PicNTell: A camcorder metaphor for screen recording

Scott Carter
FX Palo Alto Laboratory
3400 Hillview Ave, Bldg 4
Palo Alto, CA 94304
carter@fxpal.com

Laurent Denoue
FX Palo Alto Laboratory
3400 Hillview Ave, Bldg 4
Palo Alto, CA 94304
denoue@fxpal.com

ABSTRACT

PicNTell is a new technique for generating compelling screencasts where users can quickly record desktop activities and generate videos that are embeddable on popular video sharing distributions such as YouTube®. While standard video editing and screen capture tools are useful for some editing tasks, they have two main drawbacks: (1) they require users to import and organize media in a separate interface, and (2) they do not support natural (or camcorder-like) screen recording, and instead usually require the user to define a specific region or window to record. In this paper we review current screen recording use, and present the PicNTell system, pilot studies, and a new six degree-of-freedom tracker we are developing in response to our findings.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Input devices and strategies

General Terms

Design, Experimentation, Human Factors

Keywords

screencasting, natural input, multimedia recording and playback

1. INTRODUCTION

Instructional videos are increasingly popular (see http://www.5min.com/, http://www.mindbites.com/, and others). People creating videos of real world artifacts, such as those documenting how to change a tire or how to improve a golf swing, can take advantage of the natural ease of recording with a handheld video camera to move around an object with six degrees-of-freedom (6DOF) with minimal effort. However, videos of digital artifacts suffer from a tradeoff — users can either use screen recording software that provides high resolution content but sacrifices the expressiveness and ease-of-use of a video camera, or use a video camera but lose the resolution and highlighting available with screen recording. Our goal in this work is to improve screen recording by conserving the expressive power and ease-of-use of video cameras while also maintaining all of the advantages of digital content.

From our observations of common uses of screen recordings (described in the next section), it became apparent that there are two issues to address to achieve this goal: creating a full 6DOF environment for screen recording, and making use of the digital environment to support highlighting and interaction. In this paper, we describe the iterative development of a system that uses a 6DOF sensor to support the recording of static and dynamic documents. We also describe a new 6DOF sensor we are currently building that accomplishes these goals while being easy to deploy.

2. SCREEN RECORDING USE

Public video content sites such as YouTube® include a wide variety of uses of screen recordings (see Table 1). Unlike many screencasting tools, we are interested in supporting digital content generally, including both static and dynamic content. From investigating public video content, as well as video content used by more private corporate work groups, we have found that screen recording typically falls into the following categories: explanatory, documentation and review, ludic, and advertising. Explanatory, in which the author is showing the audience how to complete some task, is the most common and tends to focus on a few simple features of an application. Documentation and review videos provide much more detailed accounts of an application or other media, usually for archival purposes. Ludic videos tend to be personal videos in which the author pans and zooms over family photos. Finally, advertising videos typically use 6DOF to create engaging announcements of new web sites or applications.

Table 1 describes some important functions to support for recording different media types. Highlighting, or specifying a region-of-interest, is common to all media. For many documents and applications, it can be important to have the video lock onto a well-specified region (such as a block of text in a document or a textbox in an application). Also, it is much easier to record over lengthy documents if the recording application supports both a collection view as well as individual page or slide view; allowing the audience to interact with recorded content is especially useful for applications [1]; callouts are helpful to show a variety of focused regions while maintaining the context of the larger document; and finally past work has shown the utility of navigating through a high resolution video with a lower resolution viewport [2].

As a first step, in this paper we focus on supporting 6DOF region-of-interest (highlight) recording.

MM '08, October 26-31, 2008, Vancouver, British Columbia, Canada.
Copyright 2008 ACM 978-1-605558-303-7/08/10…$5.00.

Table 1. Functions important for recording different media

Medium	Functions
Static pictures	Highlight, seamlessly move between media
Static documents and forms	Highlight, lock-to-region, seamlessly move between pages, callouts
Video	Highlight, create sub-videos [2]
PowerPoint®	Highlight, lock-to-region, interaction, seamlessly move between slides
Web sites and applications	Highlight, lock-to-region, interaction

3. RELATED WORK

Some commercial off-the-shelf products include features that augment the expressiveness of screen recordings. For example, Camtasia Studio® includes a Smart Focus feature that automatically determines regions-of-interest in screen recordings and applies the appropriate zoom level. However, this tool is applied post hoc, uses only three degrees-of-freedom, and is designed more for applications than static documents.

Researchers have also implemented several approaches that find regions-of-interest automatically [3-5]. These techniques do not allow the user enough control to be able to craft their own story. Other researchers have used the notion of viewports onto larger documents [6, 7]. However, these approaches do not allow the user to seamlessly control the properties of the viewport itself.

Other work has augmented common devices to support 6DOF [8, 9]. None of these tools have been applied to a camcorder task, nor do they include an isomorphic implementation.

4. EXPERIMENTS WITH 6DOF SCREEN RECORDING

Because of the novelty of using a 6DOF system for screen recording, there exists no appropriate baseline condition for evaluating new designs. Furthermore, because of the exploratory nature of this work, comparisons to conventional recording techniques would be premature. Therefore, our focus is on experimenting with multiple designs rather than conducting standard usability tests. As Greenburg and Buxton point out [10], “early designs illustrate the essence of an idea” and help “make vague ideas concrete, reflect on possible problems and uses, discover alternate new ideas and refine current ones.” Importantly, though, these design sketches are not necessarily “suggestive of the finished product,” but instead serve as a means of exploration. As Schrage writes [11], these early sketches “externalize thought and spark conversation.”

As Table 2 shows, we experimented with a wide variety of different configurations to recreate the video camera experience for screen recording.

Table 2. Positives and negatives of different input techniques

Input	Pros	Cons
Mouse	Easy to deploy	Unnatural motion, tethered
Wii®	Easy to deploy, natural motion	Not true 6DOF (no yaw)
Polhemus®	Natural motion	Expensive, cumbersome
Sensing UI - upright	Natural motion	Arm fatigue, calibration
Sensing UI - table	Natural motion	Some obstruction, calibration
Webcam	Natural motion	Unknown

4.1 System design

PicNTell is composed of a 6DOF sensing system, a recorder, a database, and a player. The sensing system can be any application that provides positional and rotational information in real time. The recorder receives these positions over a socket connection and uses them to compute the geometric warp of a region-of-interest (ROI) quadrilateral, displayed over the window being captured (see Figure 2). The effect is that the user sees the quadrilateral as if it were the beam of a virtual flashlight (except it is square, not round), showing her in real-time where she is focusing. In order to paint the ROI over the captured window, the recorder window is made topmost and transparent everywhere, except for the colors used to draw the ROI quadrilateral.

The recorder also periodically (every 100ms) grabs screenshots of the window being captured and records audio continuously via an attached microphone. This data is time-stamped and sent to a database along with the ROI data.

The player application reads the ROI, image, audio, and timestamp data to generate a video. The application applies a geometric warp to each image based on the ROI to fit a predefined resolution (currently 320x240, standard resolution for YouTube® video). The player can be run synchronously with the recorder application (with a slight delay) or asynchronously. Also, the player can generate a stand-alone AVI file.

4.2 Initial input devices

We initially experimented with a camcorder control using a mouse, a Wii, and a mounted sensing system. We chose these systems because they represented a span of installation difficulties and user adoption rates. We rejected several other systems that did not adhere to the camcorder metaphor (e.g., SpaceNavigator®).

The mouse solution maps the x and y position of the mouse to the x and y position of the viewport and the mouse scroll wheel to distance from the screen (applications typically use this sensor to scroll up-and-down). The system uses keyboard arrow keys for pan and tilt, and the mouse scroll wheel (in the direction orthogonal to zoom) for roll (applications typically use this sensor to scroll left-to-right).

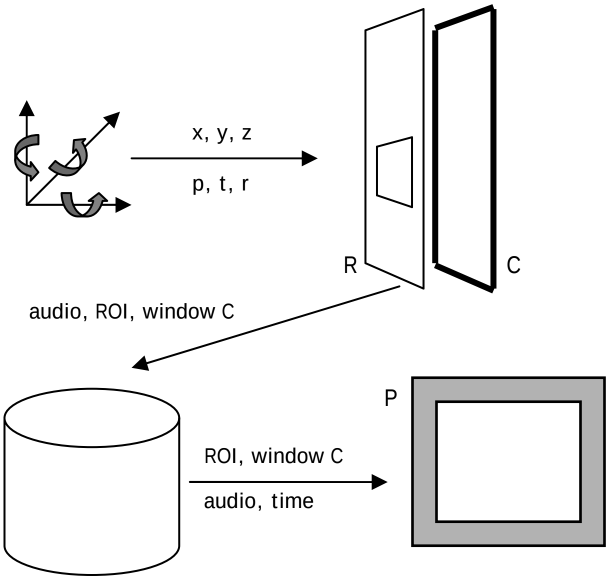

Figure 1. PicNTell. A 6DOF system sends positional and rotational information to the recorder application window (R). The recorder draws a region-of-interest (ROI) quadrilateral on top of the window of the application being captured (C), and periodically sends the ROI, a screenshot of the captured window, and audio to a database. The player (P) reads that data, warps the screen capture to match the ROI, and integrates the audio to create a video.

The Wii solution relies on the Wii's internal infrared sensor. In place of the standard Wii sensor bar, the system used two infrared flashlights to derive the x and y location of the Wii. The system uses the measured distance between the two infrared flashlights to infer zoom. Since the flashlights are a fixed distance apart, the distance increases as the Wii is increasingly closer to the screen. The Wii controller includes an accelerometer which relies on gravitational pull to measure static forces, and this works for pitch and roll but it cannot record yaw. In its place the system uses the Wii's arrow keys for both yaw and tilt. The system relies on the Wii's accelerometer to map directly the roll of the Wii to the roll of the viewport.

The Sensing UI [12] tracks a board outfitted with infrared LEDS using a camera mounted to a fixed position. While we do not consider this a final solution — it would be unrealistic for casual users to install such a system — we used this system to experiment with 6DOF screen recording because it supports natural motion while being relatively unobtrusive. It is also significantly less expensive than other similar systems [13].

4.2.1 Pilot studies

For user testing we augmented the recorder and viewer. The recorder showed two 6DOF viewports: one under the users' control and one generated automatically that users were to match, while the viewer replayed their clip projected onto a standard resolution (320x240).

We initially ran a pilot study with two users on four tasks. Because our tools are experimental, we limited tasks in the pilot to degrees-of-freedom that proved to have low recognition error rates across all devices. The tasks we chose were (1) move the viewport from one location to another in a static image; (2) pan and tilt the camera to include two different objects in a static image; (3) move the viewport to an object as that object jumps between four different locations, and (4) move the viewport to follow an object as it moves to four different locations and to match the zoom of that object as it changes. After users completed each task for each devices (a total of 12 total tasks), we asked them to describe their experience.

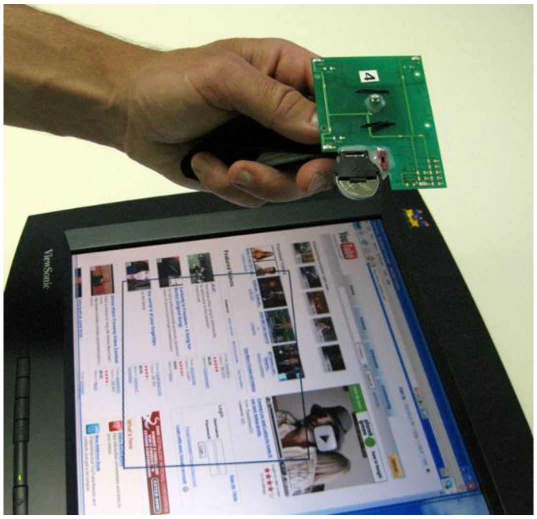

Figure 2. The recording application shown using the Sensing UI as the 6DOF input. The user holds a board outfitted with infrared LEDs (above) that is sensed by a mounted camera (not shown). The application provides feedback by tracing a rectangle that is deformed according to the positional and rotational information sensed by the input.

We found that sensor error rates on the Wii controller were still too high. Overall users took longer to complete these tasks than with the mouse because the viewport would occasionally drift when sensor readings did not match the user's intent. On the other hand, users found the Sensing UI intuitive, but users grew tired of having to hold up their arm to complete the tasks. However, users did express that they would "definitely use the tools" if the errors were removed, which encouraged us to continue.

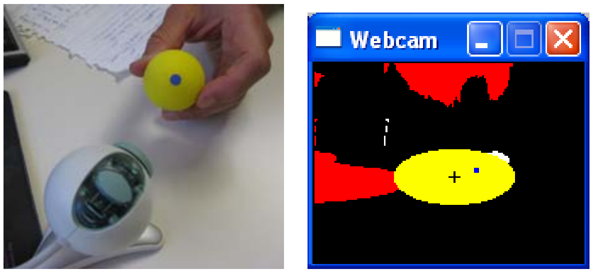

Figure 3. The webcam-based system. The user manipulates a ball in front of a webcam, and the system detects its center as well as the location of the blue dot. The dot's offset from the center gives an estimate of pan and tilt.

We followed this pilot with another experiment involving two users that tested the UI on a flat surface to mitigate arm fatigue. While users did not experience any fatigue, they did report that in the hand with which they held the sensor occasionally obscured their view of the screen, making it difficult to control exactly the location of the viewport (Figure 2). This finding led us to begin developing an entirely new 6DOF system, described next.

4.3 A webcam-based 6DOF input

One of the goals of this work is to maintain the ease-of-use of video cameras for 6DOF screen recording. As one of the core features of video cameras is that they are effortlessly mobile, it is important to develop a solution that does not require a mounted camera. For this reason, we are building a 6DOF system that tracks a small ball using a webcam (Figure 3). This system tracks the location and size of the ball to estimate x, y, and z positions, and tracks a pattern on the ball to estimate orientation. This way, the user can experience the same level of expressiveness as a video camera with the mobility of a mouse, while avoiding the hand and arm tiredness as with our earlier prototype.

Currently, our prototype tracks only 5 degrees-of-freedom. In future work, we plan to add support for rotation. For example, we could paint a green line horizontally on the ball in order to get a rough estimate of the sixth degree-of-freedom.

Also, compared to a mouse or Wii controller, this vision-based solution is less precise. In order to remove noise, the system averages the last 100 positions, but more sophisticated methods such as Kalman filtering may be more appropriate.

The original motion described by the user could also be post-processed: the rough positions and orientations captured by our imperfect tracker could be smoothed into splines before rendering. Special effects could be added, such as virtual camera shakes or sped-up zoom effects.

Furthermore, the natural first step after viewing a screencast is to try it out, usually with data relevant to the viewer. We feel that viewing and experimenting need not be separate tasks. In the next version of our tool, we plan to allow screencast creators to designate widgets as interactive, allowing viewers to adjust their properties (using the search example, the creator might allow viewers to input their own search terms). This function will require a new video player that supports the ability to swap into the video stream images captured in real-time from an off-screen instance of the application.

5. CONCLUSIONS

We described a new technique, PicNTell, that couples a 6DOF sensor with screen recording and replay applications to help users rapidly generate compelling screencasts. As described above, our pilot studies led us to focus on developing a 6DOF system that mitigates arm fatigue and screen obstruction. In future work, we plan to run full usability studies of the completed system.

6. ACKNOWLEDGEMENTS

Don Kimber configured the Sensing UI and provided general advice regarding 6DOF sensors.

7. REFERENCES

[1] Little, G., Lau, T., Cypher, A., Lin, J., Haber, E., and Kandogan, E. Koala: capture, share, automate, personalize business processes on the web. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*. 2007. 943-946.

[2] Pea, R. and Weiss, J, Chapter 55: Video-as-Data and Digital Video Manipulation Techniques for Transforming Learning Sciences Research, Education, and Other Cultural Practices. In *The International Handbook of Virtual Learning Environments*. 2006. 1321-1393.

[3] Fan, X., Xie, X., Zhou, H.-Q., and Ma, W.-Y. Looking into video frames on small displays. In *MULTIMEDIA '03: Proceedings of the ACM international conference on Multimedia*. 2003. 247-250.

[4] Liu, H., Xie, X., Ma, W.-Y., and Zhang, H.-J. Automatic browsing of large pictures on mobile devices. In *MULTIMEDIA '03: Proceedings of the ACM international conference on Multimedia*. 2003. 148-155.

[5] Rav-Acha, A., Pritch, Y. and Peleg, S. Making a Long Video Short: Dynamic Video Synopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2006. 435-441.

[6] Baudisch, P., Good, N., Bellotti, V., and Schraedley, P. Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming. In *CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems*. 2002. 259-266.

[7] Yee, K.-P. Peephole displays: pen interaction on spatially aware handheld computers. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems.* 2003. 1-8.

[8] Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., and Conway, M. The VideoMouse: a camera-based multi-degree-of-freedom input device. In *UIST '99: Proceedings of the ACM symposium on User interface software and technology*. 1999. 103-112.

[9] Zhai, S. and Milgram, P. Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices. In *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*. 1998. 320-327.

[10] Greenberg, S. and Buxton, B. Usability evaluation considered harmful (some of the time). In *CHI '08: Proceeding of the SIGCHI conference on Human factors in computing systems*. 2008. 111-120.

[11] Schrage, M., Serious Play. 2000: Harvard Business School Press.

[12] Seko, Y., Saguchi, Y., Hotta, H., Iyoda, T., and Koshimizu, Y. Position and Orientation Measurement of Small LED Cards by Firefly Catching Camera. In *DIA ’07: Dynamic Image Processing for Real Application.* 2007. 133-136.

[13] Polhemus Fastrak. Available from: http://www.polhemus.com/?page=Motion_Fastrak.