Looking at projected documents: Event detection & document identification.

Ardhendu Behera, Denis Lalanne and Rolf Ingold
Université de Fribourg
Chemin du Musée, 3
1700 Fribourg
Switzerland
{Denis.Lalanne, Ardhendu.Behera, Rolf.Ingold}@unifr.ch

Abstract

In the context of a multimodal application, this article proposes an image-based method for bridging the gap between document excerpts and video extracts. The approach, further called document image alignment, takes advantage of the observable events related to documents that are visible during meetings. In particular, the article presents a new method for detecting slide changes in slideshows, its evaluation, and a preliminary work on document identification.

1. Introduction

Recent research projects aim at recording meetings and archiving them in suitable forms for later retrieval. Meetings often involve documents, which can be discussed, projected, authored, or simply visible on the meeting table. In spite of their major role, documents have not yet fully been considered for inclusion into meeting archives, mainly because they do not provide immediate means for being time stamped. We will see in this article that document image analysis provides such a mean; a) slide change detection and further b) slide identification will help answering respectively when and which document was projected during the meeting?

We introduce in this article a new slide change detection technique that considers slide stability rather than slide change. We further present the results of an evaluation of our method. Finally, we present our preliminary work on slide identification.

2. Projected document integration

Projected documents hold a particular relationship with the meeting time; they appear at a specific time in the visual focus, which can be recorded with a camera. The corresponding captured low-resolution document images can be matched with original document images available in a repository. These matches will convey temporality to those documents.

2.1. Capture environment

The projection screen is filmed with a camera similar to other cameras used for each participant in our meeting room (figure 1). It is a firewire webcam, set at a resolution of 640 by 480, at 15 frames per second. We then use a graphical user interface in order to select manually the projected document area. This will be improved in the future with an automatic a) detection of the projected area within the video and b) correction of perspective and rotation deformations. Our goal is to develop a general method, working not only with high-resolution camera but also with low-resolution standard material. Further, we want our system to work with traditional manual presentation with transparencies. More important, we want to capture the speaker interactions with the projected documents (gesture, pointing, etc.). For all those reasons, we have chosen to use standard webcams. A major drawback of this technology, apart from its low resolution, is that there is an auto-focusing period, of roughly half-a-second, each time there is a scene change and thus, within this period, each upcoming frame is different from the previous, in term of lighting condition.

2.2. Existing slide change detection methods

2.2.1. Histogram slide change detection. This slide change detection method is based on the comparison of the color histogram of successive frames. When an important change in the color histogram is detected, i.e. a difference higher than a prefixed threshold, a

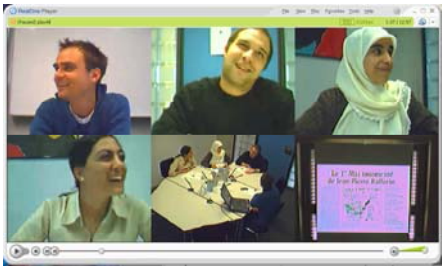

Figure 1: Meeting data captured. At the bottom right, the movie of the slideshow captured.

slide change is signaled. In the first approach we developed, we used both color and gray histograms in order to detect slide changes. Color histogram is performing well for successive slide having different background color [1]. However, in real slide presentations, most of the slides have the same background, generally corresponding to a design pattern. In this case, only the text layout and the graphical content vary. Thus, the histogram technique is not adapted to detect such changes, especially with a low-resolution camera and the resulting poor contrast level. Again, web cams are auto focusing when there is a slide change and in the following transition period, histogram techniques detect inexistent slide changes.

2.2.1. Cornell lecture browser's method. This method uses a feature-based algorithm [1]. Frames are extracted from the video, low pass filtered for noise reduction and finally adaptively thresholded to produce binary images [4]. The similarity between two successive binary images corresponds to the number of common black pixels. A slide change is signaled whenever the dissimilarity between two successive binary images exceeds the threshold, defined so that there should not be slide changes undetected. Indeed, even if some extra slide changes are detected, the redundant ones are removed after identification of the corresponding extracted images.

This method works perfectly with slideshows having high contrast, either having light background and dark text or vice versa. However, it is difficult to set a unique threshold value for various slideshows having heterogeneous background color or graphical content. Finally, in the real cases we have evaluated, many extra slide changes are generated due to the webcam's auto focusing nature.

2.3. Proposed Fribourg method

Instead of trying to detect the slide changes, the method we propose is looking for slide stability, i.e. periods during which a unique document image is displayed. To be more precise, our algorithm looks for the stability of a queue of frames, corresponding to 2 seconds of video, rather than a change in successive frames. Our assumption is that no relevant slide change occurs during the 2 seconds after a slide change has been signaled. Indeed, in real world presentation, slides that are visible less than 2 seconds should not be considered because people do not have time to read them.

First of all, our algorithm slices the slideshow movie in several queues of N frames. Then, all the frames in the queue are processed to determine whether the queue is stable or not. Frames are converted to binary images like in Cornell's method. The first frame in the queue is then compared with the rest of the frames in the queue in order to compute a statistical value, i.e. a combination of the dissimilarity distance mean, variance and standard deviation. If this value overcomes a pre-fixed threshold, the queue is considered unstable and it is analyzed in order to find out the exact slide change position.

Just after a slide change, the fade-in fade-out transition can generate slide images overlapping, i.e. the combination of two successive slide images. This is due to the relatively high frame rate of our acquired video. Furthermore, after a slide change, because of the auto-focusing problem, the dissimilarity distance gradually stabilizes. In order to get the exact slide change position, we look for the frame which dissimilarity value approaches the most the mean dissimilarity value, i.e. the average of the minimum and the maximum dissimilarity distance in the queue. This way, we'll get either the exact overlapped image or the first frame in the queue displaying the new slide.

Finally, because of the images instability during the transition period, the image extracted for further identification does not correspond to the slide change exact position. We look for a more stable frame, determined by the maximum deviation of images in the queue from the previous slide image.

2.3.1. Slide Change Vs Animation Detection. When a new slide is signaled, the animation detection method starts working simultaneously. The procedure is similar to the slide change detection but uses a finer threshold and each frame in the queue is compared with the previous and next frame, because animation is a very local feature.

2.4. Evaluation

2.4.1. Slideshow corpus and ground-truth. We have developed an application for automatically evaluating the various slide change detection algorithms. About 300 power point presentations have been first

collected, mainly from conferences, student projects and courses available on the web. More than 3 thousands slides have been accumulated this way, which represent many kinds of presentation styles. We have then built a corpus trying to equally balance various characteristics such as number of slides, background color, font color/size and background variability, graphics content, etc. Each slide has first been converted to a JPEG image (720 x 540). Further, a SMIL file has been randomly produced for each presentation stored in the database with randomly picked up images and random timestamps for each slide change. The slideshow's duration and minimum/maximum presentation time of each slide has been defined so that slideshows are realistic. The time information in this SMIL file is further used as the ground truth for evaluating the slide change algorithms.

After the creation of the SMIL file, a master PC commands the slave presenter (lap top, pc) to start playing the SMIL presentation. Simultaneously, it orders the capture box to start filming the projection screen. The master PC gets the slides' start and stop time from the ground-truth file and accordingly it controls the slave capture box and slave presenter. When the capture is over, the raw video is compressed and transferred to the server for later segmentation.

2.4.2. Metrics for evaluating segmentation. The various slide change detection algorithms, we have developed, generates an XML file containing the timestamps for each slide change and the filename of the corresponding image extracted. The ground truth being also in XML format, the comparison is straightforward. We have decided that the difference between the ground truth's timestamps and the computationally detected timestamps should not exceed 1 frame. Recall and precision has then been measured as follow:

- Recall = D / GT, Precision = D / T
- D = Number of correct slide changes detected
- GT = Number of slide changes in ground-truth.
- T = Total number of slide changes detected.

2.4.3. Results. We considered 29 slideshows in this experiment. Graphics content, background, font size and font color variations were measured on a scale of 1 to 5 and kept in a table. The performance of Fribourg's and Cornell's methods are clearly better than gray and color histograms for a tolerance of 1 frame (figure 2). For this slideshow corpus, the average recall measure of the Cornell method is 0.29 and its average precision is 0.17 (F:0.18). However, Cornell method [1] uses a slide identification mechanism for confirming the slide changes, which should considerably increase the precision but as well the processing time. Whenever there is a signal for slide change, the corresponding frame is extracted and compared to a queue containing only one image per slide in the slideshow. If the extracted frame is identified as a new slide then only the slide change is confirmed.

The high number of incorrect slide change, detected in Cornell method, drastically increases the computational work. This drawback is overcame by the Fribourg method, which does not need to perform slide identification in order to increase precision (R: 0.83, P:0.82, F:0.83). We have also tested all the methods with a tolerance of 2 frames, meaning that a slide change detected 2 frames later or before the exact ground-truth position is tolerable. In this case, the Cornell's method considerably increases its performance (factor 3, F: 0.56) and Fribourg is getting nearly perfect (F: 0.93). Further, for a tolerance of 4 frames, the Cornell's method does not get any better.

Further, we observed the whole slideshows population in order to calculate correlation coefficients between the various methods and the slideshows characteristics. The only significant correlation found is between the color histogram method, which is the only one that does not convert images to grayscale, and the background variation. Further, the color histogram method should perform better for slide change than the grayscale histogram, because of the loss of color information in the second method. In spite of this, the grayscale histogram (F:0.19) shows better performance, for tolerance 2 frames, than the color histogram (F:0.07), due mainly to the auto focusing nature of the web camera.

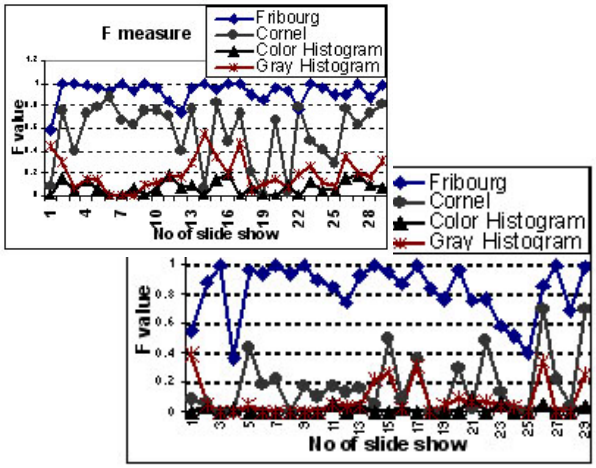

Figure 2. Slide change detection results on 30 slideshows. Tolerance 1 frame (bottom), and 2 frames

```
<VisualSignature>
    <PixelRatio Full="0.29" W1="0.41" W2="0.32" W3="0.19" W4="0.22"/>
    <BgPattern Pattern="P2"/>
    <BgColor Color="#145b243"/>
    <BoundingBox count ="7"/>
    <HasHorizontalText count ="7">
        <Sentence y="58" x="23" width="487" height="28" Words="5"
                  PxRatio ="0.55"/>
        …
    </HasHorizontalText>
    <HasImage count ="2">
        <Image y="5" x="8" width="708" height="40" PxRatio="0.95"/>
        …
    </HasImage>
    <HasHLine count ="0"/>
    <HasBullet count ="0"/>
    <HasVLine count ="0"/>
    <HasVerticalText count ="0"/>
    <HBarWithText count ="0"/>
</VisualSignature>
```

Figure 3. An example of slide visual signature.

3. Slide identification and future work

The images extracted from the slide change detection process are further compared, with the original document images in the database, in order to identify them. The slide identification method we have implemented is based on:

a) The extraction of a hierarchically structured visual signature (figure 3), containing global features, for both images extracted from the video and images of the original document. The extraction is based on several document image analysis methods such as Run Length Smearing Algorithm [3], connected components, projection profiles, etc.

b) A multi-level comparison of those visual signatures, which follows their hierarchies. The highest-level features are first compared; all the images in the database, which similarity overcomes a prefixed threshold, are kept. The comparison continues on the resulting sub-set of images with lower-level features. When all the feature's levels have been compared, the best images are kept (on a global basis, i.e. a weighted combination of all the features) and the comparison restarts at the root of the visual signature hierarchy with a more restrictive threshold.

A major advantage of this method is that it does not require any classification technique [2]. It is fast, mainly because the visual signature hierarchy guides the search towards fruitful solution spaces. Further, by alternating feature-specific matching with global distance comparison, it guaranty that no good solutions are avoided. A preliminary evaluation has shown that this simple method performs well for slideshows having a homogeneous background, without complex textures (Recall: 0.92, Precision: 0.87).

4. Conclusion

We proposed in this article a method, based on document image analysis, for integrating non-temporal documents into multimedia meeting archives. In particular, we presented a technique for slide change detection, which overcomes the auto focusing and non-uniform lightning nature of webcams. We compared the performance of this technique with all other existing methods (Cornell, color and gray histograms) and it shows the best results on a recall and precision basis. Further, our method is highly precise without having to perform slide identification, like in Cornell's method, which generally implies a significant additional processing.

We also proposed an automatic method for building a trustful ground-truth for slide change detection, using a synchronized multimedia integration language (SMIL), and we described the related evaluation procedure. Finally, the above application is not only applicable for the meetings but also for lectures, seminars, organizational presentations, etc. Further, considering the high performance of Fribourg method, the slideshow's video can be used as a control stream for indexing a meeting. In the near future, we plan to improve our identification method, so that it can handle slideshows having complex background textures. Further, we plan to detect and identify finer state changes (scrolling, zooming, etc.) and partial document identification (pointed document parts, occluded documents, etc.). In a longer term, we plan to apply this identification method to other low-resolution document' images, such as documents exhibited on the meeting table.

5. References

[1] Mukhopadhyay, S., and Smith, B. (1999) Passive capture and structuring of lectures, proceedings of the seventh ACM international conference on Multimedia (Part 1), Orlando, Florida.

[2] Shin, C., Doermann, D. and Rosenfeld, A. (2001) Classification of document pages using structure-based features. Int. J. Document Analysis and Recognition, Vol 3, pp 232-247.

[3] Wong, K., Casey, R. and Wahl, F. Document Analysis system. IBM J. R & D, Vol 26, pp 647-656, 1982.

[4] Yanowitz, S. and Bruckstein, A. A new Method for Image Segmentation, Computer Vision, Graphics and Image Processing, Vol. 46, No. 1, 1989, pp. 82-95.