

Achieving Seamlessness in Multi-Projector based Tiled Display using Camera Feedback

Pranav Kant Gaur*, Dinesh M. Sarode*, Pritam P. Shete*, Venkata P.P.K*, S.K. Bose*

*Computer Division, Bhabha Atomic Research Centre

{pranav,dinesh,ppshete,panikv,bose}@barc.gov.in

Abstract—High resolution displays utilizing an array of commodity projectors are becoming popular approach(also called *tiled displays*) for visualizing high resolution content like scientific data-sets. These systems provide a relatively cheaper and flexible alternative to displays based on single high resolution monitor or projector. This approach inherently provides a method to create displays with resolution much higher than that possible by using a single high resolution display device. However, bezels in monitor based tiled displays obstruct the geometric continuity of the rendered content. In this paper, we describe the algorithms for geometric alignment of projection regions of an array of arbitrarily placed projectors and attenuation of projected intensities in the overlapping regions of multiple projectors using camera as a feedback device. Combination of these algorithms help us achieve a seamless high resolution display. We have observed improvements in fidelity of the rendered content by utilizing *collinearity* constraint in geometric alignment calculations. We also propose a novel technique based on *cross-ratio* invariant for utilizing full projection region of individual projectors which was limited by the size of features used for geometric alignment in the earlier approaches for planar displays. This also results in more imperceptible edge blending artifacts for same physical setup of projectors.

Index Terms—geometric alignment, edge blending, collinearity, cross-ratio, multi-projector display

I. INTRODUCTION

Tiled displays provide an alternative to the visualization of high resolution content over single large scale high resolution monitor(or projector) by attaining comparable performance using combination of low cost and low resolution commodity

displays. Tiled displays have been realized either using grid of monitors[2], [3] or projectors[1], [5], [6], [7]. Using LCD monitors provides a compact, relatively cheaper and readily configurable high resolution visualization solution, though it still cannot completely imitate a single high resolution display because bezels of monitors obstruct the geometric continuity of the rendered content. Whereas in a multiprojector system, projectors may be arbitrarily placed and they inherently do not enforce any limitations of physical seams. Our work addresses *seamless* rendering of the content using projectors.

Projectors will have quad shaped projection region without necessarily ensuring geometric continuity over the projected content. Manual *keystoning* to geometrically align content from multiple projectors on the screen is time consuming and tedious. Further, it can easily become impractical for large number of projectors. Brown's[1] approach relaxes that assumption and instead allows for a casual positioning of projectors. It aims to create geometrically seamless and rectangular image in camera-space which guarantees geometrical seamlessness on the screen. But it does not guarantee rectangular output on the screen. Further, the usable projection region is limited by the size of features used for geometric alignment. Our contribution through this work is to remove the above mentioned limitations in [1] for the case when projection screen is planar. Further, by utilizing the *collinearity* constraint in geometric alignment calculations we have observed more accurate final rendering.

II. SOLUTION OVERVIEW

We assume an ideal arrangement where all projectors are identical, the principle axes of each projector is orthogonal to the display screen and all are at equal perpendicular distance from the screen. In such an arrangement, we only need to partition the image and assign individual partitions(or rectangular *tiles*) to each projector to get *geometrically continuous* projection of the content on the screen. Effectively, it emulates a single high resolution display.

However, in a practical setup of array of projectors, they are not necessarily identical. Further, they may be arbitrarily positioned and oriented. In this case, the projection on the screen and corresponding image in the projector frame buffer may not be uniformly scaled forms of each other but will depend upon the projector orientation and its position with respect to projection screen. Precisely, the projection region of each projector will be a quad unlike the rectangular one in case of the above mentioned ideal arrangement.

Therefore, to transform this arrangement back to our ideal configuration(without any physical manipulation) we need to determine the the *tile configuration*. The tile configuration for a multiprojector display describes the share of each projector(or the *tile*) in the image to be projected. Further, we also need to determine the distortion we should introduce in each tile so that when projected it remain geometrically continuous with the content projected by adjacent projectors.

To observe such deviations of physical configuration of projectors with respect to the ideal configuration, we use a camera as a feedback device and a reference pattern is projected by each projector. For each projector, camera records the distortion of this reference pattern with respect to ideal projection discussed above. Based on the boundaries of each projection region(called the *bounding box* in camera image), we determine the share of each projector in the image(or the *tile configuration*). Using the distortion information of the reference pattern captured by camera(for each projector) with respect to the image in the projector frame buffer we determine a distortion map called the *warp map*. Finally, system

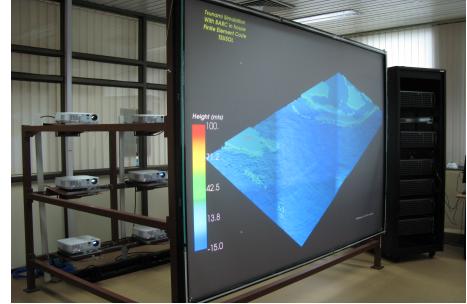


Fig. 1: Our 3X3 multiprojector display system

gives geometrically continuous content by distorting each *tile* using the *warp map*.

Further, the adjacent projectors may have overlap in their projection regions resulting in brighter regions at their boundaries, therefore it is necessary to attenuate(or *edge blend*) the intensity of projected images in such regions so that overall projected content appears *seamless*. Figure 1 shows our display screen with 3X3 rear-projection projector array and rendering cluster driving the display. We place camera in front of the projection screen to capture the projected reference patterns.

III. GEOMETRIC ALIGNMENT

Geometric alignment in [1] limits the usable projection region of each projector to the region within the quad of the outermost features used for geometric alignment. In this work, we propose to utilize *cross-ratio*[11] invariance which is preserved between perspective projection from projector to planar screen and planar screen to camera in order to recover entire projection region of individual projectors. It also eliminates another implicit limitation of [1] which requires adjacent projectors to atleast have overlap among the outermost features of their reference patterns on the screen, this limits the flexibility of positioning of individual projectors.

A. Cross-ratio invariance

Assuming planer perspective projection of features from the projector to screen and from the screen to camera, we can recover co-ordinates of an unknown point(say 'A') in camera image given the coordinates of other

three(i.e.,'B','C','D' are known) using cross-ratio invariant. Let A_p, B_p, C_p, D_p be the coordinates of 4 collinear points in projector space and A_c, B_c, C_c, D_c be the corresponding projection in camera space. Here, B_c, C_c and D_c are known whereas A_c is unknown but A_p, B_p, C_p, D_p are all known. Let us further represent the line joining A_c, B_c, C_c, D_c in parametric form as:

$$\begin{aligned} x_t &= B_c^x + t * (D_c^x - B_c^x) \\ y_t &= B_c^y + t * (D_c^y - B_c^y) \end{aligned} \quad (1)$$

where, (x_t, y_t) are parametric coordinates of an unknown point(in our case A_c)

Utilizing cross-ratio invariance yields following relation:

$$CR_p = \frac{|AC|_c * |BD|_c}{|BC|_c * |AD|_c} \quad (2)$$

Here CR_p is the cross-ratio computed in projector-space, $|ab|_c$ represents Euclidean distance between points a and b in camera-space. Using equation 2, we get quadratic equation with following possible solutions:

$$[t_1, t_2] = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (3)$$

where,

Let,

$$\text{delta} = CR_p * \left(\frac{|BC|_c}{|BD|_c} \right)^2$$

Then,

$$a = (1 - \text{delta}) * |BD|_c^2$$

$$b = 2 * (D_c^x - B_c^x) * (B_c^x - C_c^x) + (D_c^y - B_c^y) * (B_c^y - C_c^y) + 2 * |BD|_c^2 * \text{delta}$$

$$c = |BC|_c^2 - \text{delta} * |BD|_c^2$$

Based on the convention of the chosen coordinate system, only one of t_1 or t_2 will be the valid position of the unknown point A_c . Figure 2 shows a close view of a projected checkerboard(used as *reference pattern* in our work) in camera image. Specifically, in figure 2 an unknown point A_c collinear with B_c, C_c, D_c can be recovered by utilizing equation 1.

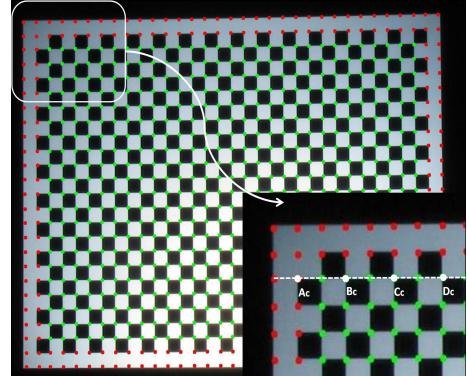


Fig. 2: Recovering A_c using B_c, C_c, D_c

B. Algorithm

Process of geometric alignment is partitioned into two major steps: *feature detection and regularization* and *warp map computation*.

1) Feature detection and regularization: It is used for recording the distortion of reference pattern for each projector. Steps are discussed below:

- 1) Camera captures the checkerboard pattern projected by each projector.
- 2) Standard OpenCV algorithm [4] is used for detecting inner checkerboard corners(*green* points in figure 2).
- 3) Since collinearity is invariant under projective transformation, features collinear in projector image must be collinear in camera space also. In order to account for this, detected locations of features in camera image are regularized by explicitly enforcing collinearity constraint. Specifically, each feature is computed as intersection of vertical and horizontal fitted lines. Figure 3 shows the difference between raw corner detection(figure 3.(a)) and improvements after applying collinearity constraint(figure 3.(b)).
- 4) Once the features are regularized, lost projection region at the boundary are recovered by determining position of *virtual* points(*red* points in figure 2) using cross-ratio constraint(equation 2).
- 5) Estimated coordinates for these *virtual* points are further modified by enforcing the

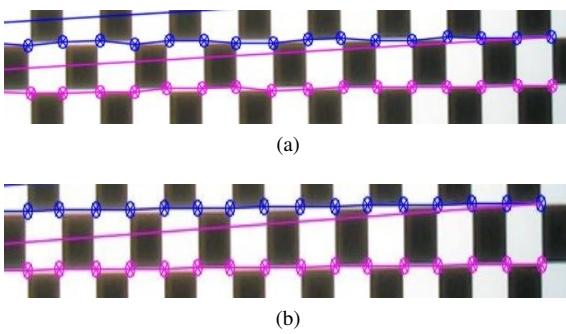


Fig. 3: Enforcing collinearity constraint in camera-space

collinearity constraint.

- 2) Warping map computation: This step computes the distortion map(or, the *warp map*) which is used for modifying the image in projector frame buffer such that when projected it retains geometric continuity of the content. This distortion information is extracted from the features detected earlier. Steps for warp-map computation are discussed below:
 - 1) Camera-to-screen homography[8] is determined to map all subsequent computations to screen space. This step ensures that the final rendered content on the screen will be free from the perspective distortions which get introduced when performing geometric alignment computations entirely in the camera-space.
 - 2) Bounding boxes are computed for each projection region in the screen-space. A bounding box is the convex hull enclosing all features of a projector recovered in feature detection and regularization phase. Figure 4 shows the bounding box for the top-left projector in our display arrangement, for rest of the projectors only bounding boxes are highlighted for the sake of clarity. It should be noted that the bounding box of top left projector has covered its entire projection region because we determined the position of *virtual* points as well.
 - 3) A *maximal* box is computed which can contain any local bounding box. Naturally, it cor-

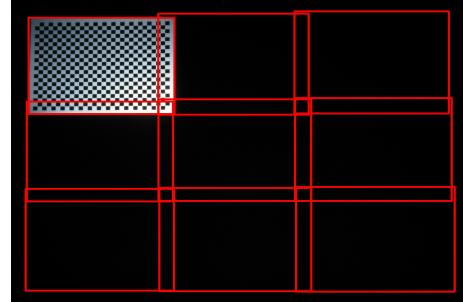


Fig. 4: Bounding boxes for all projectors

responds to the box having maximum width and height among that of all local bounding boxes. All local bounding boxes are modified to have maximum width and height.

- 4) These width and height are equated with width and height of individual projector frame buffers to maintain uniform pixel size regardless of different sizes of projection regions of the projectors on the screen.
- 5) A global bounding box is computed which is the convex hull enclosing all local bounding boxes. This global bounding box acts like a global tile coordinate system containing all local bounding boxes(or tiles) and hence all the detected features. This determines the final *tile configuration*.
- 6) All detected features are mapped to global tile coordinate system to be used for deriving *warp map* information.

IV. EDGE BLENDING

Edge blending algorithm utilizes the projector-overlap information acquired during geometric alignment phase for attenuating the projector image intensity in the overlap regions.

A. Algorithm

Projector image intensity is attenuated in the overlap region by scaling the pixel values using *alpha map*. An alpha map is a 2D array with one value(called *alpha weight*) for each projector pixel. Steps for edge blending are discussed below:

- 1) For each projector, the coordinates of detected feature relative to the global bounding box are

- computed. It maps the points from camera coordinate system to global bounding box coordinate system.
- 2) A *tightest* polygon bounding the detected features for each projector is determined.
 - 3) Each pixel (x_g, y_g) in the global bounding box(or global coordinate frame) is tagged with the index of projectors sharing it.
 - 4) For each projector p , *alpha weight* is computed at each pixel (x_p, y_p) using *distance transform* as:

$$\alpha(p, x_p, y_p) = \frac{d_p(x_p, y_p)}{\sum_{i \in \text{shares}(i, x_g, y_g)} d_i(x_i, y_i)} \quad (4)$$

where, (x_g, y_g) are global frame coordinates for pixel (x_p, y_p) in projector p and for pixel (x_i, y_i) in projector(s) i , $d_p(x_p, y_p)$ (and similarly $d_i(x_i, y_i)$) represents the perpendicular distance of the *closest* boundary of projection region of projector p (and similarly projector(s) i) from pixel (x_p, y_p) (similarly, (x_i, y_i)), proposition $\text{shares}(i)$ is *true* for any projector i which shares pixel (x_g, y_g) in the global coordinate frame,

- 5) In order to account for non-linearities in projector's output response, an additional factor *gamma*[12] is applied to alpha weights as:

$$\alpha(p, x_p, y_p)_\gamma = \alpha(p, x_p, y_p)^{\frac{1}{\gamma(p(x_p, y_p))}} \quad (5)$$

where, $\alpha(p, x_p, y_p)$, $\gamma(p(x_p, y_p))$ and $\alpha(p, x_p, y_p)_\gamma$ represent the alpha weight, gamma value and gamma corrected alpha weight for projector p at pixel (x_p, y_p) respectively.

V. PRACTICAL IMPLEMENTATION

In this section, we give brief details about the hardware setup and software implementation of our multiprojector display system.

A. Hardware

The seamless display has been achieved by using 120" rear projection screen and arranging the 9 commodity DLP projectors in 3x3 matrix form. The projectors are arranged such that each projector

projects an image of more than 40" from the rear. The 9 projected images of projectors are overlapped at the edges and collectively cover the entire screen. The display has aspect ratio of 4:3, screen dimensions of 8' x 6' and the resolution of 7 mega pixels. The display is driven by a cluster of 3 graphics workstations. Each graphics workstation drives 3 projectors through 3 graphics cards.

B. Software

We use open source software - Chromium and DMX as a software environment to realize the 3x3 projector based seamless display. We use Chromium for running OpenGL based graphics applications and DMX for running other X based desktop applications on the seamless display. We use TiledRenderer[10] - an in house MPI based parallel rendering software for viewing the high resolution scientific visualization movies or images. We have customized Chromium , DMX and Tile-dRenderer to use the warp maps and alpha maps generated by the above algorithm to pre-warp and blend the images to be displayed by individual projectors.

VI. RESULTS

On our test setup $\sim 9.6\%$ more projection region was utilized using our approach. Further, our approach allows for more inter-projector overlap leading to more realistic alpha maps as can be compared in figures 5 and 6. Consequently, boundaries are more visible in figure 5 as compared to those in figure 6. To quantify geometric alignment accuracy of our system we projected a grid on the display with a resolution of 200 cells per unit area and measured the misalignment at the junctions between adjacent projectors. We found that the average misalignment error for our system was about: $\sim 1mm$.

VII. CONCLUSION

We have described an enhanced multiprojector geometric alignment utilizing *Cross-ratio* invariance based projection region recovery. Brown's method puts a constraint in which atleast outer features of adjacent projector must overlap in order to get geometrically continuous image on the screen. This actually limits their claimed merit of



Fig. 5: Projection using Brown's[1] approach



Fig. 6: Projection using our approach utilizing cross-ratio constraint

casual alignment of projector. Further, there was no provision for using projection region beyond outermost features. Our approach allows for recovery of artificial feature points at the boundary of each projector's projection region due to which full projection region can be utilized and there is no constraint on projector positioning. Further, utilizing full projection region of each projector increases the area of overlap between adjacent projectors resulting in more uniform edge blending.

REFERENCES

- [1] Brown, M.S.; Seales, W.B., "A practical and flexible tiled display system," Computer Graphics and Applications, 2002. Proceedings. 10th Pacific Conference on , vol., no., pp.194,203, 2002
- [2] Bose, S. K.; Sarode, D.M.; Venkata, P.P.K.; Shete, P.P.; Apte, A. G., "High End Scientific Visualization with Scalable Display System," Advances in Computer Engineering (ACE), 2010 International Conference on , vol., no., pp.316,318, 20-21 June 2010
- [3] Nirmimesh; Harish, P.; Narayanan, P. J., "Garuda: A Scalable Tiled Display Wall Using Commodity PCs," Visualization and Computer Graphics, IEEE Transactions on , vol.13, no.5, pp.864,877, Sept.-Oct. 2007
- [4] Bradski, Dr. Gary Rost and Kaehler, Adrian, "Learning OpenCV", 1st Edition, 2008, O'Reilly Media, Inc.
- [5] Hereld, M.; Judson, I.R.; Stevens, R.L., "Introduction to building projection-based tiled display systems," Computer Graphics and Applications, IEEE , vol.20, no.4, pp.22,28, July-Aug. 2000
- [6] Raskar, R.; Brown, M.S.; Ruigang Yang; Wei-Chao Chen; Welch, G.; Towles, H.; Scales, B.; Fuchs, H., "Multiprojector displays using camera-based registration," Visualization '99. Proceedings , vol., no., pp.161,522, 29-29 Oct. 1999
- [7] Brown, M.; Majumder, A.; Ruigang Yang, "Camera-based calibration techniques for seamless multiprojector displays," Visualization and Computer Graphics, IEEE Transactions on , vol.11, no.2, pp.193,206, March-April 2005
- [8] Dubrofsky, Elan. "Homography estimation." PhD diss., UNIVERSITY OF BRITISH COLUMBIA, 2009.
- [9] Greg Humphreys, Mike Houston, Ren Ng, Randall Frank, Sean Ahern, Peter D. Kirchner, and James T. Klosowski. 2002. Chromium: a stream-processing framework for interactive rendering on clusters. ACM Trans. Graph. 21, 3 (July 2002), 693–702.
- [10] "TiledRenderer - an Object Oriented Framework for MPI Based Parallel Rendering", Pritam Prakash Shete, Venkata P P K, Dinesh Sarode, Mohini Laghate, Surojit Bose./Second International Conference on Parallel Distributed and Grid Computing (PDGC -2012), Shimla, India/, [2012]
- [11] Hartley, R.I. and Zisserman, A., "Multiple View Geometry in Computer Vision", Second edition, 2004, Cambridge University Press
- [12] Edge blending using commodity projectors
http://paulbourke.net/texture_colour/edgeblend/