STUDYING ON THE MOVE — ENRICHED PRESENTATION VIDEO FOR MOBILE DEVICES

Andrew Winslow (1), Qiyam Tung (1), Quanfu Fan (1,3), Juhani Torkkola (1), Ranjini Swaminathan(1)
Kobus Barnard (1), Arnon Amir (2), Alon Efrat (1), Chris Gniady(1)

(1) Department of Computer Science, The University of Arizona, Tucson AZ 85721
(2) IBM Almaden Research Center, 650 Harry Rd., San Jose, CA 95120
(3) IBM T. J. Watson Research Center, 19 Skyline Dr., Hawthorne, NY 10532

ABSTRACT

The rapid adoption of distance learning means that significant lecture video content is available on-line. However, access from mobile devices is hampered by low bandwidth and small screen size. In this paper, we address these issues by manipulating two key elements of lecture video—displayed slides and laser pointer gestures. Displayed slides need to be very crisp compared to background content. Fortunately, the needed data is available from the presentation slides, and we describe a method for splicing them into the video on the client side, increasing fidelity, and reducing bandwidth needs. This operation removes laser pointer gestures, which are often lost due to compression, and are hard to see on the small screens of mobile regardless. But these gestures are part of what makes watching lecture video different than simply looking at the slides. Hence we interpret the laser pointer gestures as we analyze the videos, creating representations that can be transmitted at a low cost. These representations can then be iconified on the client side and displayed clearly.

I. INTRODUCTION

Distance learning is widely utilized in university and corporate environments. The key component of the distance learning is the video stream prerecorded or transmitted live through Internet. Maximal quality of the video stream is desired on the client side, especially in areas where detailed information such as text and graphs is being displayed. However, good quality video is often not available due to capture, even in the case of non-mobile devices.

High quality production with multiple cameras may be costly, labor intensive, and require high bandwidth connection for delivery that may not be available on mobile devices. Lower resolution video production, on the other hand, reduces production costs and requires less bandwidth to transmit but may be frequented by common problems such as inadequate illumination, significant color distortion, and images in video that lack clarity. Subsequently, lower quality video affects readability of the text slides captured in the video, impairing learning ability of the viewer. Therefore, distance learning is likely to lose the vibrant, interactive, and insightful context that exists in the lecture room.

To improve the accessibility of the video we developed the SLIC (Semantically Linked Instructional Content) system [4], [5] at the University of Arizona in collaboration with IBM Almaden Research center. A major component of the SLIC system is an automatic time alignment of slides and video – in a video of a lecture, utilizing a set of the slides that are used in the lecture. SLIC system is capable of finding time and location of each presentation slide in the video. The mechanisms are robust, and successful even in challenging conditions such as blurry, partially occluded, and non-rectangular slides appearing in the video. Additionally, SLIC provides a slide-based video browsing tool, improving accessibility of particular topic within the video.

Recent advances in our SLIC system provided us with improved accuracy of the geometric mapping (homomorphism) between pixels of the external slides and their appearance in the video frames [3]. The improvement homomorphism allowed development of an entire spectrum of tools for improving readability and accessibility of videos for mobile devices such as PDAs or Smartphones. In particular, we can significantly improve the quality of the video and simultaneously reduce bandwidth needs by splicing in the slide images into the video stream on the client side. However, doing so, removes the laser pointer signal which is not readily visible on small screens regardless, especially with aggressive compression. This is unfortunate, as one of the differences between watching the video and looking at slides is the notion of interaction and guidance from the lecturer.

We propose mechanisms to capture variety of laser pointer gestures used by the presenter. Furthermore, we demonstrate the potential of the proposed mechanism by a relatively simple application, which highlights the word pointed to in the video. Doing this requires the accurate homomorphism mentioned above because this establishes how an video

image location maps to the original slides. Having done this analysis, we can then interpret or iconify the gesture on the client side. For example, we can add a box around the word pointed to, using color and thickness specified by the user interface. This uses little bandwidth, as one only needs to transmit the coordinates of the corners of the box.

I-A. Related Work

Systems for analysis, indexing, search, and browse of videos for education have been studied extensively in the past decade [13], [8], with a detailed survey presented in [6]. Recently, several successful e-learning systems have been developed by IBM, including CueVideo [1], and ViaScribe [18]. CueVideo provides tools to quickly convert videotaped lectures or conference presentations into searchable on-line video proceedings, as well as automatic slide matching for topical indexing [14]. ViaScribe [18] is a live captioning system, designed to make classroom lectures accessible for the hearing impaired. It enables the presenter's computer to capture and log all slide changes, runs live captioning using speech recognition, captures all the synchronized channels, and allows post-lecture editing and distribution. While each of these systems makes use of both slides and video, none of them utilizes the external slides for improving accessibility to the videos of the lectures.

II. BACKGROUND: SLIDE MATCHING AND BACKPROJECTION

The fundamental component that the proposed laser pointer enhancement mechanisms relay on is an accurate mapping between the external slides and the slides that appear in the video. This mapping — a linear transformation in homogeneous coordinates — can be used to splice high resolution slide images into video images (back-projection). This process of substituting the slide image in the video frames with high resolution images from external slides is called *backprojection*. Subsequently, the text and details of the slides that are unreadable in the original video would appear much clearer and readable after back-projecting the slides. The mapping is also required to interpret laser pointer motion in the context of the original slides, as it tells us where laser pointer locations on the screen are within the slide, which can be related to the content, as discussed in the next section.

To successfully apply backprojection, we rely on high accuracy of homography between video frame locations and external slide locations. We represent image points using homogeneous coordinates. Given two homogeneous points $\mathbf{x} = [u, v, w]^T$ and $\mathbf{x}' = [u', v', w']^T$ linked by a homography $\mathcal{H}$, the mapping between $\mathbf{x}$ and $\mathbf{x}'$ is expressed by

$$\mathbf{x}' = \mathcal{H}\mathbf{x}, \tag{1}$$

where H is a 3×3 matrix. The homography $\mathcal{H}$ has 9 elements, but only 8 degrees of freedom as scale is not relevant. Thus we can set any element of $\mathcal{H}$ as a constant (e.g., let $H_{33} = 1$). Our method for automatically matching slides to video frames, and determining H accurately is described elsewhere [4], [5], [3].

To apply back-projection we use the established homography to link video frame pixels to geometric regions in the slide. Note that there is no 1-1 correspondence between pixels. Rather video frame pixels correspond to regions. For each pixel $F_{i_{(x,y)}}$ in the frame region of frame i of the video, we must interpolate a color from the slide image S_j. A single pixel from the frame F_i can map to an area of the slide S_j whose axes are not orthogonal to those defining the pixel grid of S_j. This area can consist of full pixels and partial pixels that can be described form various polygonal areas, as seen in the shaded regions in the previous figure. However, an interpolation scheme like bilinear filtering is appropriate in the case of simple image rescaling, in the case of our transformations, bilinear interpolation yields unsatisfactory results. Instead, we map every pixel $F_{i_{(x,y)}}$ in the slide region of F_i to a weighted average of the pixels in the area $A_{j_{(x,y)}}$ of S_j as defined by transforming the corners of the pixel $F_{i_{(x,y)}}$ with the frame to slide homography H_i^{-1}. The weights in the weighted average are defined by the exact geometric areas of the polygonal regions of $A_{j_{(x,y)}}$ inside each pixel of S_j that intersects with $A_{j_{(x,y)}}$. The averaging process is done through an iterative scanline algorithm. Figure 1 demonstrates the improvement in the quality and readability of the original frame (bottom), comparing to the backprojected frame (top). Note that the projected slide region is sharper and the color is improved.

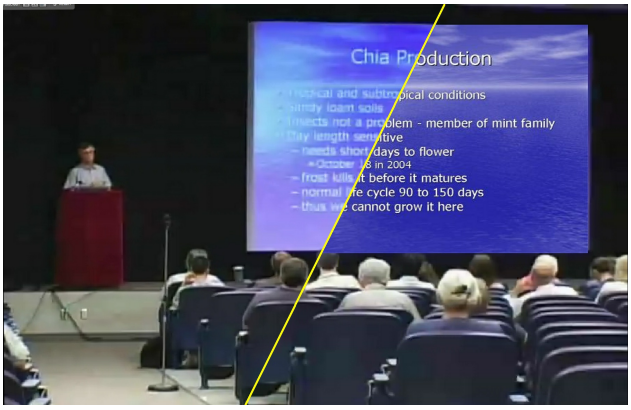

Fig. 1. *Comparison between the original video frame, vs. frame after backprojection of the slide.*

Further details are discussed in a technical report on general methods for improving educational videos [6]. We utilize the backprojection mechanisms as proposed in [6] with a significant modifications to adapt it better to video transmissions in mobile environment with low bandwidth availability. The strategy proposed [6] for using backprojection is to apply it to each frame of the original video, pixel

by pixel. The approach, where applicable, has the advantage of enabling very accurate backprojection, allowing color correction, and correspondence of other artifacts presented in the original video.

However, for PDA and smartphone applications, a different solution is preferred. We use the accurate mappings described above (accurate positions and timing) from slides to video frames to utilizing Synchronized Multimedia Integration Language (SMIL), and send to the client the original video, the slides, and metadata. SMIL is then used to integrate the slides into the video on the client side, using the accurate positions and timing we have computed, as described above. This method is much more suitable for mobile applications, where low bandwidth limits the communication. It allows a higher compression while ensuring high quality where it is needed. The loss of detail from aggressive compression can be directed to less significant parts of the video such as the audience in the classroom, while the slides appearing in the video will stay sharp. Therefore, this technique significantly improves delivery and display of educational video in mobile devices with limited bandwidth availability and small screens such as PDAs or Smartphones.

Comment: For both backprojection and laser pointer, we have used SMIL and the Ambulant player. The player's specification indicates that SMIL will soon be fully compatible also on mobile devices running different versions of Windows Mobile 5, but this is not the current state, and thus we have ran our experiments on a desktop monitor, using the only a small portion of the screen. We believe that this only a partially hampered the evaluation of the true user experience.

III. LASER POINTER TRACKING

Laser pointers are frequently used by speakers in presentations to indicate a specific area of interest on a slide. Understanding where the laser pointer is in the video frame and hence in the presentation slide provides us with the topic of focus at that instant in time. The additional context provided by the extracted keywords from the laser pointer detection will improve searching and browsing capability of the topic based search system of the instructional videos. Furthermore, highlighting the region pointed to by the laser pointer will improve the presentation delivery context that may be otherwise lost when watching the video using a PDA. In this section, we discuss how our system tracks the laser pointer, computes its corresponding positions on the slide so highlighting of the corresponding words is possible.

The majority of laser pointer tracking research has focused on incorporating laser pointers as an input devices[7], [15], [11], [12]. In this section, we describe a system that extracts laser pointer usage information from external video. Once basic pointer movements are extracted, they are used to utilized in three ways: (1) to enhance the visibility of the pointer in the existing video, (2) to condense pointer movement into gesture shapes, such as a curve circles around a paragraph in the slides, and (3) to determine what slide text (which words(s)) the laser pointer is pointed to. The last application is of particular potential for helping PDAs' users, due to the challengingly readability of the small screen and studying environment (e.g. during commuting).

III-A. Linking Slide Locations to Words

The first challenge to address is extracting the target information, namely what are the coordinates inside the slide of word that appears in the slide. Parsing the slides-file itself is possible, but is not easy, and is limited to specific file formats (e.g. PowerPoint). To deal with the various existing presentation formats, we prefer to convert them into PDF's and image files. We then use an open source program (pdftotext) to extract its text. With this information, we create PDF's and images that have a single word missing in each file and generate their corresponding image files. Finally, to find the coordinates for the box bounding the word, we subtract pixel-by-pixels this image from the image of the original slide, and seek a relatively large area where this value is above a threshold.

III-B. Detection and Enhancement

Since input video comes from external sources, scene brightness and color is unknown and frame differencing is used to remove these variations. For each frame of the video, a second frame is computed as the difference between the corresponding frame and a pixel-wise average of a set of proceeding frames in the original video. A median filter on brightness approximately equal in size to the laser pointer is then passed over each frame of the difference video. The brightest pixel from the resulting image is then selected.

Sequences of these brightest pixels are then fit to curves. The fitting is done using axis-aligned cubic equations parameterized over time. A least squares method is used to minimize the squares of the distances from each brightest pixel to the closest point on the curve. The fit of the approximating curve is used to determine final laser point locations, i.e. it coordinates in the video frame. The transformations of Section II are then used to find the corresponding location in the slide.

Preliminary experiments have shown that the algorithm described previously is able to detect laser pointer locations even in video with poor lighting and color. In these situations, a laser point may be dim and shaky. Visibility of the point is improved by superimposing a larger, brighter dot, as depicted for example in Figure 4. Shakiness is removed naturally by the algorithm through the curve-fitting process. We can combine these techniques with the back-projection method described previously to create a viewing experience, which is both true to the original video and dramatically improved.

III-C. Laser Pointer Gesture Interpretation.

Gesture interpretation focuses on distilling laser pointer movement into high-level patterns, which provide an alternative method for conveying laser pointer information. This is particularly useful for broadcasting video over mobile phones, PDAs or other low-bandwidth outlets where streaming video is not always a possibility. A combination of slide images, gesture shapes, and audio could provide a low-bandwidth lecture video alternative, which contains nearly all the instructional content of the original video in a fraction of the data.

Extracting laser pointer gestures is done in two phases: parsing and classification. Parsing is determined from point movement speed. A simple 2-means clustering divides determines the threshold between gesture (slow) speeds and non-gesture speeds. Temporally clustered frame sets containing gesture-speed pointer movement are combined. Each clump is then classified as either underlining, highlighting or circling. Gesture types are determined by a voting scheme utilizing Hough transforms. While a more precise classification is possible, these basic gesture types are sufficient for nearly all laser pointer usage in an instructional setting. Hence classification can done by simple heuristic methods. Gestures that have a width to height ratio of at least 4:1 and a maximum height of 20-30 pixels (a few times taller than the point itself) are classified as underlinings. For gestures that are not underlinings, an ellipse Hough transform is used to determine if they are circlings. The existence of an ellipse which has all points in the gesture within 10 pixels (the diameter of a point) of its boundary is used as the criterion for a gesture to be classified as a circle. All other gestures are classified as highlightings.

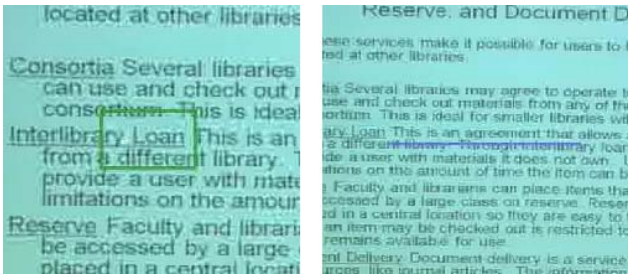

Fig. 2. Two instances of gesture recognition in a lecture video. In the left image, a highlighting gesture is indicated by a green box around the region being indicated. In the right image, an underlining gesture is indicated by a blue line along which the laser pointer traveled over several seconds.

In Fig. 2, two examples of gesture detection are seen. As discussed in Section II, when the video is sent wirelessly to a mobile device, there is a huge advantage to send an aggressively compressed video file, and a SMIL file containing the slides and their positions. We are using the SMIL mechanism to specify on the displayed video (on the client side) the information obtained from the laser tracking, such as which word is emphasized. See for example Figs. 3,4 and 5. These shapes contain nearly all of the information contained in the raw laser pointer location data, but are stored as only a handful of coordinates combined with start and end time information.

III-D. Laser Pointer Experiments

Frame differencing alone is not robust enough to provide accurate detection of a laser pointer in poorly-lit or low-quality video. Fitting curves to pointer movement over a series of video frames provides a significant increase in the detection rate of a laser pointer in these situations. We have succeed to discover the location of the pointer in each frames of a video footage shot using a small medium-quality camera and suffers from pixelation and artifacts whose size is similar in scale to the laser pointer. Using our algorithm, we can discover the location of the pointer in each frame and overlay a simulated pointer, which is larger and brighter than the original.

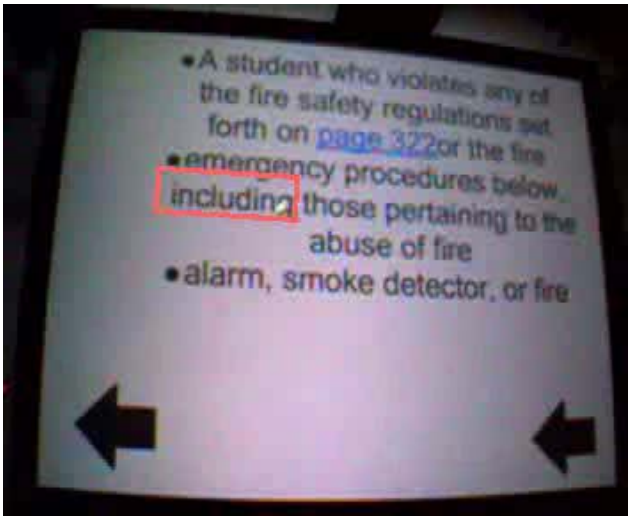

Fig. 3. In this figure, the laser pointer is replaced by a white point for clarity. After the word indicated by the lecturer is found, we emphasize this word by a box by adding a red box encapsulating this word. In this case, this box is added on the original video, which is more accurate, since it can be modified to the non-linearity of the projection transformation.

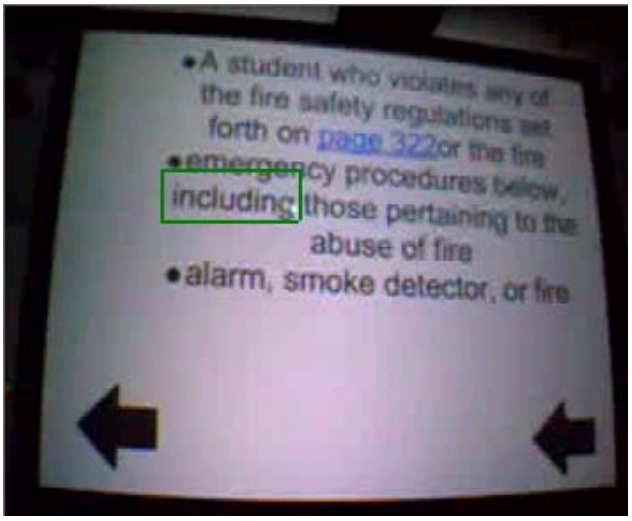

Fig. 4. In this case, the box is added using SMIL in the client side, and its shape fits slightly less accurately with respect to the geometry of the screen. It is still quite servicable for enriching the presentation for mobile devices.

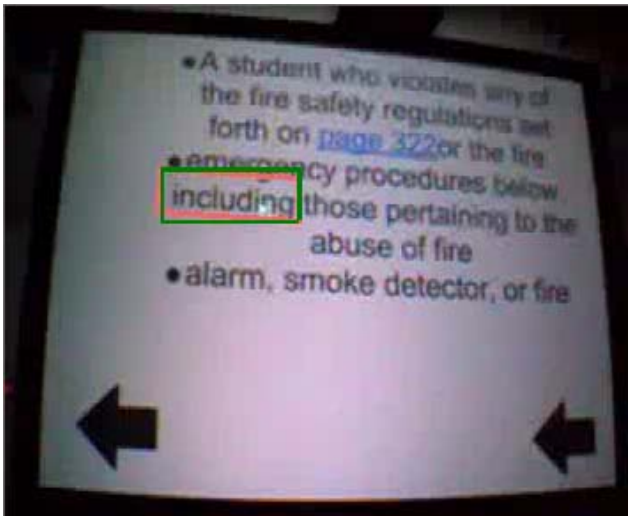

Fig. 5. Here the two boxes are drawn together for comparison, showing that minor difference in accuracy between them.

IV. REFERENCES

[1] A. Amir, G. Ashour, and S. Srinivasan, "Automatic generation of conf. video proceedings," in *Journal of Visual Communication and Image Representation, JVCI Special Issue on Multimedia Databases*, 2004, pp. 467–488.

[2] D.H. Ballard, "Generalizing the Hough Transform to Detect Arbitrary Shapes", Pattern Recognition, Vol.13, No.2, p.111-122, 1981.

[3] Q. Fan, A. Amir, K. Barnard, R. Swaminathan, and A. Efrat, "Accurate alignment of presentation slides with educational video," Tech. Rep., University of Arizona, *http://kobus.ca/SLIC/publication/SLICBundleTR.pdf*, 2009.

[4] Q. Fan, A. Amir, K. Barnard, Ranjini Swaminathan, and A. Efrat, "Temporal modeling of slide change in presentation videos," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2007.

[5] Q. Fan, K. Barnard, A. Amir, A. Efrat, and Ming Lin, "Matching slides to presentation videos using sift and scene background matching," in *8th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2006.

[6] Q. Fan, J. Torkkola, R. Swaminathan, A. Winslow, K. Barnard, A. Amir, A. Efrat, and C. Gniady, "Content enrichment in presentation video," Tech report, University of Arizona, 2009.

[7] Carsten Kirstein, Heinrich Mueller, "Interaction with a Projection Screen Using a Camera-tracked Laser Pointer," mmm, p. 191, 1998 MultiMedia Modeling, 1998.

[8] S. Mukhopadhyay and B. Smith, "Passive capture and structuring of lectures," in *ACM Multimedia (1)*, 1999, pp. 477–487.

[9] Myers, B. A., Bhatnagar, R., Nichols, J., Peck, C. H., Kong, D., Miller, R., and Long, A. C. 2002. "Interacting at a distance: measuring the performance of laser pointers and other devices". Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Minneapolis, Minnesota, USA. CHI '02. ACM, New York, NY, 33-40.

[10] Nakano, Ochi, et al. "Unified Presentation Contents Retrieval Using Laser Pointer Information". Proceedings of the 21st International Conference on Data Engineering (ICDE 05).

[11] Olsen, D. R. and Nielsen, T. 2001. Laser pointer interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Seattle, Washington, United States). CHI '01. ACM, New York, NY, 17-22.

[12] Jean-Francois Lapointe and Guy Godin. "On-Screen Laser Spot Detection for Large Display Interaction". HAVE 2005.

[13] J. M. Rowe and L. A. Gonzelez, "BMRC lecture browsers," in *http://bmrc.berkekey.edu/frame/projects/lb/index.html*, 1999.

[14] T. F. Syeda-Mahmood, "Indexing for topics in videos using foils," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. II: 312–319.

[15] Xu, Richard. "A Portable and Low-Cost E-Learning Video Capture System". Advanced Concepts for Intel-

ligent Vision Systems.
[16] Yokota, Kobayashi, et al. ”Unified Contents Retrieval from an Academic Repository”.
[17] “—”
[18] ViaScribe, “IBM ViaScribe,” in *http://www-03.ibm.com/able/solution_offerings/ViaScribe.html*, 2005.