

Ad-Hoc Multi-planar Projector Displays

Kashyap Paidimarri and Sharat Chandran

Indian Institute of Technology Bombay
<http://www.cse.iitb.ac.in/~{kashyapp,sharat}>

Abstract. High-resolution portable projectors have become commodity items now to own – but not to use. It is not always possible to find a display area where the camera can be properly aligned so that an undistorted image be seen. We present a method to project an undistorted image using a digital projector on a piecewise-planar display area.

We use uncalibrated structured light ranging to segment the unknown projection area and further compute the homographies that map the projector space to the camera space through each of the planes. The edge detection and point-correspondences are subpixel precise. Finally, we use these computed homographies to pre-warp the display image so that a distortion-free image is visible. Our results show a seamless and correct rectification with accurate segmentation of the planes.

1 Introduction

Increasingly digital projectors have become commonplace. But using a projector requires that a good display area be available. This is not always the case. The costs and maintenance, not to speak of the inconvenience suggests that “it would be nice to project on an arbitrary wall or desk corner.” Manual calibration of such a configuration is painstaking and error-prone, and give less than desirable



Fig. 1. Cluttered working environment precludes casual use of projectors. In this figure, the projection polygon has been demarcated for clarity. All images in this paper are best seen in color on a monitor.

experience. Fig. 1 shows a cluttered office environment where there is no available area of projection apart from the wall corner. In such an environment it is imperative that a method be devised to remove the distortion so that the user sees a rectangular and correct image.

1.1 Projector Camera Systems

Of late, there have been a variety of solutions to this problem and most often, a camera is used to provide visual feedback to the system (hence the field, called ‘Projector Camera systems.’). These systems range from a simple one-projector one-camera configuration to clusters of projectors and cameras intended to enlarge the available display area. The primary ideas in this field come from multiple view geometry of cameras [1]. An LCD (or DLP) projector, unlike older CRT projectors, can be modeled as a pinhole camera. Thus the projector becomes a dual of the camera and hence, both the capture and the display process can be represented as perspective transforms.

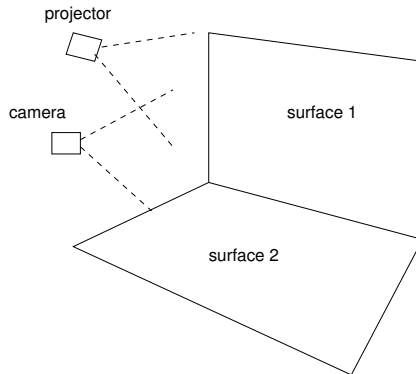


Fig. 2. Projector-Camera systems

Problem Definition: The primary components of such a system are a digital projector, an arbitrary display surface, and a digital camera. We are working with a model of the system where we have a single projector and camera projecting on a display surface formed from the juxtaposition of two or more planes as in Fig. 2. A minimal assumption is that the region of interest is covered in the field of view of both the camera and the projector.

Related Work: Given a calibrated projector-camera pair, conventional structure-light ranging can be used to reconstruct a complete 3D model of the display environment [2,3]. However, such methods are overkill because recovery of the complete geometry of the scene, the projector, and the camera is not necessary for rectification. In [4], the projection area is divided into a rectangular mesh and planar homographies are calculated from each rectangle of projection

to the display. These homographies are then stitched together and the image is pre-warped to achieve the correction. This method is applicable to any arbitrary continuous surface. A similar method [5] uses affine transforms rather than homographies followed by texture-mapping. However, these methods suffer from artifacts at sharp edges because they do not explicitly identify the planes. The work most resembling ours appears in [6]; planar homographies are estimated from projected points and metric rectification is achieved under the constraint that the display surface contains a rectangle of known aspect ratio whose boundaries are visible.

This paper: In this work, we present a method to automatically correct an ad-hoc projector display using a camera. Correction is achieved in the sense that the camera position is a ‘sweet spot’ from where the display is guaranteed to be geometrically identical to what was intended to be displayed by the projector. Regions around the camera center look acceptable (see Fig. 9). Our method can be combined with a head tracking system to further correct the display to any new position the user might move to. The techniques we contribute in putting together a working system include:

- Automatic segmentation of planar regions. This is the key challenge in ad-hoc projector usage.
- Use of lines as features instead of points or blobs in the intermediate process of homography calculation, and the use of point-line duality. The advantage is that lines are easier to detect and to a greater accuracy using voting mechanism. Thus, our system is tolerant to clutter even on the display surface (see, for example, the white board in Fig. 1).

The rest of this paper is organized as follows. Sec. 2 briefly reviews the theory of projective transforms and their application in projector-camera systems. The system design is presented in Sec. 3 followed by implementation details and results in chapter 4. We conclude in the last section.

2 Homographies for Projector Camera Systems

At the very heart of projector camera systems are planar *homographies* or *projective transforms*. A mapping $h : \mathbb{P}^2 \rightarrow \mathbb{P}^2$ is a *projectivity* [1] iff there exists a non-singular 3×3 matrix H such that for any point in \mathbb{P}^2 represented by a vector \mathbf{x} , $h(\mathbf{x}) = H\mathbf{x}$. \mathbb{P}^2 is the projective plane and points are represented in their homogeneous 3-vector representation. Projectivities form a group since the inverse of a projectivity is also a projectivity and so is the composition of two projectivities. Since in homogeneous coordinates lines are the dual of points, for $\mathbf{l} \in \mathbb{P}$ we have $\mathbf{l}' \in \mathbb{P}'$

$$\mathbf{l}' = H^{-T}\mathbf{l}$$

where we have been given a projectivity H from plane \mathbb{P} to \mathbb{P}' , and points on the line related by $\mathbf{x}' = H\mathbf{x}$.

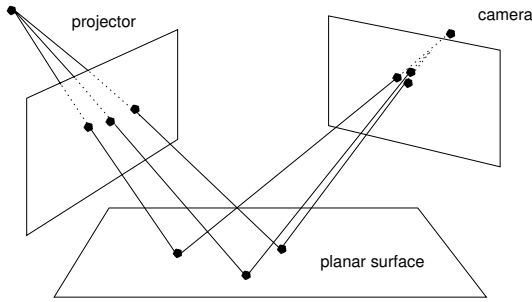


Fig. 3. Projective transform from projector to camera through a plane

Since the camera is the element that can observe images that the projector generates, albeit through the screen, it is critical to compute the homographies between these planes. A homography $_pH_s$ maps points from the projector plane to the planar display surface, and another homography $_sH_c$ maps from the surface to the camera plane. Composing these two we get the transformation from projector to camera $_pH_c = _pH_s \circ _sH_c$. Thus a point x in the projector plane P is transformed to point x' in the camera plane C . Finally $_cH_p$ is used to pre-warp the input image

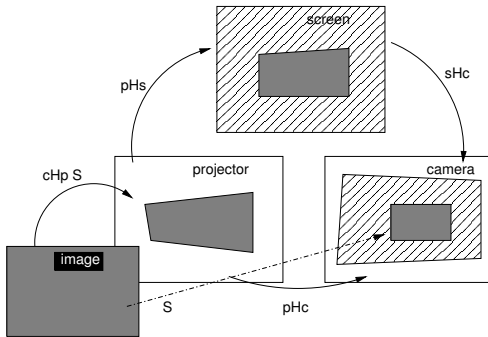


Fig. 4. Projective transforms between the camera, projector and the screen. Correction happens by applying an inverse map to the projected image.

from the application before projection. The image should now appear rectilinear in the final camera image except for the scaled rigid-body transform S that preserves shape. It is necessary to apply a correct S to ensure that the pre-warped image sent to the projector is not clipped to the projector’s resolution.

3 System Design

We now present our system. For simplicity, we assume in the description below that we are dealing with two planar surfaces, and divide the steps into three stages.

Structured light: This is the data-collection stage where we use uncalibrated structured light in the form of lines to identify kinks caused by the intersection of the planes. Further, we also identify the line segments that are formed due to multiple planar regions.

Homography: We determine the line of separation between planes, followed by a determination of the partitioning of the data. Finally point correspondences are evaluated from the data and the projective transforms from the projector plane to the camera plane through each of the surfaces of projection are found.

Image Correction: The partitioning edge is now identified in the projector space (rather than the camera space) by using each of the homographies. Then appropriate inverse homographies are applied to each partitioned region of the projected image. Finally a stitching is applied.

We now describe details of our system.

3.1 Structured Light

Our structured light registration is run in two phases with different angles for each of the phases. Common choice of angles for the two phases are 0° and 90° . For each angle we iteratively create lines and scan the entire projection space. The corresponding camera images are retrieved and a kink-detection algorithm is run to find the kink introduced in the line due to the wall corner. This facilitates the breaking of the input lines into line subsegments.

Fig. 5 shows a captured image during the structured light registration phase. A white line is projected against black background and an image of the display area is captured by the camera. The following steps are needed to identify the kink and the individual segments.

Binarize the image: Convert the image to gray-scale and apply a fixed threshold to segment the white projected line from the background. We have seen that this is quite robust to general ambient lighting. Adaptive thresholds and other techniques from the image processing literature can be used to counter the effect of unpredictable lighting.

Line fitting: We use the Hough transform to fit lines to the binary image. The Hough transform returns all identified lines and we discard lines that lie too close to each other. Effectively, this ignores clutter on the display area. The standard Hough transform however identifies only the lines and not the line-segments and further processing is required.

Kink detection: The detected lines are now intersected to find the kink. We then sample points in the original image along each line on opposite directions of the detected kink to partition the line.

Using lines over points/blobs has the advantage that lines are easier to detect and to a greater accuracy. Blobs tend to get distorted due to perspective distortion.

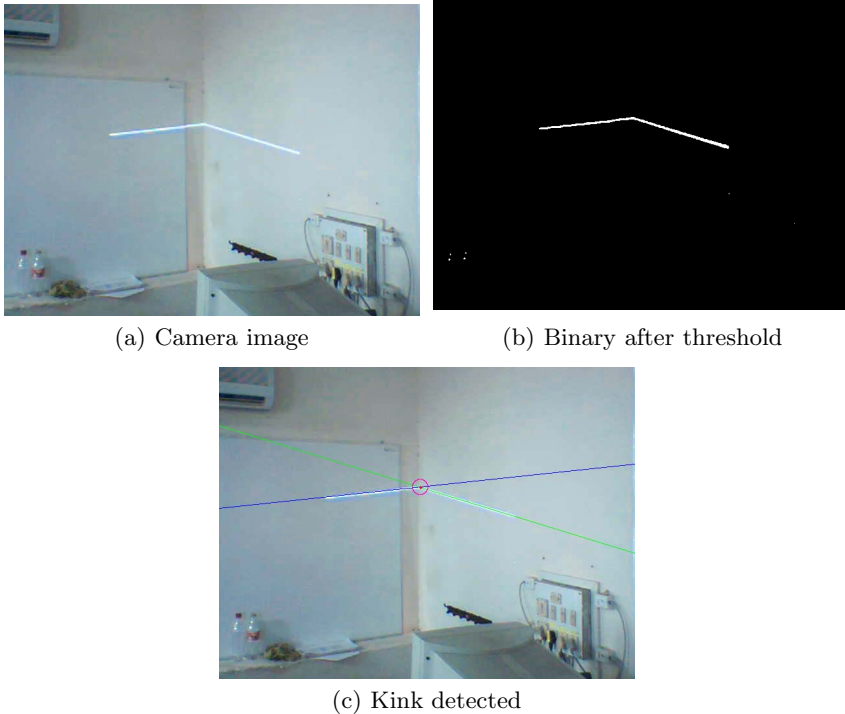


Fig. 5. Kink detection in the structured light image

3.2 Point Correspondences and Homographies

Recall that at this stage, we have individual kinks but no line of separation. We also have false positives in the kinks. We do not know which points belong to which plane and their relationship.

Line fitting: The structured light registration returns the detected kinks and the corresponding lines and line-segments in the captured camera image. Least-squares line-fitting is applied on the detected kinks and an initial estimate is found. This is further refined by removing outliers from among the detected kinks by measuring the distance to the fitted line from each kink. Experiments have proven this to be quite robust because the data tends to have low noise except for a few random outliers. Fig. 6 shows a line fitted to the detected kink points after outliers have been removed. The edge detection is highly accurate with sub-pixel precision and this shows in the results where even a low-resolution camera is sufficient to correct a high-resolution display.

Partitioning the segments: Now that the intersection of the two planes has been identified in the camera image, the previously detected segments can be partitioned onto the each of the planes. Lines that haven't been broken into

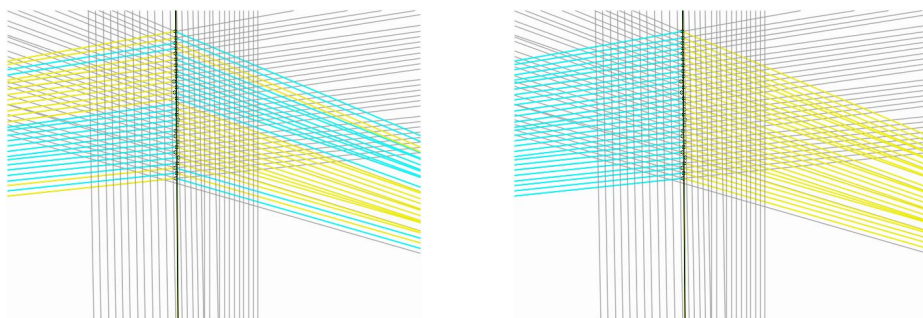


Fig. 6. (a) Fitting a line to the detected kinks, and (b) after cleaning up and reordering

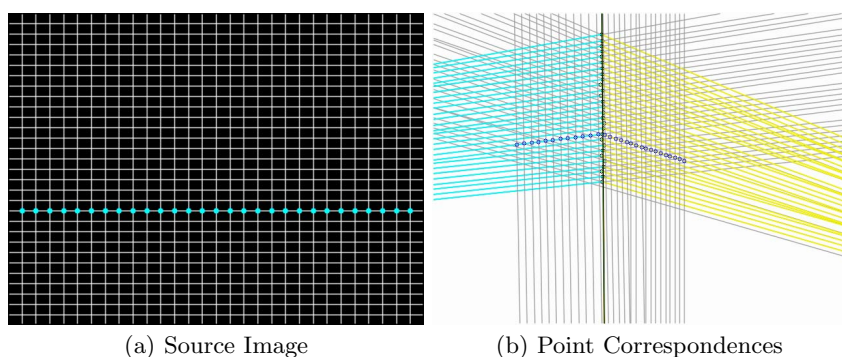


Fig. 7. Point correspondences between projector and camera image

segments because they do not cross the edge are left as such. Fig. 6 shows the plotted lines color coded to identify the partition they are in.

Correspondence: The structured light registration was run at two orthogonal angles. We now find pairwise intersections between lines from each of the first and second phase respectively. Points of intersection in the source image are unambiguous, however the calculated point of intersection in the camera captured image, needs to be again tested for sanity. The point correspondences are also partitioned onto each of the planes depending on where they occur in the captured image. Fig. 7 shows a row of corresponding points between the projected image and the captured data.

Partitioning the image space: We now use the point correspondence data for each of the planes to evaluate a homography from the projector to the camera. Homographies ${}^1_p H_c$ and ${}^2_p H_c$ map points on the projector plane to the camera plane through each of the two planes of the display surface respectively. An important property of the homographies is that they must agree on the common

intersecting lines of two planes, which was previously found. If L_c is the partitioning line in the camera image, then

$$L_p = {}^1_p H_c^{-T} L_c = {}^2_p H_c^{-T} L_c$$

3.3 Image Correction

We need to compute the inverse transformations corresponding to ${}^1_p H_c$ and ${}^2_p H_c$ to pre-warp the input image. Let us denote ${}^1_c H_p$ and ${}^2_c H_p$ as the corresponding inverse transforms. Now given an input image I , the warp H gives an image P to be projected, where $P(H * x) = I(x)$. However, the standard method for applying a warp is to once again find the inverse of the warping transform and use a linear (or bilinear or cubic) filtering to fill pixels in the destination image. For each pixel x' in the target image P , $P(x') = I(H^{-1} * x')$.

In our method, it reduces to using ${}^1_p H_c$ and ${}^2_p H_c$ for the warping. We thus generate images P_1 and P_2 from image I where $P_1(x') = I({}^1_p H_c * x')$, and $P_2(x') = I({}^2_p H_c * x')$.

All that remains is to composite together the two warped images at the partitioning line to generate the final image to be projected. In order to do this we generate a mask M where $M(x) = 1$, if x lies on the first plane in the projector coordinate frame; else $M(x) = 0$. So, in order to generate the final image P we do a masked copy of P_1 and P_2 using the mask M .

$$P(x') = P_1(x'), \text{ if } M(x') = 1 \quad P(x') = P_2(x'), \text{ if } M(x') = 0.$$

4 Implementation and Results

We have implemented the system in C++ using the OpenCV [7] libraries. OpenCV is a cross-platform library targeted towards work in computer vision and image processing. The following are the details of the hardware and the software:

Projector. Sharp XR-1S DLP projector with a native resolution of 800x600.

Camera. A 'Creative Live!' webcam with a maximum resolution of 640x480.

Note that it does not match the projector resolution.

Platform. Linux 2.6 with Video4Linux, spca5xx usb-webcam drivers and ffmpeg. We have also tested the system on Microsoft Windows XP.

4.1 Results

Fig. 8 shows the image projected from the projector and that which is seen by the camera. The camera image is rectilinear with complete agreement at the seam of the two planes (seen in magenta). The seamlessness is a measure of the correctness of the edge-detection method and the homography estimation because the partitioning edge in the projector image can only be inferred using the two computed homographies. When the two homographies do not agree, we see a pair of lines instead of one. In addition the grid lines running across the two planes stay truly

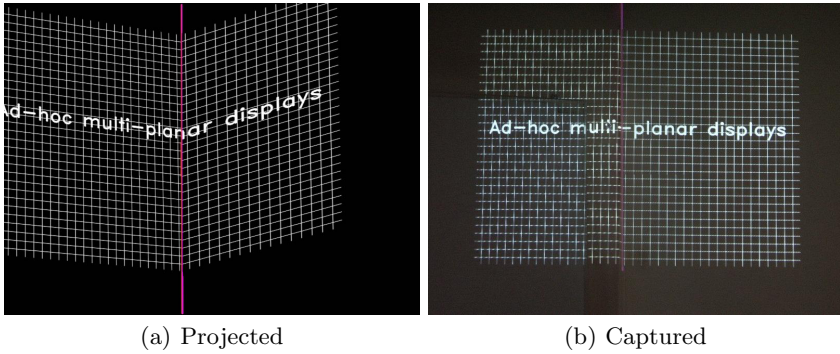


Fig. 8. The projected image and the image as seen from the camera



Fig. 9. The corrected scene



Fig. 10. Video excerpts on the wall corner

parallel. We have also demonstrated the system with video (Fig. 10) as input. Our experiments indicate a high degree of robustness and accuracy.

For completeness, we show (see Fig. 9) the motivating example of Fig. 1. The wall corner is now more usable as a projection area and for watching videos!

5 Conclusion

We have developed and implemented a method to display distortion free rectilinear images on wall corners and other multi-planar display surfaces. Results show high accuracy and the method is robust to environmental factors apart from high ambient lighting. A low-end usb-webcam is sufficient to correct a high-resolution display and previously unusable display room areas now become usable as display surfaces.

References

1. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press (2000)
2. Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L., Fuchs, H.: The office of the future: a unified approach to image-based modeling and spatially immersive displays. In: 25th annual conference on Computer graphics and interactive techniques. (1998) 179–188
3. Raskar, R., Beardsley, P.: A self-correcting projector. In: *Computer Vision and Pattern Recognition*. Volume 2. (2001) 504–508
4. Heda, N.: *Correcting projector display for multi planar surfaces*. Master’s thesis, Indian Institute of Technology, Bombay (2005)
5. Raskar, R., van Baar, J., Beardsley, P., Willwacher, T., Rao, S., Forlines, C.: iLamps: geometrically aware and self-configuring projectors. In: *ACM Trans. Graph.* Volume 22. (2003) 809–818
6. Sukthankar, R., Stockton, R., Mullin, M.: Smarter presentations: Exploiting homography in camera-projector systems. In: *International Conference on Computer Vision*. Volume 1. (2001) 247–253
7. Intel: *OpenCV: Open Source Computer Vision Library*. www.intel.com/research/mrl/research/opencv (2004)