ITC Strategic Initiative Grant Final Report
A Tablet PC-Based, Cooperative Learning Environment for a Multidisciplinary IT Course

PI: W. Richards Adrion, CS Professor adrion@cs.umass.edu
Other Participants: Paul Dickson, CS Ph.D. Student; Scott Myers, CS MS Student

Total Amount of Funds Granted: $24,847
Revised 7/21/2008

Summary

We planned to create a problem-focused, cooperative teaching and learning setting, supported by a *tablet-PC-based environment*. The strategy was to revise the curricula, materials, and pedagogy for a core course in the University of Massachusetts Amherst IT Program (CMPSCI 120), to capture and distribute multimedia records of each course session, and to evaluate the system by delivering courseware via UMassOnLine to at least one other institution. We have been very successful in extending our research on automatic classroom content capture and multimedia content delivery systems, but we ran into unexpected problems obtaining, installing and testing the necessary servers and camera equipment. These equipment difficulties (not completely overcome until spring 2008) resulted in a significant revision in our strategy for pedagogy, evaluation and assessment. We did develop a sophisticated classroom capture system and two content delivery systems. We were able to partially evaluate these systems in courses during the 2007-2008 academic year and that allowed for some pedagogy exploration. We will reach our rather ambitious goals, but at a slower pace than hoped.

The RIPPLES group at the University of Massachusetts Amherst has used the MANIC [1-2] content delivery framework for more than a decade and has evaluated several MANIC content delivery systems in a variety of educational and pedagogical settings [3, 18-21]. Instantiations of the MANIC content delivery system provide a platform by which content can be delivered in a format that includes features proven effective in distance education. Several of these features were developed to support on-campus teaching and learning, including search, navigation, notation, and collaboration tools. The MANIC framework was to be used to develop a tablet-based PC environment, however we developed a viable alternative for in-class use (AutoPresenter [7]).

Our effort to capture classroom content employs the PAOL (Presentations Automatically Organized from Lectures) system. PAOL emphasizes *transparency* achieved by using a computer screen capture device, wireless microphones, and high-resolution cameras, all of which function unobtrusively. PAOL captures a small set of images of the computer screen and whiteboard (typically under 100/hour) that represent “significant events” (a slide transition, inking on a tablet, the state of an application, a figure, or writing on a whiteboard) together with a digitally zoomed tracking video of the speaker. These images and video can be compiled, synchronized, and indexed into a FlashMANIC multimedia presentation or the images can be distributed in real-time using the AutoPresenter interface and server.

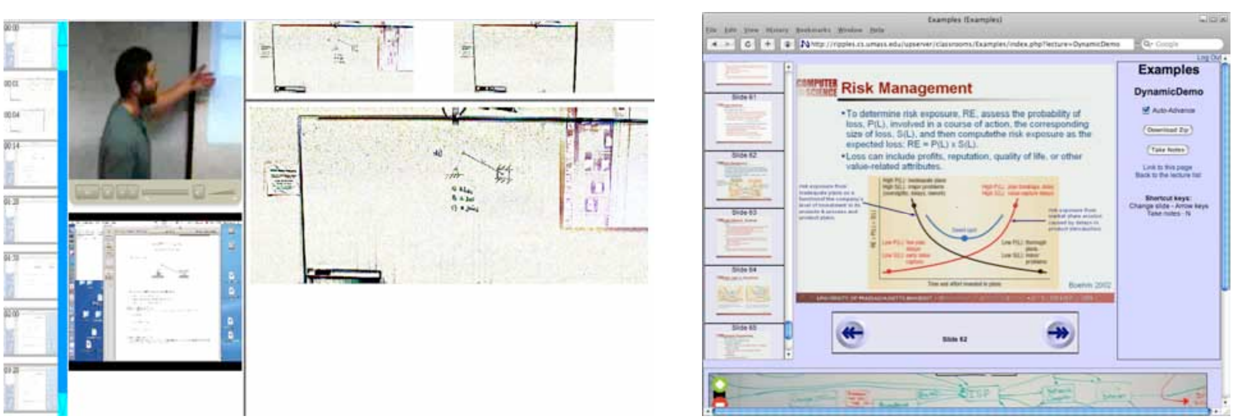
PAOL laptop capture and AutoPresenter were used in a software engineering class in fall 2007. The experiment led to some revisions of the server technology. The full PAOL system was used in spring 2008 in an operating systems class. That experience, along with various in depth focus groups with users, led to a redesign of the FlashMANIC interface. In both experiments, the

captured content proved to be valuable to the students in class. In fall 2008, we intend to capture two courses and, if we can arrange the scheduling, we will add one of the IT minor core courses.

Content Delivery

The RIPPLES group at the University of Massachusetts Amherst has used the MANIC [1] content delivery framework for more than a decade and has evaluated several MANIC content delivery systems in a variety of educational and pedagogical settings. Instantiations of the MANIC content delivery system provide a platform [2] by which content can be delivered in a format that includes features proven effective in distance education. Several of these features were developed to support on-campus teaching and learning, including search, navigation, notation, and collaboration tools. Recently, the RIPPLES group developed jMANIC [3], which we intended to use for this project. The jMANIC content delivery system has a "plug-in" architecture that allows rapid customization of the system functionality. Faced with the lack of Java Media Framework implementations that provide the necessary time-indexed video access, we were forced to develop a Flash-based interface for compiling PAOL content. We continue to work in parallel on jMANIC and on incorporating the MANIC tools for searching, notation, sharing, and collaboration into FlashMANIC. In these efforts, we hope to create a powerful learning environment around PAOL captured content.

Content delivered using the MANIC technologies typically was captured by human operators and required production crews to create high-quality presentations. While RIPPLES tried a few approaches to automate content creation and capture, the demand for quicker and less expensive production and more flexibility in content delivery led to the development of PAOL (Presentations Automatically Organized from Lectures), a content capture system. In PAOL,



(a) FlashMANIC (b) AutoPresenter

FIGURE 1 CONTENT DELIVERY SYSTEMS

transparency is achieved by using a computer screen capture device, wireless microphones, and high-resolution cameras, all of which function unobtrusively. PAOL captures a small set of images of the computer screen and whiteboard (typically under 100/hour) that represent "significant events" (a slide transition, inking on a tablet, the state of an application, a figure, or writing on a whiteboard) together with a digitally zoomed tracking video of the speaker. These images and video can be compiled, synchronized, and indexed into a FlashMANIC presentation

(Figure 1a) or the images can be distributed in real-time using the AutoPresenter [4]-[5] interface and server (Figure 1b).

FlashMANIC

FlashMANIC is directly based on the MANIC content delivery framework. MANIC systems have traditionally provided an interface consisting of three separate, synchronized frames: a video of the lecturer; "slides" (or other images, text, etc.); and an index. Controls are provided to "play" (and often fast-forward/reverse and/or single step) the synchronized video and slides. The index allows a user to access the recorded presentation at any index point, In hand-produced MANIC presentations, the index is a text table-of-contents index based on the slide titles (if Powerpoint or other presentation software is used) or hand-created text indexes.

PAOL content consists of images captured from the classroom LCD projector feed (typically sampled laptop screen capture images) and from one or more (2 in our testbed) high-resolution digital cameras. Each captured image is time stamped so that it can be synchronized with a video extracted from the digital camera streams using tracking techniques. Figure 2 shows the system diagram for our primary test capture classroom. Our original design expected the content delivery system to display only the single image for each temporal index point. After capturing a number of classes and instructors, it became obvious that we would need display the latest image in all three (computer capture, right camera, left camera) streams and to offer the user an option to enlarge one of these three as shown in Figure 1a and enlarged and labeled in Figure 3. A user can mouse over the computer, camera 1 or camera 2 image and an enlargement will appear in the lower right frame. In addition, any frame can be resized. So if the presentation primarily consists of a Powerpoint slides, one would minimize the camera image frames. If it is a lecture without AV aids (speech), then one might minimize all but the video frame.

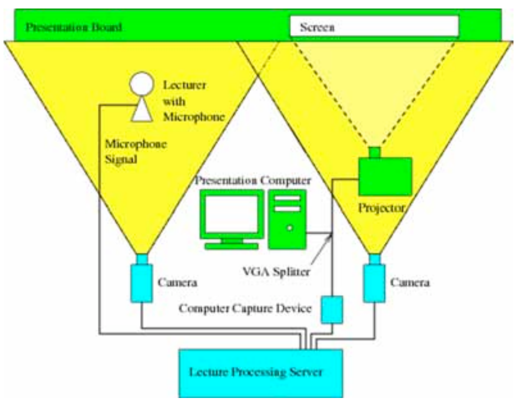


FIGURE 2 PAOL SYSTEM DIAGRAM.

When PAOL content is distributed via FlashMANIC, the system creates a thumbnail index to each significant event (image) captured and provides an index that orders these images by time stamp. Thus, the user can use the indexes to access any "significant" point in the presentation.

A user can navigate the content by playing the presentation from the beginning or from any index point. The indexes provide easy navigation to the desired points of the classroom record. This format

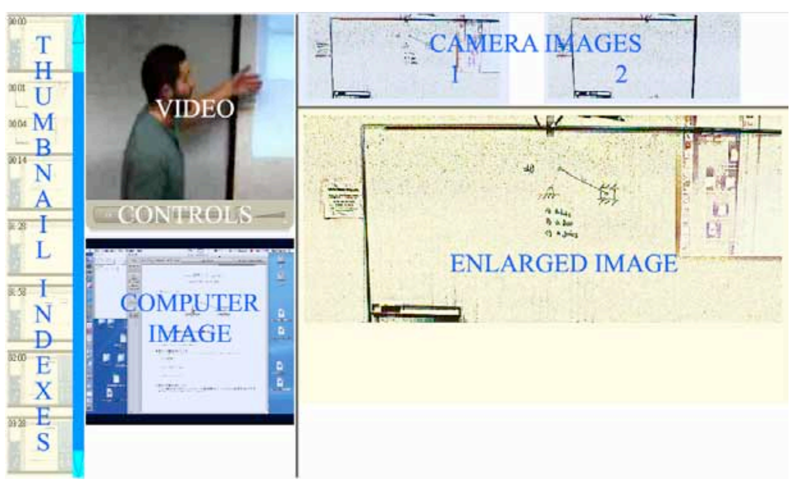


FIGURE 3 FLASHMANIC GUI.

facilitates distance education applications but also is very useful for on-campus students for study and review.

FlashMANIC was used to deliver course content for an operating systems course in spring 2008. Usability studies were carried out with the students from this class and other MANIC users. This has resulted in a redesign of the FlashMANIC layout and features.

AutoPresenter

There are a variety of research projects on classroom presentation recording and sharing. Many current systems [4-5] require both the instructor and the student to install platform-specific software onto their computers (or even use specific computers, e.g., tablets) and then learn how to use this new software. Another approach has been to allow students to view the lecture through any standard web browser but still requires the instructor to learn and use platform-specific software [6] All of these systems require the instructor to adapt his teaching style to the software, and some limit the presentation content to preexisting slide shows. PAOL allows any type of lecture content to be captured and does not require the instructor to install or use any software, but using the MANIC framework does not present lecture material in real time. It makes content available to students after class is over.

Our goal was to create a system that is platform independent, requires no special software to be learned or installed on the instructor or student computers, does not in any way restrict the style or content of the lecture, can run efficiently on commodity hardware, and can be viewed by students in real time while the lecture is being presented or at any time after class. Towards reaching that goal we created Auto Presenter [8], a real-time, platform independent system for recording and sharing of computer-based classroom presentations. The Auto Presenter system allows instructors to share their computer-based presentations (and soon whiteboard materials) with students. It requires no extra training for instructors nor special software be installed or learned. It doesn't require a specific type of computer or operating system nor special network configuration. The instructor simply plugs a video cable into their computer at the beginning of the lecture, just as they would when using a digital projector.

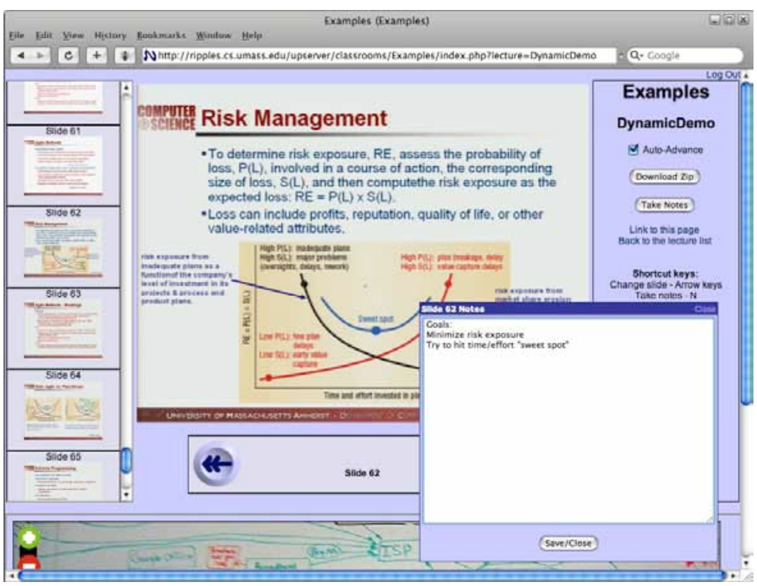


FIGURE 4 AUTOPRESENTER GUI.

Students log into the Auto Presenter website and find the lecture for the current date. As the instructor changes slides, an image of the slide is automatically loaded in the students' web browsers. Even if an instructor uses a tablet PC with digital ink, the instructor's ink markings will be sent to each student as they are written. If the instructor would like to demonstrate how to use an application, each step of the usage will be sent to the students as slides. Students are able to go back to previous slides at will, and can add their own notes to any slide. At any time after the lecture is over, students can log into the website and review the lecture slides as well as their personal annotations.

After a student has signed up for an account and enrolled in one or more classes, they can log into Autopresenter and select a class. Clicking on a lecture link brings the student to the lecture viewer. The lecture viewer (enlarged in Figure 4) presents the student with a scrollable filmstrip of slide thumbnails on the left, a full size view of the slide in the center, and lecture information and options on the right. Below the full sized view of the slide are arrow buttons that can be used to navigate between slides. Also, clicking on a thumbnail image from the filmstrip will immediately bring that slide into view. The left and right arrow keys can also be used to show the previous and next slides, respectively. When a new image is detected by the PAOL capture system, the viewer automatically adds the slide thumbnail to the bottom of the filmstrip. If the Auto-Advance option is selected, the filmstrip will automatically scroll down to this new thumbnail, and the slide will be shown in full size in the center of the screen. If Auto-Advance is not selected, the student will have to manually advance the slides.

Clicking on the Download Zip button will send a zip file of all of the current lecture slides to the student. The "Link to this page" link provides a static link to the current classroom, lecture, and slide that can be saved or shared with another student.

The Take Notes[1] button provides a pop up text box that the student uses to annotate the current slide. When the student clicks on the Save/Close button, the notes are saved to the Auto Presenter server. Clicking on the Take Notes button again will load any previously saved notes for a particular slide. Because these notes are saved onto the server, the student can log into the Auto Presenter website from any computer after class and be able to view and update the saved notes.

As noted, AutoPresenter was used successfully in a software engineering class in fall 2007. Some issues arose concerning wireless network interference in the classroom and security of the UP server software. These are being addressed.

Content Capture

Presentations Automatically Organized from Lectures (PAOL) is a classroom-capture system that is transparent to and places no constraints (teaching style or pedagogy) on a lecturer. *Transparency* to the lecturer is a key tenant of the RIPPLES group. If one is to increase the adoption of technology in education, one must keep the overhead low. PAOL is unique—no other system can capture unconstrained computer-based materials, capture whiteboard writing and drawing, and produce a speaker video without requiring preinstalled software, electronic whiteboards, or special training. PAOL uses high-resolution cameras, a screen-capture device, and a wireless microphone to automatically capture enhanced images of all material presented by computer or written/drawn/projected/mounted on a whiteboard, a video of the lecturer, and an index to support navigation. The PAOL system has been used extensively during system development in several varied settings. PAOL captured media can be delivered by one of two content delivery systems: AutoPresenter for real-time, in-class delivery and FlashMANIC for post-class and distance learning applications.

[1] Using the "N" key also accesses the Take Notes feature.

As we have noted, PAOL uses relatively inexpensive, high-resolution cameras, a simple hardware screen capture device, and a wireless microphone to produce content presented in an unconstrained manner on a computer, a whiteboard of many dimensions and aspect ratios, or a combination of both. PAOL is unique among content capture systems in its ability to capture both computer-based and whiteboard content while maintaining transparency. PAOL can handle a variety of lighting conditions—poor lighting, high-contrast projected images and the board, reflections, etc.

Lecture capture systems have been under development for years. These systems range from the basic seminar-style capture of AutoAuditorium [8] to eClass [9], a comprehensive capture system that indexes computer and whiteboard material by using electronic whiteboard and special computers to capture all material. In between are such systems as the University of California Berkeley's system [10] that captures a video of the lecturer and any slides presented and Authoring on the Fly [11], which records lectures but focuses on methods of indexing captured material.

Many other capture systems exist, and all limit the lecturer in some way, either by restricting presentation modality or requiring special software or equipment. PAOL requires neither and will be compared with the most similar systems.

PAOL is currently installed in a medium-size classroom, the setup and images of which can be seen in Figures 2 and 3. The system is also installed in a slightly smaller classroom with a single camera but with all other components the same. PAOL's computer capture device is an Epiphan Systems VGA2USB. The high-resolution cameras are a pair of Point Grey Research Flea2 color cameras that are run at 1024x410 pixels and 15 frames per second. Figure 3 shows the small size of the Flea2 camera that makes it unobtrusive when mounted in a classroom. Each camera captures a 12-foot width of the front of the room, giving a resolution of 8.5 pixels per inch, and the cameras have a 2-foot overlap of board coverage.

PAOL breaks lecture capture into 3 separate subproblems: whiteboard capture, computer capture, and video creation. Whiteboard capture and computer capture are based on the same principles. In each case, index points are created when significant content is presented. Significant content is defined as material that remains unchanged on whiteboard or computer for a set length of time. We set the time through extensive experimentation. When significant content is identified, the image (whiteboard or computer) and the time it occurred are saved. The image is displayed in the appropriate content window of the created MANIC presentation (Figure 1). Thumbnails are created from these images and the timing information is used to create a thumbnail-based index. As noted, the captured content can also be sent to students using a modified UP system [4]. The PAOL capture algorithms are described in greater detail and with greater comparison with similar systems in [12] and [13].

FIGURE 5 PAOL-EQUIPPED LECTURE ROOM AND FLEA2 CAMERA.

The whiteboard capture algorithm analyzes the stream of images captured by the camera(s), identifies significant events, enhances the associated images for greater legibility, and stores the images with timing information. To identify content to store, the algorithm first creates a whiteboard ground truth image by breaking the camera images into blocks and giving each block the average value of the brightest 25% of pixels in that block. The brightest part of each block corresponds to the whiteboard pixels in the block. The whiteboard is then made white and text, drawings and other markings are enhanced by applying equation (1) to each color channel. In equation (1), P_{out}, P_{in}, and P_{ground} are the pixel values for the enhanced image, input camera image, and whiteboard ground image, respectively, and 255 is full saturation of a color channel.

$$P_{out} = \min\left(255, \frac{P_{in}}{P_{ground}} * 255\right) \tag{1}$$

Consecutive enhanced images are compared to determine lecturer location. Taking a target enhanced image and substituting the pixels from the previous image for those pixels where the lecturer is located creates a clean whiteboard image. The whiteboard images are then compared to determine when content changes and hence when a new whiteboard image needs to be saved.

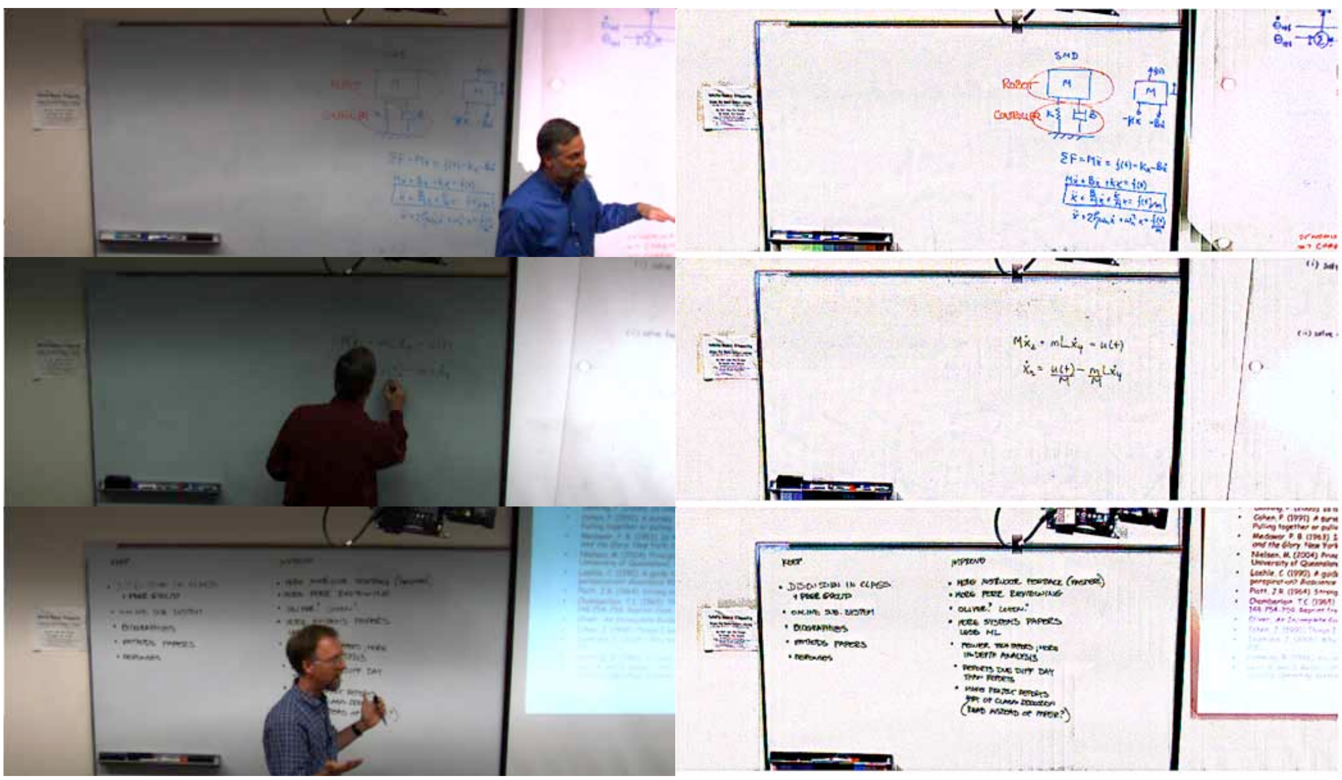


FIGURE 6 SAMPLE WHITEBOARD RESULTS.

The saved whiteboard images are then sharpened and the contrast is increased. A series of images from the cameras and the captured whiteboard content associated with them is displayed in Figure 6. These examples show the variety of lighting conditions under which PAOL functions and the quality of results despite these lighting changes. Note that in the first two examples the overhead projector material is also enhanced. The quality of the text captured is shown in Figure 7, which contains an enlargement of the text from the second example in Figure 6.

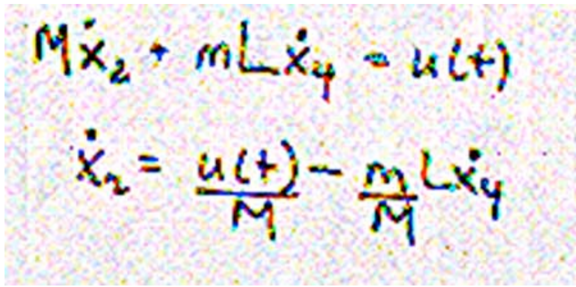


FIGURE 6 ENLARGEMENT OF CAPTURED WHITEBOARD TEXT.

One of the strengths of PAOL is the capture of the progression of ideas. An example of a concept being incrementally presented is shown in Figure 8.

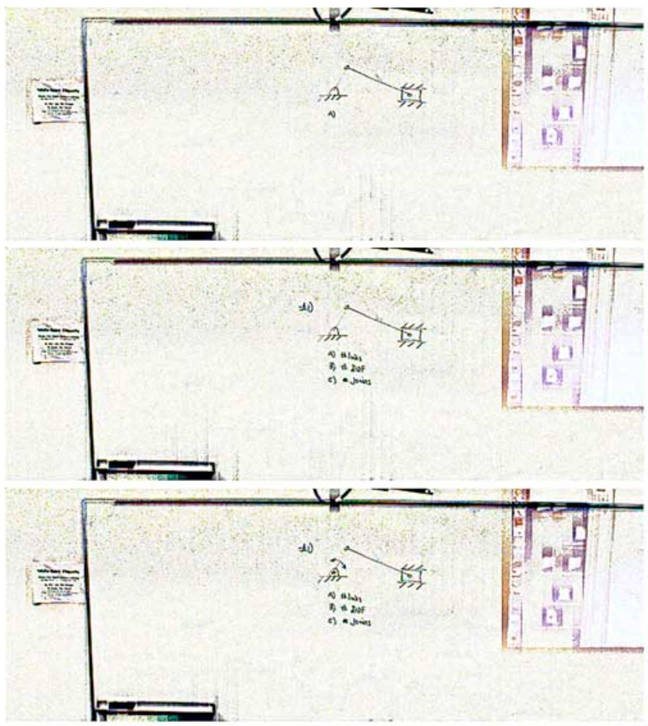


FIGURE 8 PROGRESSION OF WHITEBOARD LECTURE CONTENT.

The best systems currently available for capturing whiteboard material are Microsoft's whiteboard capture [14] and TeleTeachingTool (TTT) [15]. Microsoft's whiteboard capture provides cleaner whiteboard images than PAOL but cannot function on large whiteboards or handle faint dry-erase markers as well as PAOL. TTT requires an electronic whiteboard to be installed and requires the lecturer to start a special capture computer. PAOL is more robust than Microsoft's system and more transparent than TTT.

A video of the lecturer is created by locating the lecturer in each camera view image and creating a difference image between successive images from the same camera. The difference is used to identify a lecturer location and the location is used to crop a video frame from the original image (Figure 9). The algorithm decides which camera view to use to crop the video frame by determining which camera view contains the lecturer. A smoothing function is used to ensure that the captured video frames do not jump around.

FIGURE 9 DIFFERENCE IMAGE AND CROPPED VIDEO FRAME.

The computer capture algorithm searches for significant events to capture in the same manner as the whiteboard algorithm by looking for changes. The computer capture algorithm captures images of computer material using a VGA2USB converter. The captured image stream is processed to remove noise and identify stable (significant) screen images. As each significant event is identified, an image is saved as new content. With this method any material presented on screen can be captured and indexed (Figure 10).

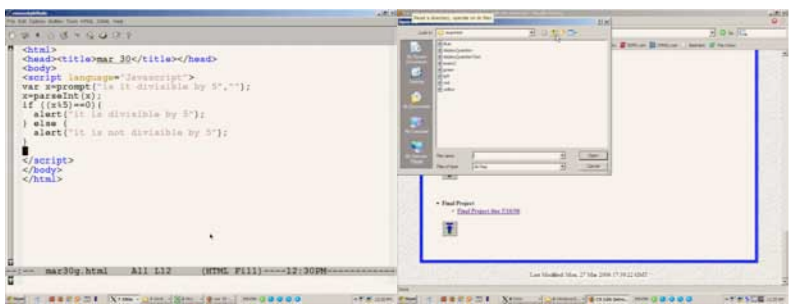


FIGURE 10 COMPUTER CAPTURE RESULTS.

Again, the progression of the material presented is important and can contain important information. An example of written content progression captured is presented in Figure 11.

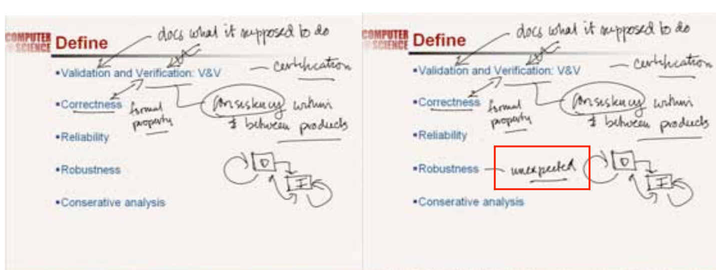


FIGURE 11 PROGRESSION OF COMPUTER CAPTURE CONTENT.

The systems most similar to the PAOL computer capture algorithm are Mediasite [16] and TTT [17]. Mediasite uses an external capture device similar to that used by PAOL to capture screen content and gets similar results. The PAOL computer capture algorithm does a better job of saving significant material whereas Mediasite can detect and display videos/animations, which PAOL cannot because it only saves stable material. On occasion PAOL does capture these videos in the lecturer video it creates. TTT gets results nearly identical to PAOL's but requires the lecturer to present the lecture by connecting to a special presentation server. PAOL provides more features overall than Mediasite and is again more transparent than TTT.

EXPERIMENTAL RESULTS

During development of the PAOL capture system, data were captured from 96 individual classes involving 9 lecturers. During the 2006-2007 academic year, data were captured using only the computer capture system. In addition, the computer capture system was demonstrated at a number of conferences (including [12]) and other venues. The computer capture system was also tested with the AutoPresenter content delivery system in a software engineering class in Fall 2007 [7]. Once the cameras were installed, a number of individual classes were captured using the computer, whiteboard, and lecture video capture system. The set of all sampled images from these classes serve as a significant base of experimental data that can be used to refine and evaluate the algorithms. The work reported in [12] and [13] describe our analysis of the quality of the captured and stored images in terms of duplicated and/or missed images, noise, etc.

In spring 2008, we attempted to capture a full course and compile it into FlashMANIC presentations with the long-term goal of evaluating the integrated PAOL capture system/MANIC delivery system for its impact on teaching and learning. We carried out usability studies to determine the most effective presentation formats and interfaces. These usability studies helped determine how well the PAOL system captures the essence and important events in the classroom and how easy it is to navigate to information students are seeking. We also evaluated the Flash MANIC GUI and tools. The usability studies involved focus groups of students enrolled in the class and students who took the class previously.

Both of these experiments (AutoPresenter in Fall 2007 and FlashMANIC in Spring 2008) helped evaluate the impact of the content capture and delivery systems on teaching and learning. In fall 2008, we will be carrying out additional experiments. We plan to measure teaching and learning outcomes by looking at student performance, gathering student evaluations, and interviewing and surveying students and instructors.

FUTURE WORK

In the current form, PAOL, FlashMANIC and AutoPresenter enable the capture and presentation of courses that include whiteboard and computer presentation of materials and do so transparently. For full adoption, the type of material captured by PAOL will need to be broadened. As a next step we plan to develop algorithms that enable PAOL to capture material presented on chalkboards. Chalkboard capture will make it possible for PAOL to be used in developing countries where chalkboards are still more prevalent then whiteboards.

In many MANIC instances, the indexes are provided as a text table of contents (TOC). The availability of text indexes supports the MANIC search tools. When manually produced, it is relatively easy to add text indexes, and RIPPLES has experimented with OCR extraction of slide titles. For PAOL-produced content, we will have to use a combination of OCR, computer vision and speech-understanding techniques to extract meaningful and accurate text indexes. This work has been underway before and in parallel with the PAOL development. Since the MANIC notation tools provide a TOC index, we can also allow learners to provide their own indexes.

ACKNOWLEDGMENTS

This work was partially funded by the National Science Foundation under grants CNS-0749756 and CNS-9979833 and by the University of Massachusetts under an ITC Strategic Initiative Grant.

REFERENCES

[1] Schapira, A, de Vries, K, Pedregal-Martin, C, "MANIC: An Open-Source System to Create and Deliver Courses over the Internet", *SAINT-W '01: Proceedings of the 2001 Symposium on Applications and the Internet-Workshops*, 2001, 21.

[2] Adrion, W, R, Kurose, J, Riseman, E, Zhu, Z, “A Multimedia Virtual Classroom,” *Proceedings of the International Conference on Information Communications Technologies in Education* (ICICTE-04), July 2004

[3] Adrion W. R., Cooper, W., Dickson, P., Ros, I., Wallace, B., and Watts, K., “Evolution of a cross-platform, multimedia courseware presentation system,” *ACM Transactions on Internet Technology*, Under review. (2006)

[4] Anderson, R., Anderson, R., Chung, O., Davis, K.M., Davis, P., Prince, C., Razmov, V., Simon, B., “Classroom Presenter – A Classroom Interaction System For Active and Collaborative Learning, *Proceedings of the First Workshop on the Impact of Pen-based Technology on Education.* (2006)

[5] Beavers, J., Chou, T., Hinrichs, R., Moffatt, C., Pahud, M., Powers, L., and Van Eaton, J., “The Learning Experience Project: Enabling Collaborative Learning with ConferenceXP,” MicrosoftResearch Technical Report MSRTR-2004-42.

[6] Wilkerson, M, Griswold, W, G, Simon, B, "Ubiquitous presenter: increasing student access and control in a digital lecturing environment", *SIGCSE '05: Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education*, Vol, 36, 2005, 116-120.

[7] Myers, S, "Real-Time, Platform Independent Recording and Sharing of Computer-based Classroom Presentations", downloaded at *http://ripples.cs.umass.edu/~adrion/Reports/Scott_Myers_Auto_Presenter.pdf*, August 2007.

[8] Bianchi, M, H, "AutoAuditorium: a Fully Automatic, Multi-Camera System to Televise Auditorium Presentations", *http://www.autoauditorium.com/nist/autoaud.html*, 1998.

[9] Brotherton, J, A, Abowd, G, D, "Lessons learned from eClass: Assessing automated capture and access in the classroom", *ACM Transactions Computer-Human Interaction*, Vol, 11, 2004, 121-155.

[10] Machnicki, E, Rowe, L, A, "Virtual Director: Automating a Webcast", Multimedia Computing and Networking 2002, Proc. IS&T/SPIE Symposium on Electronic Imaging: Science & Technology, 2002, 208-225.

[11] Hurst, W, Maass, G, Muller, R, Ottmann, T, "The 'Authoring on the Fly' system for automatic presentation recording", *CHI '01: CHI '01 extended abstracts on Human factors in computing systems*, 2001, 5-6.

[12] Dickson, P, Adrion, W, R, Hanson, A, "Automatic Capture of Significant Points in a Computer Based Presentation", *ISM '06: Proceedings of the Eighth IEEE International Symposium on Multimedia*, Vol, 8, 2006, 921-926.

[13] Dickson, P, E, Adrion, W, R, Hanson, A, R, "Automatic Creation of Indexed Presentations from Classroom Lectures", accepted, ITiCSE 2008.

[14] He, L, Zhang, Z, "Real-Time Whiteboard Capture and Processing Using a Video Camera for Remote Collaboration", *IEEE Transactions on Multimedia*, Vol, 9, Jan 2007, 198-206.

[15] Ziewer, P, Seidl, H, "Transparent teleteaching", *ASCILITE*, 2002, 749-758.

[16] SonicFoundry, "Medisite.com.", *http://www.Mediasite.com/*, Apr 2006.

[17] Ziewer, P, "Navigational Indices and Content Interlinkage on the Fly", *ISM '06: Proceedings of the Eighth IEEE International Symposium on Multimedia*, 2006, 915-920.

[18] Wallace, B, Adrion, W, R, Burleson, W, Cooper, W, Cori, J, et al, “Using Multimedia to Support Research, Education and Outreach in an NSF Engineering Research Center,” *Proceedings of the Frontiers in Education Conference*, 2005.

[19] Thampuran, S., Burleson, W, Watts, K., “Multimedia Distance Learning Without the Wait”, *Proceedings of the Frontiers in Education Conference*, 2001.

[20] Kumela, D, Watts, K , Adrion, W, R, “Supporting Constructivist Learning in a Multimedia Presentation System,” *Proceedings of the Frontiers in Education Conference*, 2004.

[21] Ray, E, Watts, K, Adrion, W, R, “Extending Record and Playback Technologies to Support Cooperative Learning,” *Proceedings of the Frontiers in Education Conference*, 2004.