Multi-Projector Image Correction on the GPU

Tyler Johnson, Florian Gyarfas, Rick Skarbez, Patrick Quirk,
Herman Towles, and Henry Fuchs
Department of Computer Science
University of North Carolina at Chapel Hill

ABSTRACT

Multi-projector displays require a variety of image correction techniques. Geometric distortions resulting from projecting onto non-planar display surfaces and non-linear properties of projector lenses must be corrected to produce usable images. Photometric issues such as varying luminance response between projectors and brightness discontinuities in projector overlap regions must also be corrected in a high-quality display. We show how the programmability of modern graphics hardware can be used to efficiently implement several image correction techniques for multi-projector displays.

1. INTRODUCTION

Multi-projector displays are faced with the challenge of using multiple display devices to produce what appears to the user to be a single display. If the devices are casually placed, the individual contributions of each device should combine to form an image that is free of geometric distortions and photometric inconsistencies.

Geometric distortions in projective displays can arise from projecting onto non-planar display surfaces and also from projector lens distortions. Lens distortions can prevent precise registration of imagery between projectors resulting in ghosting effects which blur high-frequency detail.

In addition to geometric distortions, photometric non-uniformities such as brightness discontinuities in projector overlap regions and color gamut and luminance differences between projectors can detract from the feeling of immersion the display is able to provide.

In the following sections we describe how techniques for correcting common geometric distortions and photometric non-uniformities in multi-projector displays can easily and efficiently be corrected in a common framework using programmable graphics hardware.

2. GEOMETRIC CORRECTION

In this section we consider GPU accelerated techniques for correcting distortions due to non-planar display surface geometry and non-linearities in projector lenses.

2.1 Correcting Perspective Distortions

If the display surface is more complex than a simple plane, the resulting imagery can be so distorted as to be unusable. A number of techniques have been developed for correcting this type of distortion.

Bimber performs geometric correction in [1] using look-up tables measured during calibration which map the pixels of a projector to pixels in the desired image so that distortions are eliminated at the user viewpoint. These look-up tables are stored on the GPU and used by a fragment program to perform correction each frame.

Since a major goal of our research is to use convenient display surfaces to provide an immersive experience to tracked users, we use the two-pass projective texturing approach described in [5], which easily accommodates dynamic viewpoints. The approach uses a viewing frustum assigned to the viewer to projectively texture the display surface geometry with the desired image produced in the first pass. The texture-mapped geometry is then rendered from the viewpoint of the projector using a projection matrix calculated during system calibration. The result is a pre-distorted image which will appear undistorted from the viewing position when displayed by the projector.

We perform this projective texturing step on the GPU in a vertex program. Each vertex of the display surface model is transformed by the projection matrix of the user followed by a viewport transform to produce a texture coordinate in the ideal image. The vertex is then transformed into clip space using the projection matrix of the projector.

2.2 Correcting Lens Distortions

Projector lens distortion is generally left unconsidered in multi-projector displays since projector lenses typically exhibit little distortion. Radial lens distortions of only a few pixels can nevertheless result in slight misregistrations which blur high-frequency detail. Also, radial distortions are more significant near the periphery of the lens, which in multi-projector displays is typically where projector imagery overlaps and accurate registration is the most important.

Our projector calibration process produces parameters modeling both radial and tangential lens distortion. This is the same model proposed by Brown in [2] and used by OpenCV and the Matlab Camera Calibration Toolbox. To correct lens distortion, we pre-distort the image to be projected by the inverse of the distortion it will undergo as it passes through the lens. Pre-distortion is performed each frame in a fragment program using a pre-computed look-up table

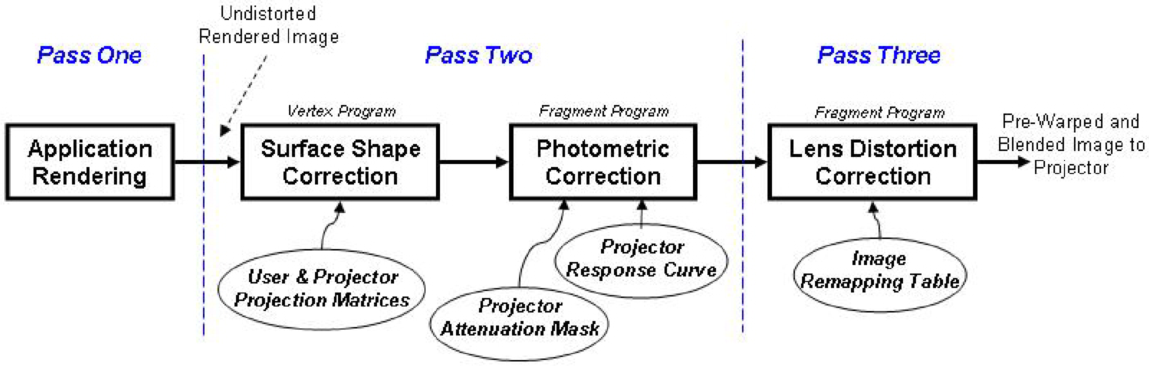

Figure 1: Block Diagram of the Image Correction Pipeline

stored on the graphics card which describes how each output pixel in the pre-distorted image maps to a sub-pixel location in the input image.

3. PHOTOMETRIC CORRECTION

Even when all geometric distortions have been corrected, photometric non-uniformities can still detract from a user's feeling of immersion in a multi-projector display environment. In this section we describe how programmable graphics hardware can be used to attenuate projected imagery to remove areas of inconsistently high-brightness in image overlap regions.

Our calibration process creates an attenuation mask for each projector which describes how its output intensity should be modified at each pixel so that overlapping imagery on the display surface has the brightness of a single projector. Methods of generating attenuation masks for use in multi-projector displays are described in [3] and [5]. Since an attenuation mask describes the amount to attenuate photonic output at each pixel, we must correct for non-linearities in the luminance response of the projector.

If R is the channel independent luminance response of the projector and i_c the intensity of channel c for a pixel in the image to be projected, the modified intensity i'_c that must be input to the projector to attenuate its output by α is

$$i'_c = R^{-1}(\alpha \cdot R(i_c)). \tag{1}$$

Methods of automatically measuring the luminance response of a projector are described in [3] and [4]. After measuring the luminance response of each projector, we additionally calculate the inverse of the response function. To apply intensity correction, we discretize the response and inverse response functions and store them as 1D textures on the graphics card along with the image-resolution attenuation mask. During run-time, the response textures and attenuation mask are passed to a fragment program which applies (1) for each channel of each projector pixel.

4. CORRECTION PIPELINE

Our research group has developed a system which combines GPU implementations of perspective distortion correction and the photometric correction technique described above in a single rendering pass. We are currently in the process of adding to this system our GPU implementation of projector lens distortion correction. The combined system we envision is depicted in Figure 1. Lens distortion correction will be incorporated into the system as an additional third pass with the previous correction techniques remaining unmodified.

5. CONCLUSIONS

We have shown how different image correction techniques typically applied in multi-projector displays can be performed efficiently in a common framework using programmable graphics hardware. The GPU natively supports the filtering of discretized data necessary to prevent aliasing when performing image correction. The most recent generation of graphics cards also support 32-bit floating point values throughout the graphics pipeline, effectively eliminating quantization errors that may occur when performing photometric correction.

While programmable graphics may not be explicitly necessary for performing some of the geometric and photometric correction techniques discussed, it allows all image correction techniques to be combined in a common, high-performance framework. Also, when implemented in an API-independent high-level shading language such as Cg, correction can be supported on multiple platforms with little or no modification.

6. REFERENCES

[1] O. Bimber, G. Wetzstein, A. Emmerling, and C. Nitschke. Enabling view-dependent stereoscopic projection in real environments. In *Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'05)*, pages 14–23, 2005.

[2] D. Brown. Lens distortion for close-range photogrammetry. In *Photometric Engineering*, volume 37, pages 855–866, 1971.

[3] A. Majumder and R. Stevens. LAM: Luminance attenuation map for photometric uniformity in projection based displays. In *ACM Virtual Reality and Software Technology (VRST)*, 2002.

[4] A. Raij, G. Gill, A. Majumder, H. Towles, and H. Fuchs. Pixelflex2: A comprehensive, automatic, casually-aligned multi-projector display. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS-2003)*, 2003.

[5] R. Raskar, M. S. Brown, R. Yang, W.-C. Chen, G. Welch, H. Towles, W. B. Seales, and H. Fuchs. Multi-projector displays using camera-based registration. In *IEEE Visualization*, pages 161–168, 1999.