Steerable Projector Calibration

Mark Ashdown and Yoichi Sato
Institute of Industrial Science, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo, 153-8505, Japan
mark@ashdown.name, ysato@iis.u-tokyo.ac.jp

Abstract

A steerable projector is a digital projector whose beam can be moved under computer control to illuminate different objects in its environment. Various projects have explored the possibilities of steerable projectors but they have not addressed the calibration of a generalized optical and mechanical system. We present a method for calibrating a device comprised of a projector and a pan-tilt mirror. It starts by obtaining the internal parameters of a camera and the projector, then the pan-tilt mirror is placed in a series of positions and for each one the reflected projector position is calculated from a pattern projected onto a planar surface. From those readings a coarse version of the steerable projector parameters is obtained, which is then iteratively refined. We describe real results from our physical setup and a range of results from simulated data to characterize the performance of our algorithm.

1. Introduction

Digital projectors have been put to various novel uses in recent years, but the most fundamental of these must be the ones in which normal objects in the environment are augmented with projected graphics since these are not possible with any other display hardware. This type of augmentation allows, for instance, the computer to answer queries over physical objects by highlighting their actual location [3]. To augment objects spread around a normal indoor environment one needs a digital projector whose beam can be moved under computer control to illuminate different objects—that is, a steerable projector.

There are two types of steerable projector: ones in which the projector rotates, and ones in which the projector remains stationary and a moving mirror directs the beam. The former type has the advantage of a wider range of projection angles. The latter should allow the physical device to be smaller, and because the mirror is much smaller and lighter than the projector the movement of the beam will be faster and its dynamics will be easier to model, which will be important when switching between projection surfaces or tracking moving objects.

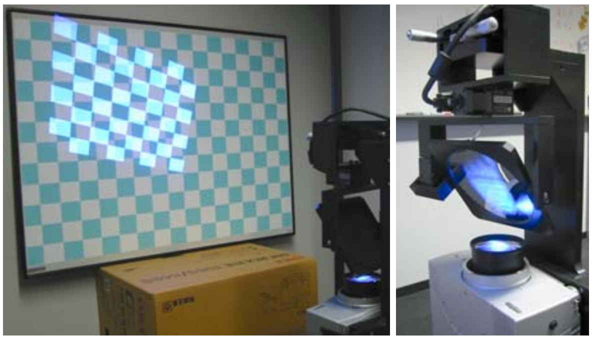


Figure 1. Steerable projector incorporating a pan-tilt mirror displaying a calibration pattern on a checkered surface.

Here we deal with the second type of steerable projector that incorporates a moving mirror (Figure 1), and present the first method for calibrating this type of device which recovers the full optical and physical configuration of the projector and mirror.

We start by reviewing some previous work (Section 2), we describe the model we use to represent the system (Section 3), then the main part of the paper explains how we calibrate the steerable projector (Section 4), we then report on the performance in speed and accuracy (Section 5), and finish with conclusions and directions for future work (Section 6.)

2. Previous Work

If a projector is used to display graphics on anything other than an flat surface aligned with the projector axes geometric calibration is necessary, and much work has been performed in recent years on this subject for fixed and portable projectors. Assuming a planar surface and pin-hole optics a projector can be calibrated using a planar homography [14], or a tree of homographies for a multi-projector display [4]. Non-planar surfaces can be augmented with graphics if the 3D configuration is known [12], and this configuration can be quickly measured with the help of the projector if objects in the environment contain RFID devices with light sensors [11].

The Everywhere Displays Project [9] at IBM Research has involved much work on applications for steerable projectors [13] and on modular computer-vision based interfaces for controlling the projected displays, but calibration for steerable projectors is less advanced. The Everywhere Displays system requires manual setting of the steerable projector parameters—pan, tilt, zoom, focus, and projector-to-surface transformation—for each of a fixed set of projection surfaces [8]. It can also utilize a 3D model of the environment to automatically select a projection surface based on the position of a tracked user. Automatic selection of projection surfaces has been demonstrated using a combination of scanning with laser and optical devices [15]. Okatani and Deguchi [7] give an algorithm related to steerable projectors which is used to calibrate a camera pointed at a featureless wall. Instead of a projector they use four laser pointers in a fixed configuration which produce four points on the wall that are captured by the camera. They manually move the device around and use the detected points in the camera to calculate a planar homography between camera and wall. Raij and Pollefeys [10] calibrate a multi-projector display wall using only the positions of projected points on a featureless wall. They make some simplifying assumptions about the projector parameters—pin-hole optics and just two internal parameters for each projector. Although this technique was designed for a static arrangement of projectors it could be combined with our technique for calibrating a steerable projector to avoid the need for feature points on the projection surface.

Recently a steerable projector designed to follow moving objects has been presented [5], although the calibration is mainly focused on the real-time constraints and dynamics of the movement to ensure registration with a moving target can be maintained, rather than on precise calibration of the projector-camera system. Borkowski *et al* [2] demonstrated a pan-tilt projector that rotates about its optical centre. Assuming that a moving projector rotates about its optical centre greatly simplifies the problem of calibration, but places precise constraints on the physical configuration and requires the centre point to be known in advance. The position of the optical centre relative to the plastic box that houses the electronics of the projector must be found via a calibration step, and it will change when the zoom and focus of the projector are changed [1]. Similar simplifying assumptions could be made for a steerable projector with a mirror, to constrain the relative pose of the projector and mirror and constrain the design of the pan-tilt mirror. However we would like to use a very general model to support any type of hardware that might be created, and in future to allow the zoom and focus of the projector to be changed during operation. Also we would like to allow for displacements between the pan and tilt axes, and between the tilt axis and the mirror, to allow compact pan-tilt mirrors to be created such as the one used in the PixelFlex project [16].

3. Steerable Projector Model

We have created our steerable projector my combining a standard digital projector with a mirror attached to a pan-tilt unit (Figure 1). Our current device is 105 cm tall and weighs about 50 kg. With readily available projector and pan-tilt components a more compact version could be made that is around 60 cm long and weighs around 8 kg, although a projector that must illuminate distant objects under normal lighting conditions requires high brightness and there is a tradeoff between the size and weight of the projector and its output in lumens. A more compact pan-tilt mirror could be made by allowing a non-zero displacement between the axes and between axis and mirror.

Our model of the system contains the internal parameters of the projector (eight parameters), the projector pose (six parameters), the pose of the pan-tilt mirror system (six parameters), and the internal details of the mirror system (three parameters), for a total of 23 parameters.

The model for the pan-tilt mirror is shown in Figure 2. The mirror is moved by changing the pan and tilt angles ϕ and θ which cause it to rotate about the pan and tilt axes respectively. The pan axis remains stationary while the tilt axis rotates around it. The pose of the mirror system in world co-ordinates is represented by a rigid transform $\mathtt{W}$, the displacements from the two axes are t_1 and t_2, and the offset angle around the tilt axis is α. During calibration we also calculate an offset angle around the pan axis (Section 4.6) but in the final result this extra rotation is included in $\mathtt{W}$. We assume that t_1 and t_2 are known before calibration starts—these will be dependent only on the measurements of the pan-tilt unit, which should be specified precisely by the manufacturer, and the thickness of the mirror, which should be known accurately.

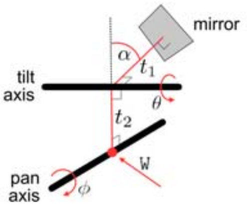


Figure 2. In our pan-tilt mirror model the unit is positioned with a rigid transform $\mathtt{W}$ and also has parameters t_1, t_2, and α. The mirror is moved by changing the pan and tilt angles ϕ and θ.

In homogeneous co-ordinates a reflection $\mathtt{D}$ of three-dimensional points in a mirror in the x-y plane is given by

$$\mathtt{D} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \quad (1)$$

For a mirror that is not in the x-y plane, but on the end of a pan-tilt unit, we use $\mathtt{D}$ as part of a function $\mathtt{M}(\theta, \phi)$ which is dependent on the pan and tilt angles and is defined as

$$\mathtt{M}(\theta, \phi) = \mathtt{Trans}(\theta, \phi)\ \mathtt{D}\ \mathtt{Trans}^{-1}(\theta, \phi)\ , \tag{2}$$

where

$$\mathtt{Trans}(\theta, \phi) = \mathtt{W}\mathtt{R}_y(\phi)\mathtt{T}_z(t_2)\mathtt{R}_x(\alpha + \theta)\mathtt{T}_z(t_1)\ , \tag{3}$$

and $\mathtt{R}_x$, $\mathtt{R}_y$, and $\mathtt{T}_z$ respectively are rotations about the x and y axes and a translation in the z direction. As described above, $\mathtt{W}$ transforms the mirror system from the origin to its actual pose in world co-ordinates. The transformation $\mathtt{Trans}$ is simply a rigid transformation, dependent on the pan and tilt angles ϕ and θ, that takes the mirror from the x-y plane to its final pose, and $\mathtt{Trans}^{-1}$ is its inverse.

Of the 23 parameters of the full system two, t_1 and t_2, are given, so calibration consists of estimating the other 21. The next section describes the stages of our algorithm to obtain the parameter values.

4. Calibration

This section describes our calibration algorithm which calculates the parameters of the projector and the pan-tilt mirror. The output of the various stages is combined to generate the final result.

4.1. Stages of the Algorithm

Calibration consists of obtaining values for the 21 parameters described in Section 3. The stages of the calibration algorithm are shown in Figure 3. The camera is calibrated, which then allows the internal parameters of the projector to be obtained (Section 4.2.) Images of projected patterns from a series of mirror positions are then used to obtain a *reflected projector pose* for each pair of θ and ϕ values (Section 4.3.) Those reflected projector poses are the input to a stage in which repeated searches followed by clustering are used to obtain the projector pose (Section 4.4.) The pose of the tilt axis is found for each value of ϕ (Section 4.5), then the position of the pan axis is found (Section 4.6.) Now the the complete set of parameters for the system can be calculated. For the final stage iterative refinement is used on the parameters of the pan-tilt mirror (Section 4.7.)

We use a Directed Perception PTU-46-17 pan-tilt unit. The real data we have captured uses six values of θ and six values of ϕ to give a total of 36 mirror positions. For each of those an image of the projected pattern is taken by the camera.

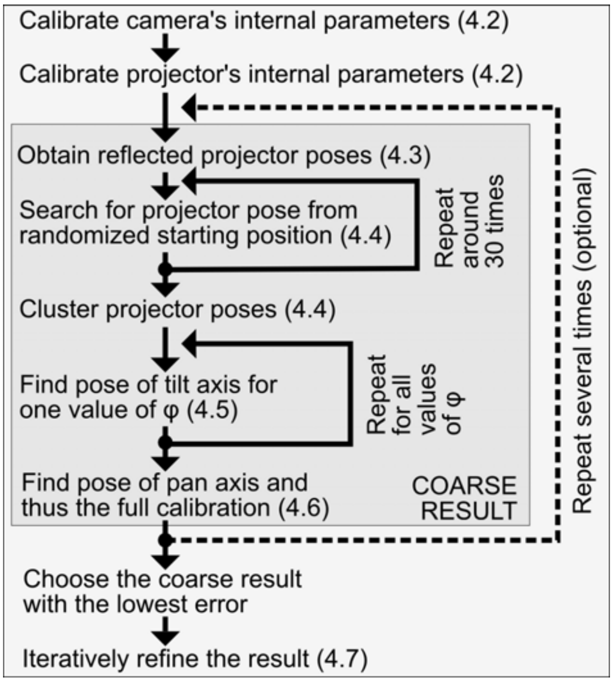

Calibrate camera's internal parameters (4.2)
Calibrate projector's internal parameters (4.2)
Obtain reflected projector poses (4.3)
Search for projector pose from randomized starting position (4.4)
Repeat around 30 times
Cluster projector poses (4.4)
Find pose of tilt axis for one value of φ (4.5)
Repeat for all values of φ
Find pose of pan axis and thus the full calibration (4.6)
COARSE RESULT
Repeat several times (optional)
Choose the coarse result with the lowest error
Iteratively refine the result (4.7)

Figure 3. The stages of the calibration algorithm. Corresponding section numbers are shown in parenthesis.

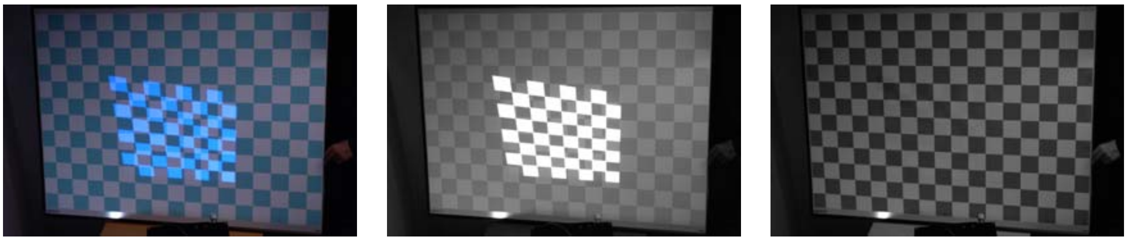
The parameters we obtain during calibration contain one rigid transform to place the projector in space and another to place the pan-tilt mirror. If the steerable projector is to be moved after calibration to a better position for illuminating the whole room this information can be split into two parts: we can define the pan-tilt mirror pose relative to that of the projector, then the information is split into an external rigid transform the specifies the pose of the steerable projector device, and a set of internal parameters that will remain fixed if that device is moved.

4.2. Camera and Projector Intrinsics

The camera is calibrated using the Camera Calibration Toolbox for Matlab.[1] We use the default settings for the tool box, for which calibration consists of obtaining eight internal parameters: focal length, aspect ratio, two parameters for the principle point, and four radial and tangential distortion coefficients. We assume zero pixel skew. We use a surface with a printed checkered pattern, and take about 25 images of it in different poses. The surface has 17×12 70-millimetre squares giving a total of 171 internal corner points. We then use the correspondences between the printed corner points and their images in the camera as input to the algorithm.

The same method can be used to calibrate the projector once we have a set of point correspondences between 2D points on the surface and 2D points in the projector. When projecting a calibration pattern the 2D location of points in

[1] http://www.vision.caltech.edu/bouguetj/calib_doc/



Original image Blue channel Red Channel

Figure 4. We use the blue and red channels of a colour image to simultaneously capture points from the projected pattern, and from the underlying surface.

the projector will be known in advance but those on the surface must be measured.

To simultaneously capture an image of the printed points and the projected points for a moving board we project a blue checkered pattern onto a cyan and white checkered surface (Figure 4 (left)). We use an 8×8 projected pattern, which has 49 internal corner points. In the blue channel the blue projected pattern is clearly visible and the printed pattern is faint because the cyan and white squares under white ambient illumination have similar—ideally identical–blue components (Figure 4 (centre)). In the red channel the projected pattern virtually disappears, leaving only the checkered surface underneath which appears dark because the cyan squares absorb most of the red component of the ambient light (Figure 4 (right)).

The known positions of the points on the surface and the points in the red channel image, compensated for distortion using the previously obtained camera parameters, are used to obtain a planar homography from normalized camera points to surface points. That homography is then used to map the points detected in the blue channel image to the surface, thus obtaining a surface point for each known projector point. We use these point correspondences in the camera calibration routine, treating the projector just like a camera, to obtain the same eight parameters for focal length, aspect ratio, principle point, and distortion, thus obtaining the first eight of the 21 parameters of the steerable projector system.

This method of projector calibration works without any alteration if the projector image is first reflected in a mirror, as we have verified with simulated data, so the presence of the pan-tilt mirror can be ignored for this stage. From now on we will assume that camera and projector points are corrected for distortion so that we can use pin-hole optics.

4.3. Reflected Projector Poses

Before calculating the projector pose in Section 4.4 we calculate a set of reflected projector poses, one for each mirror pose. If there is a mirror between the projector and the surface, and the pose of the mirror is not known, the pose of the projector cannot be calculated from the projected points on the surface. However, the pose of a reflected version of the projector can be calculated (Figure 5.) Each pair of θ and ϕ values produces a different mirror pose and thus a different set of projected corner points on the surface. For each of those mirror poses we calculate a reflected projector position as described below.

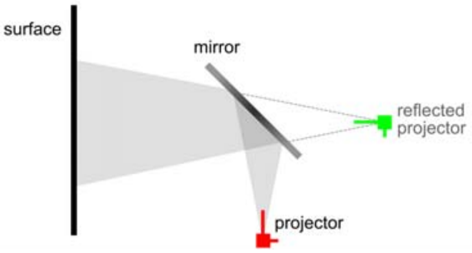


Figure 5. From the projected pattern we can retrieve the pose of the *reflected projector* but not the pose of the real projector or the mirror.

The internal parameters of the projector are obtained as described in Section 4.2 and placed in a 3×3 upper-triangular matrix $\mathtt{K}$ [6, page 157]. For a projector without a mirror, homogeneous 3D points $\mathbf{X} = (X, Y, Z, 1)^\top$ and their corresponding homogeneous 2D projector points $\mathbf{x} = (x, y, w)^\top$ are related by the equation $\mathbf{x} = \mathtt{K}[\mathtt{R} \quad \mathbf{t}]\mathbf{X}$ just as for a camera, where $\mathtt{R}$ and $\mathbf{t}$ determine the orientation and position of the projector—the orientation of the projector is $\mathtt{R}$ and its centre is $-\mathtt{R}^{-1}\mathbf{t}$ [6, page 156]. If the 3D points are on a plane, such as the checkered surface onto which we project, that can be defined to be the $Z = 0$ plane. Then $\mathbf{X}$ has the form $(X, Y, 0, 1)^\top$ and points on the two 2D planes of the surface and the projector are related by a planar homography $\mathtt{H} = \mathtt{K}[\mathbf{r}_1, \mathbf{r}_2, \mathbf{t}]$ where $\mathbf{r}_1$ and $\mathbf{r}_2$ are the first two columns of $\mathtt{R}$. We can calculate $\mathtt{H}$ from point correspondences between the two planes [6, chapter 4], and $\mathtt{K}$ is known. We can therefore calculate $\mathtt{K}^{-1}\mathtt{H}$, normalize it so that the length of the first column is unity, then calculate $\mathtt{R}$ as $[\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_1 \times \mathbf{r}_2]$ since the rotation matrix $\mathtt{R}$ must be orthogonal, and obtain $\mathbf{t}$ as the last column of the normalized matrix. Given a set of projected points on the surface this allows us to calculate the pose of a calibrated projector in world co-ordinates, which are relative to the surface.

When a planar mirror used to direct the projector beam onto the surface the projection equation becomes,

$$\mathbf{x} = \mathtt{K}[\mathtt{R} \quad \mathbf{t}] \begin{bmatrix} I - 2\mathbf{n}\mathbf{n}^\top & 2d\mathbf{n} \\ 0 & 1 \end{bmatrix} \mathbf{X} \, , \tag{4}$$

where the extra matrix performs a reflection in some plane with normal vector $\mathbf{n}$ and distance from the origin d. We express this new projection matrix as $\mathtt{K}[\mathtt{R}' \quad \mathbf{t}']$ where $\mathtt{R}'$ can be shown to be orthogonal. From point correspondences between surface and projector we calculate $\mathtt{R}'$ and $\mathbf{t}'$ in the same way as $\mathtt{R}$ and $\mathbf{t}$ above. Now they give the position and orientation of the reflected projector, except that a change is necessary to $\mathtt{R}'$. We define the projector to use a right-handed co-ordinate system, so the reflected projector will have a left-handed system which cannot be represented by a rotation matrix. To obtain the actual basis vectors for the reflected projector pose we simply reverse the z-axis, that is, we negate the final column of $\mathtt{R}'$.

To calculate the reflected projector pose a set of point correspondences between projector and surface is required. The points in the projector are known, and those on the surface are obtained in the same way as for projector calibration (Section 4.2.) Because the surface and camera remain stationary just a single image is required to calculate the camera-to-surface homography, and this homography is also used to calculate the pose of the camera which is useful in the final calibration stage (Section 4.7) when the whole system of projector, mirror, surface, and camera is simulated to obtain predicted camera points to compare with the measured points. As θ and ϕ are varied the resultant projector-to-surface homographies are used to obtain a series of reflected projector poses.

4.4. Projector Pose

We have found that the calibration process is more stable if we obtain an estimate for the projector pose before calculating the mirror parameters, rather than attempting to find both simultaneously. Therefore, after obtaining the reflected projector poses in Section 4.3 we obtain an estimate for the real projector pose—the next six of the 21 steerable projector parameters. To find these parameters we perform an iterative minimization over projector positions:

$$\min_{\mathbf{p}}(\mathrm{err}_q(\mathbf{p})) \; . \tag{5}$$

Using Levenberg-Marquardt iteration we perform a minimization over 3D projector positions $\mathbf{p}$ with the error err_q for each position being calculated in the following way. First, for each reflected pose assume there is a mirror half way between $\mathbf{p}$ and the reflected position and reflect the orientation in that mirror. This results in a set of projector orientations around $\mathbf{p}$ (Figure 6) which in the ideal case, when there is no error in the data, will all be identical. We convert these projector orientations to quaternions and use the average distance from their mean as the error value err_q.

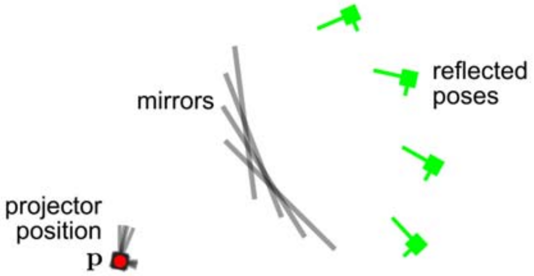


Figure 6. Iterative search for projector pose. The reflected poses are known and the error for a projector position $\mathbf{p}$ is calculated as the spread of the reflected orientations around that position.

As with any iterative minimization this method is sensitive to the starting position that is used, so we use some knowledge of the likely vicinity of the projector position in world co-ordinates to choose the set of randomized starting points. By observation of the steerable projector hardware and its position relative to the checkered surface that defines the world co-ordinate system we have created a simple function to roughly estimate the projector centre from the the reflected projector positions. This function would need to be updated if the hardware configuration was changed.

We use a randomized starting point for the iteration which is within about 150 mm of the true projector centre, and which is kept away from the reflected projector positions. We run the iterative minimization several times (currently 30 times) and perform clustering on the resultant positions to the reject outliers that sometimes occur when the iteration converges to a false minimum. The final projector orientation is then the mean of the reflected quaternions.

4.5. Rotation Using Theta

At this stage estimates of 14 of the full 21 parameters have been calculated, and the remaining seven are those of the pan-tilt mirror (Figure 2) excluding t_1 and t_2 which are given. We approach the estimation of these parameters in two stages: this section deals with subsets of the input data in which ϕ is kept constant to obtain a series of poses for the moving tilt axis, then those are used in next section in which the pose of the pan axis is calculated.

If ϕ is kept constant the tilt axis remains stationary and the mirror rotates about it as θ changes. This situation consists of a mirror rotating about a single axis in 3D which can be represented by a simplified version of equation 2 without the translation of t_2 or rotation of ϕ. An estimate for the projector pose has already been obtained and the value of t_1 is given. The aim is now to find the line about which the mirror rotates and the angular offset of that rotation. This process is repeated to get a pose in world co-ordinates for the tilt axis corresponding to each value of ϕ.

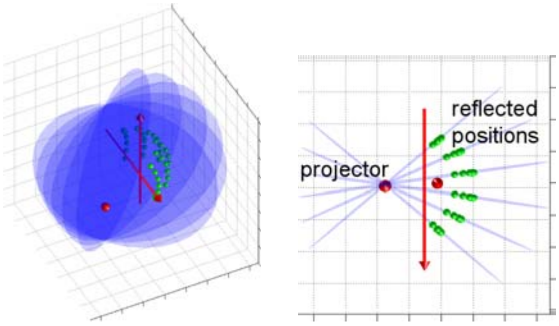


Figure 7. For a fixed value of ϕ the reflected projector positions lie in a plane. The pan and tilt axes (arrows) are shown in their home positions.

As θ changes and the mirror rotates about the tilt axis the reflected projector centre will appear at points in a plane which also contains the real projector centre (Figure 7), so we start by fitting a plane to a set of points containing the real and reflected projector centres. This plane will be defined by normal vector $\mathbf{n}$ and distance from the origin d. The direction of the tilt axis should be parallel to $\mathbf{n}$. We then create a virtual mirror between the projector and each reflected point as in Figure 8—each of these planes will be defined by normal vector $\mathbf{m}_i$ and distance d_i, where the index i selects each of the tilt angles θ_i in turn.

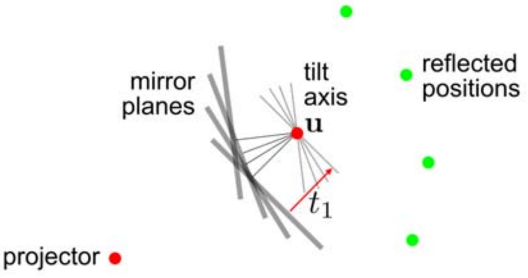


Figure 8. We calculate the position of the tilt axis, which passes through point $\mathbf{u}$, for a fixed pan angle ϕ.

Next we define a 3D point $\mathbf{u}$ through which the axis should pass. Mathematically a reflection transformation makes no distinction between points on the two sides of the reflection plane, but in reality the pan-tilt mirror is single-sided and the projector position must be in front of the mirror, not behind it. We therefore define $\mathbf{u}$ to be a distance t_1 behind the mirror, that is, on the opposite side from the projector position. In the perfect case $\mathbf{u}$ will be precisely t_1 behind each of the mirror planes in a direction perpendicular to $\mathbf{n}$, which is expressed as,

$$\frac{\mathbf{m}_i \cdot \mathbf{u} - d_2}{||\mathbf{n} \times \mathbf{m}_i||} = -t_1 \ . \tag{6}$$

This is linear in the three co-ordinates of $\mathbf{u}$ so we combine the constraints from the different values of θ and find the least-squares solution.

The full details of a 3D mirror system with a single axis of rotation have now been calculated so predicted positions in the camera of the projected corner points can be calculated. We iteratively refine the tilt axis position and direction using Levenberg-Marquardt iteration to minimize the sum of the Euclidean distances between predicted and measured corner points in the camera image.

4.6. Rotation Using Phi

The technique from Section 4.5 is used to get a position for the tilt axis for each value of ϕ. In the zero-error case these will lie in a plane, so we fit a plane to them and assume that pan axis is perpendicular to that plane. The aim is now to calculate the intersection point of the axis with the plane, and the angular offset of rotations about the axis. Figure 9 shows the situation. The pan angles ϕ_i are known, the distance t_2 is given, and the positions of the tilt axis were previously calculated. The aim is to find the three unknowns: the position of the pan axis $\mathbf{v}$ and the offset angle β. These are not explicit parameters of the mirror model (Section 3) but will be used to calculate the transform `W` that fixes the pose of the mirror system in world co-ordinates.

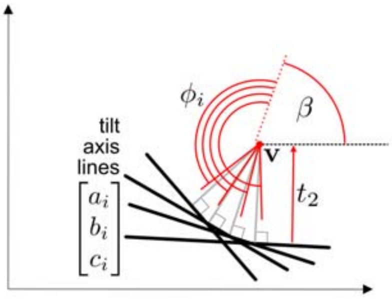


Figure 9. The tilt axis rotates in a plane around the pan axis $\mathbf{v}$ at a distance t_2 using angles ϕ_i with offset angle β.

For each pan angle ϕ_i there is a corresponding previously computed tilt axis line which can be expressed in 2D in the form $\mathbf{l}_i = (a_i, b_i, c_i)^\top$. We normalize so that $a_i^2 + b_i^2 = 1$, then $(a_i, b_i)^\top$ is the normal vector to the line [6]. In the perfect case each normal vector $(a_i, b_i)^\top$ will have an orientation angle equal to the fixed angular offset β, which is common to all the vectors, plus an angle ϕ_i. We estimate the offset angle as the mean difference between the ϕ_i values and the orientation angles of their corresponding normal vectors.

For a homogeneous 2D point $\mathbf{x}$ the signed distance from the line $\mathbf{l}_i$ is given by $\mathbf{l}_i^\top \mathbf{x}$, and for the pan axis position this distance should be t_2, so the constraint on the co-ordinates (v_x, v_y) of $\mathbf{v}$ is,

$$\begin{bmatrix} a_i & b_i & c_i \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ 1 \end{bmatrix} = -t_2 \; . \tag{7}$$

We stack these linear constraints from the set of lines and compute a least-squares solution for the axis position $\mathbf{v}$.

4.7. Final Solution

The final 7 parameter values are used to calculate the transform `W` (six values) and angular offset α (Section 3.) The results from the previous two sections (Sections 4.5 and 4.6) provide the pose in world co-ordinates of the pan axis and of the tilt axis for each value of ϕ. These are combined to calculate `W` which fixes the pose of the mirror system in world co-ordinates. Finally α is calculated from the results of Section 4.5.

A complete set of parameter values for the steerable projector has now been calculated and the camera's internal parameters (Section 4.2) and pose (Section 4.3) are known, so the whole system for projecting and capturing calibration patterns can be simulated to obtain a set of predicted points in the camera corresponding to any pair of θ and ϕ values. We run an iterative minimization over the mirror parameters and projector pose where the error metric is the sum of Euclidean distances between predicted and measured camera points. We use normalized camera points in the minimization process to avoid repeatedly applying radial and tangential distortion functions, but for the accuracy measurements below we have converted these distances into actual pixel measurements in the camera.

5. Performance

We used the algorithm described in Section 4 to calibrate our steerable projector. The projector we used is a Sanyo PLC-XP55 which has XGA resolution (1024×768 pixels) and outputs 4500 lumens, although we reduce the output by 50% during calibration to avoid saturating the camera. Our camera is a Nikon D1x with a resolution of 2000×1312 pixels. The actual setup is shown in Figure 1. The checkered surface pattern is printed on an A0 sheet of paper (approximately 4×3 feet) and the steerable projector is 1 metre from surface. We used six pan angles covering 30 degrees, and 6 tilt angles covering 10 degrees, giving a total of 36 pan-tilt positions.

The calibration procedure for camera and projector internal parameters from Section 4.2 resulted in a mean error for the camera of 0.25 camera pixels which is a typical value, and a mean error for the projector of 0.47 projector pixels which is quite accurate considering that errors in the camera calibration, the printed corner points, and the projected corner points will all adversely affect the projector calibration. The mean error for the calibration of the whole steerable projector system after iterative refinement was 9.4 camera pixels, which corresponds to approximately 8 millimetres on the surface.

We have found that maintaining reasonable focus of the projected image on the surface is important, particularly during the projector calibration stage when care must be taken to avoid the surface getting too close to the projector.

The calibration system is written in Matlab and we run it on a Pentium 4 3GHz PC. With 36 pan-tilt positions it takes around one minute to iteratively find the projector location using 30 runs of the method from Section 4.4. Because of the use of randomized starting positions this method produces varying results so we repeat it several times for higher accuracy. It then takes around 30 seconds to calculate the rest of the coarse result, and a further 1.5 minutes to obtain the refined result.

To test the algorithm on a range of input data we created a simulated calibration scenario similar to the real one used above, with randomized camera, projector, and mirror parameters. Camera points were generated corresponding to the corner points of the printed pattern and corresponding to the projected points for the set of pan-tilt positions. Uniform random noise of varying levels was added to the x and y co-ordinates of the camera points. We ran the simulation 10 times for each of 10 error levels from 0.0 to 0.9 camera pixels, and used two conditions: the one used in the experimental setup above with 30 degrees and 10 degrees for pan and tilt respectively and 36 positions, and one with 60 degrees and 20 degrees respectively and 64 pan-tilt positions. The results are shown in Figure 10.

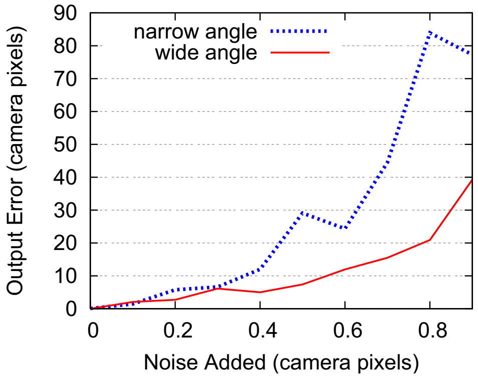


Figure 10. Error in the final calibration result against error added to input images. Narrow range is 30 and 10 degrees for pan and tilt, wide range is 60 and 20 degrees respectively.

6. Discussion

We used the narrower range of angles for the real calibration because it allows the set of projected patterns to fit within the A0 surface while also keeping the projector at a sufficient distance from the surface so focus changes across the surface are not too large. Figure 10 shows that using a larger range of angles and more mirror positions improves the accuracy of the calibration. The wider range would require a surface approximately 1.8×1.4 metres which is within the range of a normal office wall. Using our current method we would still need a homography from camera to surface, but rather than covering the whole wall with a checkered pattern this could be achieved by placing at least four cyan and white fiducial markers at known positions on the wall. Using more mirror positions would require more effort for capturing images so we would like to automate that process.

A calibrated pan-tilt system allows a steerable projector to project images in different directions, but we would also like to be able to change the zoom and focus of the projector to accommodate the size and distance of objects in the environment. Calibration for zoom and focus has been performed for cameras, essentially by calibrating the camera many times and treating each of the relevant internal and external parameters as a second-degree polynomial function of the zoom and focus settings, the parameters of which are obtained via a least-squares fitting [1]. The same approach should be possible for a projector with motorized zoom and focus, and as discussed above, it should be possible to automate the process so that the projector is calibrated for each zoom-focus setting without human assistance.

We have described a method of calibrating a steerable projector that uses a mirror to move a projected image around the environment. First we calibrate the camera and projector parameters using a board with a checkered pattern, then we use images of a projected pattern for different positions of the pan-tilt mirror to obtain a coarse calibration for the steerable projector which we then refine via iterative minimization. This technique could be combined with other techniques for camera and projector calibration to create a steerable projector that can augment surfaces and objects in a 3D environment without prior calibration to those targets, and follow moving objects.

Acknowledgments

This work was funded by postdoctoral fellowship from the Japan Society for the Promotion of Science.

References

[1] R. Atienza and A. Zelinsky. A Practical Zoom Camera Calibration Technique: An Application on Active Vision for Human-Robot Interaction. In *Proc. Australian Conference on Robotics and Automation*, pages 85–90, 2001.

[2] S. Borkowski, O. Riff, and J. L. Crowley. Projecting Rectified Images in an Augmented Environment. In *Procams 2003 Workshop on Projector-Camera Systems*, 2003.

[3] A. Butz, M. Schneider, and M. Spassova. SearchLight—A Lightweight Search Function for Pervasive Environments. In *Proc. Second Int. Conf. on Pervasive Computing (PERVASIVE 2004)*, pages 351–356, 2004.

[4] H. Chen, R. Sukthankar, G. Wallace, and K. Li. Scalable Alignment of Large-Format Multi-Projector Displays Using Camera Homography Trees. In *Proc. IEEE Visualization 2002*, pages 339–346, 2002.

[5] J. Ehnes, K. Hirota, and M. Hirose. Projected Augmentation - Augmented Reality using Rotatable Video Projectors. In *Proc. Third IEEE and ACM Int. Symp. on Mixed and Augmented Reality (ISMAR 2004)*, pages 26–35, 2004.

[6] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision 2nd edition*. Cambridge University Press, 2003.

[7] T. Okatani and K. Deguchi. Autocalibration of a Projector-Screen-Camera System: Theory and Algorithm for Screen-to-Camera Homography Estimation. In *Proc. of ICCV 2003*, pages 774–781, 2003.

[8] G. Pingali, C. Pinhanez, T. Levas, R. Kjeldsen, and M. Podlaseck. User-Following Displays. In *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME) 2002*, pages 845–848, 2002.

[9] C. Pinhanez. The Everywhere Displays Projector: A Device to Create Ubiquitous Graphical Interfaces. In *Proc. Ubiquitous Computing (UbiComp) 2001*, pages 315–331, 2001.

[10] A. Raij and M. Pollefeys. Auto-Calibration of Multi-Projector Display Walls. In *Proc. ICPR 2004*, pages 14–17, 2004.

[11] R. Raskar, P. Beardsley, J. van Baar, Y. Wang, P. Dietz, J. Lee, D. Leigh, and T. Willwacher. RFIG Lamps: Interacting with a Self-Describing World via Photosensing Wireless Tags and Projectors. In *Proc. Siggraph 2004*, pages 406–415, 2004.

[12] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. iLamps: Geometrically Aware and Self-Configuring Projectors. In *Proc. Siggraph 2003*, pages 809–818, 2003.

[13] N. Sukaviriya, M. Podlaseck, R. Kjeldsen, A. Levas, G. Pingali, and C. Pinhanez. Embedding Interactions in a Retail Store Environment: The Design and Lessons Learned. In *Proc. IFIP INTERACT 2003*, 2003.

[14] R. Sukthankar, R. G. Stockton, and M. D. Mullin. Smarter Presentations: Exploiting Homography in Camera-Projector Systems. In *Proc. ICCV 2001*, pages 247–253, 2001.

[15] Y. Tokuda, S. Iwasaki, Y. Sato, Y. Nakanishi, and H. Koike. Ubiquitous Display for Dynamically Changing Environments. In *Extended Abstracts of CHI 2003*, pages 976–977, 2003.

[16] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. S. Brown. PixelFlex: A Reconfigurable Multi-Projector Display System. In *Proc. IEEE Visualization 2001*, pages 167–174, 2001.