A Laser Pointer/Laser Trails Tracking System for Visual Performance

Kentaro Fukuchi

Graduate School of Information Systems,
The University of Electro-Communications Choufu-shi, Tokyo , JAPAN 182-8585
fukuchi@megaUI.net

Abstract. Visual performance with a large video projection screen is popular for various entertainment events such as DJ events. Some performers use computers to generate visuals, but using a keyboard or a mouse to control the visuals in front of a large screen is neither exciting nor intuitive for performers and audiences. We developed an interactive display system using camera-tracked laser pointers that enables performers to interact with the screen directly. The system can also detect shapes of the laser trails that enables the performer to move the laser pointers quickly. Most of existing systems employ color and pattern matching techniques that are not suitable for visual performance.

1 Background

Interactive techniques to improvise live visual images are an important challenge for a stage performance. For that purpose, computers are commonly used on-stage to generate real-time rendered computer graphics, and most of the systems are controlled via mice or keyboards. The problem is that using such interactive devices in front of a large screen on the stage lacks the feeling of direct manipulation for a performer. Besides, it is difficult for an audience to recognize the relationship between a body action of the performer and the result on the screen. In addition, it is not very entertaining to see a performer manipulating a mouse or tapping a keyboard. The goal of our research is to make the relationship between performer and screen clear for both the performer and the audience. We believe this goal is attained with our laser pointer tracking system.

2 Laser Trails Tracking System

2.1 System requirements

An effective laser pointer tracking system for visual performance must satisfy some requirements that differ from requirements for desktop applications or interactive presentation systems. First, its scan rate should be high and its latency should be low because the visual on the screen must be synchronized to the motion of the performer's action. Second, the system must track lasers accurately

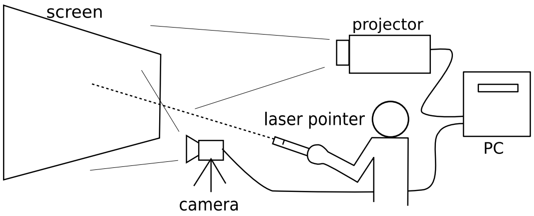

Fig. 1. An illustration of the laser tracking system.

during a performance. Third, the system should be able to apply to any kind of visual presentation without restriction of color or image.

Because of these requirements, we did not adopt the previous approaches[1][2]: background subtraction or color matching techniques are not versatile because they restrict the presentation of the visual, while pattern matching techniques are not robust if the laser spots move quickly.

2.2 Implementation

We developed a laser pointer tracking system and applied it to a live visual performance. This system was designed to meet the requirements described above.

Fig.1 shows an overview of the system. Its basic approach is the same as that of previous systems: the computer is connected to a projector and a camera that observes the screen. We used an IEEE1394 digital camera (Fire-i, Unibrain S.A.) that can deliver uncompressed 640×480 pixels at 30 frames per second. The projector was a standard 2000–2500 ANSI lumen XGA projector.

In order to bypass expensive image processing techniques for laser detection, we used very bright green laser pointers (532nm wave length, class 3a, 5mW), and attached an ND-4 or ND-8 filter that decreases the power of an incoming ray to 1/4 or 1/8. By using this filter, we could eliminate environmental light and image on the screen from the camera's view completely because the luminosity contrast is very high. All of the automatic parameter controls (brightness, white balance, exposure) of the camera were turned off to avoid unexpected parameter shifts during a performance. We set the exposure to almost 1/30 second, the same as the scan rate of the camera. This causes the image of a laser spot moved quickly to become a blurred and slightly dimmed trail (fig.2). This has been considered less suitable for laser spot tracking[3], but we feel justified in using the laser trails because the laser motion by performers is captured as trails. We discuss this approach in section 2.4.

2.3 Calibration

A calibration process is needed before a performance to map the detected positions and shapes of laser trails in the camera image to the corresponding positions and shapes on the screen. We employ a simple solution that uses a perspective transformation.

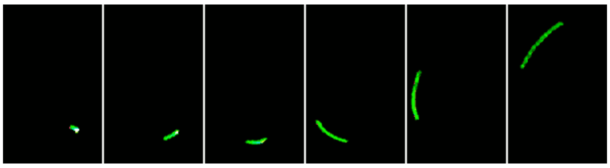

Fig. 2. Sequential shots of a fast moving laser trail.

At the beginning of the calibration process, the user points to four corners of the screen with a laser pointer for a second (30 frames) for every corner. The system measures centroids of the laser spots in the camera image and chooses a *mode* position among 30 recognized positions.

After the four corner positions are given, a three-dimensional transformation matrix is obtained by using standard linear algebra techniques.

2.4 Using laser trails

Most of the previous systems use a centroid of a laser spot to estimate its position and track its motion. In many cases this strategy works well enough to track the motion of the laser for a presentation or a desktop application. But sometimes, as seen in fig.2, a laser trail can be large and skewed. This means that a set of centroids of the trails makes a rough estimation of the motion of the laser pointer, and sometimes this is not desirable for visual performances.

Our approach to solve this problem uses bitmap image of a laser trail. If tracking positions are not needed by the application, using a bitmap has advantages over position tracking. First, it is able to use the entire information of a laser trail, some of which is omitted by the position tracking process. Second, we could avoid delay caused by position tracking process.

As we described in section 2.2, our approach allows us to get a bitmap image of a laser trail. This is sufficient for a painting application. To get a whole stroke of the laser pointer, we compose all the bitmap images of the trails by adding RGB values of pixels with saturation.

To extend this approach to Graphical User Interface, we implemented an experimental GUI button for bitmap image-based interaction. Traditional GUI widgets and interaction techniques (e.g., buttons or sliders, click or drag & drop mechanisms) are not usable with this approach because the GUI depends on a position-based input system.

In fig.3, a rectangle and a circle frame represent buttons, and a filled rectangle and a circle represent *pressed* buttons. When a trail image is captured, every button on the screen scans its region inside its shape (rectangle or circle) and counts the number of pixels of laser trails. If it is non-zero, the button is pressed, or it is *released*. It does not lose touch with the laser trails even if the laser moves quickly (fig.3). At this time, if a button is touched with two or more laser pointers simultaneously, it is not able to detect them, but there would be chance to detect them by counting the increasing number of pixels.

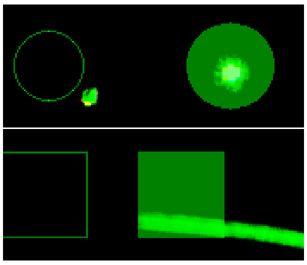

Fig. 3. Button widgets for bitmap image-based interaction.

3 Case Study

We have not evaluated this interaction technique precisely, but we have used the laser tracking system for visual performances and observed the impact of this approach for a performance.

We observed that this approach satisfied our design goal in presenting a clear relationship between a performer's action and a visual result. Especially in a club space, the room was filled with smoky air, but the beams of the laser pointers were clearly visible, and the audience understood that the visuals on the screen were controlled directly by the lasers. Some members of the audience were given laser pointers to use, and these enabled them to interact directly with the screen.

We used both position tracking-based and image-based applications. We consider that the two approaches have different advantages. Position-based interaction is good for making some geometrical graphics or for manipulating graphical objects on a screen by laser interaction. On the other hand, the image-based approach produced an 'action painting-like' performance. For that purpose, we observed that a delay of visuals is fatal for a performance. Our system always causes 1/30 second delay because of the camera interface. If there is an additional one frame delay, it causes around 0.1 second delay in total, and it is not acceptable for a performance with music.

References

1. Kirstein, C., Müller, H.: Interaction with a Projection Screen Using a Camera-tracked Laser Pointer. Proceedings of The International Conference on Multimedia Modeling (1998) 191–192
2. Sukthankar, R., Stockton, R.G., Mullin, M.D.: Self-Calibrating Camera-Assisted Presentation Interface. Proceedings of International Conference on Automation, Control, Robotics and Computer Vision (2000) 33–40
3. Oh, J.Y., Stuerzlinger, W.: Laser Pointers as Collaborative Pointing Devices. Proceedings of Graphics Interface (2002) 141–149