

Self-Calibrating Camera-Assisted Presentation Interface

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

¹Just Research
4616 Henry Street
Pittsburgh, PA 15213

²The Robotics Institute
Carnegie Mellon Univ.
Pittsburgh, PA 15213

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

Abstract

This paper presents a self-calibrating camera-assisted presentation interface that enables the user to control presentations using a laser pointer. The setup system consists of a computer connected to an LCD projector and a consumer-level digital camera aimed at the presentation screen. Although the locations, orientations and optical parameters of the camera and projector are unknown, the projector-camera calibrates itself by inferring the mapping between pixels in the camera image to pixels in the presentation slide. The camera is subsequently used to detect the position of the pointing device (such as a laser pointer dot) on the screen, allowing the laser pointer to emulate the pointing actions of a mouse. The user may then select active regions in the presentation, or even draw on the projected image. Additionally, arbitrary distortions due to projector placement are negated, allowing the projector (and camera) to be placed anywhere in the presentation room — for instance, at the side rather than the center of the room. This solution works with standard hardware, but could easily be incorporated into the next generation of LCD projector systems.

1 Introduction

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 pre-

sentation. 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 (see Figure 1). Existing systems for accepting user input from the presentation surface include expensive electronic white-boards and pointing devices such as remote mice. Electronic white-boards are not portable and either require laborious manual calibration and/or force the use of specially coded markers. Remote mice lack the transparency and immediacy of pointing actions, and suffer from other problems. For instance: infrared mice require the user to point the mouse at a small target; radio mice are subject to interference; mice with long cables are unwieldy.

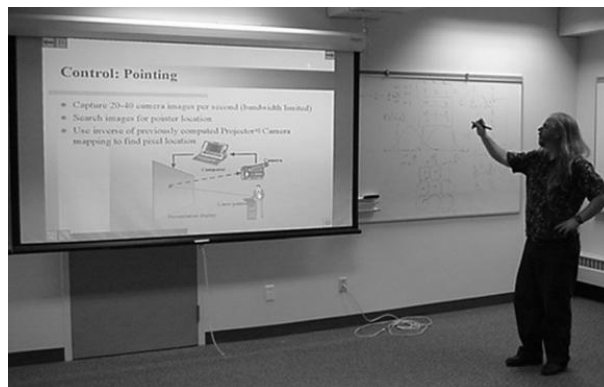


Figure 1: A photograph showing the camera-assisted presentation system in use. Although the portable LCD projector (not visible) is placed at the side of the room, the automatic vision-based keystone correction produces an undistorted image. The user is shown controlling the presentation using his laser pointer.

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

This paper presents a system that enables the user to directly control the presentation in a more natural manner, at a distance from the computer using a pointing device such as a laser or telescoping pointer. The camera may be placed anywhere in the room such that its field of view contains the presentation area. The projector may also be placed anywhere in the room since distortions due to misalignment are automatically corrected.

Figure 2 presents an overview of the camera-projector system. The system first calibrates itself by exploiting knowledge of the projected image. Subsequently, computer-vision algorithms determine where the user is pointing on the presentation surface, providing a correspondence between the user’s actions (as seen by the camera), and active regions in the image being displayed on the screen. The presentation software performs programmed actions in response to the location and characteristics of the user’s pointer actions. This enables users to activate “virtual buttons” on the projection screen simply by pointing, and also to “draw” directly on the presentation surface.

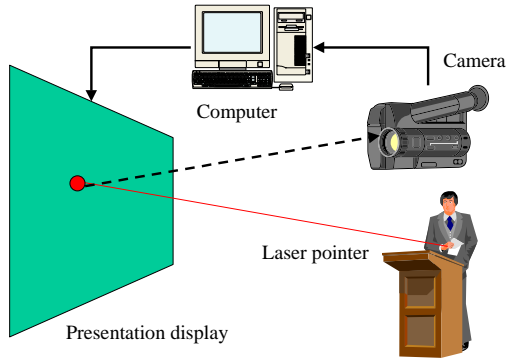


Figure 2: An illustration of the camera-projector system. The computer is connected to a presentation display (such as an LCD projector not depicted here) which is observed by the camera, also connected to the computer. Note that an uncalibrated setup is employed: positions and orientations of the physical screen, camera and LCD projector are unknown, as are the focal lengths of optical components. The position of the laser pointer dot on the screen (as observed by the camera) is used to control the computer.

2 System Calibration

The goal of the calibration process is to automatically determine the mapping between a point on the projection screen (e.g., where the laser pointer dot was detected) and the corresponding pixel in the source image that projected to that point on the screen. Let us first introduce three frames of reference: the camera image frame, the projection screen frame, and the source image frame. Points in the projection screen frame are observed using a camera mounted in an unknown location with unknown optical parameters. Furthermore, the projector’s position, orientation and optical parameters are also unknown. Surprisingly, one can still infer the mapping between the source image frame and camera image frame as follows.

First, we note that the mappings from the source image frame to the projected image frame, and from the projected image frame to the camera image frame are each perspective transforms. When these two transforms are composed (i.e., the projection of the source image is viewed through the camera), the resulting mapping, while not necessarily a perspective transform, can be expressed as a projective transform:

$$(x, y) = \left(\frac{p_1 X + p_2 Y + p_3}{p_7 X + p_8 Y + p_9}, \frac{p_4 X + p_5 Y + p_6}{p_7 X + p_8 Y + p_9} \right), \quad (1)$$

where (x, y) is a point in the source image frame, (X, Y) is the corresponding point in the camera image frame and the parameters $p_1 \dots p_9$ are the unknowns to be determined. Although there are 9 unknowns in Equation 1, there are only 8 degrees of freedom ($\sum_i p_i = 1$). Four point correspondences (where each point provides two constraints) are therefore necessary. Fortunately, in our system, these point correspondences can be automatically obtained by projecting a known rectangle into the environment, and observing the locations of its corners through the camera. Given four points, a unique solution for the parameters is obtained using standard linear algebra techniques.¹ The parameters need only be determined once, during initialization (assuming that the camera and projector will remain fixed during the presentation). Equation 1 then allows the camera-assisted presentation system to efficiently determine the regions (in the source image frame) that are of current interest to the user (as observed by pointing gestures visible to the camera).

¹If more than four point correspondences are available, a least-squares solution is used.

3 Pointer Detection

The user may control the presentation using a variety of methods including: laser pointer motions on the projected image; shadows cast by the user's fingers; or a traditional telescoping pointer (optionally augmented by a lighted or reflective tip). Naturally, control can be augmented using standard keyboard/mouse events. Different pointer types can require different image processing approaches. The methods for pointer detection must be efficient since pointer tracking is required to operate at a high rate with low latency times. Here, we discuss methods specialized at detecting laser dots and telescoping pointers with highly-visible tips.

In a dark environment, such as a presentation theater, the pointer creates a saturated region of pixels in the camera image. This spot may occupy several pixels in the image (due to camera bleeding) and can be extracted by appropriately thresholding the image (e.g., for a typical laser pointer, we examine the red channel of the image). The centroid of these saturated pixels provides an estimate of the pointer location with potentially sub-pixel accuracy. When the pointer cannot be located using simple color- or intensity-based techniques, the feature extraction phase employs methods such as template matching (searching the image for a known shape), or image differencing (comparing the current image with a prior image) to locate the pointer. Image differencing is conceptually straightforward: the current image is compared to a reference image, and the presence of the pointer is seen as a difference between the two images. In practice, creating and updating a reference image is non-trivial, particularly when the scene is not static (as slides change and if the user is visible in the camera image). Our system creates a reference image whenever the user advances to a new slide, and constantly updates the reference image over time using a weighted average.

The pointer-detection methods operate with the camera in a low-resolution mode, where images can be captured at 20Hz through a parallel-port interface². False positives rates in the detection are reduced by voting over a sequence of images. Our experiments show that the system can locate a standard laser pointer's position to within ± 3 pixels on a 1024×768 LCD projector screen using a consumer-level 160×120 greyscale camera.

Once the pointer has been located in the image, its position is converted from camera image frame coordinates to

²The camera-assisted presentation interface employs inexpensive hardware, making the system very accessible to the general user.

the computer's internal coordinates (source image frame) using Equation 1.

4 User Interface

The system provides a general method for specifying pointer actions in projected images. Using the techniques described above, the pointing device can be used to move a mouse pointer in the presentation. Event activation can be triggered by one of several strategies. The easiest approach is to activate an action when the pointer's state changes (e.g., a laser pointer dot that changes color or shape, or a hand gesture change resulting in a different shadow shape). Alternately, actions can be triggered by specific motion patterns (e.g., a virtual button can be activated when a laser pointer dot hovers over the active region for 500ms). Finally, the vision-based pointer detection can be augmented by other input modalities (e.g., voice and keyboard events, or remote mouse buttons). The pointer events enable a variety of interfaces, two of which are detailed below: active regions and freehand drawing.

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 implemented in a straightforward manner. The position of the pointing device, as detected by the camera, can be converted from camera image frame coordinates into source image frame coordinates using Equation 1. If the point falls within any of the active regions (defined as bounding boxes in source image frame coordinates), the associated action is triggered. The active region bounding box is highlighted to provide visual feedback.

Figure 3 (top left) shows a PowerPoint slide displayed using the camera-assisted presentation tool. Several active regions, or "virtual buttons" are 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 the presentation overview, shown in Figure 3 (bottom left); the buttons in the bottom left corner toggle the freehand drawing

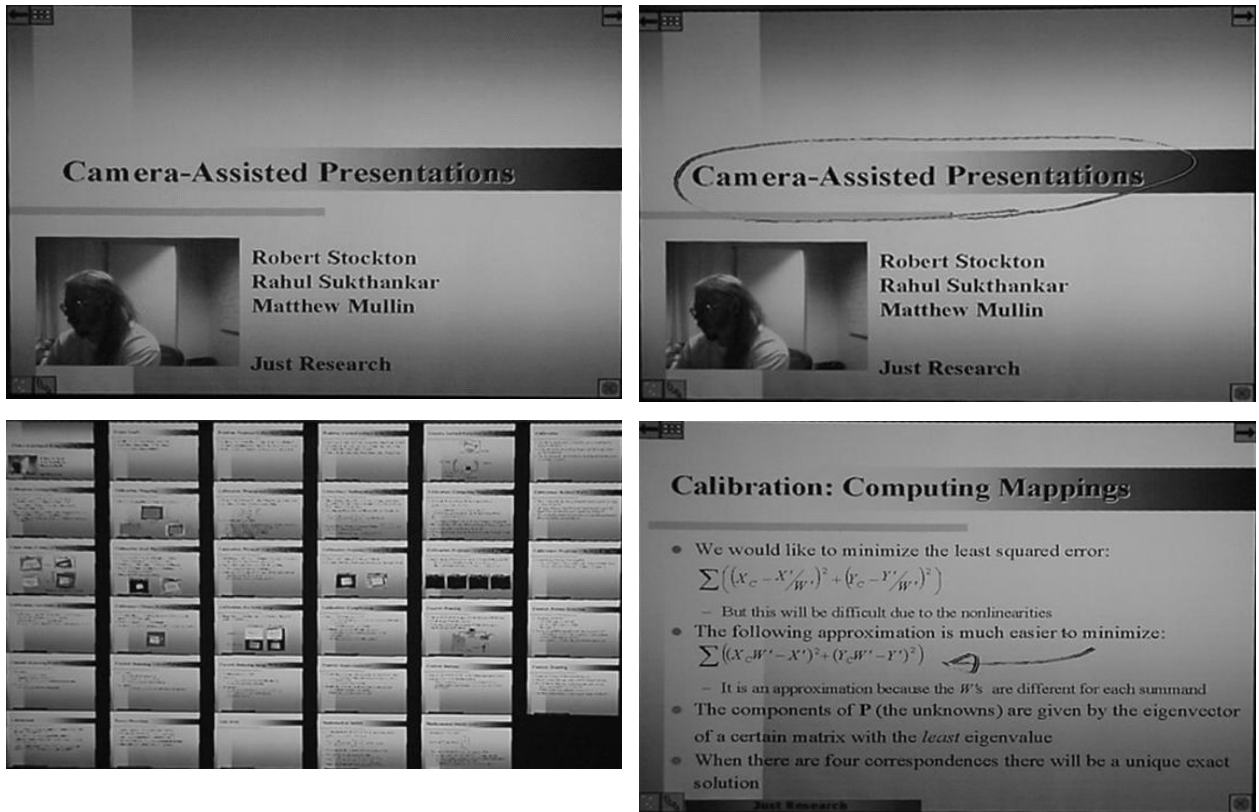


Figure 3: Screenshots from the camera-assisted presentation interface. **Top left:** several active regions (buttons) are automatically added to the corners of the PowerPoint slide when displayed using the camera-assisted presentation tool. **Bottom left:** the slide overview is invoked by pressing a 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. **Top right:** illustration of the freehand drawing interface; the user has highlighted the title text using a laser pointer. **Bottom right:** the user emphasizes an equation by drawing an arrow, using the laser pointer.

mode discussed below; finally, the button in the bottom right corner exits the presentation.

4.2 Freehand Drawing

The camera-assisted presentation system also enables users to highlight and annotate slides, as shown in Figure 3 (top right). The user can use the presentation area as a virtual sketchpad, drawing and erasing lines using the pointing device.

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. Note that simply connecting raw laser pointer positions will produce unsatisfactory results since the laser pointer magnifies any small trembles, creating a very jagged line.

To address this problem, the freehand drawing interface smoothes laser pointer input using a mass-spring-damper model inspired by DynaDraw [1]. 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. By default, the physical parameters are set for critical damping to produce responsive yet smooth curves, as shown in Figure 3 (top and bottom right).

5 Automatic Keystone Correction

Unless the projector is precisely aligned to the presentation screen, the resulting image suffers from perspective (keystone) distortions. Although some modern LCD projectors offer a form of digital keystone correction, this only rectifies the limited class of distortions caused by vertical misalignment and requires tedious manual adjustment by the user. We believe that projectors are often better placed at the side rather than the center of the room, where the projector beam is not blocked by the presenter or audience members. Our automatic keystone correction system pre-warps the image to be projected in such a way that the distortions induced by the projector-screen geometry precisely negate the warping, resulting in a perfectly rectangular image, aligned with the presentation screen (see Figure 4).

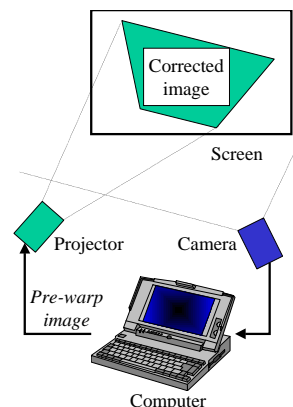


Figure 4: Due to projector misalignment, the rectangular screen appears as a distorted quadrilateral (shown shaded). However, by appropriately pre-warping the source image, the projected image can be made to appear rectilinear (shown by the white rectangle enclosed by the keystone quadrilateral). The pre-warping parameters are automatically determined by the projector-camera system calibration.

Our method for keystone correction is summarized as follows. (1) Determine the mapping between points in the computer display and the corresponding points in the camera image. (2) Identify the quadrilateral corresponding to the boundaries of the projection screen in the camera image. From this, compute a possible mapping between the projection screen and the camera image frame. (3) Infer a possible mapping from the computer display (source image frame) to the projection screen based upon the mappings computed in the previous two steps. (4) Determine an optimal placement for the corrected image on the projection screen. This is the largest rectangle that is completely contained within the projection of the computer display (i.e., the keystone quadrilateral in Figure 4). (5) Pre-warp each application image to correct for keystone.

It is important to note that the goal of keystone correction is to align the projected image to the presentation screen. This is more difficult than warping the image so that it looks rectangular in the camera image frame since the camera may not be centered with respect to the screen. Due to space limitations, the automatic keystone correction system cannot be fully described here; please see [6] for further technical details.

6 Related Work

The camera-assisted presentation system is an instance of a perceptual user interface that indirectly observes the user through his/her interactions with the audience and was motivated by research in real-time shadow gesture recognition [4]; however, our system is able to exploit the camera-projector feedback loop to enable automatic self-calibration. Recent work on multi-projector presentation systems [3], where a presentation is mosaiced onto several surfaces, is also relevant to the automatic keystone correction component of our system.

7 Conclusion

The camera-assisted presentation interface described in this paper has 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 another member of the audience. Second, our system relaxes the usual constraints on a presentation environment: by allowing the LCD 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. The prototype system has proven to be so practical that the authors have used it to deliver their presentations since December 1999, and demonstrations have evoked considerable commercial interest. Since the calibration problem becomes simpler if the camera and projector are integrated into a single device, it is likely that elements of this camera-assisted presentation interface will become standard in future generations of LCD projectors.

Acknowledgments

Thanks to Mark Kantrowitz and Terence Sim, with whom the initial ideas for pointer-based presentation control were discussed, and to Gita Sukthankar for valuable feedback on this paper. Provisional patent applications for the inventions stemming from this work have been filed by Just Research [2, 7, 5].

References

- [1] P. Haeberli. Dynadraw: A dynamic drawing technique, 1989. <<http://www-europe.sgi.com/grafica/dyna/>>.
- [2] M. Mullin, R. Sukthankar, and R. Stockton. Calibration method for projector-camera system. U.S. Patent Filing, 2000. Provisional patent filed in 1999.
- [3] 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.
- [4] J. Segen and S. Kumar. Shadow gestures: 3D hand pose estimation using a single camera. In *Proceedings of CVPR*, 1999.
- [5] R. Sukthankar, R. Stockton, and M. Mullin. Automatic keystone correction. U.S. Patent Filing, 2000. Provisional patent filed in 1999.
- [6] R. Sukthankar, R. Stockton, and M. Mullin. Automatic keystone correction for camera-assisted presentation interfaces. In *Proceedings of International Conference on Multimodal Interfaces*, 2000.
- [7] R. Sukthankar, R. Stockton, M. Mullin, and M. Kantrowitz. Vision-based coupling between pointer actions and projected images. U.S. Patent Filing, 2000. Provisional patent filed in 1999.