

# Ubiquitous Pointing and Drawing

Kristian Jantz<sup>1</sup>, Gerald Friedland<sup>2</sup>, and Raúl Rojas<sup>1</sup>

<sup>1</sup> Institut für Informatik, Freie Universität Berlin, Germany

<sup>2</sup> International Computer Science Institute, Berkeley, USA

**Abstract**—This paper presents a system that enables a presenter to work directly on the projection without the typical interruption of the presentation flow being caused by a mouse or keyboard activity. The lecturer simply uses his fingers as a natural and intuitive pointing device. The system consists of an LED, a standard webcam, a video projector, and a small software system running on a notebook computer. The camera is positioned such that it views the projection area. After a few calibration clicks, the mouse emulation is functional at any place of the projection area.

**Index Terms**—Mobile Presentation Assistant, Camera Driven Pointing Device

## I. INTRODUCTION

In a typical slide-based presentation, an instructor stands in front of the audience presenting his or her ideas using a notebook and a video projector. He or she navigates through the slides by either using the keyboard or the mouse. This setup forces a slight interrupt every time the lecturer wants to progress to a new slide. The situation gets even worse when the presenter wants to outline certain aspects on the slides, for example, by using the drawing function of the presentation software because this requires a dexterous hand-eye coordination to control the mouse. An interruption of the presentation flow also happens when the presenter wants to perform an operation outside the presentation software. Furthermore these activities are hard to be seen by the auditory that cannot keep track of any mouse movement, and even less its activity. Watching the finger of the presenter on the other hand, is nothing less than natural and intuitive. Because today nearly every operation can be performed by using a mouse, a mobile mouse-like device also has many advantages for other applications. An instructor is able to start programs, copy/move files, or navigate through websites in front of a class without having to concentrate on the movements of a small mouse device. When using a drawing tool or chalkboard emulation software such as [3, 4, 9], one is able to convert any wall into an electronic whiteboard. We present a system where the presenter simply uses any light source as a natural and intuitive pointing device. The system can be used anywhere, is easy to setup, and does not need any expensive hardware. Figure 1 shows an example usage.

## II. RELATED WORK

### A. Interactive Datawalls

Even though many pointing devices for presentation exist, such as interactive whiteboards or retro projectors, pen computing in the large still seems to be expensive and

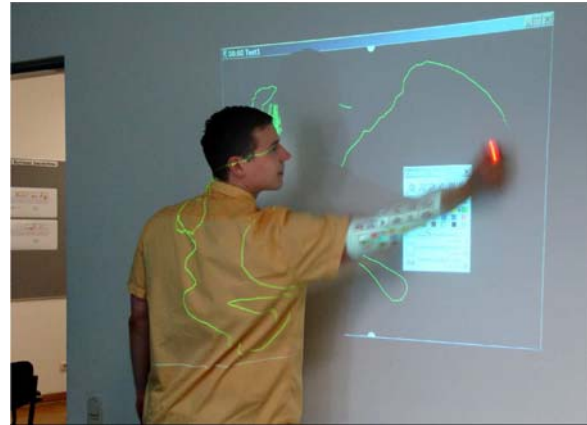


Figure 1. Our system in action: A presenter drawing on a wall without the need of an expensive interactive whiteboard or datawall system.

stationary. The few mobile solutions, such as [8] cost about thousand USD.

The emulation of a mouse device by tracking a light spot with cameras has been researched and successfully implemented by a number of projects. However, all the approaches we know focus on the design of a fixed location (i.e. non-mobile) solution. In most systems, a variable number of video projectors present the desktop on a compound projection area, the projectors may either project directly onto the working area or indirectly using a mirror. Usually, there is a camera located behind each part of the projection area. The camera tracks a laser spot on the surface and the instructor uses a laser pointer as pen substitute.

The first prototype of such a datawall system has been developed by the US Army in August 1998 [6]. The Army developed a very large display that consists of three smaller displays. Cameras located behind each display track the moving laser spot. The presenter is able to work directly on the board using a standard laser pointer as a pen. The Army has developed special tracking hardware that allows tracking the light spot at  $512 \times 480$  pixels with 50Hz. The system works very quick and precise. However, it requires a lot of hardware and needs a time-consuming calibration on its first run. The army also developed a portable version of the datawall, the *Deployable Interactive Datawall*. This device consisted of the projection area, the video projectors, and the tracking devices in one moveable box. Still, a lot of hardware has to be carried around and though it is a *mobile* display it is not as flexible as the solution presented here.

### B. Mobile Phones as Pointing Devices

An interesting application of mobile phones as pointing devices has been presented by Ballagas, Rohs, and

Sheridan [2]. In their work the authors present two techniques that allow turning a mobile phone into a pointing device for public terminals and large public displays.

1) *Absolute Positioning using Point & Shoot*

In the *Point & Shoot* mode a mobile phone is used as a targeting device to select objects. The user moves the phone over the area of interest on the large display. A magnified view of the area and a selection cursor are displayed on the mobile phone. The user may move the cursor and press the *Shoot*-button to finish the selection.

The authors use an overlaid tagging mechanism to locate the desired position. As soon as the user finishes his or her selection, a grid containing graphical coded region information is superimposed on the large display for a short time. The mobile phone evaluates the grid to determine the exact position of the selection. Rohs [7] describes this mechanism in further detail.

2) *Relative Positioning using Sweep*

The second mode of operation allows a mouse cursor directly be moving the mobile phone around. The phone movement is calculated from the optical flow in a series of images taken by the camera of the cellular phone. This method allows for relative movement of the mouse cursor.

According to the authors, the main disadvantage of the system is currently its speed. The movement of the cursor in sweep mode is subject to a perceptible delay, since all image processing operations have to be performed on the mobile phone. The main purpose of the system seems to be the selection of single objects -- for example, to start a program, to open a file, or to control a character in a video game. Using the method for drawings or writings in the drawing mode of presentation software seems to be rather awkward because this requires a quick transition between absolute and relative cursor positioning, the ability to quickly press and release a mouse button, and -- most importantly -- a very small reaction delay of the computer. The direct control of the mouse cursor by using the built-in joystick of the mobile phone also provides insufficient possibilities to draw and is not comparable to a pen-like device.

### III. POINTING AND DRAWING

The mobile system presented here consists of a standard web camera, a notebook, and a projector. Any red light spot can be used as pointing device. One can use a laser pointer, a key-ring pendant with an LED, or any other light emitting device that can be used for pointing. The notebook runs the tracking software and any other application used during the presentation. The tracking software feeds the mouse events directly into the operating system's event queue. Before the system is operational it needs to be calibrated to determine the projection geometry for the calculation of exact mouse coordinates.

#### A. Hardware Setup

The technical setup of our system is shown in Figure 2. The video projector is used to project the computer image onto any surface.

The camera is positioned next to the projector so that it is able to see the entire projected working area. The

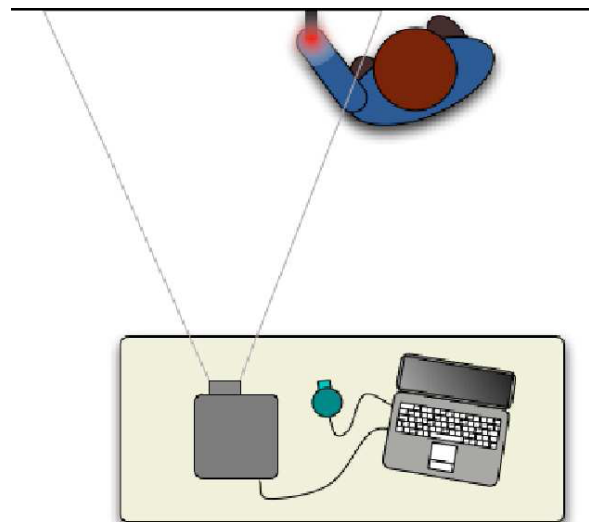


Figure 2. An illustration of the conceptual setup of the mobile pointing and drawing system. The presenter uses a light source as pointer substitute. The webcam tracks the lights as it is moved over the projection area. Our software generates the appropriate mouse events.

lecturer uses an LED as pointing and drawing device which is tracked by the camera.

The actual mouse position is calculated as described in Section 3.3. The pointing device is used like a one-button mouse. Every time the light is tracked by the camera a mouse click event is generated at the corresponding position.

#### B. Calibration

In order to work correctly, the system needs to be calibrated before use. The calibration is needed to take care of the distortion of the camera image caused by the lens and the translation of the projected image to the working area.

The mobile system presented here uses a calibration mechanism derived from the implementation of a fixed place datawall [5]. The calibration is done using a few mouse clicks. The tracking program shows the corners of a grid of variable size (by default  $3 \times 3$ ) on the projection area and asks the user to point at each cross of the grid using the LED. The user does not have to place the LED exactly over the cross the calibration automatically takes care of an offset. Figures 4 show an example where the lecturer uses an LED mounted on the backside of his hand and the system is calibrated such that the mouse cursor is placed on his fingertip and the lecturer writing a text using that calibration. Using an offset calibration presumes that the offset for a given point remains constant in time.

The information gathered during the calibration step is used to transform the coordinates of the light spot into proper mouse coordinates. The points marked by the user define a grid. The grid is triangulated and any coordinate in the distorted camera image is assigned to its surrounding triangle. The calculation of the undistorted mouse coordinates is done by mapping each triangle of the camera image to its corresponding triangle in the undistorted screen coordinate system. The corresponding point within the triangle is found by a linear approximation using two of the sides of the triangle as base vectors. This method is simple to implement, is

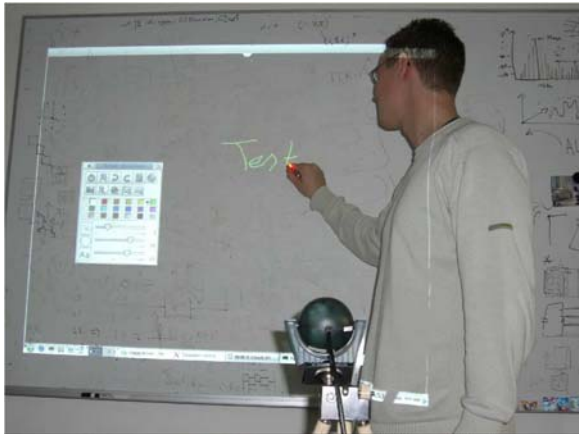


Figure 3. In combination with a drawing tool our system turns any wall into an interactive electronic whiteboard.

efficient, and works well for cameras that are positioned in front of the projection area and do not use special curved lenses. The performance of the tracking is very important as it has to operate at least with 30 frames per second to provide a smooth user interaction. However, it must not consume too much CPU power because the user applications have to run on the same machine as the mouse emulator. We therefore decided to store the mappings in a hash table. The hash table is build right after the calibration of the pen device.

An obvious issue with this model is that the camera distortion is approximated by a linear model. However, the distortion observed by camera lenses is radial. If the camera delivers a very distorted image the grid must be very fine for exact mouse cursor positioning. We found that for most cases the default setting of  $3 \times 3$  is a good trade-off between precision and calibration overhead. Figure 3 shows a scenario where an instructor is using a small key-ring pendant LED to write on a virtual chalkboard [3, 4, 9]. The calibration overhead for the scenarios was less than two minutes. The precision of the tracked pointer is sufficient to draw sketches.

### C. Tracking the Light Spot

Of course, the first step in calculating the mouse coordinates is to track the light spot. The intensity of every pixel is examined and compared to a sensitivity threshold. The Firewire standard [1] defines several methods to manipulate the camera image and allows to dampen or amplify certain colors of the image by reducing parameters like brightness, saturation, gain, or exposure. Using this technical trick provides a good pre-processing and allows to simplify and speed-up the tracking algorithm. The light spot is observed as an area and not as a single point by the camera. We use the centroid of the area to map the image coordinates to screen coordinates.

## IV. CONCLUSION

Operating a mouse driven, graphical desktop system like Linux or Windows during a slideshow presentation or annotating a slide by hand-drawings requires the usage of a mouse-like device. This paper presents a solution for a low-cost, mobile pointing device where the presenter uses any light emitting device as a pointer. Only an LED and a web camera are needed in addition to the hardware that is required for a slide-based presentation anyway. The

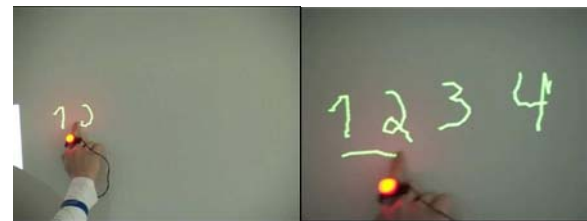
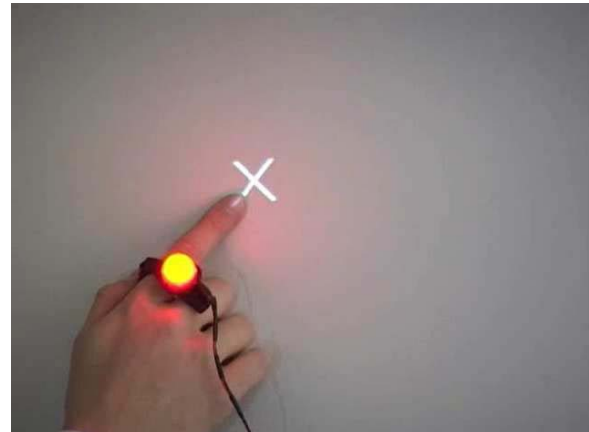


Figure 4. Using a simple setup (above), the system can be calibrated to any pointing device. In the picture, an LED ring with micro switch is used and the system is calibrated such that the mouse pointer position is at the fingertip of the presenter (below)

system operates in real-time at 30 fps on a 2.4GHz single core notebook running Microsoft Windows XP leaving enough CPU resources for any other application. The camera tracks every movement of the LED. After a light spot has been found the program calculates the corresponding mouse coordinates and generates the appropriate mouse events. In order to achieve a high cursor positioning precision, a calibration procedure is needed. Anyway, the overhead for the calibration of the light pen is small and reduces to a few clicks that are easily performable during the usual set up of the presentation.

## V. FUTURE WORK

A disadvantage of the setup is that the presenter might stand in the projector beam. However, this is an inherent problem when using front-projectors. The lecturer may also occlude the light spot so that the camera is not able to track it. If the lecturer points at a corner of the calibration grid using the LED, the light spot may not be observed exactly over that corner, due to perspective reasons and the way the lecturer holds the pen. For example if the lecturer tries to click a button on the left border of the screen, he will most likely hold the pen in a way, it is tracked a few pixels next to the intended spot. Though the problem with the offset is handled autonomously on manual calibration, it needs special treatment if the geometries are determined automatically. We also plan to evaluate the camera position during the calibration. At the moment the user has to place the camera manually, so that it has a proper view onto the screen. In future the program could try to detect the borders of the projection area itself and output a message if the camera position is not optimal. We also plan to track different colors to emulate different mouse buttons. This, of course, requires a more

specialized hardware - a key-ring pendant equipped with a multi-color LED. Ideally, the pointing device presented here should be operable immediately without calibration. Although the calibration procedure does not require much effort, it may still be optimized by learning the projection geometry automatically. The distortion grid could be determined by projecting a checkerboard onto the surface during the calibration step. The system could then find the corners automatically.

#### REFERENCES

- [1] 1394 Trade Association. *TA Document 2003017, DC1394-based Digital Camera Specification, Ver.1.31*, February 2004.
- [2] R. Ballagas, M. Rohs, and J. G. Sheridan. Sweep and point and shoot: phonecam-based interactions for large public displays. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1200–1203, New York, NY, USA, 2005. ACM Press.
- [3] G. Friedland, L. Knipping, R. Rojas, and E. Tapia. Web-based Education as a Result of AI-supported Classroom Teaching. In *Knowledge-Based Intelligent Information and Engineering Systems: 7th International Conference, KES 2003 Oxford, UK, September 3-5, 2003 Proceedings, Part II*, volume 2774 of *Lecture Notes of Computer Sciences*, pages 290–296. Springer Verlag, Heidelberg, September 2003.
- [4] G. Friedland, L. Knipping, E. Tapia, and R. Rojas. Teaching With an Intelligent Electronic Chalkboard. In *Proceedings of ACM Multimedia 2004, Workshop on Effective Telepresence*, New York, October 2004.

- [5] K. Jantz. Ein Stift-Treiber für eine interaktive Multiprojektionswand. Master's thesis, Fachbereich Mathematik und Informatik, Freie Universität Berlin, January 2006.
- [6] P. A. Jedrysik, J. Moore, R. Sweed, and M. Brykowsytc. The Interactive DataWall. Technical Report, Air Force Research Laboratory (AFRL) Rome, NY, 1998.
- [7] M. Rohs. Real-world interaction with camera-phones. In *2nd International Symposium on Ubiquitous Computing Systems (UCS 2004)*, pages 39–48, 2004.
- [8] Virtual Ink. Virtual Ink: Mimio. <http://www.mimio.com>, 2006.
- [9] Website of the E-Chalk project: <http://www.echalk.de>, 2006

#### AUTHORS

**K. Jantz** is with the Institut für Informatik, Freie Universität Berlin, Takustr. 9, 14195 Berlin, Germany (e-mail: [jantz@inf.fu-berlin.de](mailto:jantz@inf.fu-berlin.de)).

**G. Friedland**, was with Freie Universität Berlin. He is now with the International Computer Science Institute, Berkeley, CA 94720, USA (e-mail: [gfriedland@computer.org](mailto:gfriedland@computer.org)).

**R. Rojas** is with the Institut für Informatik, Freie Universität Berlin, Takustr. 9, 14195 Berlin, Germany (e-mail: [rojas@inf.fu-berlin.de](mailto:rojas@inf.fu-berlin.de)).

Manuscript received 17 October 2006.