DOI: 10.1111/j.1467-8659.2011.01965.x
Eurographics / IEEE Symposium on Visualization 2011 (EuroVis 2011)
H. Hauser, H. Pfister, and J. J. van Wijk
(Guest Editors)

Volume 30 (*2011*), *Number 3*

Automatic Registration of Multi-Projector Domes Using a Single Uncalibrated Camera

B. Sajadi[1] and A. Majumder[1]

[1]University of California, Irvine

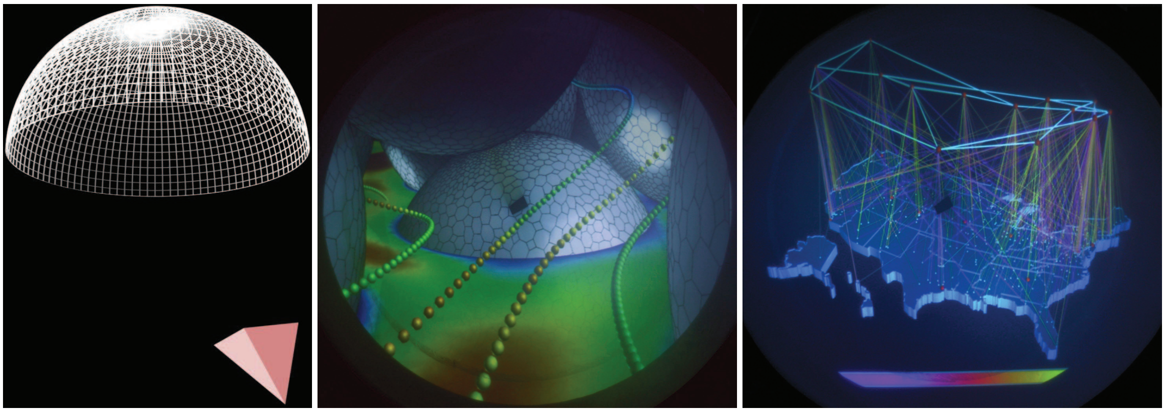

Figure 1: *Left: Reconstruction of our dome geometry and the camera parameters for our 4 projector front projection dome; Center: Our dome with view-dependent registration for flow simulation; Right: Our dome with view-independent registration of a visualization of early internet backbone, using orthographic map projection techniques.*

Abstract

In this paper we present a novel technique for easily calibrating multiple casually aligned projectors on spherical domes using a single uncalibrated camera. Using the prior knowledge of the display surface being a dome, we can estimate the camera intrinsic and extrinsic parameters and the projector to display surface correspondences automatically using a set of images. These images include the image of the dome itself and a projected pattern from each projector. Using these correspondences we can register images from the multiple projectors on the dome. Further, we can register displays which are not entirely visible in a single camera view using multiple pan and tilted views of an uncalibrated camera making our method suitable for displays of different size and resolution. We can register images from any arbitrary viewpoint making it appropriate for a single head-tracked user in a 3D visualization system. Also, we can use several cartographic mapping techniques to register images in a manner that is appropriate for multi-user visualization.

Domes are known to produce a tremendous sense of immersion and presence in visualization systems. Yet, till date, there exists no easy way to register multiple projectors on a dome to create a high-resolution realistic visualizations. To the best of our knowledge, this is the first method that can achieve accurate geometric registration of multiple projectors on a dome simply and automatically using a single uncalibrated camera.

Categories and Subject Descriptors (according to ACM CCS): I.3.7.g [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality

Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and
350 Main Street, Malden, MA 02148, USA.

1. Introduction

Domes can create a tremendous sense of immersion and presence and hence are becoming popular in many edutainment and visualization applications, including planetariums and museums. Tiling multiple projectors on domes is a common way to increase their resolution. However, the challenge lies in registering the images from the multiple projectors in a seamless fashion to create one seamless display. Though current proprietary software solutions allow manual or semi-automatic ways to achieve this, these are complicated procedures involving calibrated stereo cameras and 3D depth reconstruction facilitated via many physical fiducials. As a result, they are time consuming, difficult to maintain and deploy. Hence, such environments are still quarantined to niche super-expensive entertainment applications when they should be easily accessible to regular commonplace venues like schools, universities and public malls or marketplaces.

In this paper we present a novel technique to register images from multiple projectors on a spherical dome using a single uncalibrated camera. We avoid using calibrated stereo cameras and complex 3D reconstruction thereof by using the prior knowledge that the display surface is a dome. We use an image of the dome itself and a projected pattern from each projector to reconstruct the camera properties (both intrinsic and extrinsic) via a non-linear optimization. Since spherical domes are rotationally symmetric we need to use a single physical fiducial to define a unique coordinate system for the dome. When the whole display can not be seen in a single camera view or the resolution of the display is much higher than the resolution of the camera, our method allows using multiple pan and tilted views of the uncalibrated camera to register the display. Therefore our method is suitable for displays of various resolution and size even when the camera can not be placed far enough to see the display in a single view.

After recovering the camera parameters, the images of the projected patterns are used to relate the projector coordinates with the display surface coordinates. We represent this relationship using a set of rational Bezier patches and use it to segment the appropriate parts of the image for each projector to register them on the dome. The images can be registered for any arbitrary viewpoint of a single head tracked user in 3D visualization applications. However, since domes are often used for multi-user visualizations (e.g. planetariums), we can also use cartographic mapping techniques to wrap the image on the dome making it conducive for multi-user viewing. Our method is accurate, automated, and extremely easy to deploy. Further, the corrections required to achieve the registration can be applied in real-time via our GPU implementation making it suitable for interactive visualizations. To the best of our knowledge, this is the first method that can achieve registration of multi-projector domes in an automated manner without using complex stereo setups, but simply using a single uncalibrated camera.

2. Related Work

There has been a large amount of work on registering images on planar multi-projector displays using a single uncalibrated camera via linear homographies enabled by the planar screen [Ras00, CSWL02, RvBB*03, RP04, YGH*01, YMB05,RGM*03,AFSR04]. Such registration has also been achieved in the presence of projector non-linearities using rational Bezier patches [BJM07]. Some early work on non-planar displays used a single camera [RBWR04], but in these cases the geometric registration looks correct from only one 'sweet spot', the location of the calibrating camera. More recently, [HCS*06, SSC*08] have tried to remove this restriction of a 'sweet spot' while using a single camera for the special case of cylindrical surface rather than a general non-planar surface. They recover the 2D surface parametrization using the fact that cylindrical displays are ruled surfaces. This is achieved by using a piecewise planar representation of the display surface in the camera space and linking them to the piecewise planar representation of the projector image plane in the camera space. However, this demands precise correspondences between the physical display and the observing camera coordinate to reconstruct a piecewise planar representation of the display surface in the camera space. To achieve this, a precisely calibrated physical pattern is pasted along the top and bottom rims of the cylinder. Thus, images can be wallpapered on the cylinder facilitating multi-user applications. However, since the application cannot support fiducials at a high spatial density and can only sample the rims of the display, these methods result in distortions or stretching, especially towards the middle of the display.

It is evident from the above explorations that to achieve an accurate registration, one needs to recover the 3D geometry of the display, not just its 2D parametrization. Hence, Raskar et al. in [RBWR04] uses a stereo camera pair to reconstruct special non-planar surfaces called quadric surfaces (spheres, cylinders, ellipsoids and paraboloids) and propose conformal mapping and quadric transfer to minimize pixel stretching of the projected imagery when registering the display for multi-viewing purposes. Raskar et al. in [RBY*99] uses special 3D fiducials to achieve a complete device (camera and projector) calibration and 3D reconstruction of the display surface using a large number of structured light patterns, later used to achieve geometric registration. Aliaga et al. in [AX08, Ali08] uses a single camera without any physical fiducials, and hence needs to constrain the system in other ways to recover the 3D geometry of the display. These constraints are achieved by completely superimposing projectors and validating results from photometric and geometric stereo, resulting in a self-calibrating system. However, this cannot allow tiled projectors with small overlaps across the projectors as is relevant for visualization on domes.

Our work is close to a body of literature that uses some prior knowledge of the display geometry to constrain the problem sufficiently such that the camera properties and the

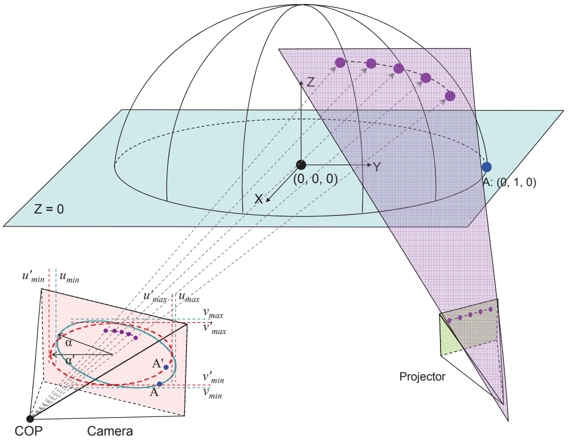

Figure 2: *This illustrates the world coordinate system and the display surface and camera setup with respect to it. The captured image of the boundary of the screen is shown in the picture (cyan ellipse) and also the reprojected boundary on the image plane of the estimated camera (dashed red ellipse). Also one of the projected sets of points which are collinear in the projector space is shown. The 3D position of the detected points is estimated using ray-shooting and then tested for coplanarity (constraint 4).*

3D display geometry can both be recovered without using calibrated stereo cameras, but simply using a single uncalibrated camera. [SM09] shows that for smooth vertically extruded surfaces of known aspect ratio, a single uncalibrated camera can be used to register images from multiple projectors on them. The prior facilitates accurate camera parameter estimation and 3D display geometry reconstruction which are then used for registration of the images from multiple projectors. [SM10c] extends this work to allow multiple views from a single uncalibrated camera when the display surface is too large to fit in a single view. [SM10b] shows that when dealing with perfect linear projectors, multi-projector registration can be achieved even when the vertically extruded surface is not smooth, but piecewise planar. This allows overlapping projectors on the corner of CAVES, a popular VR display choice. Finally, [SM10a] shows that when assuming a prior knowledge of swept surface, the prior can facilitate a 3D display geometry and camera property estimation which can then lead to multi-projector registration on complex swept surfaces, even self-intersecting ones.

Drawing inspiration from these previous works, we also use the prior knowledge of the display surface having a hemispherical shape. However, our work has a key difference from these previous works. In all of these works, a single image of the display (when no projectors are turned on) is used to estimate the camera properties and display geometry. Following this projected patterns are used to relate the projector coordinates with the display coordinates to achieve the geometric registration. However, since a dome has much less boundary features than either a vertically extruded or a swept surface, it is not possible to constrain the problem sufficiently to find the camera parameters without using the projectors. So, unlike all previous work in this direction, in this paper we propose an entirely different and more involved non-linear optimization that uses the projectors to enable enough constraints in order to make the problem tractable (Section 3). This is critical in enabling the use of a completely uncalibrated camera while registering projectors on domes making this method very accessible to a lay-person user. Finally, we expand our method to work with multiple pan and tilted views of an uncalibrated camera using an approach similar to [SM10c], but with higher accuracy, to allow displays of different size and resolution (Section 4).

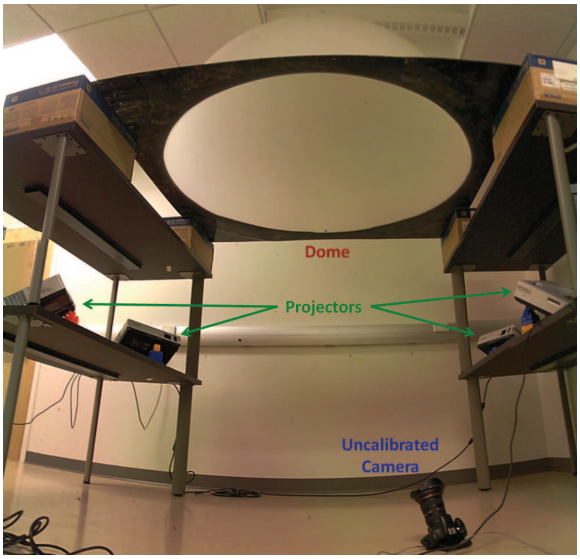

Figure 3: *Image of our 4 projector dome setup. The calibrating camera can be seen at the bottom.*

3. Single-View Registration Algorithm

Let the radius of the hemisphere (a more formal term for dome) be 1. We define a world coordinate system where the equatorial plane of the hemisphere is the Z=0 plane and the center of the hemisphere is at $(0,0,0)$. The hemisphere is rotationally symmetric. Hence, we need one fiducial to define the coordinate system unambiguously. We use a small fiducial A on the equator of the hemisphere and assume its coordinate to be $(0,1,0)$. This defines a world coordinate system (Figure 2). Let the image planes of the camera and the projectors be parameterized by (u,v) and (s,t) respectively.

We assume that our projectors and camera are linear devices with no radial distortion. Projectors are considered the dual of a pin-hole camera. We assume that our camera is completely uncalibrated, i.e., both its intrinsic and extrinsic parameters are unknown. For a system of n projectors, our

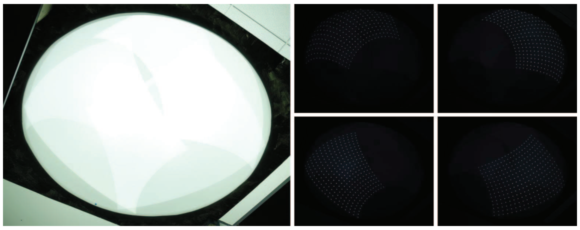

Figure 4: *The images used for calibrating our 4 projector dome. One image from the dome and one image for each projector are used.*

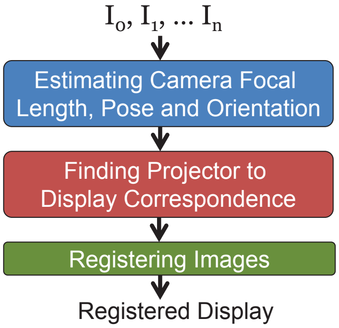
I₀, I₁, ... Iₙ
Estimating Camera Focal Length, Pose and Orientation
Finding Projector to Display Correspondence
Registering Images
Registered Display

Figure 5: *The pipeline of our algorithm.*

registration algorithm takes $n+1$ images as input. The first image, I_0, is of the hemispherical display with no projectors turned on. Next, for each projector i, $1 \leq i \leq n$, we take a picture I_i of the same display surface with projector i projecting blobs that form a grid of vertical and horizontal lines in the projectors image space. We assume that the total number of such grid lines are m. The set of $n+1$ images for a four projector setup of Figure 3 is shown in Figure 4.

Our algorithm has three steps. First, we recover the intrinsic and extrinsic camera parameters using a non-linear optimization (Section 3.1). Next, we find the display to projector correspondences (Section 3.2). Finally, we register the images from the multiple projectors on the hemisphere (Section 3.3). The complete pipeline of our method is illustrated in Figure 5. Note that it is common to have domes which are not exactly hemispheres, i.e. the vertical field-of-view (FOV) is less than 90 degrees. Our method can handle such cases given an extra prior – the ratio of the height of the dome to its radius (Section 3.4).

3.1. Estimating Camera Parameters

The input to this step is the set of images, $I_0, I_1, \ldots, I_n$, and the output is the 3×4 camera calibration matrix of the camera. We assume that the equator of the hemisphere is distinct from its surroundings and can be segmented easily in I_0. In each image I_i, we detect the 2D coordinates of the blobs from projector i using a blob detection technique that is robust in the face of distortions created by the hemispherical display surface. These 2D blobs coordinates are then organized in groups L_{ij} (line j in projector i) such that the blobs in each group fall either on a vertical or a horizontal line in the projector image plane. Let the total number of blobs in line L_{ij} be m_{ij}. Hence, the total number of blobs from projector i is given by m_i, where $m_i = \sum_j m_{ij}$. This blob detection and organization is described in details in Section 5.

Let $M = K(R|RT)$ be the camera calibration matrix comprising of the 3×3 intrinsic parameter matrix K and the 3×4 extrinsic parameter matrix $(R|RT)$ that provides the pose and orientation of the camera. $(R|RT)$ comprises of six parameters including three rotations to define the orientation and the 3D center of projection (COP) of the camera to define the position. In most cameras today the principal center is at the center of the image, there is no skew between the image axes, and the pixels are square and not rectagular. Using these assumptions, as in [SSS06,SM10b,SM10a,SM09], we assume the camera intrinsic parameter matrix K to have only one unknown, the focal length f, i.e.,

$$K = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1}$$

Hence, in this step we are seeking to estimate the seven parameters of the camera – the focal length, the three rotation angles of its orientation and the three coordinates of its COP.

We estimate these parameters using a non-linear optimization with the following constraints.

1. *Fiducial Constraint:* This constraint seeks to minimize the reprojection error E_1 of fiducial A. Let (u_A, v_A) be the detected coordinate of A in I_0. The projected coordinate (u'_A, v'_A) is given by applying M to the 3D coordinates of A. We define the error $E_1 = (u_A - u'_A)^2 + (v_A - v'_A)^2$. This constraint allows us to resolve the ambiguity resulting from the rotational symmetry of the hemisphere.
2. *Boundary Size Constraint:* This is a constraint on the size and position of the image of the equator of the dome. To measure the size in the image I_0, we fit an axis-aligned bounding box given by (u_{min}, v_{min}) and (u_{max}, v_{max}). The equator of the dome in the world coordinate system is defined as $X^2 + Y^2 = 0, Z = 0$. We reproject it on the camera image plane using M to get (u'_{min}, v'_{min}) and (u'_{max}, v'_{max}). We define the error $E_2 = (u_{min} - u'_{min})^2 + (v_{min} - v'_{min})^2 + (u_{max} - u'_{max})^2 + (v_{max} - v'_{max})^2$.
3. *Boundary Orientation Constraint:* This is a constraint on the orientation of the boundary. The image of the equator in I_0 will be an ellipse in general. We identify the major axis of this ellipse, given by vector α. Next we reproject the equator on the camera image plane using matrix M and identify its major axis α'. We seek to minimize the angular deviation between α and α'. Hence, we define the error $E_3 = (1 - |\alpha \cdot \alpha'|)^2$. This constraint together with the the previous constraint assure that the captured image of the equator and the reprojection of the equator on the image plane are identical.

4. *Coplanar Lines Constraint:* This constraint is on the image of each line L_{ij} in image I_i to resolve the scale factor ambiguity and hence to help us in finding the focal length of the camera. For this, we first use ray casting to back project the 2D images of all the m_{ij} blobs in L_{ij} using M and find the corresponding 3D locations of the blobs on the hemisphere. Note that all these 3D points should be coplanar since they are the projections of collinear points in the projector image plane. In order to evaluate this we create a $m_{ij} \times 4$ matrix P_{ij} using these 3D coordinates where the first three elements of each row are the 3D back-projected coordinates of a 2D blob lying on the image of L_{ij} and the last element is 1. The coplanarity of these points is assured if the fourth eigenvalue of matrix P_{ij} is zero. Hence, to enforce the coplanarity constraint for each line L_{ij}, we define the error metric E_{ij} as the square of the fourth eigenvalue of P_{ij} for each line L_{ij}. The total deviation of all the lines from coplanarity defines the final error metric E_4, i.e. $E_4 = \frac{1}{w}\sum_i \sum_j E_{ij}$, where the weight w is given by $1/\sum_i \sum_j m_{ij}$. This allows us to give the same importance to E_4 as the previous error metrics irrespective of the number of blobs used.

Using the above constraints we seek to minimize $E = \sqrt{E_1+E_2+E_3+E_4}$ in our non-linear optimization. To solve this, we use standard gradient descent methods. To assure faster convergence we (a) apply a pre-conditioning to the variables so that the range of values assigned to them is normalized; and (b) use decaying step size. We initialize the optimization assuming the view direction of the camera to be aligned with the Z-axis. To initialize the distance of the camera we use an estimate of the vertical FOV covered by the screen in the camera image. For this we find the height H of the image of the equator in pixels and then initialize the center of projection to be at $(0,0,\frac{H}{2f})$. For the initial value of f we use the EXIF tags of the captured image, as in [SM09].

3.2. Finding Projector to Display Correspondences

In this step we use the recovered camera parameters and the 2D blobs identified on each image I_i from projector i to find the correspondences between the projector coordinates (s,t) and the 3D display coordinates (X,Y,Z). Each blob Q_k, $1 \leq k \leq m_i$, in I_i is back-projected on the display surface by casting rays from the COP of the camera using the recovered camera parameters and finding their intersection with the hemispherical display surface in 3D. Let's assume the back-projected position of blob Q_k is (X_k,Y_k,Z_k) and the position of the blob in the projector coordinate system is (s_k,t_k). In order to relate the 2D coordinate system of the projector to the 3D coordinate system of the display we fit three rational Bezier patches, $B_X(s,t)$, $B_Y(s,t)$, and $B_Z(s,t)$, using these correspondences such that

$$(X,Y,Z) = (B_X(s,t), B_Y(s,t), B_Z(s,t)). \tag{2}$$

To fit the rational Bezier patches we use a non-linear least squares fitting solved efficiently by the Levenberg-Marquardt gradient descent optimization. Using perspective projection invariant rational Bezier patches for interpolation instead of a simple linear interpolation allows us to achieve accurate registration even with a sparse set of correspondences. This also enables the use of a low resolution camera to register the higher resolution hemispherical display.

3.3. Registering Images

We perform geometric registration in two different ways depending on the application, view-dependent and view-independent, described in details as follows.

For Single User: For single user applications such as 3D visualization, flight simulation, and 3D games, we register the image in a view-dependent manner that looks correct for an arbitrary desired viewpoint. For this, we use the standard two-pass rendering approach used in [Ras00]. We first render the scene from a virtual camera at the desired viewpoint. In the second pass, for every projector pixel (s,t) , we use Equation 2 to find the corresponding (X,Y,Z) display coordinate, and then project this 3D point on the image plane of the virtual camera to assign the desired color.

For Multiple Users: View-dependent registration is not suitable for multi-user applications such as planetariums and maps. For these applications we need to wrap the image on the surface of the hemisphere in a manner appropriate for multi-viewing. Though this depends largely on the application, we can borrow a large number of techniques from the domain of map projections in cartography for this purpose. This can be a simple orthographic or stereographic projection or more complex Lamberts conformal conic or azimuthal equidistant projection. Such projections do not look right for a single viewpoint but provide sensible information from all views making them suitable for multi-user viewing.

3.4. Handling Non-Hemispherical Domes

Our method described so far assumes the display surface to be a perfect hemisphere. However it is common to have non-hemispherical domes which are truncated from the bottom (and not from the side of the pole) with a plane parallel to the equator. We can also handle this more general scenario with a little more prior information. A non-hemispherical dome results in an ambiguity between the focal length and the depth of the dome since the relative height of the dome with respect to its radius is unknown. In order to overcome this ambiguity, we need to know the ratio of the height of the dome with respect to its radius, β, which can be easily provided by the user. β is taken into consideration to define the global coordinate system. Following this, the algorithm proceeds as before since the surface of the screen is well-defined in the world coordinate system. The only difference is that while estimating the camera parameters the intersection of the rays from the camera is performed with a partial hemisphere instead of a complete one.

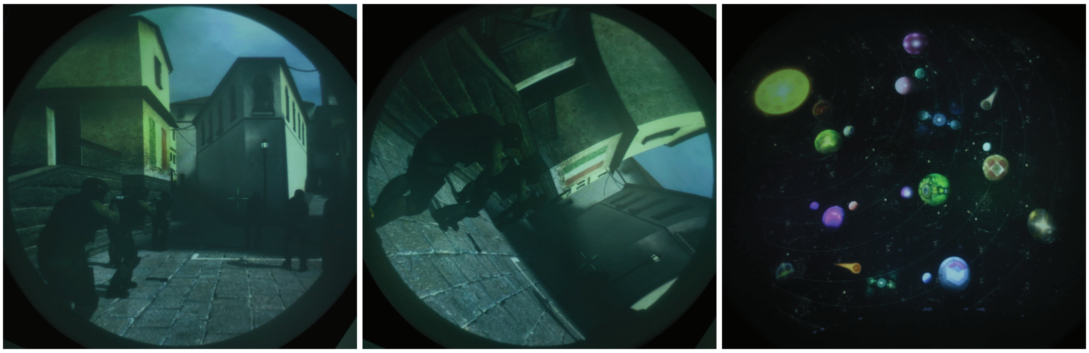

Figure 6: *Left: View-dependent registration for a gaming application while the capturing camera is located at the virtual viewpoint. Middle: View-dependent registration of the same scene when the capturing camera is away from the virtual viewpoint and hence the visible distortions of the straight lines of the buildings and on the pathway. Right: View-independent registration for a planetarium like setup using stereographic projection.*

4. Multi-View Registration Algorithm

The single-view registration algorithm requires the entire display to be visible from the single camera view. This cannot be assured when the display is large. Therefore we adapt the method in [SM10c] that uses multiple overlapping partial views from an uncalibrated camera, mounted on a pan-tilt unit (PTU), to register multiple projectors on a vertically extruded display. However, [SM10c] assumes the camera center of projection (COP) to be coincident with the center of rotation of the PTU. This is not true for non-nodal pan-and-tilt units, especially when using a camera with a zoom lens. This results in inaccuracies in the estimation of the camera view parameters and hence in registration. Hence, we improve this method by allowing a translation between the center of rotation of the pan-and-tilt unit and the COP of the camera.

As in [SM10c] we assume the camera is panned and titled to capture Q views of the display. Each view, $V_i, 1 \leq i \leq Q$, can see only a part of the display. We assume a translation vector T_R between the center of rotation of the pan-and-tilt unit and the COP of the camera. For each camera view, V_i, a set of images as in Section 3 is captured only for the projectors visible in that view. As in [SM10c], we assume same zoom and therefore same intrinsic matrix for all the views. Therefore the unknowns are the common intrinsic matrix K, and the extrinsic matrices $C_i, 1 \leq i \leq Q$.

First, we use angular constraints on pairs of correspondences across multiple views, as in [SM10c], to recover an initial estimate of the focal length, and therefore K. The extrinsic matrices C_i and C_j of two adjacent views V_i and V_j respectively can be related to each other by

$$C_j = C_i \left(\begin{array}{ccc|c} & R_{ij} & & R_{ij}T_R - T_R \\ 0 & 0 & 0 & 1 \end{array} \right) \qquad (3)$$

where R_{ij} is the 3×3 rotation around the center of rotation of the PTU. If K and T_R are known, R_{ij} can be recovered using at least 4 pairs of correspondences between V_i and V_j using a linear least square optimization, assuming the correspondences are not collinear. However, in our case, T_R is not known and only a rough estimate of K is available. So, we use a non-linear optimization, solved using a gradient descent optimizer as in Section 3.1, to recover T_R and R_{ij} and to refine the estimate of K archived using the angular constraints. In each iteration, the optimizer assigns new values to f and T_R and recover the R_{ij} matrices for the adjacent views using a linear least square optimization.

Inaccuracy in the estimate of f or T_R results in R_{ij}s that are not perfect rotation matrices. Perfect rotation matrices should have orthonormal row vectors. We measure this using the deviation of the magnitude of each row vector of R_{ij} from unity and the deviation of the dot product of each pair of row vectors of R_{ij} from zero. Our error metric is defined by the square root of the sum of these deviations across all the R_ijs. We seek to minimize this error metric in each iteration of the non-linear optimization until it is smaller than a certain threshold, usually 0.01 for our setups.

Using the recovered center of rotation and R_ijs we find all the C_is by considering C_1 as the reference view and finding a minimum spanning tree in a graph formed by considering the views as nodes where edges connect adjacent overlapping camera views. More details of this process can be found in [SM10c]. The rest of the method is similar to single-view registration but all the back-projections and reprojections happen only in the related cameras views.

5. Implementation

We have implemented our method in MATLAB and tested it on a 5 ft diameter front projection dome tiled with four casually aligned projectors. We use Epson 1825p projectors ($600). We use two types of sensors: (a) a high-end high-resolution (13 Megapixel) Canon Rebel Xsi SLR camera ($800); and (b) a low-end low-resolution (1.3 Megapixel) Logitech QuickCam Orbit MP ($120). The latter demon-

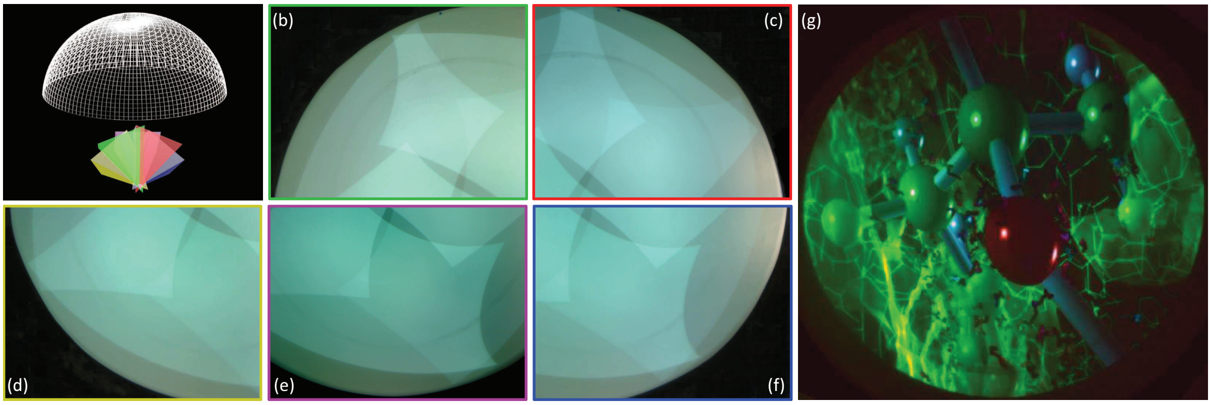

Figure 7: *(a) Reconstruction of the display and the camera view-frustums. (b)-(f) Images of the boundary of the screen captured for multi-view registration. The colors of the borders of the images and the view-frustums show which image is captured from which view. (g) A registered image on the display using this multi-view registration.*

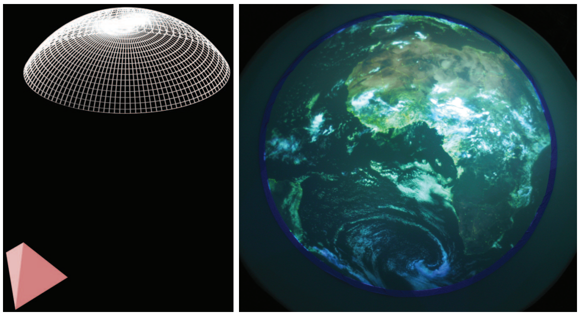

Figure 8: *Left: Reconstruction of our non-hemispherical dome and the camera view frustum. Right: Registration of a map data on the non-hemispherical dome.*

strates the advantage of our Bezier based interpolation scheme. For photometric blending of the overlap region, we simply divide the color projected by each projector pixel by the number of contributing projectors at that pixel. This generates an attenuation map for each projector. To assure this division happens in a linear color space, we apply a gamma correction to these attenuation maps.

The registration is done offline and takes about 2 minutes for single-view and 4 minutes for multi-view registration using 5 camera views. This generates the rational Bezier patches B_X, B_Y and B_Z for each projector which are then used for image correction. We have implemented a real-time image correction algorithm using modern GPUs through Chromium - an open-source distributed rendering engine for PC clusters [HHN*02]. A module for Chromium is written that first pre-computes the coordinate-mappings of all pixels using the rational Bezier parameters. This per-pixel projector to screen lookup table is used by a fragment shader to map pixels from the projector coordinate space to the screen coordinate space during the rendering process. However, the same process can be adapted easily to other rendering software solutions like TouchDesigner or Equalizer [EMP09].

To detect the blobs in I_i for projector i, we consider the m_i blobs in the projector space forming a 2D array or grid G of size $m_p \times m_q$. Each blob is identified by a 2D array coordinate in the projector space $G(p,q)$. Let the corresponding position of the blob in the camera image coordinates be $I_i(G(p,q))$. We use white color for all the blobs except for $I_i(G(1,1))$, and the two blobs on its immediate right and below, essentially $I_i(G(1,2))$ and $I_i(G(2,1))$. We find the location of these three blobs automatically using their distinct color. The center of the blobs are refined using a non-maximum suppression (NMS) method in a small window around the recovered location. Next, we consider two distance vectors: d_s is the distance vector in camera space between two adjacent blobs in a row in the projector space and d_t is the same for two adjacent blobs in a column. We then use the following algorithm to detect the blobs row by row by updating d_s and d_t from the more recently detected blobs.

```
$d_s = I_i(G(1,2)) - I_i(G(1,1))$;
$d_t = I_i(G(2,1)) - I_i(G(1,1))$;
for $k = 1$ to $m_p$ do
    if $k > 1$ then
        Find $I_i(G(k,1))$ using NMS around $I_i(G(k-1,1)) + d_t$;
        $d_t = I_i(G(k,1)) - I_i(G(k-1,1))$ ;
    endif
    for $l = 2$ to $m_q$ do
        Find $I_i(G(k,l))$ using NMS around $I_i(G(k,l-1)) + d_s$;
        $d_s = I_i(G(k,l)) - I_i(G(k,l-1))$ ;
    endfor
endfor
```

Using this method we can find all the blobs automatically. We have found this method to be effective even in face of severe perspective distortion due to the hemispherical shape of the screen. However, this method assumes all the blobs in

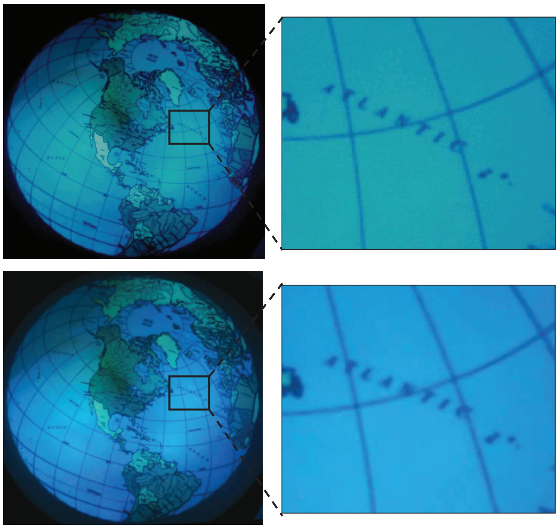

Figure 9: *Here we compare the geometric registration using our high-end SLR camera (13 Megapixels) (top) with that achieved with our low resolution webcam (1.3 Megapixels)(bottom). Note that there is no degradation in the accuracy of registration, even for challenging contents like text and single pixel lines for latitudes and longitudes.*

the projector space are visible in the captured image, which is not the case when using multiple partial camera views. For such cases, we project the blobs in a time sequential manner and recover their position by capturing multiple frames of binary coded blobs as used in [SM10c]. This increases the number of input images per projector to $log(m_i)$.

6. Results

Figures 1(middle) and 6(right and middle) show three examples of registration from a single arbitrary viewpoint. In Figure 6 we demonstrate the correctness of the view by showing the distortions resulting when the capturing camera is moved away from the virtual viewpoint. Figures 1(left) and 6(left) show examples of view-independent registration on our display. This demonstrates the accuracy of our geometric registration. To demonstrate the multi-view method, we register our dome using 5 overlapping views from a camera mounted on a PTU. The 5 partial views of the unlighted display, the reconstructed display and camera view-frustums, and an image registered using these multiple overlapping views are shown in Figure 7. Note that the quality of the registration is comparable with the registration using a single camera view. Our supplementary video demonstrates real-time applications where we achieve interactive frame rates using the GPU for image correction.

To demonstrate the effectiveness of our method on a non-hemispherical dome, we covered the bottom portion of our dome to make it non-hemispherical and applied our registration technique (Figure 8). In this particular example, the radius of the boundary circle of the non-hemispherical dome is 2.25 ft and the height of it is 1.41 leading to a β of 0.626.

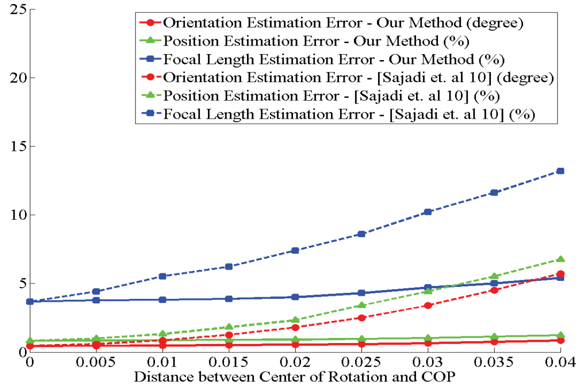

Figure 10: *Comparison between the accuracy of the recovered extrinsic parameters using our method and the method proposed by Sajadi et. al. [SM10c] for setups with 4 to 10 camera views. The results show considerable improvement when the distance between the center of rotation of the PTU and the COP of the camera is larger than 0.01 which is the case for our setups in practice.*

We compare the accuracy of the geometric calibration achieved when using a low-resolution webcam with that achieved using a high-end SLR camera (Figure 9). The accuracy of our Bezier based interpolation is shown by the lack of degradation in the quality of registration when using a low-resolution webcam, even on particularly challenging contents like text and single pixel latitudes and longitudes.

7. Discussions

In this section, we discuss the dependency of our method on various parameters like imprecision in the display surface construction, projector non-linearities and so on.

Accuracy and Sensitivity: Our system makes two assumptions: (a) the display surface is a dome; and (b) the projectors are perfect linear devices. Hence, it is important analyze the accuracy of our calibration in the presence of imperfections in the construction of the display and slight nonlinear distortions in the projectors. It is difficult to analyze all the above issues in real systems, hence we have conducted extensive analysis in simulation for this purpose.

Figure 11 shows a quantitative analysis of the error in extracting the camera parameters with respect to the deviation of the surface from being a perfect dome. We simulated several such deviations which not only affect the surface of the dome but also the shape of the equator of the dome. We measure the deviation of the surface from being a perfect dome by the maximum difference from the radius at any point on the dome. This plot shows that even in presence of large deviation (almost 10%) our method can achieve a reasonably good estimation of the camera pose and orientation (less than 5%) and focal length (less than 9%).

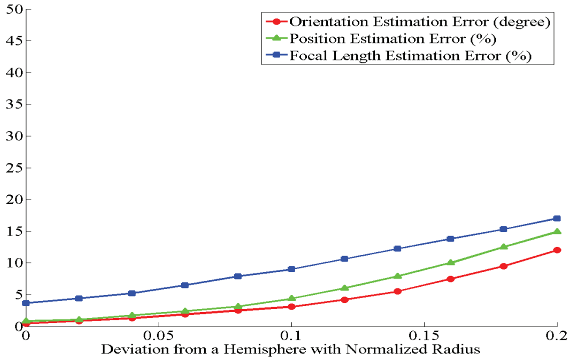

Figure 11: *Accuracy of camera parameter estimation in presence of deviation of the screen from being a dome.*

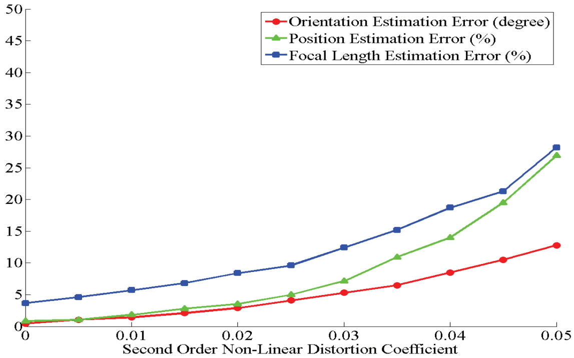

Figure 12: *Accuracy of camera parameter estimation in presence of projector non-linearity.*

We also analyzed the accuracy of our method with slight non-linear distortions in projectors. Empirically, we have seem that most commodity projectors only demonstrate second order radial distortion. Since projectors are dual of cameras we use the standard models for camera non-linearities [Zha99] to model such projector non-linearities. Figure 12 shows the error in extracting the camera parameters with respect to the second order nonlinear distortion of the projectors. In our simulation we have analyzed this for setups with different number (as large as 20) and arrangement of projectors. Even for severe second order distortion of 0.02, the estimation errors are less than 9%.

For the multi-view reconstruction, we compared the accuracy of the estimated camera parameters with respect to the distance between the center of rotation of the PUT and the COP of the camera using our approach and the approach presented in [SM10c] (Figure 10). We used 4 to 10 partially overlapping camera views and the distances are normalized such that the radius of dome is 1 unit. For our cameras and PTUs, the distances between the center of rotation and the COP are between 0.005 and 0.02 of the radius of the dome. Figure 10 demonstrates considerable improvement for our method for such distances.

Camera Non-linearity and Placement: Finally we assume the camera to be a linear device devoid of any non-linear distortion. The effect of slight camera nonlinearities will produce error in the estimation of the camera parameters similar to the effect of projector nonlinearities. However, even in presence of error in estimation of the camera parameters our method will not show any pixel misregistration since our rational Bezier patches provide a particularly robust framework for handling such errors. This is due to the fact that a small error in extracting the camera parameters will lead to an erroneous projector to display mapping but the overlapping pixels from the multiple projectors will still map to the same (X,Y,Z) location. Hence, such errors will create small image distortions but will not lead to any misregistration. Fortunately, human visual system can tolerate such minor deviation in the image. This is evident in our results when using low-end Logitech QuickCam (Figure 9) where slight nonlinear distortions in the camera produce almost imperceptible image distortion and no misregistration.

Our method relies on the image of the equator of the dome to be an ellipse. However, when the view direction of the camera is exactly along the Z axis the image of the equator will be a circle. In such a case the boundary orientation constraint is no longer applicable since a major axis is not defined for a circle. But this situation ensures that the camera orientation is along the Z axis reducing the number of unknowns by two. Hence, the problem is still tractable using the remaining constraints.

Boundary Segmentation: Our method needs to detect the bounding box and orientation of the image of the equator of the dome. Since the screen is usually the most distinct white object in the environment, segmenting it is relatively easy if the background is of reasonable contrast. Further, even if the boundary of the screen is partially blocked or not segmentable due to low contrast between the screen and the background color, one can let the user detect the bounding box and orientation of the image of the equator. All other steps of our method are completely automated as long as the screen is entirely within the FOV of the camera.

8. Conclusion

In summary, we have presented the first work for easy registration of multi-projector domes using a single uncalibrated camera. Our method requires only one fiducial due to the inherent rotational ambiguity of the domes. Our method provides an user-friendly and cost-effective way to sustain such displays for both single user applications such as 3D visualization or training and simulation; and multiuser applications such as planetariums. Further, we demonstrated realtime image correction using the GPUs. We believe that our work has the potential to make the dome technology accessible to the masses and commonplace in low-cost visualization and edutainment applications.

References

[AFSR04] Ashdown M., Flagg M., Sukthankar R., Rehg J. M.: A flexible projector-camera system for multi-planar displays. *Proc. of IEEE CVPR* (2004). 2

[Ali08] Aliaga D.: Digital inspection: An interactive stage for viewing surface details. *Proc. ACM Symp. on I3D* (2008). 2

[AX08] Aliaga D., Xu Y.: Photogeometric structured light: A self-calibrating and multi-viewpoint framework for accurate 3d modeling. *Proc. of IEEE CVPR* (2008). 2

[BJM07] Bhasker E., Juang R., Majumder A.: Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. *IEEE TVCG* (2007). 2

[CSWL02] Chen H., Sukthankar R., Wallace G., Li K.: Scalable alignment of large-format multi-projector displays using camera homography trees. *Proc. of IEEE Vis* (2002). 2

[EMP09] Eilemann S., Makhinya M., Pajarola R.: Equalizer: A scalable parallel rendering framework. *IEEE Transactions on Visualization and Computer Graphics 15* (2009), 436–452. 7

[HCS*06] Harville M., Culbertson B., Sobel I., Gelb D., Fitzhugh A., Tanguay D.: Practical methods for geometric and photometric correction of tiled projector displays on curved surfaces. *IEEE PROCAMS* (2006). 2

[HHN*02] Humphreys G., Houston M., Ng R., Frank R., Ahem S., Kirchner P., Klosowski J.: Chromium : A stream processing framework for interactive rendering on clusters. *ACM Transactions on Graphics (SIGGRAPH)* (2002). 7

[Ras00] Raskar R.: Immersive planar displays using roughly aligned projectors. In *Proc. of IEEE VR* (2000). 2, 5

[RBWR04] Raskar R., Baar J. V., Willwacher T., Rao S.: Quadric transfer function for immersive curved screen displays. *Eurographics* (2004). 2

[RBY*99] Raskar R., Brown M., Yang R., Chen W., Towles H., Seales B., Fuchs H.: Multi projector displays using camera based registration. *Proc. of IEEE Vis* (1999). 2

[RGM*03] Raij A., Gill G., Majumder A., Towles H., Fuchs H.: Pixelflex 2: A comprehensive automatic casually aligned multi-projector display. *IEEE PROCAMS* (2003). 2

[RP04] Raij A., Polleyfeys M.: Auto-calibration of multi-projector display walls. *Proc. of ICPR* (2004). 2

[RvBB*03] Raskar R., van Baar J., Beardsley P., Willwacher T., Rao S., Forlines C.: ilamps: Geometrically aware and self-configuring projectors. *ACM Transaction on Graphics (SIGGRAPH)* (2003). 2

[SM09] Sajadi B., Majumder A.: Markerless view-independent registration of multiple distorted projectors on vertically extruded surface using a single uncalibrated camera. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* (2009). 3, 4, 5

[SM10a] Sajad B., Majumder A.: Automatic registration of multiple projectors on swept surfaces. *ACM Virtual Reality and Software Technology* (2010). 3, 4

[SM10b] Sajadi B., Majumder A.: Auto-calibration of cylindrical multi-projector systems. *IEEE Virtual Reality* (2010). 3, 4

[SM10c] Sajadi B., Majumder A.: Scalable multi-view registration for multi-projector displays on vertically extruded surfaces. *Proceedings of EuroVis* (2010). 3, 6, 8, 9

[SSC*08] Sun W., Sobel I., Culbertson B., Gelb D., Robinson I.: Calibrating multi-projector cylindrically curved displays for "wallpaper" projection. *IEEE/ACM Workshop on PROCAMS* (2008). 2

[SSS06] Snavely N., Seitz S. M., Szeliski R.: Photo tourism: Exploring photo collections in 3d. *ACM Transactions on Graphics (SIGGRAPH)* (2006). 4

[YGH*01] Yang R., Gotz D., Hensley J., Towles H., Brown M. S.: Pixelflex: A reconfigurable multi-projector display system. *Proc. of IEEE Vis* (2001). 2

[YMB05] Yang R., Majumder A., Brown M.: Camera based calibration techniques for seamless multi-projector displays. *IEEE TVCG* (2005). 2

[Zha99] Zhang Z.: Flexible camera calibration by viewing a plane from unknown orientations. *International Conference on Computer Vision* (1999). 9