

Smarter Presentations: Exploiting Homography in Camera-Projector Systems

Rahul Sukthankar^{1,2}, Robert G. Stockton¹, Matthew D. Mullin¹

¹Just Research
4616 Henry Street
Pittsburgh, PA 15213

²The Robotics Institute
Carnegie Mellon
Pittsburgh, PA 15213

{rahuls, rgs, mdm}@justresearch.com *

Abstract

Standard presentation systems consisting of a laptop connected to a projector suffer from two problems: (1) the projected image appears distorted (keystoned) unless the projector is precisely aligned to the projection screen; (2) the speaker is forced to interact with the computer rather than the audience. This paper shows how the addition of an uncalibrated camera, aimed at the screen, solves both problems. Although the locations, orientations and optical parameters of the camera and projector are unknown, the projector-camera system calibrates itself by exploiting the homography between the projected slide and the camera image. Significant improvements are possible over passively calibrating systems since the projector actively manipulates the environment by placing feature points into the scene. For instance, using a low-resolution (160x120) camera, we can achieve an accuracy of ± 3 pixels in a 1024x768 presentation slide. The camera-projector system infers models for the projector-to-camera and projector-to-screen mappings in order to provide two major benefits. First, images sent to the projector are pre-warped in such a way that the distortions induced by the arbitrary projector-screen geometry are precisely negated. This enables projectors to be mounted anywhere in the environment – for instance, at the side of the room, where the speaker is less likely to cast shadows on the screen, and where the projector does not occlude the audience’s view. Second, the system detects the position of the user’s laser pointer dot in the camera image at 20Hz, allowing the laser pointer to emulate the pointing actions of a mouse. This enables the user to activate virtual buttons in the presentation (such as “next slide”) and draw on the projected image. The camera-assisted presentation system requires no special hardware (aside from the cheap camera)

and runs on a standard laptop as a Java application. It is now used by the authors for all of their conference presentations.

1. Introduction

Presentation systems consisting of a computer projector connected to a laptop have largely replaced overhead projectors in most business environments. However, the current systems are deficient in two main areas.

First, unless the projector’s optical axis is carefully aligned to be perpendicular to the screen, the resulting image appears distorted due to perspective effects. This greatly constrains the placement of the projector in the conference room, often requiring the projector to be mounted in a manner that obstructs the audience’s view of the speaker or the presentation. These problems are particularly severe for the increasingly popular microportable projectors which are simply placed on a table at the start of the presentation. Some projectors now offer optical or digital *keystone correction* that allows users to counter the symmetric trapezoidal distortions caused by small amounts of vertical misalignment (typically within $\pm 12^\circ$). Unfortunately, these systems require the user to manually adjust the keystone parameters and the projectors cannot correct any distortions due to lateral misalignment nor projector roll.

Second, traditional methods of controlling computer-based presentations (such as PowerPoint talks) require the user to send commands to the computer using either the keyboard or the mouse. This can be awkward because it diverts the attention of the presenter and the audience from the presentation. A better interface would enable the presenter to perform actions directly on the presentation area, effectively treating the computer as a member of the audience. Existing systems for accepting user input from the presentation surface include expensive electronic white-boards and devices such as remote mice. Electronic white-boards are not portable and either require laborious manual calibration

*Rahul Sukthankar (rahuls@cs.cmu.edu) is now affiliated with Compaq Cambridge Research Lab and Carnegie Mellon; Robert Stockton and Matthew Mullin ({rstock, mmullin}@whizbang.com) are now with WhizBang! Labs.

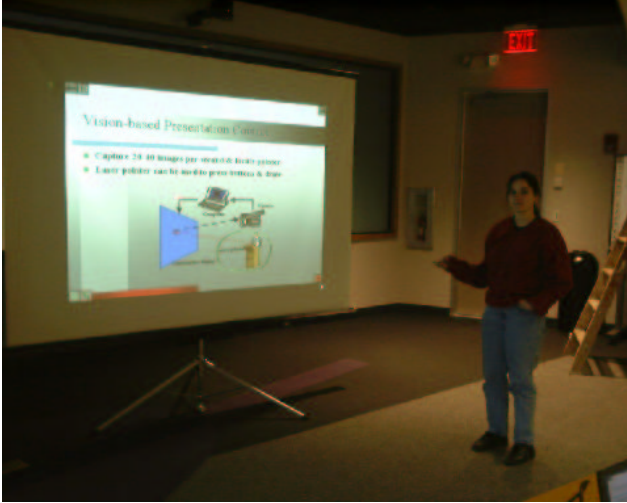


Figure 1: This photograph shows the camera-assisted presentation system in use. The portable projector (not visible) has been placed at the side of the room, yet the image on the screen is undistorted. The user is shown controlling the presentation using a standard laser pointer.

and/or force the use of specially coded markers while remote mice lack the transparency and immediacy of pointing actions.

This paper presents a self-calibrating presentation system that solves both of these problems simultaneously (see Figure 1). A low-resolution digital camera observes the presentation screen and the system automatically pre-warps the image to be displayed so that the distortions induced by arbitrary projector placement precisely cancel the warping, resulting in a perfectly aligned and rectilinear image. This allows the projector to be placed anywhere in the room, for instance to the side where it is less likely to interfere with the speaker or the audience. The same camera is also used to track laser pointer dots (or high-contrast physical pointers) on the projection screen with sub-pixel accuracy, enabling the user to directly control the presentation by activating virtual buttons (or “drawing” on the slide). The camera may be placed anywhere in the room such that its field of view includes the projection screen.

2. Projector-Camera Homography

This section details key ideas regarding projector-camera calibration that are critical for both automatic keystone correction (Section 3) and laser pointer-based presentation control (Section 4). Our assumptions are that: the positions, orientations and optical parameters of both camera and projector are unknown; camera and projector optics can be modeled by perspective transforms; the projection screen

is flat.¹

Consider a point (x, y) in the projector slide. This point is projected to some unknown point on the projection screen (by a perspective transform whose parameters depends on the unknown position and orientation of the projector relative to the screen, and the unknown focal length of the projector optics). The point on the screen is then observed by a camera at pixel location (X, Y) (undergoing a second perspective transform whose parameters depend on the unknown position and orientation of the camera relative to the screen, and the unknown focal length of the camera optics). Our goal in this section is to recover the mapping between (x, y) and (X, Y) without any additional information about the unknowns; in other words, given that we observe a feature at (X, Y) in the camera image (e.g., a bright laser pointer dot), we would like to know the point (x, y) in the projector slide that corresponds to this feature (e.g., to determine whether the laser pointer dot lies within the boundaries of a virtual button drawn on the slide).

At first glance, it may appear that this mapping is impossible to determine in the presence of so many unknowns. Fortunately, we can exploit the fact that all of the observed points in the scene lie on some unknown plane (the flat projection screen), and this establishes a homography between the camera and projector frames of reference. Thus, we can show that the compounded transforms mapping (x, y) in the projector frame, to some unknown point on the projection screen, and then to pixel (X, Y) in the camera frame, can be expressed by a *single* projective transform,

$$(x, y) = \left(\frac{p_1 X + p_2 Y + p_3}{p_7 X + p_8 Y + p_9}, \frac{p_4 X + p_5 Y + p_6}{p_7 X + p_8 Y + p_9} \right),$$

with eight degrees of freedom, $\vec{p} = (p_1 \dots p_9)^T$ constrained by $|\vec{p}| = 1$. The same transform is more concisely expressed in homogeneous coordinates as:

$$\begin{pmatrix} xw \\ yw \\ w \end{pmatrix} = \begin{pmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \\ p_7 & p_8 & p_9 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

\vec{p} can be determined from as few as four pixel correspondences²; when more than four correspondences are available, the system finds the best estimate in a least-squares sense. Given n feature point matches, $\{(x_i, y_i), (X_i, Y_i)\}$,

¹Our method has been extended to non-planar screens, but this is not detailed here due to space constraints.

²Four correspondences are sufficient provided that no three points are collinear.

let A be the following $2n \times 9$ matrix:

$$\begin{pmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -X_1x_1 & -Y_1x_1 & -x_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -X_1y_1 & -Y_1y_1 & -y_1 \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -X_2x_2 & -Y_2x_2 & -x_2 \\ 0 & 0 & 0 & X_2 & Y_2 & 1 & -X_2y_2 & -Y_2y_2 & -y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_n & Y_n & 1 & 0 & 0 & 0 & -X_nx_n & -Y_nx_n & -x_n \\ 0 & 0 & 0 & X_n & Y_n & 1 & -X_ny_n & -Y_ny_n & -y_n \end{pmatrix}$$

The goal is to find the unit vector \vec{p} that minimizes $|A\vec{p}|$, and this is given by the eigenvector corresponding to the smallest eigenvalue of $A^T A$.

The projector-camera system offers a significant advantage in determining these pixel correspondences: the projector is used to actively manipulate the scene. The presentation system displays a series of calibration slides to recover the projector-camera mapping parameters. First, an initial estimate is obtained by projecting a bright rectangle against a contrasting background. The locations of the corners of the projected rectangle (in the camera image) are automatically determined and used as features to recover the projector-camera homography according to the procedure outlined above. This homography is applied (as a pre-warp) to a calibration slides consisting of bright circles on a dark background. The system estimates the center of each circle by calculating the centroid of the observed bright region in the camera image, and uses this sub-pixel estimate to refine its estimate of the projector-camera homography. Robust feature extraction is made simpler since the system knows where and when to look for a feature in the camera image. The system can also determine when the mapping is no longer accurate and trigger automatically recalibration as necessary.

3. Automatic Keystone Correction

Once the mapping between projector and camera has been established, one could propose the following method for automatic keystone correction: (1) determine the rectangular region in the camera image where the contents of the slide should appear (naturally, this is also constrained by the maximal projection area of the projector); (2) use the equation from Section 2 to back-project the corners of this rectangle into projector coordinates; (3) determine the projective transform that warps the slide to this desired quadrilateral. The implementation of such a system is straightforward, but the results are undesirable! This is because the naive approach to keystone correction creates a projected slide that appears rectangular *only from the viewpoint of a misaligned camera*. Human audiences expect the projected slide to be aligned with the *presentation screen* (or the walls and ceilings of the room) and this only happens with the

naive method if the camera is perfectly aligned with respect to the screen. Replacing the tedious process of projector alignment with an equally tedious process of camera alignment would be completely unsatisfactory; therefore, we must tackle the more challenging problem of aligning a presentation with the projection screen.

Figure 2 summarizes relationships between the frames of reference that are relevant to the problem of automatic keystone correction. The transformations between these frames can all be modeled by homographies. Our objectives are: (1) determine a model for the distortion between the projector and the projection screen; (2) determine the pre-warp that maps the slide to a suitable target rectangle on the projection screen. The process is slightly complicated by the fact that the slide and projection screen do not appear as rectangles in the camera image since the camera is misaligned with respect to the projection screen.

3.1. Modeling projector-screen distortions

To achieve the first objective, we observe that the (unobservable) mapping between projector and projection screen, denoted P , can be expressed as a composition of two observable mappings, $P = C^{-1}T$, where T is the projector-to-camera mapping (recovered in Section 2) and C is the mapping between the projection screen and the camera image. To obtain C , we must be able to reliably locate the presentation screen in arbitrary camera images (a non-trivial task).

Our method for locating the projection screen relies on the assumption that screens appear in the camera image as uniformly-light objects with edges that are visible against the background of the room. Once again, the projector's ability to manipulate the scene is exploited. By projecting a white slide, the projector can be made to illuminate the (typically darkened) presentation environment with a bright light source. We have found that this significantly improves the detection process. The camera image is then roughly segmented using a greedy region-growing procedure that groups adjacent pixels with similar intensities into connected components. The components in the brightly-illuminated region are merged, and line-fitting [1] is performed on the edges of the quadrilateral. The intersections of these lines correspond to the corners of the projection screen, and are determined with sub-pixel accuracy (necessary because the camera image is only 160x120). These four point correspondences fully specify the eight parameters required for the mapping C .

3.2. Generating and applying the pre-warp

The second objective in keystone correction is to determine a pre-warp that results in a rectangular image on the projection screen. We can indirectly observe the projected image

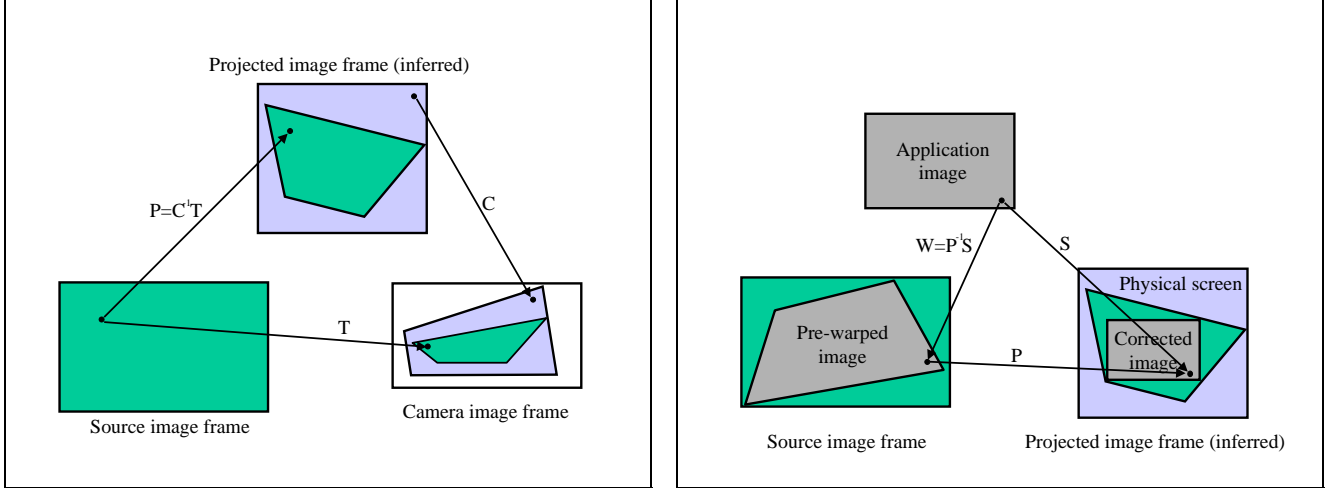


Figure 2: **Left:** the relationships between the three frames of reference corresponding to the laptop display (source image frame), camera (camera image frame) and projection screen (projected image frame). Note that the last can only be indirectly observed by our system, through the camera. T is obtained using the calibration method; C is obtained by locating the screen within the camera image. Finally, the mapping responsible for the keystone distortion, is given by $P = C^{-1}T$. **Right:** the application image can be appropriately distorted (pre-warped), using the mapping W so that it appears rectilinear after projection through a misaligned projector (modeled by the mapping P).

frame (to a scale factor) by applying the mapping C^{-1} to the camera image frame. In this frame of reference, both the projection screen and the desired corrected slide should appear as rectangles, and the corrected slide should be as large as possible, within the constraints of the projector’s display area (which appears as a quadrilateral in this frame). Our system employs a heuristic optimization to compute the dimensions and placement of the desired corrected slide (expressed as the mapping S in Figure 2); note that this is just a scaled rigid body transform since S maps a rectangle to another rectangle of the same aspect ratio. We obtain the pre-warp, W by applying P^{-1} to the coordinates of the desired corrected slide in the projected image frame; this is equivalent to applying $W = P^{-1}S$ to the original slide.

The pre-warped image is generated as follows. Each pixel in the pre-warped image is back-projected to a point in the presentation slide using W^{-1} and the four pixels closest to this point are blended using bilinear interpolation. Any point that back-projects to a point outside the slide is set to black; therefore, the pre-warped image consists of a warped quadrilateral corresponding to the user’s slide, embedded in a region of black pixels. This pre-warped image, when projected through the misaligned projector results in an undistorted image. This may be confirmed by observing that the compounded transformations on the application slide are given by $PW = PP^{-1}S = S$ and S , as noted above is simply a scaled rigid body transform. Results are presented in Figure 3.

Pre-warping an image does induce a slight loss of im-

age quality. However, since modern microportable projectors typically support XGA (1024×768) resolution, this is rarely an issue for most presentations. Users observe that our system’s image quality is superior to that of the projector’s built-in vertical keystone correction. This is because our system employs bilinear interpolation while the built-in keystone correction only does nearest-pixel resampling of the original image. Image quality can also degrade in two other circumstances: (1) when the projector is placed at extreme angles to the screen, the difference in distance to the near and far edges of the screen may be greater than the projector’s field of depth (preventing the entire screen from being in sharp focus); (2) the projector may be placed such that most of the projection area is wasted (forcing the system to display its image on a very small fraction of the available pixels).

This concludes the discussion on automatic keystone correction; the next section shows how the homography described in Section 2 can be exploited to enable laser-pointer control of presentations.

4. Pointer-based Presentation Control

This section describes how the projector-camera system enables the user to control presentations using a pointing device (e.g., a laser pointer). Conceptually, the idea is straightforward: the pointer is tracked in the camera image and the mappings derived in the previous section are used to determine the pixel in the presentation slide corresponding to this point. The real-time laser-pointer tracking algorithms used

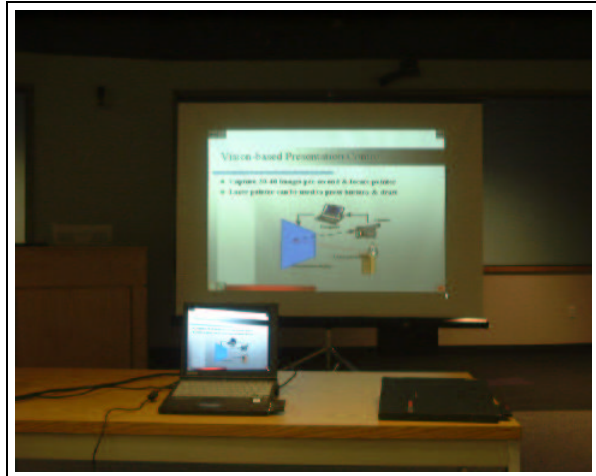
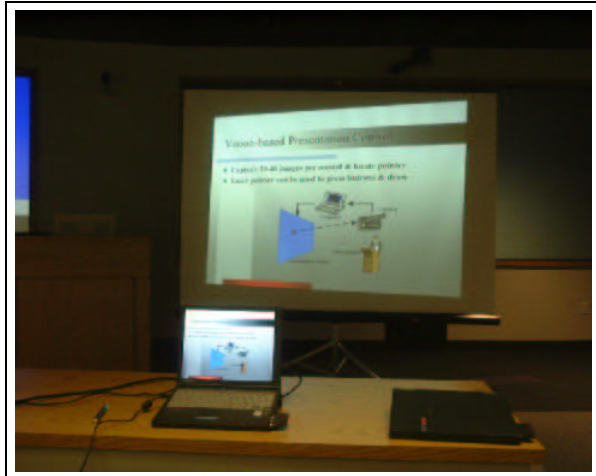
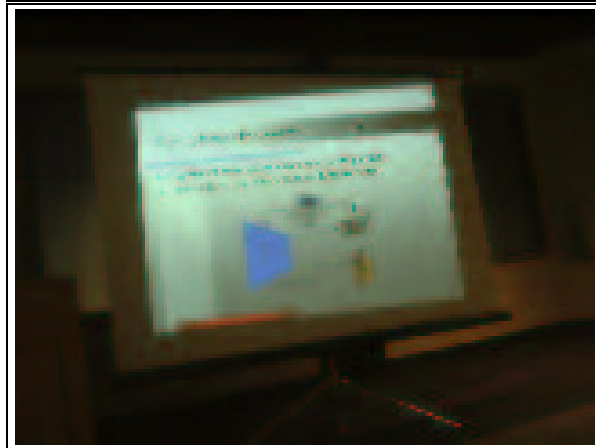
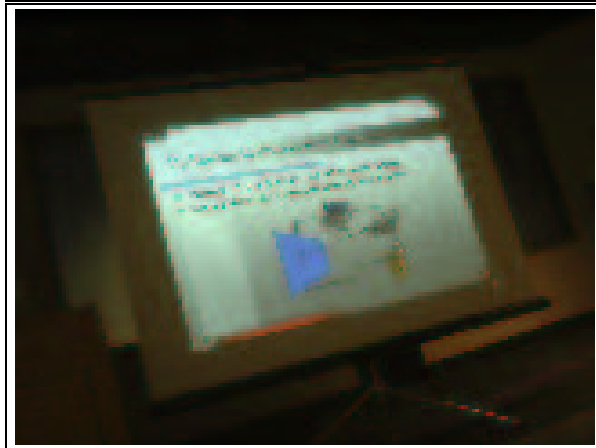
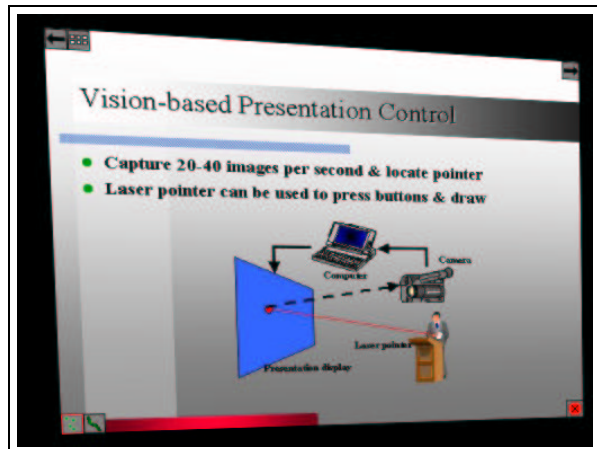
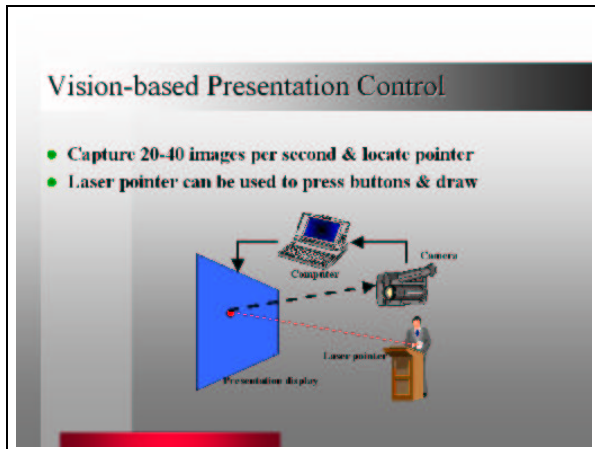


Figure 3: A rectangular source image (top left) appears distorted when projected from an off-center projector (middle & bottom left). Using our method, the source image is pre-warped (top right). The resulting projected image is rectilinear and perfectly aligned to the projection screen (middle & bottom right). The middle row of photographs was captured from the viewpoint of the laptop's 160×120 resolution camera (note the poor image quality), and the bottom row was captured from the viewpoint of an audience member; the pre-warped image is visible in the laptop screen in the bottom right photo.

in our system are not detailed in this paper due to space limitations; they are efficient implementations of standard image differencing techniques with adaptive thresholds. However, some implementation details are noteworthy. First, since the interface is very sensitive to errors in the camera-projector mapping, the system employs a multi-stage calibration procedure where the accuracy of the mapping can be confirmed and refined by displaying additional features on the projection screen at specific locations. Second, since the system employs a low-resolution mode (160x120) in order to achieve frame rate image acquisition on a standard parallel port camera, achieving accurate pointer tracking is challenging. The base accuracy of the pointer tracking algorithm is only ± 7 pixels in the 1024x768 projected slide. However, by exploiting the bleeding effect of the bright laser pointer dot and accumulating information over multiple images, the system is able to achieve an accuracy of ± 3 pixels at 20 Hz over the 1024x768 slide, which is sufficient for the applications described below.

4.1. Active Regions

Active regions allow users to deliver a presentation (e.g., changing slides) without physically interacting with the computer. By pressing virtual buttons in the presentation (using either a laser pointer, pointing stick or finger), the user can manipulate slides and activate menus in a natural manner. Active regions are specified as geometric regions on the presentation slide (source image frame coordinates) and may be constant across all slides (e.g., a “Quit” virtual button) or specific to the semantic information on the slide (e.g., a piece of text on the slide, such as a URL). Each active region is also associated with a particular action (e.g., pressing the “Next Slide” button advances the presentation).

The position of the pointing device, as detected by the camera, is converted from camera image frame coordinates to source image frame coordinates using the camera-projector homography described in Section 2 and the inverse of the automatic keystone correction pre-warp transformation. If the point falls within one of the active regions specified for the current slide, the action associated with the region is triggered (and the region is highlighted to provide visual feedback).

Figure 4 (top left) shows a PowerPoint slide displayed using the camera-assisted presentation tool. Several virtual buttons have been automatically added to the slide: the buttons in the top left and right corners change to the previous and next slide, respectively; the second button in the top left corner pops up a contact sheet with thumbnails of the slides; the buttons in the bottom left corner toggle the freehand drawing mode discussed below; finally, the button in the bottom right corner exits the presentation.

4.2. Freehand Drawing

The camera-projector system also enables users to highlight and annotate slides during the presentation, as shown in Figure 4 (top right). The speaker can use the presentation area as a virtual sketchpad, drawing and erasing lines using the laser pointer.

Implementing freehand drawing is simple in concept: a transparent overlay is created over the presentation slide, upon which successively detected positions of the pointing device (converted to the source image frame) are connected by thick line segments; the line is terminated if no pointer is sensed within the drawing area. Unfortunately, simply connecting raw laser pointer positions produces very unsatisfactory results since the laser pointer magnifies small jitters in the pointer position (whether due to speaker nervousness or tracking errors), creating an unpleasantly jagged line. Therefore, the freehand drawing interface smoothes laser pointer input using a mass-spring-damper model inspired by DynaDraw [3]. In this scheme, the tip of the line is modeled as a physical object with mass, connected by a spring to the last-observed position of the laser pointer, being dragged through a viscous fluid. As shown in Figure 4, the freehand drawing interface generates responsive yet smooth curves that are well-suited for highlighting material on the slide during the presentation.

5. Related Work

The authors are not aware of any related work in automatic keystone correction involving camera-projector systems. Researchers in the computer graphics community [6, 7] have used *calibrated* cameras to align displays consisting of multiple projectors; our work differs in several ways: (1) the physical and optical parameters for camera and projector in our system are unknown; (2) our system is able to exploit the camera-projector feedback loop to detect inaccuracies in the mapping and trigger self-calibration. Related work in innovative interfaces includes [2, 4, 5]. More about camera calibration can be found in [8].

6. Conclusion

The camera-assisted presentation interface described in this paper offers two main benefits. First, it enables the user to deliver presentations in a more natural manner: by interacting with the computer as if it were a member of the audience. Second, our system relaxes the usual constraints on a presentation environment: by allowing the projector to be mounted anywhere in the room, interference between the projector and the audience is minimized. Finally, our system requires no specialized hardware: a popular “eyeball” camera connected to a standard laptop over the parallel or USB port is sufficient, along with an ordinary laser pointer.

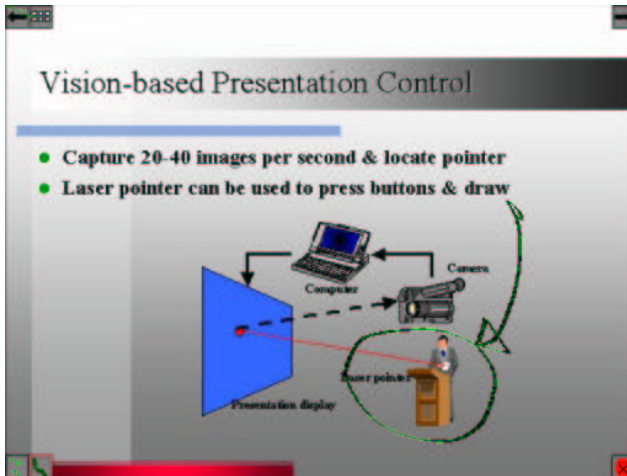


Figure 4: Screenshots from the camera-assisted presentation interface. **Left:** several active regions (buttons) are automatically added to the corners of the PowerPoint slide when displayed using the camera-assisted presentation tool. This image also illustrates the freehand drawing tool; the user has emphasized a point on the slide using a laser pointer. **Right:** the slide overview is invoked by pressing a virtual button; each of the thumbnails in this overview are active regions, allowing the user to quickly jump to the appropriate point in the presentation using the laser pointer.

The prototype system has proven to be so practical that the authors are now using it to deliver most of their presentations.

Acknowledgments

Discussions with Rich Caruana and Terence Sim were important in formulating the distortion correction problem. Mark Kantrowitz was involved in early discussions on laser pointer-based presentation control. Thanks to Gita Sukthankar for generating figures for this paper, and to one of the anonymous reviewers for detailed feedback.

References

- [1] D. Ballard and C. Brown. *Computer Vision*. Prentice-Hall, 1982.
- [2] W. Buxton, G. Fitzmaurice, R. Balakrishnan, and G. Kurtenbach. Large displays in automotive design. *Computer Graphics and Applications*, 20(4), 2000.
- [3] P. Haerberli. Dynadraw: A dynamic drawing technique, 1989. <<http://www-europe.sgi.com/grafica/dyna/>>.
- [4] D. Hall, C. Le Gal, J. Martin, O. Chomat, T. Kapuscinski, and J. Crowley. MagicBoard: a contribution to an intelligent office environment. In *Proceedings of Symposium on Intelligent Robotic Systems*, 1999.
- [5] C. LeGal, J. Martin, and G. Durand. An intelligent and interactive environment. In *Proceedings of Workshop on Managing Interaction in Smart Environments*, 1999.
- [6] R. Raskar, M. Brown, R. Yang, W. Chen, G. Welch, H. Towles, B. Seales, and H. Fuchs. Multi-projector displays using camera-based registration. In *Proceedings of IEEE Visualization*, 1999.
- [7] R. Surati. *A Scalable Self-Calibrating Technology for Seamless Large-Scale Displays*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1999.
- [8] R. Tsai. An efficient and accurate camera calibration technique for 3D machine vision. In *Proceedings of Computer Vision and Pattern Recognition*, 1986.