations: The candidate translations relating to the matched pairs of connected components are further analyzed to check the amount of overlapping content. Some of these candidate translations are false positives because two connected components may be similar though they are not really corresponding connected components.

The number of candidate translations is at most mn. However, in real experiments, this number is very small because most of the connected component pairs do not match. For each candidate translation, we align the two frames with the translation. Then we measure the number of matching content pixels in the overlapping region using the Hausdorff-distance-based matching technique introduced in Section IV-A. We compute the number of matching pixels for each candidate translation and pick the one that results in the maximum number of matching pixels.

Suppose t is the maximum number of matching pixels in f_1 and f_2. If $t \geq 0.8 \, \min\{|f_1|, |f_2|\}$, we consider that one of the $\{f_1, f_2\}$ is redundant and remove the frame with lesser number of content pixels. If $0.25 \, \min\{|f_1|, |f_2|\} < t < 0.8 \, \min\{|f_1|, |f_2|\}$, we consider that f_1 and f_2 have a significant amount of content overlap and mosaic them together. In the overlapping region, the frame that contains more content pixels is retained.

Fig. 6 shows a redundant key frame detected using our connected component based approach. Note that there is a significant spatial variation between the content in these two key frames. The number of connected components in Fig. 6(c) and (d) are 53 and 81, respectively. Only 231 pairs out of 4293 (53 × 81) passed the test for similar number of pixels. There are 16 matching pairs among these connected components. Out of these matching pairs, eight are identified as true positives. The matching components in both the frames are shown in Fig. 6(e) and (f). Out of the remaining eight false positives, three pairs are also shown in Fig. 6(g) and (h).

Fig. 7 shows the mosaicking of key frames when there is a content overlap. In Fig. 7, the frames in (a), (b), and (c) are three

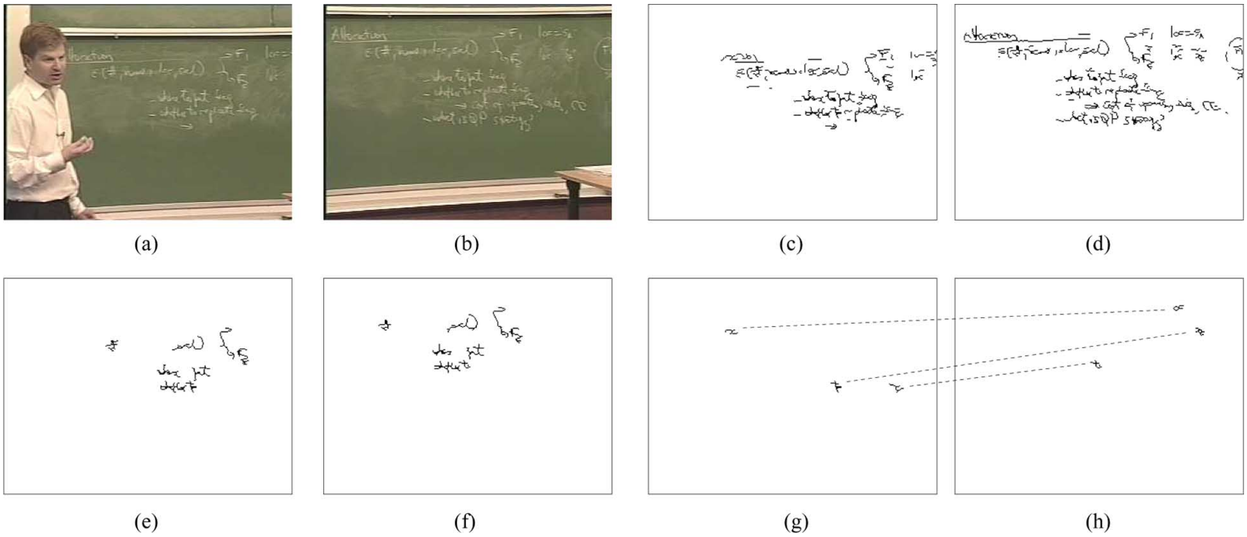


Fig. 6. The connected-component-based matching is able to detect redundant frames in spite of camera panning. (a) and (b) are two successive key frames; (c) and (d) are the extracted content of (a) and (b), respectively; (e) and (f) are the connected components matched in (c) and (d); (g) and (h) are some false matching pairs (joined with dashed lines). By matching the connected components in (c) and (d), and further analyzing the candidate translations, the redundant frame (c) is identified and removed.

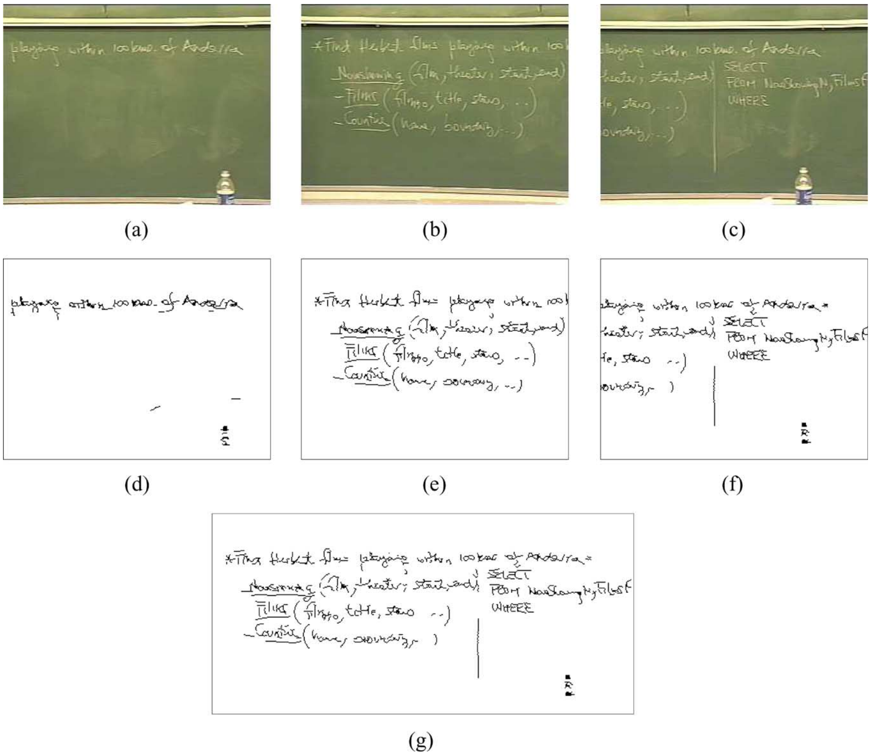
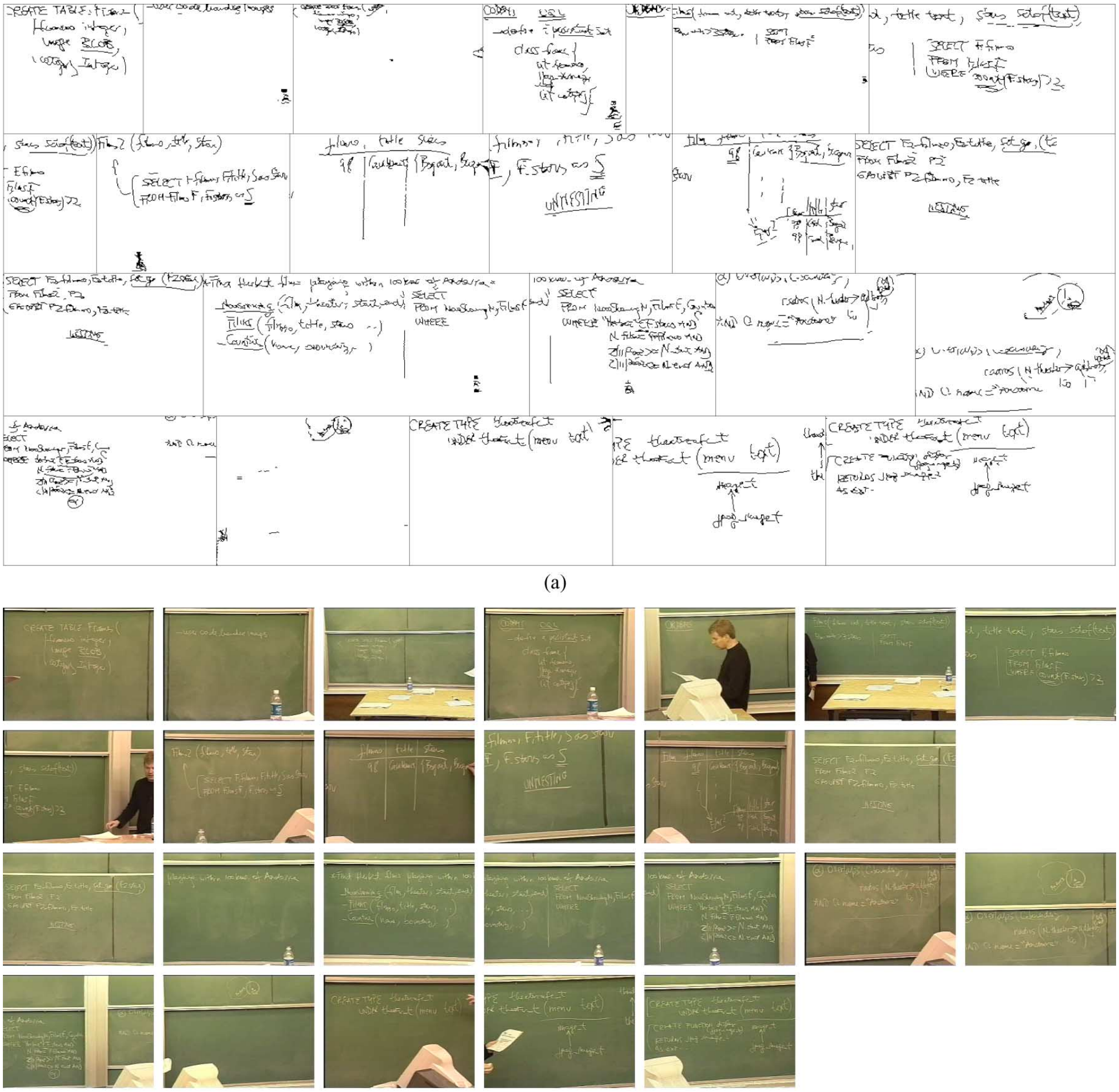


Fig. 7. Connected-component-based matching is able to detect and mosaic the frames with overlapping content. (a), (b), and (c) are three successive key frames of a video; (d), (e) and (f) are the extracted content in (a) and (b) and (c), respectively; (g) is the mosaicked frame of the three key frames.

successive key frames with overlapping content. The frame in (g) is obtained by mosaicking the content in all the three frames.

C. Limitations of Content Matching

In matching content text and figures, we mainly consider the translation transform, for the following reasons. The instructional videos are usually recorded by the camera mounted at the back of the classroom, and most of the camera movements are pans. Since the camera is far away from the board, the projective transform can be approximately modelled as translation. In real classroom environments, the rotation of the board content almost never occurs. But, there may exist camera-zooming operations, which make our content matching process less effective. However, in real classroom videos, a zoom-out operation usually relates to the video segment of classroom overview. In such case, the chalk text on board usually is not discernable. A standard camera zooming detector can be added to remove the video segments with the zooming operations.


(a)

(b)

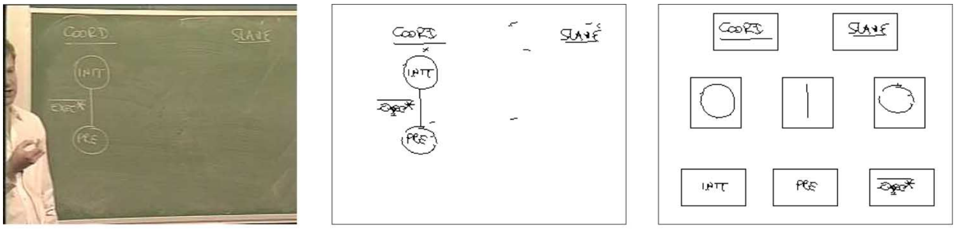
Fig. 8. Key frames are stitched together to form a summary image of the instructional video content. (a) summary image of a 37-min video; (b) the original video frames corresponding to each row in the summary image. In row 3, the second, third, and fourth frames correspond to a mosaicked frame in the summary image.

V. Experimental Results

We evaluated the performance of our content summarization algorithm on four full-length instructional videos: 1) Video 1: a 77-minute video of the course "Computer Architecture"; 2) Video 2: a 37-minute video of the course "Database Systems"; 3) Video 3: a 78-minute video of the course "Database Systems"; 4) Video-4: a 113-minute video of the course "Database Systems." All these videos were recorded in real classrooms by amateur cameramen under poor recording conditions and were used for the distance learning programs in the university. They have a significant amount of light condition changes, and occlusions of the board region. In the classrooms one camera was mounted at the back, and camera movements and panning occurred frequently in the videos. Due to the wear of the board and the erasure of written chalk content, the board regions show significant nonuniformity and variation in color and luminance. The instructors used a lot of figures, formulae and tables apart from text to teach the course content.

We refer to the final disjoint key frames (including the mosaicked frames) as **summary frames**. We get the bounding boxes for the binary content in the summary frames and stitch them together, making a **summary image** of the instructional video content. Fig. 8(a) shows a summary image for one full-length video of 37 min. The only redundancy in the summary image in Fig. 8(a) is caused by camera zooming. There are four rows in the summary image. In making the summary image, the width of the layout is predetermined, and the bounding box for the right most frame in each row is extended to fit the predetermined width. In each row, the bounding box with maximum height determines the height of the row, and the other boxes are extended to fit the row height. In Fig. 8(b), we also provide the original video frames corresponding to each row in the summary image.

To evaluate the performance of content summarization, we introduce the concept of **content element**. We define a word as a content element for the text; for figures and drawings, we define a stroke (e.g., a line segment) as a content element. Such



(a) (b) (c)

Fig. 9. Illustration of the content elements in a video frame. (a) the original frame; (b) the extracted content; (c) the eight content elements in this video frame (shown in rectangular boxes).

TABLE II
PERFORMANCE OF OUR CONTENT SUMMARIZATION METHOD ON FOUR FULL-LENGTH INSTRUCTIONAL VIDEOS

	# of Frames	# of Key Frames	# of Summary Frames	Ground-Truth Content Elements	Extracted Content Elements	Content Missing Rate	Time taken in minutes
Video 1	139K	87	60	910	885	2.7%	150
Video 2	67K	45	23	278	268	3.6%	72
Video 3	140K	86	46	259	244	5.8%	113
Video 4	203K	132	54	579	535	7.6%	140

a content element is the basic unit for high-level applications such as indexing and recognition. In Fig. 9, we show the content elements in the extracted content of an instructional video frame. With the introduction of the content element, the **content missing rate** of the summarization algorithm for a video is defined as Content Missing Rate = The number of missing content elements in the summary frames/ The total number of content elements in the video.

The three processing steps in our summarization method affect the content missing rate for a video as follows. 1) The content analysis step may not extract all the content in the frames and may also result in noise, i.e, false positives. This could be because of errors in background separation and the content extraction. 2) The candidate key frames retrieved by the content-sensitive temporal sampling technique may not contain all the content in the video. 3) The matching techniques may consider nonredundant frames as redundant frames. Note that, the temporal sampling technique and the matching techniques are affected by the content analysis phase. The content missing rate defined above measures the performance of all the three processing steps in a unified approach.

Table II shows the performance of our algorithm on the four tested instructional videos. The ground truth is obtained by manually counting the number of content elements in the videos. The content elements in the summary frames are also manually counted. The content missing rate of the video is calculated as defined above. On average, the proposed summarization technique achieves a compression ratio of 3000 (around one summary frame per one minute and 40 s). The content missing rate for all the four videos is less than 10% at such a high compression ratio.

The proposed methods are also computationally efficient. The system has been implemented using MatLab and Image Processing Toolbox on a Windows computer equipped with a 2.6 GHz Pentium-4 CPU and 1 GB RAM. The last column in Table II shows the total time taken for processing each video. The average time taken for content analysis of one frame is 6.25 s. The time taken for matching and mosaicking the key frames is 21 s per key frame on average.

We compare the performance of our summarization algorithm with three well-known key frame selection techniques namely, the fixed rate video sampling, the tolerance band [9] and, the unsupervised clustering [10] methods. The fixed-rate video sampling is a trivial method as key frames are selected by sampling the video at a fixed-rate. The tolerance band method uses a distance threshold to select key frames. The clustering method groups the video frames into clusters based on similarity of the visual content in the frames. In our implementation of tolerance band and the clustering methods, the color histogram distance is used as the similarity measure.

The performance of the three existing methods is compared with our content summarization method in terms of content missing rate. Obviously, the number of key frames in these methods depends on the input parameters (sampling rate and histogram distance threshold). To retrieve a certain number of key frames, we vary the parameters in these methods to find the best-performance parameters, which are then used for performance evaluation. Then we calculate the content missing rates in these key frames. For the four experimental videos, Fig. 10 shows the plots of content missing rates against the number of key frames. The content missing rates of our summarization method against the numbers of summary frames are also shown in Fig. 10. In Fig. 11, we also compare the visual quality of our summary frames with the sample key frames obtained using different methods in a test video.

The comparison results in Figs. 10 and 11 clearly show that our method outperforms the conventional key frame selection methods in summarizing the visual content in instructional videos. Our method performs better than the other methods in the following three aspects. First, the conventional methods are based on image dissimilarity measures, so the occlusions, light condition changes, and camera movements negatively affect the resulting key frames. Comparatively, our summarization method can effectively handle these noises. To achieve the same content-missing rates, the other methods need to retrieve much more key frames than those of our summarization method. Second, though the other key frame selection methods can also achieve a low content missing rate at sufficiently large

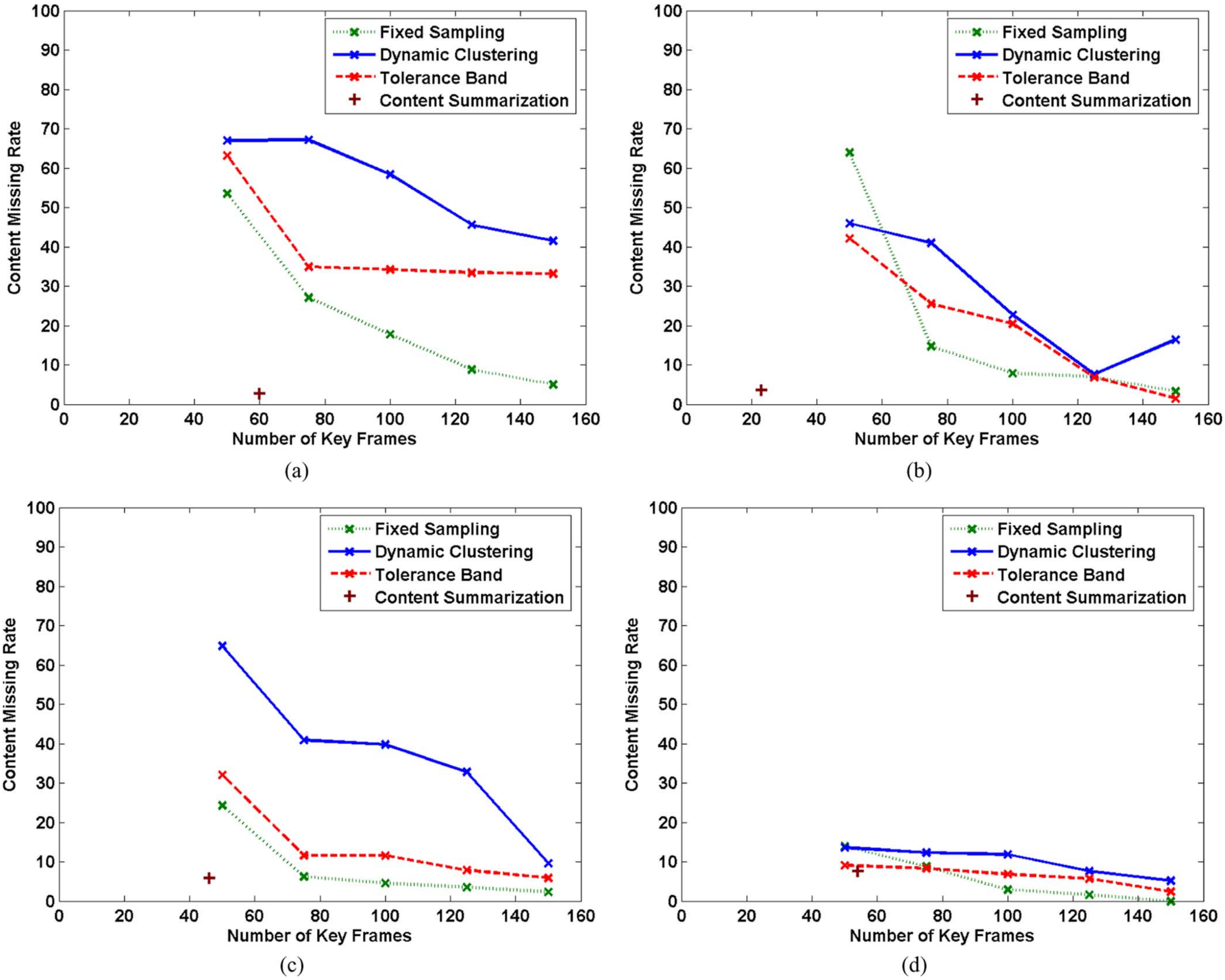


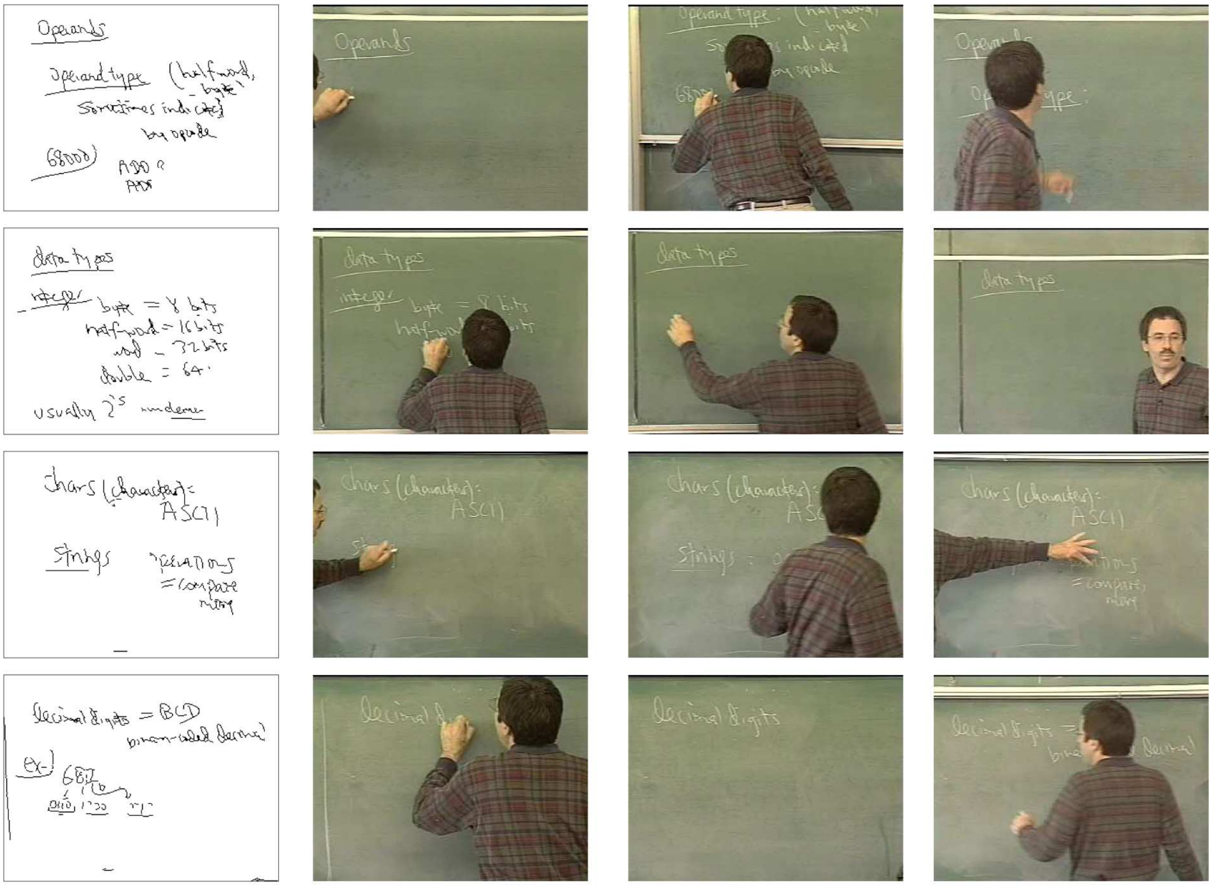
Fig. 10. Comparison of our summarization method with three different key frame selection methods in terms of content missing rate. (a)–(d) correspond to the four videos listed in Table II.

number of key frames, those key frames are not rich in content, and there are a lot of occlusions. In comparison, our summary frames contain most of the content text and figures and are thus more appealing. Third, in the dynamic clustering and the tolerance-band methods, the content missing rate does not necessarily decrease with increasing number of key frames, and they are not better than the trivial fixed-rate sampling. Compared with the fixed-rate sampling method, our summarization method retrieves 61% less key frames at the same content missing rate.

In Table III, we also compare the performance of our method with the dynamic clustering and tolerance band methods, in terms of the total time taken for processing each test video. For the fixed-rate sampling method, the computational time is negligible. Table III shows that our method takes more time than the existing methods because it has more complicated computation. Since the summarization of instructional video is an off-line process and the content missing rate is the more important factor, our approach is still suitable for this application.

VI. Conclusion

In this paper, we presented a new approach to summarize the visual content in instructional videos. We first provided a probabilistic model to extract the written content on board, then we computed key frames of instructional videos by locating the local maxima in content fluctuation curves with a shifting window algorithm. We further introduced techniques based on Hausdorff-distance and connected-component-decomposition to reduce content redundancy of the key frames by matching the content and mosaicking the frames. Experiments on four instructional videos recorded in real classrooms show that our algorithm is highly effective in extracting and summarizing the visual content in instructional videos. The comparison of our algorithm with three existing key frame selection methods demonstrated the superior performance of our method. Future work includes the analysis of the events in instructional videos and the development of indexing and retrieval systems based on the content summarization results.



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

Fig. 11. Comparison of our summary frames with the key frames obtained using different key frame selection methods in a test video. (a) our summarization algorithm; (b) fixed sampling; (c) dynamic clustering; (d) tolerance band. Our summary frames are rich in content and more appealing.

TABLE III
COMPARISON OF THE TOTAL COMPUTATIONAL TIME (IN MINUTES) AMONG THREE VIDEO SUMMARIZATION METHODS ON FOUR INSTRUCTIONAL VIDEOS

	Our method	Dynamic Clustering	Tolerance Band
Video 1	150	63	70
Video 2	72	39	33
Video 3	113	62	67
Video 4	140	93	94

REFERENCES

[1] A. Mittal, P. V. Krishnan, and E. Altman, "Content classification and context based retrieval system for E-learning," *Int. J. Educ. Technol. Soc.*, vol. 9, no. 1, pp. 349–358, 2006.

[2] C. Dorai, V. Oria, and V. Neelavalli, "Structuralizing educational videos based on presentation content," in *Proc. ICIP*, 2003, vol. 3, pp. 1029–1032.

[3] D. Phung, S. Venkatesh, and C. Dorai, "Narrative sturcture analysis with education and training videos for E-learning," in *Proc. Int. Conf. Pattern Recognition*, 2002, vol. 2, pp. 835–838.

[4] E. Altman, Y. Chen, and W. C. Low, "Semantic exploration of lecture videos," in *ACM Multimedia*, 2002, pp. 416–417.

[5] H. S. Chang, S. Sull, and S. U. Lee, "Efficient video indexing scheme for content-based retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1269–1279, 1999.

[6] P. Campisi, A. Longari, and A. Neri, "Automatic key frame selection using a wavelet based approach," in *Proc. SPIE*, 1999, vol. 3813, pp. 861–872.

[7] D. DeMenthon, V. Kobla, and D. Doermann, "Video summarization by curve simplification," in *ACM Multimedia*, 1998, pp. 211–218.

[8] T. Liu and J. R. Kender, "Time-constrained dynamic semantic compression for video indexing and interactive searching," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2001, vol. 2, pp. 531–538.

[9] M. Yeung and B. Liu, "Efficient matching and clustering of video shots," in *Proc. ICIP*, 1995, vol. 1, pp. 338–341.

[10] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra, "Adaptive key frame extraction using unsupervised clustering," in *Proc. ICIP*, 1998, pp. 866–870.

[11] A. Divakaran, R. Radhakrishnan, and K. A. Peker, "Motion activity-based extraction of key-frames from video shots," in *Proc. ICIP*, 2002, vol. 1, pp. 932–935.

[12] W. Wolf, "Key frame selection by motion analysis," in *Proc. ICASSP*, 1996, pp. 1228–1231.

[13] H. Li, D. Doermann, and O. Kia, "Automatic text detection and tracking in digital video," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 147–156, 2000.

[14] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images and videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 4, pp. 256–268, 2002.

[15] C.-W. Ngo and C.-K. Chan, "Video text detection and segmentation for optical character recognition," *Multimedia Syst.*, vol. 10, no. 3, pp. 261–272, 2005.

[16] M. R. Lyu, J. Song, and M. Cai, "A comprehensive method for multilingual video text detection, localization, and extraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 243–255, 2005.

[17] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imag.*, vol. 13, no. 1, pp. 146–168, 2004.

[18] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents:A survey," *Int. J. Document Anal. Recognit.*, vol. 7, no. 2-3, pp. 84–104, 2005.

[19] S. X. Ju, M. J. Black, S. Minneman, and D. Kimber, "Summarization of videotaped presentations: Automatic analysis of motion and gesture," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 5, pp. 686–696, 1998.

[20] T. Liu and J. R. Kender, "Rule-based semantic summarization of instructional videos," in *Proc. ICIP*, 2002, vol. 1, pp. 601–604.

[21] T. Liu and J. R. Kender, "Semantic mosaic for indexing and compressing instructional videos," in *Proc. ICIP*, 2003, vol. 1, pp. 921–924.

[22] Y. Chen and W. J. Heng, "Automatic synchronization of speech transcript and slides in presentation," in *Proc. Int. Symp. Circuits and Systems*, 2003, vol. 2, pp. 568–571.

[23] F. Wang, C.-W. Ngo, and T.-C. Pong, "Synchronization of lecture videos and electronic slides by video text analysis," in *ACM Multimedia*, 2003, pp. 315–318.

[24] L. He, E. Sanocki, A. Gupta, and J. Grudin, "Auto-summarization of audio-video presentations," in *ACM Multimedia*, 1999, pp. 489–498.

[25] T. Liu, R. Hejelsvold, and J. R. Kender, "Analysis and enhancement of videos of electronic slide presentations," in *IEEE International Conference on Multimedia and Expo*, 2002, vol. 1, pp. 77–80.

[26] T. Syeda-Mahmood and S. Srinivasan, "Detecting topical events in digital video," in *ACM Multimedia*, 2000, pp. 85–94.

[27] S. Mukhopadhyay and B. Smith, "Passive capture and structuring of lectures," in *ACM Multimedia*, 1999, pp. 477–487.

[28] C.-W. Ngo, F. Wang, and T.-C. Pong, "Structuring lecture videos for distance learning applications," in *Proc. IEEE Int. Symp. Multimedia and Software Engineering*, 2003, pp. 215–222.

[29] M. Onishi, M. Izumi, and K. Fukunaga, "Production of video image by computer controlled camera operation based on distribution of spatiotemporal mutual information," in *Proc. Int. Conf. Pattern Recognition*, 2000, vol. 4, pp. 102–105.

[30] Q. Stafford-Fraser and P. Robinson, "Brightboard:A video-augmented environment," in *Proc. Conf. Computer Human Interface*, 1996, pp. 134–141.

[31] W. J. Heng and Q. Tian, "Content enhancement for E-learning lecture videos using foreground/background separation," in *IEEE Workshop on Multimedia Signal Processing*, 2002, pp. 436–439.

[32] T. Liu and C. Choudary, "Content adaptive wireless streaming of instructional videos," *Multimedia Tools Applicat.*, vol. 28, no. 1, pp. 157–171, 2006.

[33] A. Mittal, S. Gupta, S. Jain, and A. Jain, "Content-based adaptive compression of educational videos using phase correlation techniques," *Multimedia Syst.*, vol. 13, no. 3, pp. 249–259, 2006.

[34] M. N. Wallick, R. M. Heck, and M. L. Gleicher, "Marker and chalkboard regions," in *Mirage 2005*, 2005, pp. 223–228.

[35] T. Yokoi and H. Fujiyoshi, "Generating a time shrunk lecture video by event detection," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2006, pp. 641–644.

[36] F. Wang, C.-W. Ngo, and T.-C. Pong, "Lecture video enhancement and editing by integrating posture, gesture, and text," *IEEE Trans. Multimedia*, vol. 9, pp. 397–409, 2007.

[37] L. He, Z. Liu, and Z. Zhang, "Why take notes? use the whiteboard capture system," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, pp. 776–779.

[38] L. He and Z. Zhang, "Real-time whiteboard capture and processing using a video camera for teleconferencing," in *Proc. ICASSP*, 2005, pp. 1113–1116.

[39] M. Wienecke, G. A. Fink, and G. Sagerer, "Toward automatic video-based whiteboard reading," *Int. J. Doc. Anal. Recognit.*, vol. 7, no. 2-3, pp. 188–200, 2005.

[40] Z. Zhang and L. He, "Notetaking with a camera: Whiteboard scanning and image enhancement," in *Proc. ICASSP*, 2004, vol. 3, pp. 533–536.

[41] A. Haubold and J. R. Kender, "Analysis and visualization of index words from audio transcripts of instructional videos," in *Proc. IEEE Sixth Int. Symp. Multimedia Software Engineering*, 2004, pp. 961–964.

[42] Y. Li and C. Dorai, "Instructional video content analysis using audio information," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 6, pp. 188–200, 2006.

[43] M. Lin, M. Chau, J. Cao, and J. F. Nunamaker, Jr, "Automated video segmentation for lecture videos: A linguistics based approach," *Int. J. Technol. Human Interact.*, vol. 1, no. 2, pp. 27–45, 2005.

[44] I. Malioutov and R. Barzilay, "Minimum cut model for spoken lecture segmentation," in *Int. Conf. Computational Linguistics and 44th Annu. Meeting of the ACL*, 2006, pp. 25–32.

[45] S. Repp and C. Meinel, "Semantic indexing for recorded educational lecture videos," in *IEEE Int. Conf. Pervasive Computing and Communications Workshops*, 2006.

[46] N. Yamamoto, J. Ogata, and Y. Ariki, "Topic segmentation and retrieval system for lecture videos based on spontaneous speech recognition," in *Proc. Eurospeech*, 2003, pp. 961–964.

[47] B. Erol and Y. Li, "An overview of technologies for E-meeting and E-lecture," in *IEEE Int. Conf. Multimedia and Expo*, 2005.

[48] C. Choudary and T. Liu, "Extracting content from instructional videos by statistical modeling and classification," *Int. J. Pattern Anal. Applicat.*, vol. 10, no. 2, pp. 69–81, 2007.

[49] D. Comaniciu and P. Meer, "Mean shift: A robust approach towards feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, 2002.

[50] D. Huttenlocher, D. Klanderman, and A. Rucklige, "Comparing images using the Hausdorff distance," *IEEE Trans. Pattern Anal Mach. Intell.*, vol. 15, no. 9, pp. 850–863, 1993.

Chekuri S. Choudary (S'03) received the B.Tech. degree in electrical engineering from Jawaharlal Nehru Technological University, India, in 2000 and the M.E. degree in computer engineering from the University of South Carolina, Columbia, in 2002. He is pursuing the Ph.D. in computer science in the Department of Computer Science, University of South Carolina.

During 2002, he was a Visiting Research Assistant at the University of Southern California's Information Sciences Institute (ISI), Arlington, VA. His research interests include image processing, multimedia, and e-learning technologies.

Tiecheng Liu (M'03) received the Ph.D. degree in computer science from Columbia University, New York, in 2003.

He is an Assistant Professor in the Department of Computer Science and Engineering, University of South Carolina, Columbia. His main research interests include computer vision, image and video processing, multimedia and advanced learning technologies. He has published over 30 refereed papers in the area of computer vision and multimedia technology.

Dr. Liu is a member of the ACM and has served as a committee member for IEEE CBAR'04, CIVR'05, and other conferences.