MONITORING AND CORRECTION OF GEOMETRIC DISTORTION IN PROJECTED DISPLAYS

R. Matt Steele
Stephen Webb
Christopher Jaynes

Department of Computer Science
University of Kentucky
Lexington, KY 40506
steelerm@netlab.uky.edu

ABSTRACT

Off-axis placement of light projectors induces significant planar parallax on the display surface. Although commodity solutions exist for removing this distortion, they involve iterative, menu-driven user interaction or physical alignment of the projector, and in either case interrupt the use of the display. User interaction is infeasible in a number of scenarios including mechanically aligned multi-projector displays that are subject to mechanical drift and situations in which projectors are often reconfigured.

We present a general technique for continuous rectification of arbitrary off-axis distortions that does not require user interaction. A camera automatically detects when the projector's orientation has changed, without requiring explicit fiducials or targets in the world. The method runs in concert with interactive display applications and has minimal impact on framerate. An initial rectifying transform is recovered automatically by projecting target points and observing them in the camera. The display is then warped and passively monitored for calibration error and motion of the projector. The technique distinguishes between distortions due to miscalibration and intentional framebuffer changes. A consistency score is measured by generating a predicted view based on the current framebuffer contents and correlating this prediction with the camera's captured image. Poor correlation scores indicate that the projector has moved and re-calibration and geometric correction is required. Initial experiments show that the calibration consistency measures are sufficiently robust to distinguish small motion of the projector from continuously changing imagery.

Keywords: display calibration, projection, homography

1 INTRODUCTION

The use of light projectors for information display is both widespread and common. In recent years, researchers have begun to explore the utility of light-projectors for a number of non-traditional tasks. These include augmented-reality applications [9], interactive displays [2, 1, 8], and groups of projectors that collectively render high-resolution immersive environments. In both traditional and experimental displays, the placement of projectors is somewhat constrained by the amount of display distortion that can be tolerated. Keystoning, the most familiar type of image distortion, results from tilting the projector in the vertical plane. Arbitrary placement of the projector induces a more general projective distortion. Projectors must either be placed so that the optic axis is perpendicular to the display surface plane or the display distortion must be corrected by modifying the optics or prewarping the framebuffer contents.

Although commercial solutions exist to correct for keystone and general planar projective distortions, they rely on user interaction to carefully adjust the projected image. These user-intensive solutions are typically menu-driven, involve iterative, subjective adjustment, and interrupt the normal use of the projected display. In multi-projector display environments, interactive refinement of the projected image is infeasible. In large multi-projector systems, automatic detection and removal of distortion is desirable, particularly if calibrated projectors are subject to mechanical drift or are periodically reconfigured.

We introduce a technique that requires no user interaction to detect and correct geometric distortion and performs continuous monitoring of the corrected image without interrupting the display. In the event that a projector is moved, the system automatically recalibrates and removes the resulting display distortion. A camera monitors the display by accurately predicting the appearance of the framebuffer from the camera's known view and comparing it to captured images.

Prior to display use, the intrinsic parameters of the projector and camera are recovered. In addition, the relative rotation between the projector and camera is recovered and fixed throughout the process. This assumption is reasonable because the camera can be rigidly attached to the projector that it is monitoring. Once initial calibration is complete, the system computes a projector to camera transform (a homography for the display geometry discussed here) that is then monitored and updated as necessary. This homography is computed by projecting known projector points and observing their corresponding camera pixels. Additionally a color transfer function is computed which provides a mapping from framebuffer pixel color values to image sensor color values.

Both the homography and the color transfer functions govern the predicted appearance of the projector framebuffer as seen from the camera. Predicted images can then be constructed while the display is in use for monitoring and correction of the display. A consistency measure is applied to captured and predicted camera images to determine when calibration is no longer correct.

Given the fixed rotation between the projector and camera as well as the homography between the two, a correcting transform for the display can be computed (See Section 2.1). Subsequent display of the framebuffer is pre-warped according to this transform. Figure 1 shows the geometric framework of the problem and a general setup of an experimental monitoring system.

Although the general monitoring and correction approach is applicable to arbitrary surfaces, the current technique assumes a planar surface and exploits this knowledge to simplify the pre-warp computation. Once calibrated and rectified, the system operates in a closed-loop mode, and passively monitors projected images with minimal impact on framerate. Projector movement is automatically detected using the consistency measure and the system automatically recalibrates the parameters that are required to derive a new correcting transform.

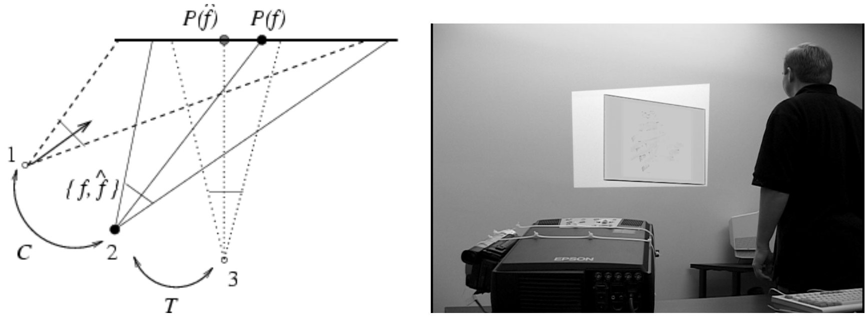
1.1 Previous Work

Several techniques have been introduced which address the problem of image distortions due to display surface geometry, projector position, and viewer position [1, 2, 10, 8]. With these approaches, the projectors, cameras, and user position are assumed to be known at all times, enabling the system to project appropriate images that do not appear distorted to the user. In particular the projector's position, once calibrated, is assumed to be fixed; changes in position result in unmodelled distortions in the projected image. In the case where multiple projectors are used to produce large-scale displays [1, 2, 7, 11], projection error as small as one pixel can produce noticeable artifacts in projector overlap regions.

Maintaining calibration for all of the devices in a projector-based display represents a significant challenge to wide-scale deployment of such systems. Mechanical solutions such as [2, 7] require skilled technicians to perform the calibration, and are subject to environmental (structural, vibrational) requirements [4]. Software-based approaches address these limitations, but still require user intervention to initiate and conduct system calibration. Our system, once calibrated, will detect calibration error due to mechanical drift, projector repositioning, and reorientation of the display surface.

In earlier work, [3] describes a technique for computing the correcting homography for a projected display. The approach requires that physical fiducials are placed in the environment allowing a camera to compute the warp that maps the four corners of the projected frustum to the physical targets. The method precludes arbitrary movement of the camera-projector pair (unless new fiducials are mounted) and does not address the problem of continuous monitoring and correction.

Continuous monitoring of an active display is a methodology that has been used by [4] to acquire a model of the display surface. A fixed



(a) (b)

Figure 1: General monitoring and correction setup. (a) A framebuffer **f** is projected to a planar surface from a projector (**2** in figure). A camera (**1** in figure) observes the distortion and computes a framebuffer warp $\hat{\mathbf{f}}$ that simulates a projector at position **3**,whose optic axis is parallel to the display surface normal. Subsequent images are then monitored from **1**. (b) Depiction of an experimental setup. Camera-projector rig (in foreground) and corrected display. Although the intersection of the projected frustum with the display surface forms is not rectangular (illuminated quadrilateral), the framebuffer has been correctly pre-warped and the displayed image is geometrically correct.

and calibrated camera detects changes in surface shape and reconstructs a model using projected imagery in a technique similar to that of structured light. By approaching the problem using an active monitoring approach, a consistent surface shape can be computed even as the display is in use. In contrast, our approach is intended to correct for distortions due to the relative positioning of the projector with respect to the display surface. The method detects changes in projector motion and/or display surface changes and corrects for both. This "display-sense-correct" framework runs with minimal degradation in the performance of the display application. The monitoring phase is transparent to the user, and recalibration occurs automatically without user interaction.

2 CORRECTION AND MONITORING FRAMEWORK

The projection function defined by the intrinsic and extrinsic parameters of a projection device, along with the display surface geometry, define a function P that maps the framebuffer f into points on the display surface, $P(f)$. This function is a warp related to the position of the projector with respect to the display surface, the display surface shape, and intrinsic properties of the projector. P is a projective distortion that must be removed in order to produce an image in the world that correctly reflects the contents of the framebuffer.

Distortions induced by P can be removed through a transform T that pre-warps the framebuffer f into $\hat{f}$ prior to projection. Subsequently projected images, $P(\hat{f})$, then, are geometrically correct. T is determined by observing projected points on the display surface from the camera and deriving a set of correspondences.

Once T has been computed and the framebuffer has been transformed accordingly, it is necessary to verify that the projected and warped image, $P(T(f))$, or $P(\hat{f})$ continues to satisfy our original geometric constraints. For example, the user may move a projector or calibration may degrade over time due to mechanical drift. Verification should not require modification to the framebuffer because the display may be in constant (and potentially critical) use. In addition, modification of the environment, such as physical placement of fiducials [3] is not necessary and infeasible in a number of different applications.

Because the display is actively changing, it is insufficient to simply compute a difference between successive camera images. Therefore, pixels in the framebuffer must be related to those in the camera through a mapping, C. The relationship C allows the system to predict what it should observe by relating pixels in the current framebuffer to their expected locations in a captured image, c. Figure 1a depicts this situation for the specific case in which the display surface is planar. Given an initial T and C, a display iteration involves building the contents of a framebuffer, f, (traditional rendering), applying the correcting warp, $T(f)$, to compute a new framebuffer $\hat{f}$, projecting the resulting buffer, $P(\hat{f})$, capturing an image of the display, c, and ultimately comparing c to $C(\hat{f})$.

A calibration consistency measure, $S = C(\hat{f}) \times c$, compares captured images to predicted images based on the current calibration estimate. If S falls below a threshold (0.9 for results shown here), we assume that the current estimate of T is wrong and the image is no longer being correctly unwarped. T is then recomputed by the system automatically and all corresponding warps (P and C) are updated and the system begins re-monitoring the newly corrected display.

In addition to the geometric warp C, a transfer function that models the color and intensity change between the devices is estimated. This transfer function, γ, is applied to points in $C(f)$ to more accurately predict the expected view of the framebuffer from the camera's location.

2.1 Geometric Specifics

Although the approach described in Section 2 is valid for arbitrary display surface shape, here a specific system is developed that exploits the extra constraints available for a planar display surface. This configuration guarantees that C is a homography, as is P, which ensures that any T required is also a homography. With these constraints, T can be computed from the intrinsics and relative orientation of the camera and the projector, and an extra constraint involving the vector that defines up for the image on the wall. T can be computed from the rotation between the projector position and a "virtual" projector whose optic axis is perpendicular to the display surface. This rotation is partly determined by the display surface normal expressed in projector coordinates. An extra constraint defining the orientation of the image on the wall fully defines the correcting rotation.

Given the three-dimensional coordinate frames of world, projector, and camera, their relationships can be described by a set of Euclidean rotations and translations.

$$\mathbf{x}_p = \mathbf{R}_{pw}\mathbf{x}_w + \mathbf{t}_{pw} \tag{1}$$

$$\mathbf{x}_c = \mathbf{R}_{cp}\mathbf{x}_p + \mathbf{t}_{cp} \tag{2}$$

Because the camera is fixed to the projector, $\mathbf{R}_{cp}$ is manually determined in advance through traditional calibration techniques [12]. However, the correcting rotation, $\mathbf{R}_{pw}$, that is used to ultimately recover T, is determined dynamically.

Given a planar display surface, T can be written as the homography from the world to the framebuffer, $\mathbf{H}_{pw}$. Likewise, C, the mapping from framebuffer to the camera's image plane can be written as $\mathbf{H}_{cp}$. Now, assuming that our homographies operate on points in normalized image coordinates, these 2D mappings are determined up to a scale factor by the three-dimensional configuration of the display[6],

$$\mathbf{H}_{pw} \propto \mathbf{R}_{pw} + \frac{\mathbf{t}_{pw}\mathbf{n}_w^T}{d} \tag{3}$$

$$\mathbf{H}_{cp} \propto \mathbf{R}_{cp} + \frac{\mathbf{t}_{cp}\mathbf{n}_p^T}{d} \tag{4}$$

where $\mathbf{n}_w$ and $\mathbf{n}_p$ denote the surface normal of the display surface in the world and projector coordinate frames respectively.

The surface normal of the display surface in the projector's coordinate system, $\mathbf{n}_p$, can be recovered from Equations 3 and 4. Assuming that $d = 1$ and the display plane is at $z = 0$ in the world coordinate system then, Equation 3 becomes:

$$\mathbf{H}_{pw} \propto \mathbf{R}_{pw} + \begin{bmatrix} 0 & 0 & t_{pw_x} \\ 0 & 0 & t_{pw_y} \\ 0 & 0 & t_{pw_z} \end{bmatrix} \tag{5}$$

The display plane's surface normal in projector coordinates, $\mathbf{n}_p$, defines a rotation which aligns the projector's z axis with the surface normal of the display plane. This rotation defines a family of homographies that correctly pre-warps the framebuffer. A final pre-warp, T, can then be selected from among these homographies.

Equation 4 determines $\mathbf{n}_p$. $\mathbf{H}_{cp}$ is estimated from four matchpoints between f and c. Because the rotation from the projector to the camera, $\mathbf{R}_{cp}$, is known (calibrated and fixed during an initial phase), we can compute the constant of proportionality in Equation 4.

$$\lambda\mathbf{H}_{cp} - \mathbf{R}_{cp} = \frac{\mathbf{t}_{cp}\mathbf{n}_p^T}{d} \tag{6}$$

Because the right hand side of Equation 6 is an outer product of two vectors, it has a rank of 1, and yields the following family of equations for λ, corresponding to the non-zero rows and columns of the right hand side.

$$\frac{\lambda H_{cp_{ij}} - R_{cp_{ij}}}{\lambda H_{cp_{il}} - R_{cp_{il}}} = \frac{\lambda H_{cp_{kj}} - R_{cp_{kj}}}{\lambda H_{cp_{kl}} - R_{cp_{kl}}} \qquad (7)$$

For these equations to be valid, at least one must involve components of $\mathbf{n}_p$ and $\mathbf{t}_{cp}$ that are non-zero. This means that the projector must not be normal to the display plane or parallel to it and the projector and camera cannot have nodal points in common.

Having computed λ, a surface normal in the projector frame, $\mathbf{n}_p = [n_{p_x} n_{p_y} n_{p_z}]^T$, can then be derived from Equation 6.

$$\frac{n_{p_x}}{n_{p_y}} = \frac{\lambda H_{cp_{i1}} - R_{cp_{i1}}}{\lambda H_{cp_{i2}} - R_{cp_{i2}}} \qquad (8)$$

$$\frac{n_{p_x}}{n_{p_z}} = \frac{\lambda H_{cp_{i1}} - R_{cp_{i1}}}{\lambda H_{cp_{i3}} - R_{cp_{i3}}} \qquad (9)$$

$\mathbf{n}_p$ is now used to construct a rotation matrix $\mathbf{R}_{pw}$, that aligns the projector's optic axis with the surface normal of the display. It is important to note that $\mathbf{n}_p$ is a normalized unit vector with only two free parameters and the specification of $\mathbf{R}_{pw}$ is under-determined. A third parameter must describe the orientation of the projected image on the display surface. In many cases, a user can simply provide this parameter by defining horizontal in the world coordinate system. However, for the system to remain fully automatic, horizontal in the world is defined by backprojecting the projector's x-axis to the display surface. This is nearly horizontal in the world given common projector setup. As the camera and projector rig rolls about its optic axis, however, the image on the wall will rotate as well.

Finally, T is constructed from $\mathbf{R}_{pw}$ and a translation and scale. Translation and scale are chosen so that pixels are mapped into valid framebuffer coordinates while simultaneously maximizing the number of projector pixels used. The corners of the framebuffer are warped using the inverse of T to produce a quadrilateral that is a representation of the frustum of the projector intersected with the display plane. The biggest box bounded by this quadrilateral that is square with the x and y axes of the framebuffer and preserves f's aspect ratio is computed. The vertices of this box are then associated to the corners of the framebuffer f to construct a new homography.

Figure 2a shows a pre-warped and rectified display while the system is in use. In this case, we colored the unused portion of the framebuffer white to show the true shape of the projected image. The goal of the approach is to recreate the largest rectified image that can be fit into the quadrilateral induced by the distortion. Figure 2b depicts the contents of the framebuffer after they have undergone a pre-warp to account for display distortion.

3 CONTINUOUS MONITORING

Given a correctly pre-warped and projected image, the goal of continuous monitoring is to detect changes in the display due to calibration error that arises from projector motion, changes in display surface orientation, or both.

A separate process runs in conjunction with the display loop and is responsible for camera control and image acquisition. This process captures an image of the camera's view of the display, and saves the corresponding pre-warped framebuffer contents, $\hat{f}$. It then uses C (in the case of a planar surface, C is the homography H_{cp}) to construct a predicted image, $C(\hat{f})$, that can be compared to the corresponding captured image c.

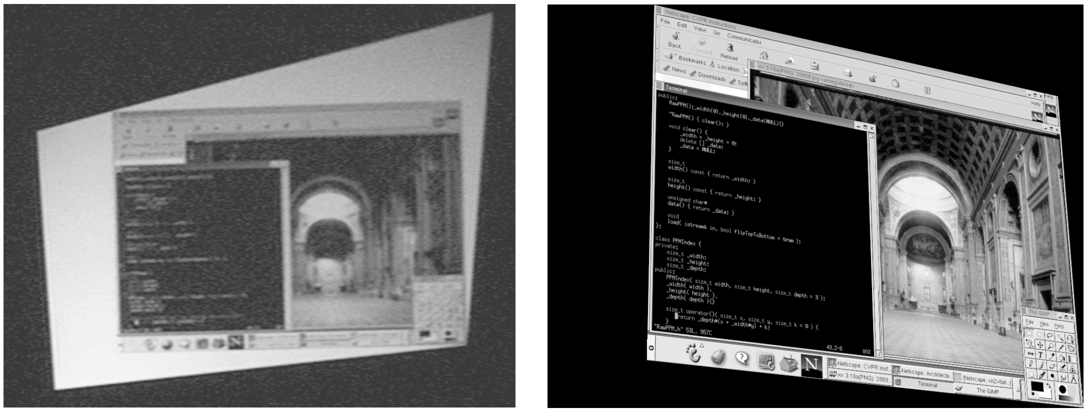
3.1 Color Calibration

C encodes the geometric mapping between observed pixels in the camera and rendered pixels in the framebuffer but does not account for radiometric effects. Although the predicted image should be a function of many factors including material properties of the display surface, currently, only the color/intensity bias between the devices is calibrated by the system.

For simplicity, the color of a point in c is modeled as a function only of the color of its corresponding point in $C(\hat{f})$. Furthermore, it is assumed that the three color channels do not interact, giving three color functions that are mutually independent. Color transfer curves for each color component are modeled by [8]:

$$\gamma(I(i,j)) = \frac{A}{1 + e^{a(I(i,j)+b)}} + k \qquad (10)$$

In order to determine the free parameters of Equation 10, four shades of gray, including white and black, are displayed during the calibration process. By splitting each shade of gray into its component colors, we determine the above parameters for each of the three transfer functions.



(a) (b)

Figure 2: (a) A photograph of the display while it is in the monitoring phase. Although the projector's frustum is warped onto the display surface, the framebuffer contents are correctly rectified. (b)The framebuffer contents in (a).

3.2 Detecting calibration error

The similarity measure that detects when T is no longer valid is a correlation function that operates on pairs of corresponding values in the predicted and color corrected image, $\gamma(C(\hat{f}))$, and the captured image, c. The three color channels of each pixel are correlated separately and summed to compute an overall correlation score.

Although the standard normalized cross correlation is capable of correctly matching images features with sufficient structure, it may yield undesirably high values in featureless image regions. We cannot guarantee a priori what either $\hat{f}$ or c will contain and must develop a correlation method that performs well for somewhat arbitrary images and does not require explicit search of $\hat{f}$ for good features to match.

Consequently, we normalize our correlation function differently:

$$ncc' = \frac{2\sum_i x_i y_i}{\sum_i x_i^2 + \sum_i y_i^2} \tag{11}$$

An overall consistency measure is derived through randomly sampling k points in $\gamma(C(\hat{f}))$ and correlating them with their corresponding points in the observed image c. For each pair, a small mask of size NxN from $\gamma(C(\hat{f}))$ is correlated over a MxM neighborhood in c. M is related to the expected geometric error of calibration and for the results shown was fixed to be 3 pixels. The maximum correlation in this neighborhood is taken to be the correlation score for that pair. A final consistency measure, then, is the average of the pairwise correlations of the random samplings:

$$S = \sum_{i=1}^{k} \frac{ncc'(X)}{k} \tag{12}$$

where X is a random variable that returns a pixel in $\gamma(C(\hat{f}))$.

Geometric error is not the only factor that influences the correlation values. Given the same geometric error, images with high contrast and high frequency components tend to have lower correlation scores than images without these features. In addition to the current contents of the framebuffer, ambient lighting contributes to these effects. For example, high levels of background light reduce the overall contrast of the projected imagery. All of these factors degrade the robustness of the consistency measure. Search for a maximum correlation score over the MxM neighborhood alleviates these effects somewhat.

4 EXPERIMENTAL RESULTS

A single projector and camera system based on the techniques discussed in this paper was developed and used to study the approach empirically. The initial relative position of the camera with respect to the projector was computed and remained fixed throughout these experiments. Once initially calibrated, the projector/camera pair is placed at an arbitrary position with respect to a planar display surface and the monitoring system is initiated.

A stand-alone application that loads an openGL program then runs in the foreground on the personal computer. The openGL display loop functions normally except that the framebuffer is pre-warped by the recovered T. Every two seconds the framebuffer is sampled and a consistency measure, using fifty randomly selected 3x3 templates, is computed according to Equation 12 on the color corrected values. If this exceeds a predefined threshold (0.90 for the results shown here), the system automatically re-initiates the calibration phase and new warping functions and the color transfer curves are recomputed.

4.1 Error Detection Latency

Given an incorrectly pre-warped image, the time delay before the system correctly detects and begins to correct the display depends on two things: the time length of one iteration of the monitoring loop, and the probability that a good correlation score is returned despite the error. The display loop is not significantly affected by the monitoring system as it executes asynchronously. However, the monitoring system itself requires approximately 0.5 seconds to construct and correlate templates in order to derive a consistency score. In future work, we expect that much of this can be optimized away– we spend a significant amount of time copying the framebuffer into main memory.

The number of iterations before detection depends on the image being displayed, ambient light conditions which affect the overall image contrast, as well as the configured correlation parameters. Experimentation shows that, on the average, calibration error is detected in a single iteration, corresponding to a mean time-to-correction of 2.5 seconds.

4.2 Geometric Accuracy

Geometric accuracy of the displayed image can be measured directly on the display surface. Given that the displayed image should be a rectangle with known aspect ratio, accuracy of the image is reflected in the relative lengths of the opposite sides of the image. Measurements indicate a geometric error of approximately 3% for a range of configurations in which the camera to projector baseline was moved from 0.2m to 3m.

In future work, we are studying the accuracy of the resulting image as a function of the relative orientation of the projector to the display and differing configurations of the camera and projector pair.

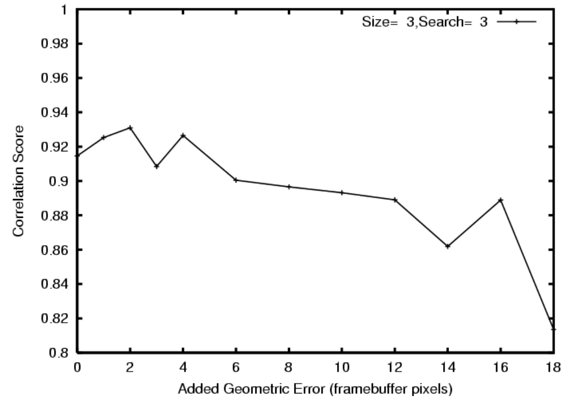


Figure 3: Correlation scores for various configurations of the monitoring loop. Each data point in this figure represents an average of trials taken at a random sequence of 50 points in the framebuffer.

4.3 Correlation Robustness

Varying amounts of geometric error under constant lighting conditions were simulated by performing a vertical shift in framebuffer coordinates. The calibration score as a function of simulated error is shown in Figure 3. The results suggest that under these conditions a single threshold can be selected to initiate re-calibration in the presence of error while avoiding unnecessary re-calibrations. It should be noted that correlation is a function of the image being displayed and that high correlation scores, in general, correspond to images more likely to appear perceptually correct to a human observer. As an extreme example, consider a constant intensity image.

It is clear that color correction improves system robustness. Figure 4 plots the standard deviation of the correlation values of the system with and without the benefit of the color transfer functions. Color correction enables tighter threshold values to be set and consequently allows the system to recalibrate in the presence of smaller error with less risk of spurious re-calibration.

5 CONCLUSIONS

We have presented a technique that automatically maintains geometric calibration in a projector-based display. Previous approaches that have addressed the problem of display distortion, whether mechanically or electronically, assume that the system remains calibrated. Experience from [1, 2, 7], however, indicates that these systems are fragile. Vibrations common to most environments (ventilation systems, doors closing,

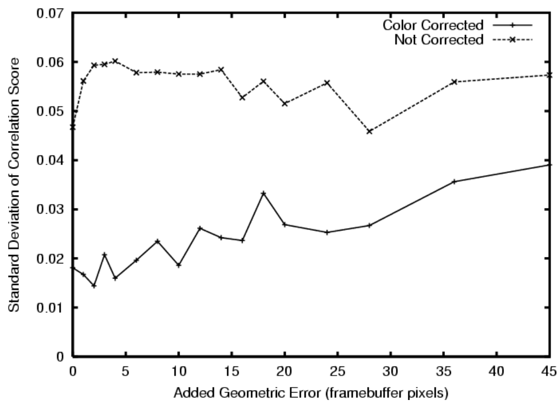


Figure 4: Standard deviation of color corrected versus non-color corrected correlations. Both of these data sets were gathered with 3 by 3 correlation regions and 3 by 3 search regions.

etc.) can cause display mis-calibration. By incorporating a camera into the display system, we now continuously monitor the state of calibration, and automatically re-calibrate without user interaction.

Error in the projected image is largely governed by error in the relative calibration between the projector and camera. Another contributor to the error is radial distortion in the projector. The radial distortion in our projector causes an estimated maximum error of two pixels. We are exploring methods to improve overall calibration by addressing both of these issues.

In addition, we are extending the "display-sense-correct" loop to multi-projector and multi-camera systems in which one or more of the devices may become miscalibrated. The general framework presented here is also being extended to continuously monitor and correct for radiometric error across multiple projectors.

REFERENCES

[1] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, H. Fuchs " The Office of the Future: A Unified Approach to Image-Based Modeling and Spatially Immersive Displays", University of North Carolina at Chapel Hill, SIGGRAPH Conference Proceedings, 1998.

[2] C. Cruz-Neira, D. Sandin, T. DeFanti " Surround-Screen Projection-Based Virtual Reality: The Design and Implmentation of the CAVE", Comptuer Graphics, SIGGRAPH Annual Conference Proceedings, 1993.

[3] R. Sukthankar, R. Stockton, M. Mullin " Automatic Keystone Correction for Camera-assisted Presentation Interfaces", Proceedings of ICMI, 2000.

[4] R. Yang, G. Welch " Automatic Projector Display Surface Estimation Using Every-Day Imagery", Conference in Central Europe on Comptuer Graphics, Visualization and Computer Vision, 2001

[5] W. Chen, H. Towles, L. Nyland, G. Welch, and H. Fuchs, " Toward a Compelling Sensation of Telepresence: Demonstrating a portal to a distant (static) office", presented at 11th Ann. IEEE Visualization Conference (Vis) 2000, 2000

[6] O. Faugeras, *Three-Dimensional Computer Vision.* London, England: The MIT Press, 1993.

[7] M. Herald, I. R. Judson, and R. L. Stevens, " Introduction to Building Projector-Based Tiled Display Systems," IEEE Comptuer Graphics and Applications, pp. 22-29, 2000.

[8] C. Jaynes, S. Webb, R. M. Steele, M. Brown, and B. Seales. "Dynamic Shadow Removal from Front-Projection Displays", *Proceedings of IEEE Visualization*, San Diego, 2001.

[9] C. Pinhanez, 'Augmenting Reality with Projected Interactive Displays", *Proc. International Symposium on Virutal and Augmented Architecture*, Dublin, Ireland. pp. 93-100, 2001.

[10] R. Raskar, "Oblique Projector Rendering on Planar Surface for a Tracked User," University of North Carolina at Chapel Hill, Chapel Hill, North Carolina 1997.

[11] R. Surati " Scalable Self-Calibrating Display Technology for Seamless Large-Scale Displays," : Massachusetts Institute of Technology, 1999.

[12] E. Trucco and A. Verri, *Introductory Techniques for Three-Dimensional Computer Vision*, Prentice-Hall, 1998.