Locating A Projector Using The Strength of Beams Reflected on A Screen

Yukio Ishihara

Faculty of Engineering
Yamaguchi University,
2-16-1, Tokiwadai, Ube,
Yamaguchi, 755-8611 Japan,
i.yukio@yamaguchi-u.ac.jp

Makio Ishihara

Faculty of Information Engineering
Fukuoka Institute of Technology,
3-30-1, Wajiro-Higashi, Higashi-ku,
Fukuoka, 811-0295 Japan,
m-ishihara@fit.ac.jp

Abstract

In this paper, we propose a real-time calibration technique for a projector-camera system, which enables the system to locate the projector even while it is moving. Usually fiducial points are stuck on the screen and the camera tracks both the fiducial points and the quadrilateral illuminated by the projector. Then the projector is located by the tracking. Instead of the fiducial points, our real-time calibration technique uses the strength of the beams reflected on the screen. So, the projection area is not limited in size and position by fiducial points and also it can be extended over a wall as far as the reflected beams are observed. This advantage allows a user to turn the projector at any position on the screen to view the position-dependent image. We take the advantage to assemble a map viewer by connecting our projector-camera system to Google Earth, which enables a user to view the map by moving the projector.

Keywords: Beam strength, real-time calibration, projector-camera system, human computer interaction

1 Introduction

Because of its size becoming small, a projector is easily installed anywhere and widely used in various ways as well as common use such as in presentations. In usual cases, a projector is fixed on-centered to the screen. If the projector is put angled, the projected image will be distorted on the screen. Sukthankar *et al.* (2000) built a projector-camera system where the projector is allowed to be off-centered to the screen. In the system the projector projects a calibration pattern and the camera captures it on the screen. From the correspondences between the calibration pattern and the observed calibration pattern, the mapping between camera and projector coordinates is obtained. And also the mapping between camera and screen coordinates is obtained by observing the outline of the screen. These two mappings give the desired mapping between projector and screen coordinates. By using the desired mapping, the projected image is corrected to appear not-distorted on the screen. In contrast to using one projector, other researches focus on multiple projectors to display a large tiled image on a large screen. Chen *et al.* (2002) aligned 24 projectors to project a seamless large image. Likewise this system needs to be calibrated by obtaining the mappings mentioned above.

On the other hand there is another use of a projector that is allowed to move in operation. In this case, real-time calibration, or quick calibration, is required whenever the projector moves. If the mapping between projector and screen coordinates is obtained every time the projector moves, the calibration pattern will interrupt the projection. So, the calibration pattern cannot be used. Instead, the outline of the quadrilateral illuminated by the projector is used. Beardsley *et al.* (2005) and Rehg *et al.* (2002) built a projector-camera system capable of real-time calibration. Four fiducial points are stuck on the screen, which show the four corners of a rectangle where the projected image is supposed to appear. The camera captures both the outline of the quadrilateral illuminated by the projector and the four fiducial points. Before an image is projected, it is pre-warped first, and then it is actually projected on the screen. This pre-warping is always performed. As a result the image appears not-distorted within the four fiducial points even when the projector moves.

In addition, a projector-camera system capable of real-time calibration is useful for Human Computer Interaction or HCI because the current orientation and position of the projector are available. Beardsley *et al.* (2005) also placed a mouse cursor at the center of the illuminated quadrilateral while projecting the desktop image of MS-Windows within the four fiducial points. The cursor moves when a user moves the projector. At the time, as mentioned above, the desktop image is kept within the four fiducial points. So the projector can be used as a mouse. In our study, we focus on HCI based on a projector-camera system.

The two systems by Beardsley *et al.* (2005) and Rehg *et al.* (2002) use fiducial points for real-time calibration. Our study proposes another real-time calibration technique without those fiducial points. Our technique enables us to real-time calibrate a projector by just viewing the projected image. How can it be done? The basic idea comes from Photometric calibration (Peng & Tat-Jen 2005). While a projector is projecting an image on the screen, the color of the projected image is affected or darkened mainly by the distance between the projector and screen. This photometric calibration can compensate for the spatially darkened color due to the distance, so that the projected image exactly looks like its original image. Instead of the compensating, our technique finds out the distance between the projector and screen by using the extent of the darkness on the projected image. That is, the orientation and position of the projector are obtained by measuring how strong the beams are, which have been projected by the projector and reflected on the screen. This technique needs no fiducial

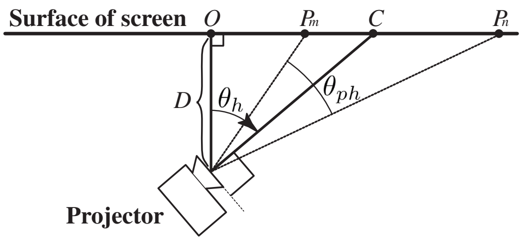

Figure 1: Variables θ_h and D when a projector is angled horizontally.

points. So, the screen can be extended over a wall as far as the reflected beams are observed.

The rest of this paper is organized as follows. Section 2 describes our real-time calibration technique and Section 3 illustrates the framework of a projector-camera system coupled with the real-time calibration technique. Then Section 4 shows an application of the system that is connected to Google Earth to view the map by moving the projector. Section 5 describes strengths and weaknesses of the system, and finally Section 6 gives the concluding remarks.

2 Real-time Calibration Technique

In this section, we discuss how to obtain the orientation and position of a projector from the strength of the beams reflected on the screen. We also measure the strength at each position on the screen to complete our technique.

2.1 How Our Technique Works

We assume that a projector projects the same strength of beams in every direction within the projector's perspective when a one-colored image, such as a white-filled image, is projected. Let θ_h be the angle at which the projector is angled horizontally to the screen, and D be the distance between the projector and screen as shown in Figure 1. O is the nearest position on the screen from the projector, and C is the position on the screen, that is in the center of the projector's perspective. $f_S(\theta_h, D)$ shows the strength of the beam reflected at C. Now we describe how to derive θ_h and D from the observed strength of beams and the function $f_S(x, y)$. If the strength of the beams reflected at two positions P_m, P_n is observed as S_m, S_n respectively, then θ_h and D are derived from the following simultaneous equations.

$$S_m = f_S\big(\theta_h - \frac{\theta_{ph}}{2}, D\big) \tag{1}$$

$$S_n = f_S\big(\theta_h + \frac{\theta_{ph}}{2}, D\big) \tag{2}$$

where θ_{ph} represents the horizontal angle of the projector's field of view. Note that on the assumption, S_m is the same as the strength when the projector is panned left by $\theta_{ph}/2$. Likewise, S_n.

Next this is extended to allow vertical pan. Let θ_v be the angle at which the projector is angled up as shown in Figure 2. The gray quadrilateral on the screen is the area illuminated by the projector. The projector projects three white squares at the top P_t and in the bottom left corner P_l, and in the bottom right corner P_r. θ_t, θ_l, and θ_r are the angles between the line from the projector to O and the line from the projector to P_t, P_l, and P_r respectively. These three angles θ_t, θ_l, and θ_r are given as follows.

$$\theta_t = f_\theta\Big(\theta_h, \theta_v, \quad 0, \quad \frac{\theta_{pv}}{2}\Big) \tag{3}$$

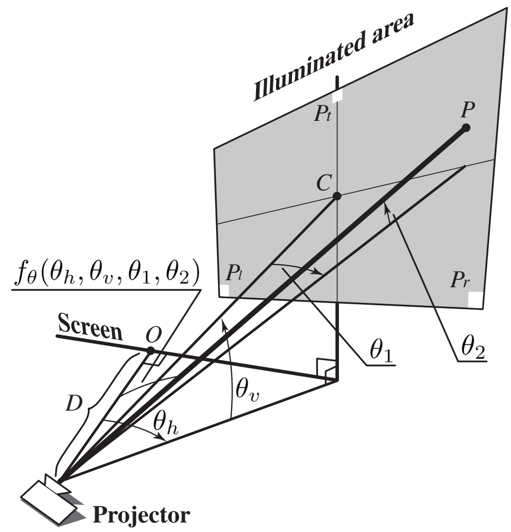

Figure 2: Variables θ_h, θ_v, and D when a projector is angled horizontally and vertically.

$$\theta_l = f_\theta\Big(\theta_h, \theta_v, -\frac{\theta_{ph}}{2}, -\text{atan}\big(\cos\frac{\theta_{ph}}{2}\tan\frac{\theta_{pv}}{2}\big)\Big) \tag{4}$$

$$\theta_r = f_\theta\Big(\theta_h, \theta_v, \quad \frac{\theta_{ph}}{2}, -\text{atan}\big(\cos\frac{\theta_{ph}}{2}\tan\frac{\theta_{pv}}{2}\big)\Big) \tag{5}$$

where,

$$\begin{aligned} f_\theta(\theta_h, \theta_v, \theta_1, \theta_2) = \; & \cos\theta_h \cos\theta_v \cos\theta_1 \cos\theta_2 \\ & - \cos\theta_h \sin\theta_v \sin\theta_2 \\ & - \sin\theta_h \sin\theta_1 \cos\theta_2 \end{aligned} \tag{6}$$

$f_\theta(\theta_h, \theta_v, \theta_1, \theta_2)$ gives the angle between the line from the projector to O and the line from the projector to P. θ_1 and θ_2 are horizontal and vertical angles respectively to P from the center C as shown in Figure 2. So, the function f_θ gives each of θ_t, θ_l, and θ_r when θ_1 and θ_2 are set properly as in Equation (3)–(5). θ_{pv} represents the vertical angle of the projector's perspective. Now we describe how to derive θ_h, θ_v, and D from the observed strength of beams and the function $f_S(x, y)$. Similarly to Equation (1) and (2), given the strength of the beams reflected at P_t, P_l, and P_r as S_t, S_l, and S_r respectively, these three values θ_h, θ_v, and D are derived by solving the following simultaneous equations.

$$S_t = f_S(\theta_t, D) \tag{7}$$

$$S_l = f_S(\theta_l, D) \tag{8}$$

$$S_r = f_S(\theta_r, D) \tag{9}$$

Note that $f_S(x, y)$ is used even when x is not a horizontal angle because $f_S(x, y)$ is the same between two cases that the projector is angled diagonally by x and it is angled horizontally by x. In the next subsection, we describe the function $f_S(x, y)$.

2.2 Strength of Reflected Beams

Now we make up the function $f_S(x, y)$ by measuring the strength of the beams reflected at the 15 positions on the screen, or the 15 pairs of $(x, y) = (0, 60)$, $(0, 80)$, $(0, 100)$, $(0, 120)$, $(0, 140)$, $(0, 160)$, $(0, 180)$, $(0, 200)$, $(10, 100)$, $(20, 100)$, $(30, 100)$, $(40, 100)$, $(50, 100)$, $(60, 100)$, $(70, 100)$ in degree and cm respectively, while projecting a white-filled image. In order to observe the strength of the beams, the lens of the camera should be placed at the same position as one of the projector. For simplicity, the camera was attached to the left side of the projector, which is the

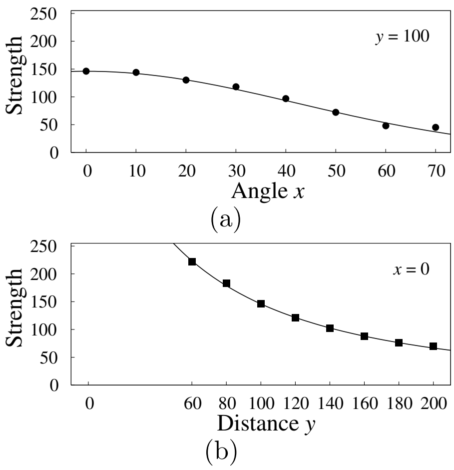

Figure 3: Strength of the reflected beams. (a) shows the change of the strength as angle x is set to 0^o to 70^o with constant $y = 100$, while (b) shows the change of the strength as distance y is set to 60 to 200cm with constant $x = 0$.

closest position to the projector's lens. And then the strength of the beams was observed.

Figure 3 presents the results of the observed strength as a function of angle x in (a), and distance y in (b) respectively. The strength ranges from 0 to 255. From the results, it seems to be reasonable that the following equations (10), (11) fit the plots in (a), (b) respectively.

$$f_1(x) = \delta e^{-\alpha x^2} \tag{10}$$

$$f_2(y) = \frac{\gamma}{(y-\beta)^2} \tag{11}$$

where α, β, γ, and δ are parameters. The other strength besides those 15 plots is complemented as follows.

$$\begin{aligned} f_S(x,y) &= \frac{f_1(x)}{f_1(0)} \times f_2(y) \\ &= e^{-\alpha x^2} \times \frac{\gamma}{(y-\beta)^2} \end{aligned} \tag{12}$$

$f_1(x)$ and $f_2(y)$ are multiplied after $f_1(x)$ is normalized. Then let the result be $f_S(x,y)$ that gives complemented strength besides the 15 plots. In order to determine the three parameters α, β, and γ in Equation (12), we used three observed strength $f_S(0,100) = 146$, $f_S(0,160) = 88$, and $f_S(50,100) = 72$. Then the parameters were set to $\alpha = 0.00028$, $\beta = -108.29$, and $\gamma = 6334296.72$. The two curves of $f_S(x,100)$ and $f_S(0,y)$ are shown in Figure 3(a) and (b) respectively.

3 Set-up of Projector-Camera System

Our projector-camera system capable of real-time calibration includes three PCs with 1GHz×2 and 860MHz processors, Logicool Qcam QV-700NHS (320×240 in reso., 30fps), and BenQ Digital Projector P2120 (1024×768 in reso., 60Hz, $\theta_{ph} = 26.9^o$, $\theta_{pv} = 20.4^o$). Figure 4 is the look of our projector-camera system. The camera and projector are fixed to the white container, so this relative position is constant. The camera is connected to a PC through USB 1.1 port, while the projector is connected to another PC through Monitor port. And the three PCs are connected on 100Mbps LAN. Figure 5 presents the framework of the system, which consists of Image

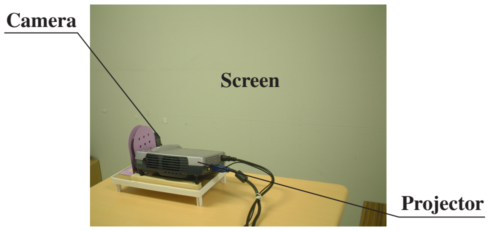

Figure 4: Look of our projector-camera system. A web camera is attached to the left side of the projector.

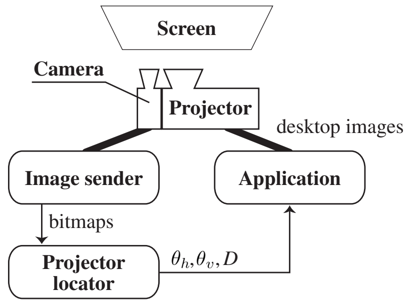

Figure 5: Framework of our system.

sender, Projector locator, and Application. Each of these three modules runs on each of the three PCs. These modules are summarized below.

Image sender takes bitmaps on the screen and sends them to Projector locator.

Projector locator receives the bitmaps from Image sender, and searches for the three white squares on them. This image processing is quite easy and simple because the relative position of the camera and projector is constant, which means that the white squares do not move a lot within the bitmaps. Afterwards θ_h, θ_v, and D are calculated from the strength of the white beams reflected on the screen using Equation (7)–(9) and (12). In fact it is not easy to solve the simultaneous equations (7)–(9). So, we used the method of steepest descent to find optimal values of θ_h, θ_v, and D. It is done by minimizing R.

$$\begin{aligned} R = &\left(S_t - f_S(\theta_t, D)\right)^2 + \left(S_l - f_S(\theta_l, D)\right)^2 \\ &+ \left(S_r - f_S(\theta_r, D)\right)^2 \end{aligned} \tag{13}$$

Unfortunately how long it takes to obtain the optimal values depends on S_t, S_l, and S_r. So the processing time can be too long to react to changes in projector's orientation and position. In order to prevent the problem, iteration steps of the method of steepest descent were fixed constant 50. Although the obtained values of θ_h, θ_v, and D are not optimal yet, these values are sent to Application at regular intervals.

Application receives θ_h, θ_v, and D. These values can be used in two ways. One is for the compensation for image distortion, while the other for HCI. In this paper, these values are used for HCI. That is, a user can operate the PC running Application just by moving the projector. Finally the desktop image of the PC is projected with the three white squares. Details of Application are given at the next section.

4 Map Viewer with Projector-Camera system

We assembled a map viewer by connecting our projector-camera system to Google Earth, which enables a user to view the map by moving the projector.

(a) (b) (c)

Figure 6: Map viewer based on our projector-camera system and Google Earth. (a) A user was holding the projector perpendicular to the screen and viewing the map of Venezia, Italy and then (b) the user panned the projector left. Finally (c) the user stepped back to see the position of Venezia on the Italian Peninsula.

4.1 Mouse Events Specific to Map Viewer

After Application receives θ_h, θ_v, and D, two mouse events are issued. When θ_h increases, wm_mousemove is issued, so that the mouse cursor moves left. Likewise wm_mousemove is issued when θ_v does, but the mouse cursor moves down. How many pixels the mouse cursor moves depend on how many degrees the projector is panned. When the projector is panned left by $\theta_{ph}/2$, for example, the mouse cursor will move right by the half of the illuminated quadrilateral. When D increases, wm_mousewheel is issued to rotate the mouse wheel forward.

4.2 Demonstration

By the specific mouse events, a user is allowed to scroll/zoom in/zoom out the map on Google Earth by moving the projector. The map is scrolled right/left when the user pans the projector left/right, while the map is scrolled up/down when the user pans it down/up. And also the map is zoomed in/zoomed out when the user steps forward/back with the projector.

Figure 6 presents a series of screenshots when a user was viewing the map of Venezia, Italy. Note that these screenshots were not taken by the camera attached to the projector. As mentioned at **2.1**, three white squares were projected at the top, in the bottom left corner and right corner, while the window of Google Earth was projected in the center. The blue bar at the bottom was the minimized task bar. The user was holding the projector perpendicular to the screen at (a), then he panned the projector left and the map scrolled right simultaneously. As seen in (b), the zigzag river was on the right side, which had been scrolled right from the left side at (a). Finally the user stepped back from the screen, and the map was zoomed out at (c).

The system averaged 16fps in frame rate while viewing the map. In Figure 6, the strength of the beams reflected at the three points seems not changed depending on the orientation and position of the projector. But in reality the projected image was totally darkened at (c) compared to (a).

5 Strengths and Weaknesses

Strengths of our projector-camera system include no fiducial points needed on the screen for real-time calibration, which provides easy setup of the system, and also does a large screen as far as the beams reflected on the screen are observed.

Weaknesses include (i) no support for the tilt of the projector in contrast to the systems by Beardsley *et al.* (2005) and Rehg *et al.* (2002), and (ii) the blur caused by the distance between the projector and screen. In our future work, (i) will not be covered because the projector was not tilted much during the demonstration while the user was holding it. And (ii) will be improved because the blur terribly affects the usability of the system. That is, a user cannot see letters clearly if the user steps back.

6 Conclusions

In this paper, we proposed a real-time calibration technique for a projector-camera system, which enables the system to locate the projector in real-time. To do real-time calibration, fiducial points are usually used on the screen. But in our study we did not use them. Instead we measured the strength of the beams reflected on the screen to locate the projector. This approach leads to the advantage that the screen can be extended over a wall as far as the reflected beams are observed because the projection area is not limited in size and position by fiducial points. Taking the advantage, we assembled a map viewer by connecting our projector-camera system capable of real-time calibration to Google Earth. The map viewer allows a user to scroll/zoom in/zoom out the map on Google Earth by moving the projector. Finally, in future work, the system will be extended to work on more walls rather than one, so that the walls including ceiling in a room are used as one large surrounding screen.

7 References

Beardsley, P., van Baar, J., Rasker, R. & Forlines, C. (2005), Interaction using a handheld projector, *in* 'IEEE Computer Graphics and Applications', Vol. 25, pp. 39–43.

Chen, H., Sukthankar, R., Wallace, G. & Li, K. (2002), Scalable alignment of large-format multi-projector displays using camera homography trees, *in* 'IEEE Visualization 2002', pp. 339–346.

Peng, S. & Tat-Jen, C. (2005), A theory for photometric self-calibration of multiple overlapping projectors and cameras, *in* 'IEEE Computer Society Conference on Computer Vision and Pattern Recognition', Vol. 3, p. 97.

Rehg, J.M., Flagg, M., Tat-Jen, C., Sukthankar, R. & Sukthankar, G. (2002), Projected light displays using visual feedback, *in* 'International Conference on Control, Automation, Robotics and Vision', Vol. 2, pp. 926–932.

Sukthankar, R. Stockton, R.G. & Mullin, M. (2000), Automatic keystone correction for camera-assisted presentation interfaces, *in* 'International Conference on Multimedia Interfaces', pp. 607–614.