GVU-PROCAMS: Enabling novel projected interfaces.

Jay W. Summet, Matthew Flagg
James M. Rehg and Gregory D. Abowd
GVU Center & College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
{summetj,mflagg,rehg,abowd}@cc.gatech.edu

Neil Weston
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332
neil.weston@ae.gatech.edu

ABSTRACT

Front projection allows large displays to be deployed relatively easily. However, it is sometimes difficult to find a location to place a projector, especially for ad-hoc installations. Additionally, front projection suffers from shadows and occlusions, making it ill-suited for interactive displays. The GVU-PROCAMS system allows programmers to deploy projectors and displays easily in arbitrary locations by enabling enhanced keystone correction via warping on 3D hardware. In addition, it handles the calibration of multiple projectors using computer vision to produce a redundantly illuminated surface. Redundant illumination offers robustness in the face of occlusions, providing a user with the experience of a rear-projected surface. This paper presents a stand-alone application (WinPVRP) and a programming system (GVU-PROCAMS) that easily allows others to create projected displays with enhanced warping and redundant illumination.

Categories and Subject Descriptors

B.4.2 [**Input/Output and Data Communications**]: [Devices - Image display]; D.2.13 [**Reusable Software**]: Reusable libraries

General Terms

Human Factors, Design

Keywords

Projection, Warped projection, Redundant projection

1. INTRODUCTION

Computer-generated visual output is a key aspect of most user interfaces. Non-projected display technologies (such as LCD and plasma display panels) can be used to produce small-to medium-sized displays, but become prohibitively expensive as the display size increases. Projectors (DLP/LCD) are an economical method to produce larger displays. Front projection allows easy deployment of large-scale displays, but suffers from the problem of occlusions which can create shadows on the display. Rear projection is the accepted method for delivering digital output on large scale interactive surfaces such as electronic whiteboards due to its ability to produce a shadow-free display. Unfortunately, rear projection is expensive. Expensive translucent screens, the cost of installing these screens, and the cost of space for the projector rooms behind the screens make rear projection installations cost prohibitive.

MM'06, October 23–27, 2006, Santa Barbara, California, USA.
ACM 1-59593-447-2/06/0010.

Figure 1: An interactive game using redundant illumination provided by GVU-PROCAMS. The redundant illumination prevents shadows from hampering the game-play.

Alternatively, front projected displays allow for more flexible deployments that can be installed, moved, or removed quickly and integrated into pre-existing spaces. Front projection allows for displays to be deployed on pre-existing (walls, floors) or improvised (sheets, screens, paper) surfaces. It also allows the augmentation of objects in the environment such as retail displays or painting canvases [8, 3]. When quickly deploying a projected display, it can sometimes be difficult to mount the projector in a position that allows pre-existing surfaces to be used optimally. In addition, front projection can suffer from shadows cast by users as they move within the environment, especially if they interact with the display.

In an effort to allow users to develop and deploy projected interfaces easily, we have developed a software application, WinPVRP, which provides redundant illumination of a standard PC desktop, build on top of the GVU PROjector/CAMeraS (GVU-PROCAMS) system. It allows one or more projectors to be placed in arbitrary positions with respect to the intended display surface. Multiple projectors can be calibrated via computer vision to provide redundant illumination for a display, providing a brighter display for use in illuminated environments, as well as providing robustness in the face of occlusions to prevent shadows. Figure 1 shows a redundantly illuminated display being used for an interactive game. Both the WinPVRP application and GVU-PROCAMS are available to download at: *http://www.cc.gatech.edu/cpl/procams.*

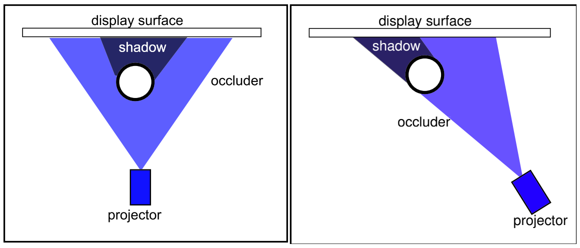

Figure 2: Illustration of a Front Projected Display (FP) and a Warped Front Projection (WFP) display. The enhanced keystone correction allows more freedom in projector placement.

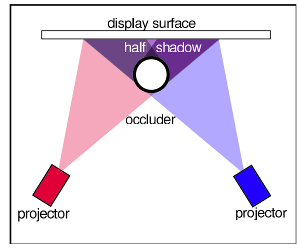

Figure 3: A redundantly illuminated display (Passive Virtual Rear Projection) uses two or more projectors to increase brightness and provide robustness in the face of occlusions and shadows.

1.1 GVU-PROCAMS Abstractions

GVU-PROCAMS supports two main features: enhanced keystone correction via warping, and the calibration needed to align multiple redundant projectors into a redundantly illuminated display. It abstracts the 3D programming, camera access API's, and computer vision techniques needed by programmers to deploy novel projected applications quickly. These programming abstractions allow a programmer to concentrate on the application functionality, not the graphics and computer vision programming needed to display it from multiple, arbitrarily-positioned projectors.

In the simplest case, GVU-PROCAMS allows a programmer to warp the output of a single projector onto an arbitrary planar surface using a projective transform performed by the accelerated 3D video card (See section 2.2, and Figure 2). This *warped front projection* (WFP) allows a projector to be placed in an arbitrary location with respect to the display surface.

Although WFP can be a useful tool to easily position projectors, redundant illumination is the key feature provided by GVU-PROCAMS that can not be implemented with other hardware or software. Redundant illumination allows users to approach the display surface without completely occluding the display with their own shadows, providing a user experience similar to rear projection. Figure 1 shows users interacting with a redundantly illuminated display which is robust to shadows. This display was created by adding a camera and second projector to the system. GVU-PROCAMS handles the computer vision needed to calculate a homography (a mathematical function which transforms points from one plane to another using a projective transform) between each projector and the camera. By using the camera's view as a frame of reference, multiple projectors can be calibrated so that their output overlaps on the display screen (Figure 3). We call this redundant illumination *passive virtual rear projection* (PVRP) as it provides a user experience similar to a rear projected screen[10].

2. GVU-PROCAMS APPLICATIONS

We have used the GVU-PROCAMS to build dedicated applications (such as the interactive game in Figure 1, and the banner display in section 2.2) as well as the WinPVRP application. The WinPVRP program is a solution for end-users attempting to implement a warped front projection or passive virtual rear projection display. Programmers can download and use the underlying C++ GVU-PROCAMS system to experiment with multi-projector systems and build custom applications.

2.1 Redundant Illumination - WinPVRP

At our institution, the School of Aerospace Engineering has retrofitted a classroom into a COllaborative DEsign environment (CODE). The CODE provides student design teams experience solving design problems in collaborative team rooms, which are becoming more common in the workplace. The design of the CODE includes several interactive wall sized computer displays. However, because of space and cost constraints, rear projection screens could not be installed. We used GVU-PROCAMS to build a Windows tray application that allows a standard Windows desktop to be projected using passive virtual rear projection (Figure 3).

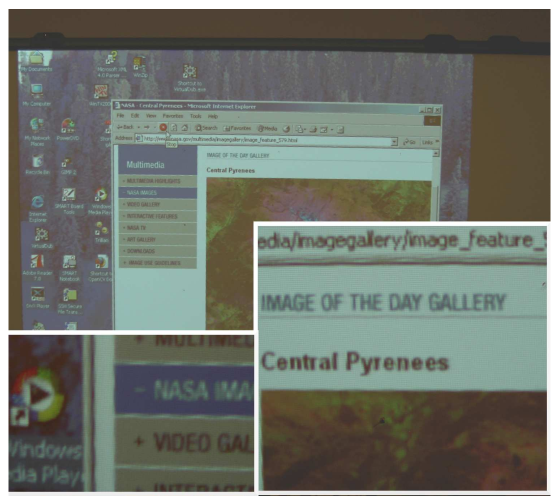

Figure 4: The WinPVRP application provides camera based calibration of dual projectors to provide a display redundantly illuminated from two projectors (top). The calibration accuracy can be seen in the two enlarged call outs (bottom).

The WinPVRP application allows users with a Windows desktop and two projectors (3 total video ports) to create a passive virtual rear projected display using any Video for Windows device (such as a USB webcam) to calibrate the two projectors[1]. A Pentium M 1.4Mhz laptop can run the WinPVRP application (in WFP mode) at 10fps, and the majority of the CPU cycles are used in making copies of the windows desktop and moving it over the PCI bus. Projecting a static image can be accomplished at 60fps, limited by the vertical refresh rate of the projector. If the WinPVRP application only detects a single projector, it will automatically fall back into

[1] Manual calibration of two projectors is also possible, but use of a camera greatly speeds the process.

warped front projection mode. WinPVRP provides an easy way to take an existing Windows application and project it onto a touch-sensitive interactive surface using passive virtual rear projection so that user's shadows do not occlude the display.

2.2 Warped Front Projection - Banner Display

The Banner program reads lines of text from a file and renders the text onto a sign. We used it to implement a Trolley Timer (Figure 5), which displays the predicted wait time for the next few trolleys at the stop outside of our building (using GPS data). The best place to locate the Trolley Timer sign was on a hallway wall at a "T" intersection. This location was chosen due to the location of windows and doors that precluded other locations, as well as the normal traffic flow patterns in the building. Unfortunately, the hallway at right angles to the chosen wall had no good locations to place a projector. The banner application, created using GVU-PROCAMS allows the user to position the display at the desired location, while placing the projector at an extreme off-axis angle. The projector was placed in an existing cabinet, and GVU-PROCAMS allowed the sign to be projected correctly in the desired location.

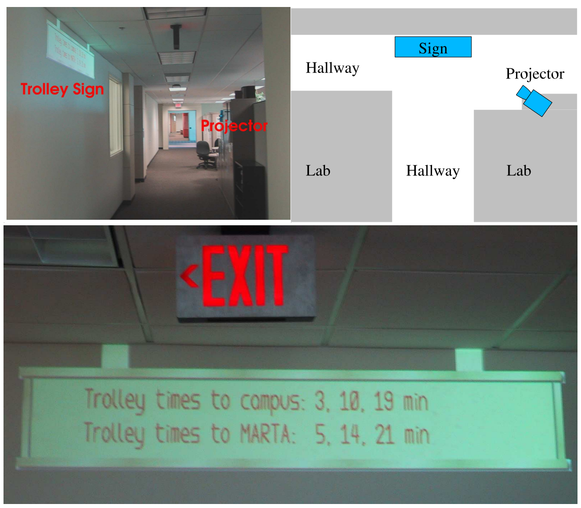

Figure 5: Trolley Timer sign environment and floor-plan.

3. GVU-PROCAMS ARCHITECTURE

GVU-PROCAMS has three main functional components with which a programmer interacts:

1. MultiProjectorSurface - This object represents a single display "surface" which can be made up of one or more projected outputs. The user adds cameras and projected outputs to this object, and it handles the computer vision needed for calibrating multiple projectors. The MultiProjectorSurface also provides user interface mechanisms for an end user to position the display interactively using the mouse.

2. GenericInput - GVU-PROCAMS supports three different camera API's: Video For Windows, Matrox Imaging Library (MIL), and the CVCam interface provided by OpenCV. This allows various USB webcams and more professional IEEE 1394 (Firewire) cameras to be used. Each camera interface is a subclass of GenericInput. A user creates an object to interface with the specific camera they have, and passes it to the MultiProjectorSurface via the *addCamera* method after casting it as a GenericInput.

3. WinD3DOutput - This object handles full-screen window creation and image warping using the 3D graphics card. Programmers use the WinD3DOutput object to "grab" one or more video ports (connected to projectors) in full-screen mode. The WinD3DOutput object is then given to the MultiProjectorSurface, which creates the display.

Figure 6 shows the data flow through these three components. In addition to these three programmer visible objects, the math and vision routines needed to calibrate multiple projectors and calculate the appropriate projective transform to warp their outputs are encapsulated within three objects that are used internally by GVU-PROCAMS. These objects are hidden from the casual programmer:

1. Homography - These objects encapsulate the math needed to calculate a homography between two planes. It is used by the Cameras2Screen object to calculate the relationship between projectors and cameras, as well as by the WinD3DOutput object to calculate the appropriate warping for a projected image. The Homography object will also be useful to advanced programmers who wish to calibrate any two planes, such as an input surface and a projected display.

2. BgsDotFinder - This object uses GenericInput objects to access a camera feed and encapsulates a background subtraction and "Dot Finder" computer vision algorithm. It is used by the Cameras2Screen object to detect projected calibration patterns. Advanced programmers can use the background subtraction routines from this object, useful as the first step in detecting human activity.

3. Cameras2Screen - This object handles the projection of calibration patterns, their detection via a camera, and the calibration and alignment of multiple projectors into a redundantly illuminated display.

To create a projected application, programmers allocate one or more projectors (via the WinD3DOutput object), an optional camera (via one of the Input objects, cast to a GenericInput) and give these objects to a MultiProjectorSurface, which handles the calibration and user interface for display placement. From that point forward, the programmer is free to create the desired graphics which are handed off to the MultiProjectorSurface via a *drawImage* method. GVU-PROCAMS also allows programmers to save calibration state between program executions. This allows projector calibration and/or display placement to be done only on initial setup or when projectors are moved.

4. RELATED WORK

A few commercial projectors such as the 3M Digital Wall Display [11] and the NEC-WT600 [12] use specialized optics to warp their output. High end projectors with lens-shift or geometry processing video chips support similar warping to correct keystone artifacts, but do not support warping to as full an extend as GVU-PROCAMS. All of these projectors are much more expensive than commodity projectors intended for the general business and home users. The latest nVidia graphics card drivers support a feature called nvKeystone, which can be used to implement warped front projection. However, this warping can not be programaticly controlled, and hence is limited to a single surface. It also requires that you use a specific brand of graphics card. None of these solutions provide redundant illumination.

Researchers at the University of Kentucky have demonstrated software that corrects for shadows on multi-projector displays [5,

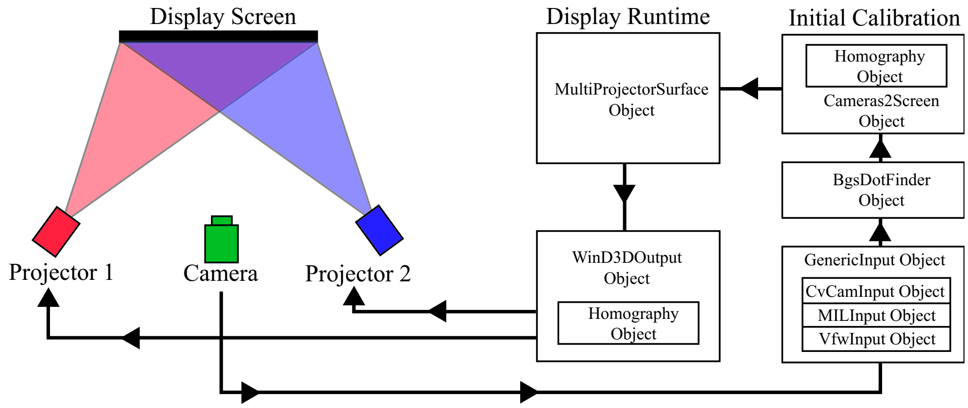

Figure 6: Architecture diagram of GVU-PROCAMS showing data flow for calibration and use.

6]. Raskar *et al.* have demonstrated projection on curved surfaces and multiple adjoining planar surfaces [9]. These research efforts have not yet resulted in the release of source code or libraries for reuse by others. The CHT multi-projector display calibration toolkit from Princeton allows the construction of large multi-projector display walls, but does not provide for redundant illumination[1].

Rear-projection systems support seamless interaction (e.g. the Xerox Liveboard [2], or Stanford Interactive Mural [7]), but can be prohibitively expensive and difficult to deploy because of the need for custom installation of a large screen and the significant space required behind the screen.

We have developed an algorithm that dynamically switches illumination from an occluded projector to an unoccluded projector, preventing light from falling on users[4]. This algorithm will be integrated into distribution 4 of the toolkit as part of the MultiProjectorSurface object, requiring minimal coding changes for existing applications to add dynamic shadow elimination.

5. SUMMARY

GVU-PROCAMS allows programmers to create applications that use one or more projectors in arbitrary positions and orientations. It allows programmers to use a camera to automatically calibrate multiple projectors into a redundantly projected display. These virtual rear projection displays provide the dual advantages of enhanced brightness and reduced shadows. GVU-PROCAMS takes advantage of accelerated 3D hardware and cameras attached to the computer, allowing programmers to use them without knowing graphics programming or camera APIs. It abstracts the computer vision needed to calibrate multiple projectors using the cameras into a single function call. It allows the user to adjust the calibration of projectors or the position of the display. The WinPVRP application allows users to easily set up passive virtual rear projected and warped front projected displays with no programming.

6. REFERENCES

[1] Han Chen, Rahul Sukthankar, Grant Wallace, and Kai Li. Scalable alignment of large-format multi-projector displays using camera homography trees. In *Proceedings of Visualization 2002*, pages 339–346, 2002.

[2] Scott Elrod, Richard Bruce, Rich Gold, David Goldberg, Frank Halasz, William Janssen, David Lee, Kim McCall, Elin Pedersen, Ken Pier, John Tang, and Brent Welch. Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 599–607. ACM Press, 1992.

[3] Matthew Flagg and James M. Rehg. Projector-guided painting. In *Proceedings of User Interface Software and Technology (UIST)*, To Appear - Montreux, Switzerland, October 2006.

[4] Matthew Flagg, Jay Summet, and James M. Rehg. Improving the speed of virtual rear projection: A GPU-centric architecture. In *IEEE International Workshop on Projector-Camera Systems (PROCAMS), held in conjunction with IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 25th, 2005.

[5] Christopher Jaynes, Stephen Webb, and R. Matt Steele. Camera-based detection and removal of shadows from interactive multiprojector displays. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 10(3):290–301, May/June 2004.

[6] Christopher Jaynes, Stephen Webb, R. Matt Steele, Michael Brown, and W. Brent Seales. Dynamic shadow removal from front projection displays. In *Proceedings of the conference on Visualization 2001*, pages 175–182. IEEE Press, 2001.

[7] Brad Johanson, Armando Fox, and Terry Winograd. The interactive workspaces project: Experiences with ubiquitous computing rooms. *IEEE Pervasive Computing*, 1(2), 2002.

[8] Claudio Pinhanez and Mark Podlaseck. To frame or not to frame: The role and design of frameless displays in ubiquitous applications. In *Proceedings of Ubiquitous Computing (Ubicomp)*, pages 340–357, 2005.

[9] Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. ilamps: geometrically aware and self-configuring projectors. *ACM Trans. Graph.*, 22(3):809–818, 2003.

[10] Jay Summet, Gregory D. Abowd, Gregory M. Corso, and James M. Rehg. Virtual rear projection: Do shadows matter? In *CHI'05 Extended Abstracts*, pages 1997–2000, 2005.

[11] 3M Digital WallDisplay. http://www.3m.com/.

[12] NEC WT-600. http://www.nec-pj.com/products/wt/.