Robust Alignment of Presentation Videos with Slides

Xiangyu Wang Mohan Kankanhalli

National University of Singapore

Abstract. Many presentations consisting of video, audio and slides are being recorded for wider dissemination purposes. Video slide alignment is necessary for efficient review and hence has attracted much attention. However, the recorded video style varies greatly because of different capturing systems, and most existing alignment approaches deal with one of the video styles. In this paper, a more general approach is proposed to make the alignment be applicable to all major video styles. We mainly combine the SIFT (scale invariant feature transform) keypoints and color features to match between video and electronic slides for alignment, and use texture features as a complement. The approach improves the alignment performance, and is able to handle many kinds of video.

Key words: video slide alignment, SIFT features, color features, texture

1 Introduction

With the advances in technology in the last two decades, there is an increasing trend to record events in many situations. In conferences, universities and corporations, many presentations and lectures are recorded for wider dissemination. Indexing for presentation videos becomes necessary for effective review. One efficient mechanism is to synchronize the presentation video with the slides. Matching slides to video segments provides an intuitive way of indexing video by slides for searching and browsing. Then, the users can watch the corresponding video segment of any particular slide. The users can also find out which slide the lecturer is talking about to help the understanding while watching the video. Moreover, it can improve the quality of the video through projecting the high-resolution slides back into the video [2].

Major research issues in the synchronization of presentation video and slides are the spotting of slide position in the frames of the video, the detection of slide transition, and the matching between video segments and slides. Related works include [7], [10], [8], [11], [2].

Depending on the capturing systems and the authoring methods, the slides may appear dramatically different in the video. The video can be captured with one or more cameras. The cameras can be fixed or allowed to switch, pan, tilt, and zoom. Thus, the slide may appear small, full-frame, or clipped and might suffer from partial occlusion, e.g., by the speaker. Moreover, the authoring method

imposes additional challenges to slide-matching algorithms. In general, there are 3 major styles for video production. The first style shows one panel in the window and it switches between the presenter and the slides, figure 1(a). The second kind shows presentation video and captured corresponding slide in two panels side by side in one window, as shown in figure 1(b). The third style shows the presentation platform with the projected electronic slides in the background, figure 1(c).

(a) Style 1: switch between slide and presenter

(b) Style 2: both slide and presenter

(c) Style 3: slide in background

Fig. 1. Examples of different frame types

Over the past decade, researchers have been exploring methods for matching slides to video. Early systems such as the Classroom 2000 project [1] and BMRC Lecture Browser [9] match the slides to the video segments by manually editing the time stamps. Some automatic approaches [7], [10], [8], [5], [11] also have been proposed. However, these methods are only style specific solutions and may not be flexible for other video styles. For example, [7] can only deal with fixed camera video. Ngo *et al.* [8] proposed a method to detect the slide transition for topic indexing. The method takes into account the background vs foreground information, figures and caption regions in slides when detecting transitions. However, the method can only deal with the scenario that the camera is fixed and stays stationary throughout a lecture. Jones *et al.* [5] introduced an audio-

visual method to align slides and video with both audio and visual features. But the automatic speech transcription is of high error rate using automatic speech recognition system without individual acoustic models for lecturers.

Fan *et al.* [2] developed a framework to automatically match electronic slides to the presentation videos. The approach combined SIFT (scale invariant feature transform) with the temporal and camera cues to improve the performance in ambiguous cases. But the recognition error rate increases in the following scenarios: the use of many video and animations, duplicated identical slides, slides with very little content, and low quality video with defocused projector screen as shown in figure 4(a). To the best of our knowledge, there is no one existing method that can deal with all the video styles.

Another work of ours [12] that integrates different video styles into a more uniform framework has been proposed in [12]. That work uses optical flow and Gabor analysis to deal with videos containing de-focused slide content, speaker occlusion as well as camera pan, tilt and zoom sequences. But the accuracy for some videos is not as good as the approach in this paper.

In this paper, we propose a general approach for video slide alignment. By combining both the SIFT keypoints features and color features, and adopting texture features as a complement, the approach improves the alignment performance and can work for different video styles. The remainder of the paper is organized as follows. Section 2 briefly defines the alignment problem and introduces the features used. Section 3 presents our video slides matching algorithm in details. Section 4 shows the experiment results. Finally, section 5 concludes our approach.

2 Overview

2.1 Problem Formulation

The synchronization problem of electronic slides and presentation videos can be formulated as follows: Given a presentation video $V = \{t_1, t_2, ..., t_n\}$ and the electronic slides $S = \{s_1, s_2, ..., s_m\}$ associated with it, here $t_i, i = 1, 2, ..., n$ denotes the homogeneous segments in the video such that the projected slide image does not change in each segment, and $s_j, j = 1, 2, ..., m$ denotes the slide images captured from the projector or generated from electronic files. The aim is to find the correspondence function $f : V \rightarrow S$ such that for any t_i, $f(t_i) = s_j$ if the video segment t_i contains slide s_j and $f(t_i) = null$ if there is no corresponding slide for the segment t_i.

In order to be robust for different capture systems and authoring methods, not many assumptions should be used and only the information extracted from the video should be used for slide frame matching. Moreover, in order to avoid matching slide with each frame, the video should first be segmented into shots. There should not be slide transition in each shot. In the following part, the details of the video slide alignment will be presented.

2.2 Descriptors

The SIFT keypoints [6] are introduced by Fan *et al.* [3] to perform reliable matching between different views of a slide image. The SIFT features are invariant to image scale and rotation, and are shown to provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. Thus, SIFT features are quite suitable for the video slide alignment application. The work in [3] shows the effectiveness of SIFT features. However, SIFT features are local gray-levels features, and may not work well in the following cases.

– First, the SIFT features can not differentiate the different outline slides whose contents are usually the same with different highlighted section titles, as shown in figure 2.

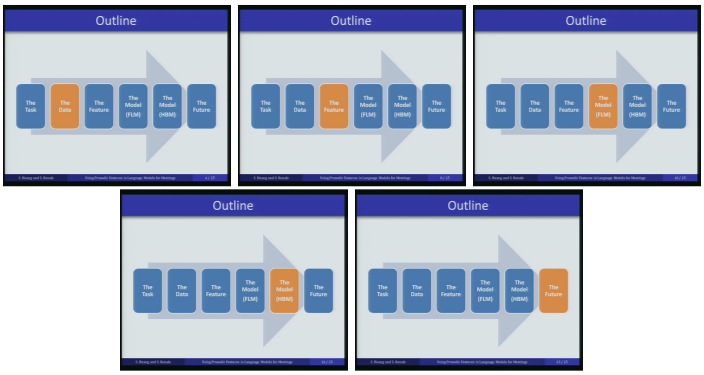


Fig. 2. Example of outline slides

– Another case is the animated slide whose whole content appears step by step as shown in figure 3. If the electronic slides set contains both the partial slide A and whole slide B, a partial slide in video frame may be misaligned to either A or B by the SIFT keypoints.
– The biggest problem is the defocused video as shown in figure 4. The camera focuses on the presenter at the foreground and thus the projector screen is defocused at the background, so that the SIFT keypoints fail to recognize the slides on the projector screen.

As indicated above, the color and texture features are also adopted as global features in our approach.

3 The Alignment Framework

The flow of the video slide alignment algorithm is shown in figure 5. The details will be presented in the following subsections.

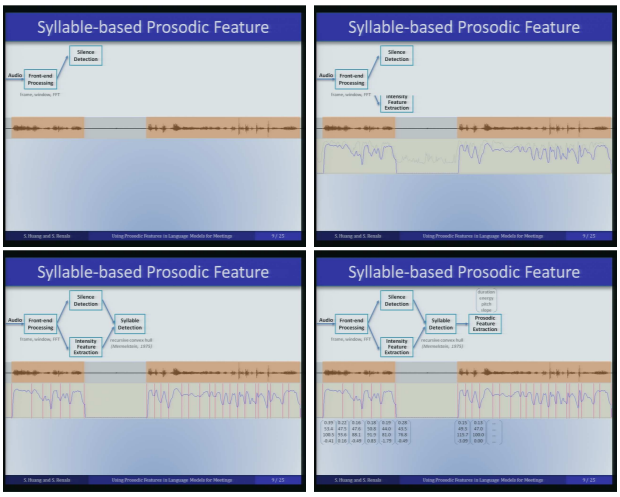


Fig. 3. Example of animated slides

3.1 Check For Defocus

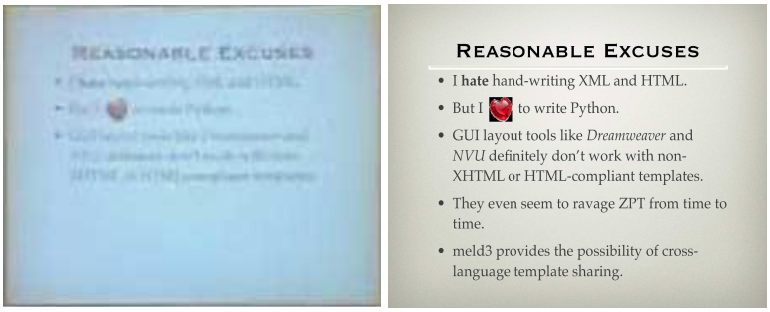
For some video captured by a single fixed camera, the camera focuses on the presenter and thus the slide region in the frame is defocused. Because of the distortion of the text in the slide region, SIFT method fails in this case. Thus, the alignment method for defocused slide region video should be different, and the video type should be first identified. This problem happens only in video of style 3. For video of other styles, the slide region generally will be in focus since the cameras can move. Thus for video of style 3, if one frame is defocused, the whole video will be considered as defocused. The principle is that the defocused blurry images have dense gradients. Given the slide region in frame A and corresponding slide image B, resize them to the same size and compute their gradients. Denote the number of nonzero elements in gradients as N_A and N_B. A is considered blurred if $\frac{N_A}{N_B} \geq \tau$. We have empirically determined τ value as 1.5. Manual modification can be done if needed.

3.2 Preprocessing

Video Segmentation The presentation video is first segmented since it is inefficient and inaccurate to compare all the electronic slides with each frame. The most commonly used grayscale histogram with chi-square distance method is adopted.

– The video segmentation method first calculates 64 bin gray level histograms of images, then computes the histogram difference using chi-square distance measurement. Given two grayscale histograms h_1 and h_2 of two frames, the frame difference is

(a) frame (b) slide region



(c) extracted slide region and slide image

Fig. 4. Example of defocused video

$$fd_{chi} = \begin{cases} \frac{1}{N^2}\sum_i \frac{(h_1[i]-h_2[i])^2}{h_2[i]}, & h_2[i] \neq 0 \\ \frac{1}{N^2}\sum_i \frac{(h_1[i]-h_2[i])^2}{h_1[i]}, & h_2[i] = 0 \end{cases} \quad (1)$$

where N is the number of pixels in a frame.

– Comparing the color histogram of successive frames, and a scene change is found when a large enough change is detected.

Unlike shot transitions, slide transitions do not show significant color changes in most cases [8] since most presenters tend to apply the same design to all slides in one presentation. Thus, the threshold value should be set conservatively, say 3×10^{-9} in this experiment, such that all slide transitions get detected. As a result, there are many false positives. But they do not matter because what we want to make sure is that the slide does not change during each segment. Moreover, the computation is reduced.

For defocused video, the method is different:

– The first step is to obtain the unwarped slide region in the frame. Since the camera and projector screen are fixed, the corner points of the quadrilateral are constant and can be obtained by using Hough transform, as shown in 4(b). The quantization of θ and ρ are empirically determined to be 1 and

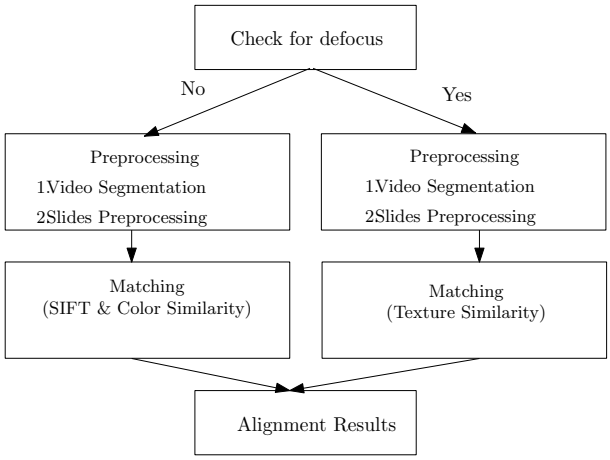


Fig. 5. Flow of alignment algorithm

3 respectively. Manual modification can be done when needed. Then, a homography H that performs the mapping from slide region to the electronic slide can be computed [7]. Thus, the slide region in the frames can be extracted, and thus the unwarped slide region image is obtained by undoing the projection H on it.

– Then, the similarity between successive unwarped slide region images using Hausdorff distance are computed to detect slide transitions. The procedure is as follows:
  - For an image G, the Canny edge detection is first applied on it. The edge image G_b is then dilated by 3 to give another binary image G_d.
  - Then, for two successive slide region images A and B, four numbers a, a', b, b' are computed. Here, a is the number of white pixels in A_b, and a' is the number of white pixels in A_b whose corresponding pixels in B_d are also white. b is the number of white pixels in B_b, and b' is the number of white pixels in B_b whose corresponding pixels in A_d are also white.
  - The similarity between A and B is $M_{AB} = \min(\frac{a'}{a}, \frac{b'}{b})$

  If the similarity is less than the empirical threshold 0.75, a slide transition is considered to happen and the video is segmented.

Slides Preprocessing In the past works, researchers mainly focus on the preprocessing of video. However, some preprocessing of electronic slides may be easier and can improve the matching result significantly. There may be duplicated partial content slides in electronic slides set due to animation. In our approach, the electronic slides are preprocessed to remove them to avoid misalignment.

– First, the SIFT keypoints are detected for all the slide images in the set of the electronic slides S_{init}.

- Second, a simple matching scheme is applied on the keypoints of two images to get a set of putative correspondences. Given the keypoints detected in two images A and B, Lowe [6] presents a nearest neighbor matching scheme: Suppose P_B is a keypoint in image B, and P_{A1}, P_{A2} respectively are the nearest neighbor and second nearest neighbor of P_B in the descriptor's feature space in image A. Then, P_{A1} is accepted as a match to P_B if $\frac{d(P_{A1},P_B)}{d(P_{A2},P_B)} < distRatio$. Here, $d(\cdot,\cdot)$ denotes the Euclidean distance between the descriptors of the two keypoints. The *distRatio* is set to be 0.6. After the simple matching, the matched keypoints in the two images A and B are M_A and M_B respectively.
- Finally, the RANSAC algorithm [4] is used to determine the homography H between two images by solving $M_B = H * M_A$. At the same time, the inliers can also be got. Then, the match ratio $matchRatio = \frac{N_{inlier}}{N_{match}}$ is calculated. A is considered as part of B if *matchRatio* larger than 0.8.
- After the animated slides removal, the set of the electronic slides becomes S, which is a subset of S_{init}.

3.3 Matching

After preprocessing of the presentation video and electronic slides, the presentation video is segmented and one keyframe is extracted from each segment for matching with the slides to find the corresponding slide of the segment.

The first phase is the SIFT keypoints matching. First, SIFT keypoints of all the keyframes and electronic slide images are detected. Then for each keyframe F, the similarities with all the slide images S are measured using SIFT keypoints. The details are as follows:

- Given a keyframe image F and an electronic slide image E, the keypoints of F and E are denoted as P_F and P_E, respectively.
- First, the simple nearest neighbor matching scheme proposed in [6] is used to find the putative matching points M_F and M_E.
- Then, the RANSAC algorithm is used to search for the true keypoint correspondences by imposing a homography on the putative correspondences.

The second phase is the color matching. The slide region in the frame is first extracted with the homography derived from the SIFT matching. Then for the corresponding regions, the color histograms are computed and the similarity is measured. The details are as follows:

- At first, the image is divided into 3 by 3 grid, and the different cells are weighted by the following filter

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{2}$$

- In each region, the color histogram is computed and the similarity is measured using the Bhattacharyya distance. Suppose the color distribution is p and q, then the Bhattacharyya distance is calculated as: $BC(p,q) = \sum_{x \in X} \sqrt{p(x)q(x)}$. Here, X is the color domain.

– Finally, the color similarity between two images A and B is:

$$Color(A,B) = \frac{1}{\sum_{i=1}^{3}\sum_{j=1}^{3} w_{ij}} \sum_{i=1}^{3}\sum_{j=1}^{3} w_{ij} color_{ij} \tag{3}$$

Here, w_{ij} denotes the weight at (i,j) in the filter, and $color_{ij}$ denotes the color similarity between the region $A(i,j)$ and $B(i,j)$ measured using Bhattacharyya distance.

With both the matched SIFT keypoints and color similarity, the similarity between two images A, B is computed using an empirical formula:

$$Similarity(A,B) = N_{AB} * N^{Color(A,B)} \tag{4}$$

Here, N_{AB} is the number of inliers in matched SIFT keypoints, N is the maximum number of inliers in SIFT keypoints between one frame F and each slide $S = \{s_1, s_2, ..., s_m\}$, and $Color(A,B)$ is the color similarity. The formula has several advantages:

– The similarity considers both the SIFT keypoints and color features.
– The similarity is consistent with both the keypoints similarity and color similarity.

By comparing the frame with all the slide images, the slide that is most similar to the slide region in the frame is chosen as the corresponding slide of the frame. Here, a hidden markov model (HMM) is adopted to increase the accuracy. According to the observations, there is a temporal locality for the order of showing slide in presentations. That is, if a frame f_t is showing slide s_i, there is a high probability that frame f_{t+1} will show slide s_i, s_{i+1}, or s_{i-1}. In our experiment, we are giving higher probability (0.2) the nearby slides, and lower probability (equally divided for the remaining) to the others.

For the defocused slide region video, the two-phase SIFT & Color matching fails. Thus, the layout information is employed. For each segment of the presentation video, the last slide region image is chosen to match with all the electronic slide images. The similarity is measured using the Hausdorff distance.

4 Experiment and Discussion

Five presentation data sets are tested in total, as shown in table 4.

The slides from the data sets are tested for slide preprocessing. The animated slides removal result is shown in table 2.

Then, the video segmentation is tested with acceptable deviation of 2 seconds. For focused presentation, all the slide transitions are detected, but the accuracy is only about 10%. For the defocused slides video, the result of slide transition detection is shown in table 3.

Our slide video alignment using SIFT keypoints and color histogram is compared with the SIFT only method in [3], and the results are given in table 4(SP

Table 1. Data Set

Test Set	Description	Duration	Slides	Style
1	MLMI'07[1]	29min	63	2
2	MLMI'07	25min	28	2
3	MLMI'07	17min	13	2
4	CMU lecture	63min	39	1
5	Plone Symposium'06[2]	37min	14	3
6	Plone Symposium'06	39min	21	3

[1] MLMI'07 source: `http://www.idiap.ch/mmm/talk-webcast/mlmi/mlmi-07`

[2] Plone Symposium'06 source: `http://plone.org/events/regional/nola06/presentation-material-and-video`

Table 2. Animated slides removal

Test Set	Total Slides	Animated Slides	Removed Slides
1	63	36	37
2	28	6	6
3	13	0	0
4	39	1	1
5	14	0	0
6	21	0	0

denotes slides preprocessing, S denotes SIFT, C denotes color). The accuracy is the ratio of correctly aligned video segments and total video segments.

It can be seen that the preprocessing improves the alignment results by about 5% in average. Moreover, in both cases (with or without preprocessing), the SIFT & color method works better than SIFT only method. In test set 4, the SIFT & color method significantly improves SIFT only method. That may be because the threshold for keypoints is not set appropriately so that the frames without slide region are misaligned to some other slide instead of remaining unchanged. There is no improvement in test set 3 in both cases (with or without preprocessing) because there is no animated slide in the data. An example of the result is shown in figure 6.

The results of slide video alignment using texture similarity for defocused slides video are given in table 5. Some examples are shown in figure 7.

Though the texture features work when the SIFT method fails and are better than the color features, the error rate is still around 45%. It is because the defocused slides are inherently difficult to identify and many slide layouts are similar. Moreover, the texture method can only deal with fixed camera case and will also suffer from occlusion problems.

Table 3. Slide transition detection

Test set	Transition	Detected Transition	
		total	correct
5	14	15	14
6	21	70	21

Table 4. Accuracy of Alignment using SIFT & color

Test set	w/o SP		with SP	
	S	S & C	S	S & C
1	78.2%	95.2%	84.4%	97.7%
2	78%	81.8%	83%	98.1%
3	38.5%	54.5%	38.5%	54.5%
4	27.5%	85.2%	28.8%	90.6%

5 Conclusions

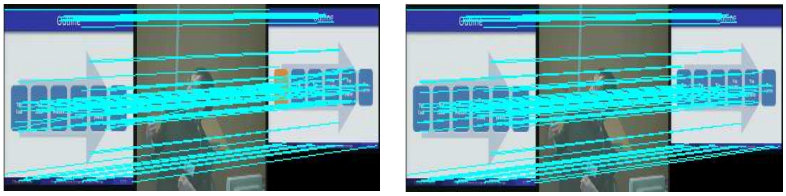
In this paper, a general approach has been proposed for video slide alignment. Preprocessing on the electronic slides is introduced into the alignment algorithm. Animated slides are removed and the ambiguity is reduced. The approach finds a way to combine both the SIFT keypoints features and color features, and adopts texture features as a complement for defocused video. Moreover, a HMM is used to improve the results. As a result, it improves the alignment performance and can work for different video styles. However, the alignment approach still needs different methods for the defocused video and the accuracy is not good enough, and more work will be done.

References

1. G. D. Abowd. Teaching and learning as multimedia authoring: the classroom 2000 project. In *MULTIMEDIA '96*, pages 187–198, 1996.
2. Q. Fan. Temporal modeling of slide change in presentation videos. In *ICASSP '07*.
3. Q. Fan, K. Barnard, A. Amir, A. Efrat, and M. Lin. Matching slides to presentation videos using sift and scene background matching. In *MIR '06*, pages 239–248, 2006.
4. M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
5. G. J. F. Jones and R. J. Edens. Automated alignment and annotation of audio-visual presentations. *ECDL'02*, pages 276–291, 2002.

Table 5. Comparison of Alignment Accuracy using Color and Texture

Test set	Transition	Error	
		Color	Texture
5	15	14	6
6	70	70	35



(a) wrong alignment with SIFT only method (b) correct alignment with SIFT and color information

Fig. 6. Example of alignment results

(a) wrong alignment

(b) correct alignment

Fig. 7. Example of alignment results for defocused video: the left image is the frame, the middle one is the slide region in the frame, and the right one is the aligned slide in the results

6. D. G. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003.
7. S. Mukhopadhyay and B. Smith. Passive capture and structuring of lectures. In *MULTIMEDIA '99*, pages 477–487, 1999.
8. C.-W. Ngo. Detection of slide transition for topic indexing. In *ICME '02*, pages 533–536, 2002.
9. L. A. Rowe and J. M. González. A lecture browser and production system. Technical report, 2000.
10. T. F. Syeda-Mahmood. Indexing for topics in videos using foils. In *CVPR'00*, pages 312–319, 2000.
11. F. Wang, C.-W. Ngo, and T.-C. Pong. Synchronization of lecture videos and electronic slides by video text analysis. In *MULTIMEDIA '03*, pages 315–318, 2003.
12. X. Wang, R. Subramanian, and M. Kankanhalli. A robust framework for aligning lecture slides with video. In *ICIP '09*, 2009.