Quadric Transfer for Immersive Curved Screen Displays

Ramesh Raskar, Jeroen van Baar, Thomas Willwacher

Abstract

Curved screens are increasingly being used for high-resolution immersive visualization enfironments. We describe a new technique to display seamless images using overlapping projectors on curved quadric surfaces such as spherical or cylindrical shape. We exploit a quadric image transfer function and show how it can be used to achieve sub-pixel registration while interactively displaying two or three-dimensional datasets for a head-tracked user. Current techniques for automatically registered seamless displays have focused mainly on planar displays. On the other hand, techniques for curved screens currently involve cumbersome manual alignment to make the installation conform to the intended design. We show a seamless real-time display system and discuss our methods for smooth intensity blending and efficient rendering.

Publication History:

1. First printing, TR-2004-34, 02 2004

Quadric Transfer for Immersive Curved Screen Displays

Paper ID: 1104

Abstract
Curved screens are increasingly being used for high-resolution immersive visualization environments. We describe a new technique to display seamless images using overlapping projectors on curved quadric surfaces such as spherical or cylindrical shape. We exploit a quadric image transfer function and show how it can be used to achieve sub-pixel registration while interactively displaying two or three-dimensional datasets for a head-tracked user. Current techniques for automatically registered seamless displays have focused mainly on planar displays. On the other hand, techniques for curved screens currently involve cumbersome manual alignment to make the installation conform to the intended design. We show a seamless real-time display system and discuss our methods for smooth intensity blending and efficient rendering.

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

1. Introduction

Large seamless displays using overlapping projectors is an emerging technology for constructing high-resolution immersive visualization environments capable of presenting high-resolution images from scientific simulation, large format images for entertainment and surround environment for instruction. Such overlapped projector systems complement non-overlapping multi-projector technologies such as the CAVE [CNSD93], Blue-C [SGKM00] and well defined tiled planar [Jup02] or curved displays [Tri02]. In the last few years, we have seen a number of ideas for creating seamless displays on planar screens using electro-optic approaches such as vignetting [LC99] or using camera in the loop [Sur99, CCC*00, YGHT01, BS02, Fut02] to determine the registration and blending parameters. In this paper, we extend the camera-based techniques to deal with curved screens.

1.1 Overview

Accurate estimation of geometric relationship between overlapping projectors is the key for achieving seamless displays (Fig. 1). They influence the rendering and intensity blending algorithms. General techniques to support casually installed projectors and exploit geometric relationship between projectors and display surface eliminate cumbersome manual alignment and reduce maintenance costs. While camera-based parametric approaches for planar screens have exploited the homography, a 3 by 3 matrix, induced due to the plane of the screen [YGHT01, CSWL02, BS02], to our

knowledge, there has been no similar work to exploit a parametric relationship for curved surfaces for immersive environments.

The relationship for surfaces that adhere to quadric equations, such as spheres, cylinders, cones, paraboloids and ellipsoids, can be defined using a quadric image transfer [ST97].

Contributions In this paper, we present a complete set of techniques to generate seamless displays on curved quadric surface. Our technical contributions are as follows.

- Adaptation of quadric transfer, in a semi-permanent setup using a stereo camera pair

- Calibration methods, to estimate quadric transfer using partial Euclidean reconstruction

- Head tracking support for immersive display

- Intensity blending scheme using parametric approach

- Single pass rendering, that can exploit rendering hardware while maintaining high image quality

Conventional non-parametric approaches for displaying multiprojector images on curved screens sample discrete image features to build a look-up table for bilinear interpolation for pixel locations [Jar97, RWF98] and hence errors in feature localization are immediately visible. We propose a completely parametric approach that fits a quadric model to the discrete features reducing them to a few parameters. Similar to the advantages of using a homography for planar screens,
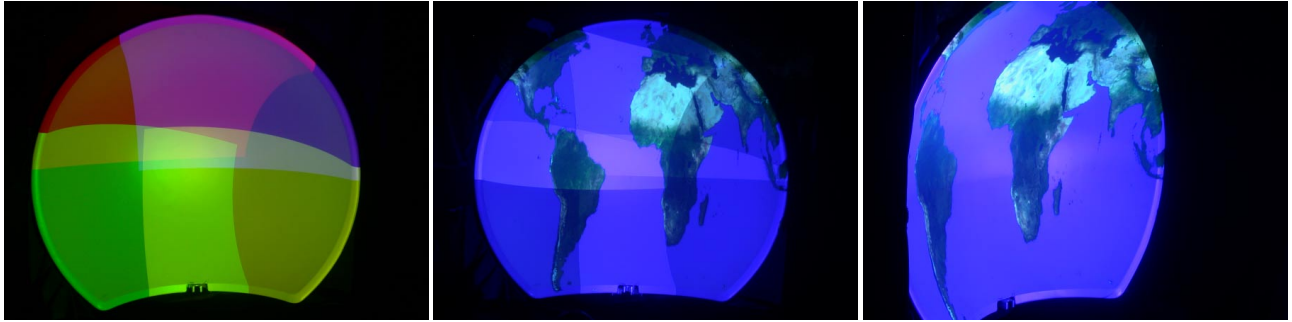
Figure 1: *Visualization on curved quadric surfaces. (Left) Dome with casually aligned overlapping projectors (Middle) Registered images (Right) Displayed with intensity correction. The parametric approach leads to accurate geometric registration and perceptually seamless images. (Embedded images are high-resolution. Please zoom with PDF browser.)*

the quadric transfer based parametric approach leads to reduced constraints on camera resolution, better tolerance to pixel localization errors, faster calibration and finally a simpler single pass parameterized warping process. Another advantage is that our cameras do not need to be placed at or near the 'sweet-spot' i.e. the ideal viewer location.

The scope of the paper is limited to description of algorithms related to quadric transfer, its estimation and use with graphics hardware. We do not focus on photometric issues [MHTW00], data distribution [HHN*02], interaction [FBZ*99] and load balancing [SZF*99] during rendering. Our approach, in the current form, lacks ease of use because of the time involved in non-linear optimization involved in estimating the parameters for image transfer, but this is an active area of research.

1.2 Related Work

Seamless Displays   In commercially available planar displays, alignment is typically performed manually. However, many research groups have exploited camera-based non-parametric (look-up table based) [Sur99] and parametric (homography based) [YGHT01, CSWL02, BS02] approaches to automate this process for planar screens.

Multi-projector alignment for curved screens is sometimes aided by projecting a 'navigator' pattern [Tri02, Jar97]. Then all the overlapping projectors are manually aligned with the grid [Tri02]. We have heard that, at Hayden Planetarium [Hay02] in New York, two technicians spend one hour each morning trying to adjust the registration between the seven overlapping Trimensions Prodas projectors.

An automated approach for planar or non-planar screens, using a non-parametric process, involves putting a camera at the sweet-spot. The camera observes the structured patterns projected by projector. The sampled readings are then used to build an inverse warping function between the input image and projected image by means of interpolation [Jar97, RWF98, Sur99].

We opt for a parametric method and extend the homography-based approach for planar screens to an approach based on quadric transfer. However, curved screens present new challenges. Many Euclidean quantities are required to be calibrated early in the process and non-linear optimization of parameters makes robustness an issue.

Quadric Surfaces   Quadrics for image transfer have been introduced in computer vision literature [ST97, CZ98]. In multi-projector systems, however, there has been little or no work on techniques for parameterized warping and automatic registration of higher order surfaces. Quadrics appear in many projector-based configurations. Large format flight simulators have traditionally been cylindrical or dome shaped [SM99], planetariums and OmniMax theaters use hemispherical screens [Alb94], and many virtual reality setups use a cylindrical shaped screen.

As part of applications of handheld autonomous projectors, [RvB*03] described a quadric transfer based approach where a camera is embedded in each projector. In an extended text abstract, [vWRR03], at a workshop speculated about how this embedded camera approach could be used for a display system. The large number of cameras lead to a significant increase in the number of pairwise Euclidean 3D parameters. In the absence of an environmental camera, the pairwise parameters are refined to be globally consistent. While this approach eliminates the need for external cameras, which is appropriate for self-contained projectors, we focus on semi-permanent installations.

For several reasons, including (i) supporting immersive rendering for a head-tracked user, (ii) exploiting single pass rendering using graphics hardware, and (iii) building practical solution for semi-permanent installations, one needs to use a different approach. We present a simpler and efficient solution using a stereo camera pair appropriate for such installations. In addition, we propose a complete solution for a practical calibration, intensity blending and efficient rendering. We address the inherent problems in using quadric transfer and present a solution to overcome the robustness

issues by exploiting a partial Euclidean reconstruction. To our knowledge, a parameterized solution for a head-tracked system for curved screen is being presented for the first time.

2. Quadric image transfer

We present the notations and basics of quadric transfer, and describe how we adapt it to register overlapping projector images.

2.1 Basics

Mapping between two arbitrary perspective views of an opaque quadric surface, Q, in 3D can be expressed using a quadric transfer function, Ψ. While planar homography transfers can be computed from 4 or more pixel correspondences, quadric transfer requires 9 or more correspondences. The quadric transfer can be defined in a closed form using the 3D quadric surface, Q and additional parameters that relate perspective projection of the two views. The quadric transfer in our case means image transfer from first view to the second view.

The quadric, Q, is a surface represented by a 4×4 symmetric matrix, such that 3D homogeneous points X (expressed as a 4×1 vector) that lie on the surface satisfy the quadratic constraint,

$$X^T Q X = 0$$

The quadric, Q, has 9 degrees of freedom corresponding to the independent elements of the matrix. The matrix is symmetric and defined up to an overall scale.

The homogeneous coordinates of the corresponding pixels, x in the first view and x' in the second view are related by

$$x' \cong Bx - \left(q^T x \pm \sqrt{(q^T x)^2 - x^T Q_{33} x} \right) e$$

From pixel correspondences (x, x'), we compute the 21 unknowns: the unknown 3D quadric Q, a 3x3 homography matrix B and the epipole, e, in homogeneous coordinates. The epipole, e, is the image of the center of projection of the first view in the second view. The sign $\cong$ denotes equality upto scale for the homogeneous coordinates. Matrix Q is decomposed as follows.

$$Q = \left[\begin{array}{cc} Q_{33} & q \\ q^T & 1 \end{array} \right]$$

Thus, Q_{33} is the top 3×3 symmetric submatrix of Q and q is a 3 vector. $Q(4,4)$ is non-zero if the quadric does not pass through the origin i.e. the center of projection of the first view. Hence, it can be safely assigned to be 1.0 for most

display surfaces. The final 2D pixel coordinate for homogeneous pixel x' is $(x'(1)/x'(3),\ x'(2)/x'(3))$.

The transfer described by 21 parameter has four dependent parameters [ST97, WS99]. This ambiguity is removed [RvB*03] by defining

$$A = B - eq^T \qquad\qquad E = qq^T - Q_{33}$$

so that,

$$x' \cong Ax \pm \left(\sqrt{x^T E x} \right) e$$

The equation $x^T E x = 0$ defines the outline conic of the quadric in the first view. (The Outline conic can be geometrically visualized as the image of the silhouette or the points on surface where the view rays are locally tangent to the surface, e.g. the elliptical silhouette of a sphere viewed from outside the sphere.) A is the homography via the polar plane between the first and the second view. The ambiguity in relative scaling between E and e is removed by introducing a normalization constraint, $E(3,3) = 1$. The sign in front of the square root is fixed within the outline conic in the image. The sign is easily determined by testing the equation above by plugging in coordinates for one pair of corresponding pixels.

Note that the parameters of the quadric transfer, $\Psi = \{A, E, e\}$, can be directly computed from 9 or more pixel correspondences in a projective coordinate system. So it is tempting to follow a approach similar to estimating planar homography for planar displays without computing any Euclidean parameters. However, as described later, in practice it is difficult to robustly estimate the epipolar relationship in many cases. Hence, we follow a pseudo-Euclidean approach as described below.

2.2 Approach

All registration information is calculated relative to a camera stereo pair. We assume that the stereo camera pair can see the entire 3D surface. One of the cameras is arbitrarily chosen as the origin. The cameras here are used to determine only the 3D points on the display surface and not for any color sampling. Hence, any suitable 3D acquisition system can be used. The outline of our technique is as follows. The details are in Section 3 and 4.

During **pre-processing**, the following steps are performed.

- For each projector i
  - ▷ Project structured light pattern with projector
  - ▷ Detect features in stereo camera pair and reconstruct 3D points on the display surface
- Fit a quadric, Q, to all the 3D points detected
- For each projector i
  - ▷ Find its pose wrt the camera using the correspondence between projector pixels and 3D coordinates of points

they illuminate
▷ Find the quadric transfer, Ψ_{0i}, between the camera and projector i
▷ Find intensity blending weights, Φ_i, in overlap regions

At **run time**, the rendering for 2D images or 3D scenes follows these steps.

- Read head tracker and update quadric transfer Ψ_{h0} between virtual view and camera
- Read the input image in the 2D video or compute the input image by rendering a 3D scene from the virtual viewpoint
- For each projector i
  ▷ Pre-warp input image into projectors framebuffer using quadric transfer, $\Psi_{0i} \circ \Psi_{h0}$
  ▷ Attenuate pixel intensities with blending weights, Φ_i

This approach, however, involves several issues. The quadric transfer estimation, although a linear operation, requires non-linear optimization to reduce pixel re-projection errors. It is difficult to estimate projector pose (external parameters) because the 3D points projected on the quadric are usually nearly planar leading to a degenerate condition. These plus other issues, and practical solutions are discussed below.

3. Calibration

The goal is to compute the parameters of quadric transfer, $\Psi_{0i} = \{A_i, E_i, e_i\}$, so that the projected images are geometrically registered on the display surface. The method to calculate quadric transfer parameters directly from pixel correspondences involves estimating the 4x4 quadric matrix, Q, in 3D [ST97], [CZ98] using a triangulation of corresponding pixels and a linear method. If the internal parameters of the two views are not known, all the calculations are done in projective space after computing the epipolar geometry, i.e. the epipoles and the fundamental matrix. However, we noticed that when projectors rather than cameras are involved, the linear method produces very large re-projection errors in estimated 3D quadric, Q. The errors are of the order of 30 pixels for XGA projector. In addition, the fundamental matrix is inherently noisy given that the 3D points on the quadric surface illuminated by a single projector do not have significant depth variation in display setting such as segments of spherical or cylindrical surfaces. We instead follow a pseudo-Euclidean approach where the internal and external parameters of the camera and the projectors are known *approximately*. and are used to estimate Euclidean rigid transformations. Hence, unlike the planar case, computation of accurate image transfer for curved screens, involves three-dimensional quantities. We describe our approach and use of approximate Euclidean parameters for estimating warping parameters.
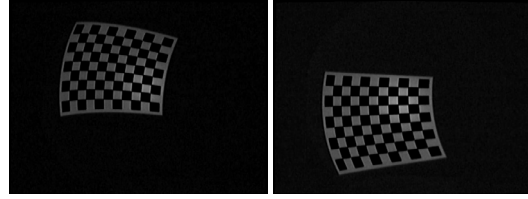

Figure 2: *Images captured by the 640x480 resolution camera during calibration. The resolution of each projector is significantly higher at 1024x768 and yet is captured in only a small part of the camera view.*

3.1 Quadric Surface

We use a rigid stereo camera pair, C_0 and C_0', for computing all the geometric relationships. We arbitrarily choose one of the cameras to define the origin and coordinate system. We calibrate the small baseline stereo pair with a small checkerboard pattern [Zha99]. Note that the cameras do not need to be near the sweet-spot in this setup which is an important difference with respect to some of the non-parametric approaches.

The stereo pair observes the structured patterns projected by each projector (Fig 2) and using triangulation computes a set of N 3D points, $\{X_j\}$, on the display surface. The quadric, Q, passing though each X_j is computed by solving a set of linear equations, $X_j^T Q X_j = 0$, for each 3D point. This equation can be written in the form

$$\chi_j V = 0$$

where, χ_j, is a 1×10 matrix which is a function of X_j only and V is a homogeneous vector containing the distinct independent unknown variables of Q. With $N \geq 9$, we construct a $N \times 10$ matrix X and solve the linear matrix equation

$$\mathsf{X}V = 0$$

Given points in general position, the elements of V (and hence Q) are the one dimensional null-space of X.

3.2 Projector View

In addition to the quadric, Q, we need to estimate the internal and external parameters of each projector with respect to the camera origin. We use the correspondence between the projector pixels and coordinates of the 3D points they illuminate to compute the pose and internal parameters.

However, finding the pose of a projector from known 3D points on a quadric is error-prone because the 3D points are usually quite close to a plane leading to an unstable solution [Fau93, FP02]. Dealing with near-planar points is a difficult problem. If points are distributed in depth, we can easily use a linear method to estimate the internals as well as externals of the projector. On the other hand, if the points are known

to be planar, we can estimate the externals if some of the internals are known.

How about bringing in a temporary surface to add non-planar points for the sake of calibration? In most cases this is impractical or cumbersome. The temporary surface will have to be approximately the same size as the display surface. Our goal is to compute the quadric transfer completely automatically.

For dealing with near-planar surfaces, we are required to use an iterative algorithm. If we know the projector internal parameters, we can first find an initial guess for external parameters based on homography and then use an iterative algorithm described in [LHM00]. We use Powell's method for nonlinear refinement of re-projection error. However, estimating projector internals is equally difficult. If the projectors cannot be easily moved, as mentioned above, calibrating them usually requires large surfaces illuminated in two or more positions.

Our strategy is to use projector internal parameters that are approximately known. We find internal parameters of just one projector and use these internal parameters for all projectors for all future settings. The same projector at a later time and other projectors will clearly have different zoom settings and have other mechanical or optical deviations. In addition, the external parameters computed by iterative method in [LHM00], will also be approximate.

3.3 Camera to Projector Transfer

The idea is to use the perspective projection parameters of the camera along with approximate projection matrix of the projector to find the camera to projector quadric transfer using linear methods. Then refine the solution using non-linear optimization.

The quadric transfer parameters between camera and projector i, $\Psi_{0i} = \{A_i, E_i, e_i\}$, are easy to calculate from Q, camera projection matrix $[\,I\,|0]$ and projector projection matrix $[\,P_i\,|e_i]$.

$$A_i = P_i - e_i q^T \qquad\qquad E_i = qq^T - Q_{33}$$

In our tests the parameters found by the linear method turned out to be far too imprecise for our purpose for reasons described in the next subsection. We observed misalignment between the projectors of 15-30 pixels on screen. Since seamless displays require sub-pixel accuracy, we apply a nonlinear minimization to refine the results obtained via the linear method. The j'th pixel in camera, x_0^j, physically corresponds to pixel x_i^j, in projector i, while the transferred pixel location $\hat{x}_i^j$ for projector i is given by

$$\hat{x}_i^j = A_i x_0^j \pm \sqrt{x_0^{j^T} E_i x_0^j}\, e_i$$

The objective function for nonlinear optimization is the total squared transfer error for all pixels,

$$\varepsilon_i = \sum_j \left(\frac{x_i^j(1,2)}{x_i^j(3)} - \frac{\hat{x}_i^j(1,2)}{\hat{x}_i^j(3)} \right)^2$$

Note that the sign in front of the square root found using the linear method, which is same for all the pixels, remains the same during the nonlinear optimization. We used Nelder-Mead Simplex and obtained very satisfactory results.

3.4 Partial Euclidean Reconstruction

There are two questions here: (i) Why not ignore Euclidean approach altogether and directly go for projective space and non-linear optimization and (ii) If we have accurate projector internal parameters, can we avoid non-linear optimization stages?

As mentioned earlier, ignoring Euclidean viewing parameters and solving the quadric transfer purely from pixel correspondences leads to poor re-projection errors. The estimated 3D quadric, Q, cannot be used as a good guess for further non-linear optimization. In fact, in most of our tests, the solution did not converge.

Using accurate projectors internals only reduces the re-projection errors but does not eliminate them. This is because, many kinds of errors are propagated in the three dimensional Euclidean calculations, including estimating 3D points on the display surface by triangulation, estimating the 3D quadric using linear methods and finding the projector pose. The non-linear optimization attempts to minimize the physical quantity we care about the most, i.e. pixel re-projection error in image transfer from camera to projector for known corresponding set of pixels. Since the correspondence between overlapping projector pixels is indirectly defined by this image transfer equation, minimizing pixel re-projection errors ensures geometric registration between the displayed projector pixels.

4. Rendering

The rendering involves a two-step approach. For 2D data, we extract the appropriate input image. For 3D scenes, we first render the 3D models from the head-tracked viewer's viewpoint. In the second step, the resultant image is then warped using the quadric image transfer into the projector image space.

4.1 Virtual View

When 3D scenes are displayed on a curved screen, the images are perspectively correct from only a single point in space. We track this 3D location, the virtual viewpoint, with a head-tracking system and update the view as user moves.

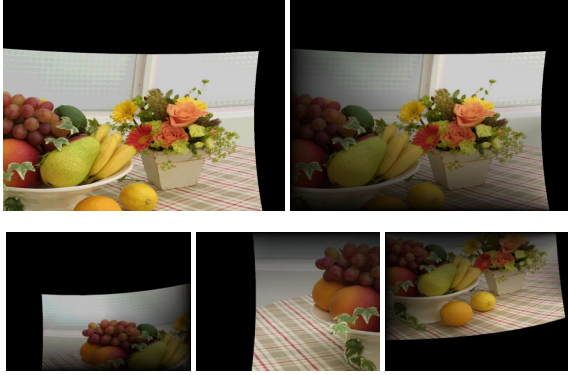We recalculate the quadric transfer, Ψ_i, between virtual

Figure 3: *Top row: Image in top-right projector's framebuffer, before and after attenuation with alpha map Φ_i. Bottom row: Other three projectors with intensity correction. Note the outer black areas which are automatically generated after quadric transfer.*

view image space and projector i framebuffer by cascading two transfers, $\Psi_{0i} \circ \Psi_{h0}$, where Ψ_{0i} is calculated during preprocessing and Ψ_{h0} is updated as the user moves. Ψ_{h0} is calculate using a linear method from Q plus the projection matrices of the camera and the virtual view.

4.2 Display Region

The view frustum for the virtual view is defined using the head-tracked position and the extents of an oriented bounding box (OBB) around the display surface. The look-at vector is from the virtual viewpoint toward the center of the OBB (Note, the view frustum is only used for computing the VirtualViewProjection matrix and Ψ_{h0}, and not for rendering).

We crop the view frustum to an aesthetic shape such as a disk or a screen space rectangle. For 3D applications, we draw a set of black quadrilaterals to cutout the areas outside the desired display region. For example, for a rectangular view, the viewport is made by four large quads near the outer edge of the viewport in the framebuffer. The black quads along with rest of the 3D models get rendered and warped as described below (Figure 3). For 2D applications, areas outside the input image to be displayed are considered black.

4.3 Image Transfer using a Single Pass Rendering

We present a single-pass rendering solution to pre-warp rendered images of a 3D scene before projection on curved screen. A single-pass solution avoids the costly post-rendering warp and it also eliminates the aliasing artifacts common in texture mapping of limited resolution input images. A single-pass solution is possible due to the parametric approach that does not require a look-up table and involves only a small number of parameters.

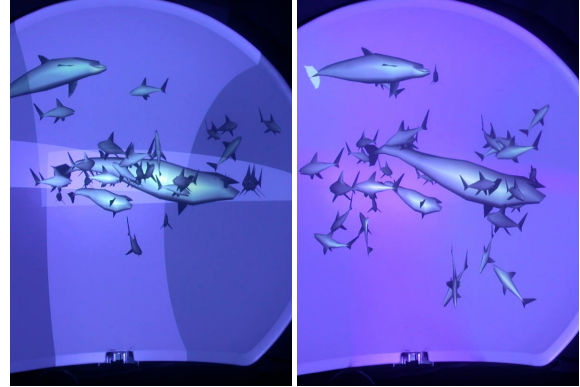Given a 3D vertex, M, in the scene to be rendered, we find

Figure 4: *An unmodified 3D rendering application displayed with correct depth sort and texture mapping without and with intensity blending (Please see the video).*

its screen space coordinates, m, in the virtual view. Then, we find the transferred pixel coordinate, m_i, in the framebuffer of projector i, using the quadric transfer, $\Psi_i = \{A_i, E_i, e_i\}$. The polygons in the scene are then rendered with vertices M replaced with vertices m_i. Thus the rendering process at each projector is the same. Each projector framebuffer automatically picks up the appropriate part of the virtual view image and there is no need to explicitly figure out the extents of the projector.

- At each projector, i
  - ▷ For each vertex M
    - Compute pixel m via VirtualViewProjection(M)
    - Compute warped pixel m_i via quadric transfer $\Psi_i(m)$
  - ▷ For each triangle T with vertices $\{M^j\}$
    - Render triangle with 2D vertices $\{m_i^j\}$

There are two issues with this approach. First, only the vertices in the scene, but not the polygon interiors, are accurately pre-warped. Second, visibility sorting of polygons needs a special treatment.

After quadric transfer, the edges between vertices of the polygon, theoretically, should map to second-degree curves in projector frame buffer. But scan conversion converts them to straight-line segments between the warped vertex locations. This problem will not be discernible if only a single projector is displaying the image. But, overlapping projectors will create individually different deviations from the original curve and hence the edge will appear mis-registered on the display screen. Therefore, it is necessary to use sufficiently fine tessellation of triangles. Commercial systems are already available that tessellate and pre-distort the input models on the fly [Eva02, IZRB97, KQST94] so that they appear straight in a perspectively correct rendering on the curved screen. So our method is compatible with fine tessellation provided by such systems. Pre-distortion of the scene

geometry in commercial systems is used to avoid the two-pass rendering, which involves texture-mapping result of the first pass. In our case, instead of pre-distorting the geometry, we pre-distort the image space projection. Our approach, arguably, is more practical thanks to the programmable vertex shaders now available (see Appendix).

Scan Conversion Issues When pixel locations in the projection of a triangle are warped, information needs to be passed along so that the depth buffer will create appropriate visibility computations. In addition, for perspectively correct color and texture coordinate interpolation, the appropriate 'w' values need to be passed. Our solution is to post-multiply the pixel coordinates with 'w'.

$$
\begin{aligned}
m(x, y, z, w) &= \text{VirtualViewProjection}(M(X)) \\
m_i'(x_i', y_i', w_i') &= \Psi_i(m(x/w, y/w), 1) \\
m_i(x_i, y_i, z_i, w_i) &= [wx_i'/w_i', wy_i'/w_i', z, w]
\end{aligned}
$$

Thus, the z and w values are preserved using explicit assignment. Homogeneous coordinates of m_i have the appropriate final pixel coordinate $(x_i/w_i, y_i/w_i) = (x_i'/w_i', y_i'/w_i')$ due to quadric transfer along with original depth, $(z_i/w_i) = z/w$, and $w_i = w$ values. The corresponding code is in the Appendix and Figure 4 shows a photo of the dome displaying a sample interactive animation that is synchronized and displayed on the dome with four overlapping projectors.

For rendering 2D input images, we densely tessellate the virtual view image space into triangles and map the image as a texture on these triangles. Vertex, m, of each triangle is warped using the quadric transfer into vertex (and pixel) m_i as above. Scan conversion automatically transfers colors and texture attributes at m to m_i and interpolates in between. If required, two-pass rendering of 3D scenes can also be achieved in this manner, by first rendering the input image due to VirtualViewProjection.
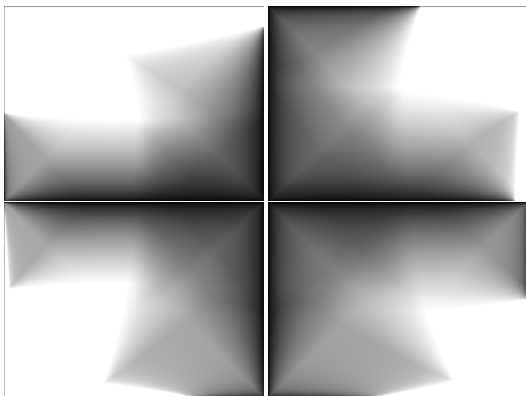

Figure 5: *Calculated alpha maps, Φ_i, for intensity correction of four overlapping projectors.*
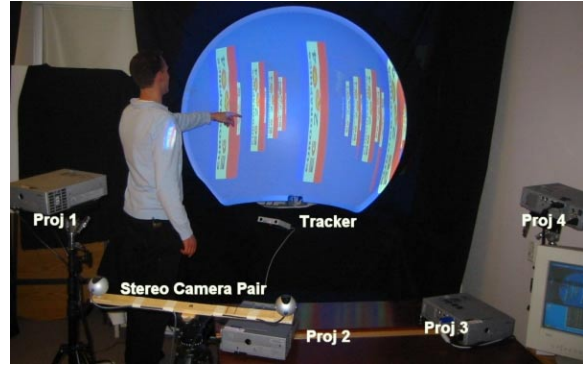
Figure 6: *The setup of four casually installed projectors, stereo camera pair and tracker with a concave dome. The camera pair is not near the sweet spot.*

4.4 Intensity Blending

Pixel intensities in the areas of overlapping projectors are attenuated using alpha blending of the graphics hardware. Using the parametric equations of quadric transfer, the alpha maps are calculated robustly and efficiently.

For every projector pixel, x_i in projector i, we find the corresponding pixels in projector k using the equation

$$
x_k \cong \Psi_{0k}\left(\Psi_{0i}^{-1}(x_i)\right)
$$

For cross-fading, pixels at the boundary of the projector frame buffer are attenuated. Hence, the weights are proportional to the shortest distance from the frame boundary. The weight assigned to pixel x_i, expressed in normalized window pixel coordinates (u_i, v_i) which are in the range $[0, 1]$, is

$$
\Phi_i(x_i) \cong d(x_i)\Big/\sum_k d(x_k)
$$

where, $d(x)$ is $min(u, v, 1 - u, 1 - v)$ if $0 \le u, v \le 1$, else $d(x) = 0$. Thanks to a parametric approach, we are able to compute corresponding projector pixels, x_k, and hence the weights at those locations at sub-pixel registration accuracy. The sum of weights at corresponding projector pixels accurately adds to 1.0.

At each projector, the corresponding alpha map, Φ_i, is loaded as a texture map and rendered as screen aligned quads during the last stage of the rendering (Fig 5).

4.5 Distributed Rendering

We use a Chromium [HHN*02] framework to build a distributed rendering system. Chromium does not support vertex position manipulation or post-rendering warp. So we modified the Chromium server code to include a one-pass rendering scheme implemented via a vertex shader as described above. This allows us to render client applications without modification or re-compiling (Fig. 4). Each server
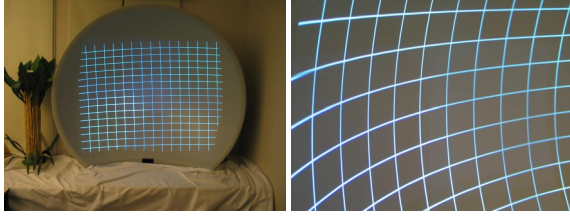
Figure 7: *Registration accuracy. (Left) Four overlapping projectors projecting a grid (Right) Closeup of display region where all four projectors overlap.The embedded images are 2000x1500 pixels. Please zoom with your PDF browser.*

is connected to a projector. We believe one-pass rendering for curved screen is an important development where off-the-shelf applications can be displayed in curved immersive environments without any loss in image quality or decrease in rendering performance.

5. Results

Please see anonymous website http://www.4gotzname.com for more images and higher resolution videos of the results. (The submitted images and videos are lower resolution due to file size limits.)

Our **prototype system** (Fig 6) involves four Mitsubishi X80 LCD projectors, XGA (1024x768) resolution at 1500 lumens. The stereo camera pair is built with two cheap Logitech USB VGA (640x480) resolution webcams which suffer from Bayer color patterning. The color interpolation negatively affects the accuracy of feature location. We intentionally chose the low quality cameras to demonstrate that a parametric approach can overcome feature localization errors as long as the errors are Gaussian distributed and independent. Each projector is assigned to a PC with corresponding Chromium server. The client running on a master PC runs an OpenGL application, unmodified. We used a Origin DynaSight 3-DOF tracker optical tracker [Ori], which tracks a user wearing a retro-reflective dot on the head, to update the rendering at the client. To demonstrate our techniques we use an eLumens VisionStation hemispherical dome with a diameter of 1.5 meter. The accompanying video shows registration and intensity correction for both a concave spherical segment, a well as a convex spherical segment (the back-side of the eLumens dome).

Our **calibration process** is slow compared to the planar homography based systems [BS02] which take just two seconds per projector. We compute the projector pose from near-planar 3D points using an iterative scheme in [LHM00], and the non-linear refinement of $\Psi_{0i} = \{A_i, E_i, e_i\}$ to minimize pixel re-projection errors. Currently our non-optimized code takes about 45 - 60 seconds per projector. The linear optimization for projector pose takes about 4 seconds and the non-linear optimization takes about 40-55 seconds. The time needed for the non-linear optimization increases with the number of features detected from the projected checker
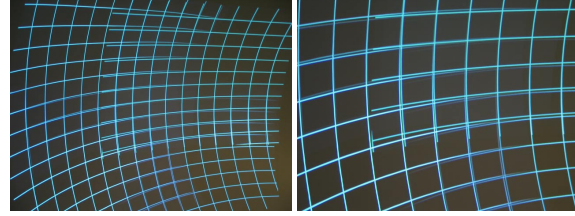

Figure 8: *Registration after linear estimate of quadric transfer. The errors on screen are 15 to 30 pixels.*

patterns. Typically around 100-200 iterations are necessary and we noticed that the iterations converge in most cases.

To determine how well the registration of projectors using pose estimation from near-planar 3D points performs, we compare our results to the pixel **re-projection errors** of a fully calibrated system. In the fully calibrated system we determine internals and externals for the two cameras and the projectors. We use OpenCV library [Int02] to determine the camera internals. The externals are determined from point correspondences of planar surfaces at multiple depths and orientations. Projector internals and externals are then found from triangulated 3D points and corresponding projector image checker corner pixels. Table 1 shows the results of theoretical pixel re-projection errors for the linear estimation of Q before and after non-linear refinement. The setup contains four projectors, Proj1 and Proj3 were closer to the camera(-pair). As mentioned earlier, temporarily installing large surfaces at multiple depth and orientations in not practical in most of the real settings. Hence, in our pseudo-Euclidean method, we assume the same approximate projector internal matrix for all projectors and compute the projector external parameters from the near-planar points observed in the given display setting. The RMS re-projection errors after linear estimation are large (Fig. 8), but this is a good initial guess. After nonlinear refinement, the error is about one pixel which is acceptable. It is important to note that the computed pixel re-projection errors do not directly predict pixel mis-registration errors on the display surface. As seen in Fig. 8 and in the video, after linear estimation, the visible errors on display surface are 15 to 30 pixels although predicted projector to projector re-projection error is about 12 pixels (twice the camera to projector transfer error of about 6 pixels).

We used a 11×9 checker pattern which was optimal for generating sufficient feature points on the screen with reliable detection (Fig. 2). The casual placement of the dome surface results in a small deviation from its intended spherical form. However we demonstrate that our technique still produces accurate results (Fig. 7).

We have also applied our technique for registering the projections on the convex side of the dome surface and on a planar surface (which is a degenerate quadric). This demonstrates the generality of our technique. Projecting on the convex side would be particularly useful for rear-projection. The

	Fully Calibrated		Pseudo-Euclidean	
	Linear	NL Optim	Linear	NL Optim
Proj1	3.5	0.3	5.4	0.73
Proj2	5.3	0.5	7.1	1.02
Proj3	3.7	0.35	5.6	0.79
Proj4	4.8	0.45	6.5	0.91

Table 1: *Comparing results with an ideal system. RMS re-projection pixel error in quadric transfer of camera pixels to projector pixels.*
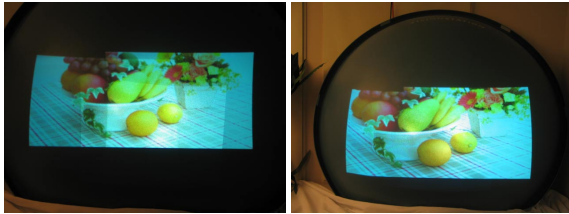

Figure 9: *Three projector system on a convex quadric surface, before and after intensity blending.*

rear-projection with user inside the dome will provide full immersive experience without blocking projectors. Fig. 9 shows a setup of three overlapping projectors. The left image shows registration, the right image also has intensity correction applied. We observed that projected patterns are more distorted due to the convexity of the surface in all directions. Therefore feature detection using checker patterns is more difficult in this case.

6. Conclusion

We have presented techniques that are ideal for building multi-projectors curved displays for 2D or 3D visualization that do not require expensive infrastructure and can be achieved with casual alignment. Our automatic registration exploits quadric image transfer and eliminates tedious setup and maintenance, and hence reduces cost of the system. The techniques can simplify the construction, calibration and rendering process for widely used applications in flight simulators, planetariums and high-end visualization theaters. Our method can also make casual use of multi-projector screens in game arcades or entertainment possible.

The new parametric approach allows an elegant solution to a problem that has so far been solved by discrete sampling. An advantage is that, unlike look-up table based non-parametric approaches [Jarvis97][Raskar98], our cameras do not need to be placed at the sweet-spot. This is important in real-world applications where sweet-spot is rightfully reserved for the human user not cameras or expensive equipment. In some cases, keeping a camera at the sweet-spot

means using a very wide-field of view camera which is expensive and typically suffers from radial or fish-eye distortion.

We are currently working on extending our technique to general curved surfaces or surfaces that have minor deviations from quadric properties. We also would like to extend to the case where quadric transfers cannot be easily cascaded using a small number of cameras, such as wrap-around rear or front projection displays.

(Please see paper videos and images at anonymous website http://www.4gotzname.com)

7. Appendix

Code for Vertex Shader for quadric transfer (in Cg language)

```
vertout main( appin IN, uniform float4x4 modelViewProj, uniform float4 constColor, uniform float3x3 A, uniform float3x3 E, uniform float3 e) {

    vertout OUT;
    float4 m1 = float4(IN.position.x, IN.position.y, IN.position.z,
        1.0f );
    float4 m, mi ; float3 m2,mp; float scale;

    m = mul( modelViewProj, m1);
    m2.x = m.x/m.w; m2.y = m.y/m.w; m2.z = 1;
    scale = mul(m2, mul(E,m2));
    mp = mul(A,m2) + sqrt(scale)*e;
    mi.x = m.w * (mp.x)/(mp.z);
    mi.y = m.w * (mp.y)/(mp.z);
    mi.zw = m.zw;
    OUT.position = mi;
    OUT.color0 = IN.color0; //Use the original per-vertex color specified
    return OUT;

}
```

References

[Alb94]   ALBIN E. F.: Planetarium special effects: A classification of projection apparatus. In *The Planetarian* (1994), vol. 23, pp. 12–14. 2

[BS02]   BROWN M. S., SEALES W. B.: A Practical and Flexible Large Format Display system. In *The Tenth Pacific Conference on Computer Graphics and Applications* (Beijing, China, October 2002), pp. 178–183. 1, 2, 8

[CCC*00] CHEN Y., CHEN H., CLARK D. W., LIU Z., WALLACE G., LI K.: Automatic Alignment of High-Resolution Multi-Projector Displays Using An Un-Calibrated Camera. In *IEEE Visualization 2000* (Salt Lake City, UT, October 8-13 2000). 1

[CNSD93] CRUZ-NEIRA C., SANDIN D., DEFANTI T.: Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE. In *SIGGRAPH 93 Conference Proceedings* (Aug. 1993), vol. 27, pp. 135–142. 1

[CSWL02] CHEN H., SUKTHANKAR R., WALLACE G., LI K.: Scalable Alignment of Large-Format Multi-Projector Displays Using Camera Homography Trees . In *IEEE Visualization* (2002). 1, 2

[CZ98] CROSS G., ZISSERMAN A.: Quadric Reconstruction from Dual-Space Geometry. In *Proc. 6 th Int. Conf. Computer Vision* (1998). 2, 4

[Eva02] EVANS AND SUTHERLAND:, 2002. http://www.es.com. 6

[Fau93] FAUGERAS O.: *Three-dimensional computer vision: a geometric viewpoint*. MIT Press, 1993. 4

[FBZ*99] FRÖHLICH B., BARRASS S., ZEHNER B., PLATE J., GÖBEL M.: Exploring geo-scientific data in virtual environments. *IEEE Visualization* (1999). 2

[FP02] FORSYTH, PONCE: *Computer Vision, A Modern Approach*. 2002. 4

[Fut02] FUTURESLAB:, 2002. ActiveMural, Argonne National Labs. 1

[Hay02] HAYDEN PLANETARIUM:, 2002. http://www.amnh.org/museum/press/feature/rosefactsheet.html. 2

[HHN*02] HUMPHREYS G., HOUSTON M., NG Y., FRANK R., AHERN S., KIRCHNER P., KLOSOWSKI J.: Chromium: A stream processing framework for interactive graphics on clusters. In *Proceedings of SIGGRAPH 2002* (2002). 2, 7

[Int02] INTEL:, 2002. Open Source Computer Vision Library (OpenCV), http://www.intel.com/research/mrl/research/opencv/. 8

[IZRB97] IDASZAK R. L., ZOBEL J., RICHARD W., BENNETT D. T.:. US Patent 6,104,405, Alternate Realities, now Elumens, USA, Feb 26, 1997, 1997. Systems, methods and computer program products for converting image data to nonplanar image data. 6

[Jar97] JARVIS K.: Real Time 60Hz Distortion Correction on a Silicon Graphics IG. *Real Time Graphics 5*, 7 (Feb. 1997), 6–7. 1, 2

[Jup02] JUPITER INC:, 2002. http://www.jupiter.com. 1

[KQST94] KELLY W. A., QUICK L. T., SIMS E. M., TACKABERRY M. W.:. US Patent 5,319,744, General Electric Company, USA, June 7, 1994, 1994. Polygon fragmentation method of distortion correction in computer image generating systems. 6

[LC99] LI K., CHEN Y.: Optical Blending for Multi-Projector Display Wall System. In *Proceedings of the 12 th Lasers and Electro-Optics Society 1999 Annual Meeting* (November 1999). 1

[LHM00] LU C.-P., HAGER G. D., MJOLSNESS E.: Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 6 (2000), 610–622. 5, 8

[MHTW00] MAJUMDER A., HE Z., TOWLES H., WELCH G.: Color Calibration of Projectors for Large Tiled Displays. In *IEEE Visualization 2000* (Salt Lake City, UT, USA, October 2000). 2

[Ori] ORIGIN INSTRUMENTS CORPORATION, 854 GREENVIEW DRIVE, GRAND PRAIRIE, TEXAS 75050-2438 USA:. http://www.orin.com/. 8

[RvB*03] RASKAR R., VAN BAAR J., BEARDSLEY P., WILLWACHER T., RAO S., FORLINES C.: ilamps: Geometrically aware and self-configuring projectors. *ACM Trans. Graph. 22*, 3 (2003), 809–818. 2, 3

[RWF98] RASKAR R., WELCH G., FUCHS H.: Seamless Projection Overlaps Using Image Warping and Intensity Blending. In *Fourth International Conference on Virtual Systems and Multimedia* (Gifu, Japan, Nov. 1998). 1, 2

[SGKM00] STAADT O., GROSS M., KUNZ A., MEIER M.: The blue-c: Integrating real humans into a networked immersive environment. In *ACM Collaborative Virtual Environments 2000* (2000). 1

[SM99] SCOTT K., MCCOLMAN R.: Report of the IPS Technical Committee: Full-Dome Video Systems. In *The Planetarian* (1999), vol. 28, pp. 25–33. 2

[ST97] SHASHUA A., TOELG S.: The quadric reference surface: Theory and applications. *International Journal of Computer Vision (IJCV) 23*, 2 (1997), 185–198. 1, 2, 3, 4

[Sur99] SURATI R.: *Scalable Self-Calibrating Display Technology for Seamless Large-Scale Displays*. PhD thesis, Massachusetts Institute of Technology, 1999. 1, 2

[SZF*99] SAMANTA R., ZHENG J., FUNKHOUSER T., LI K., SINGH J. P.: Load Balancing for Multi-Projector Rendering Systems. In *SIGGRAPH/Eurographics Workshop on Graphics Hardware* (August 1999). 2

[Tri02] TRIMENSION SYSTEMS LTD:, 2002. http://www.trimension-inc.com/ , Cited Dec 2002. 1, 2

[vWRR03] VAN BAAR J., WILLWACHER T., RAO S., RASKAR R.: Seamless multi-projector display on curved screens. *Eurographics Workshop on Virtual Environments (EGVE)* (2003). 2

[WS99] WEXLER Y., SHASHUA A.: Q-warping: Direct Computation of Quadratic Reference Surfaces. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), June, 1999* (1999). 3

[YGHT01] YANG R., GOTZ D., HENSLEY J., TOWLES H.: PixelFlex: A Reconfigurable Multi-Projector Display System. In *IEEE Visualization 01* (San Diego, CA, Oct 2001). 1, 2

[Zha99] ZHANG Z.: Flexible camera calibration by viewing a plane from unknown orientations. In *7th Int. Conference on Computer Vision* (1999), pp. 666–673. 4