A Foveal Inset for Large Display Environments

Benjamin A. Ahlborn Oliver Kreylos Bernd Hamann Oliver Staadt

University of California, Davis
Department of Computer Science
Institute for Data Analysis and Visualization
{ahlborn,kreylos,hamann,staadt}@cs.ucdavis.edu

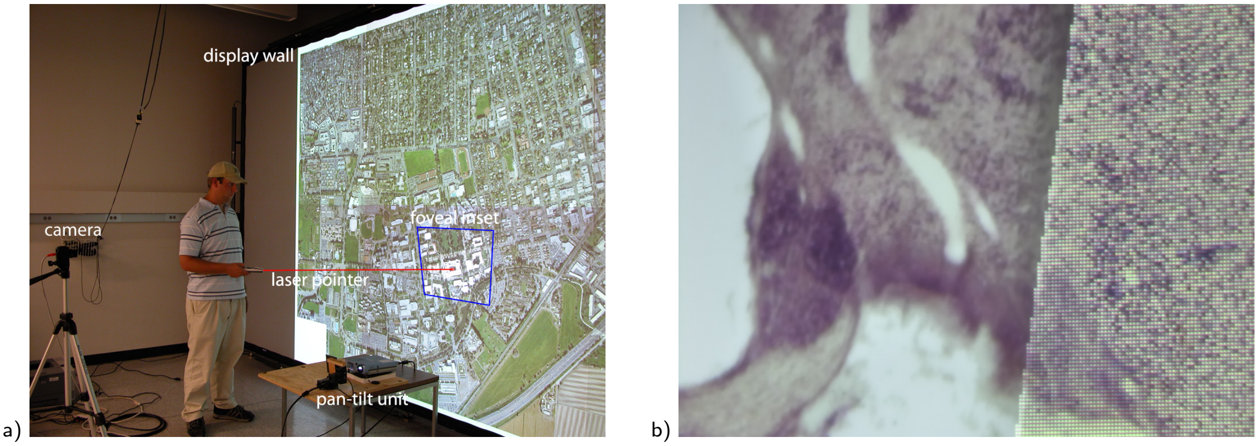

Figure 1: Foveal inset. a: The user directs the projection of the foveal inset using a laser pointer. The size of the projected inset is significantly smaller than the tiles of the rear-projected display wall, thus providing a higher resolution. b: Magnified view of a high-resolution slice of a cryosection of a monkey brain. The boundary between the high-resolution foveal inset (left) and the lower-resolution display wall (right) is clearly visible. Note that the pixel dimensions of the foveal inset projector and the display wall projector are identical. (Aerial photographs courtesy of the City of Davis, CA. Monkey brain data set courtesy of E.G. Jones, UCD Center for Neuroscience.)

ABSTRACT

We introduce a system that adds a foveal inset to large-scale projection displays. The effective resolution of the foveal inset projection is higher than the original display resolution, allowing the user to see more details and finer features in large data sets. The foveal inset is generated by projecting a high-resolution image onto a mirror mounted on a pan–tilt unit (PTU) that is controlled by the user with a laser pointer. Our implementation is based on Chromium and supports many OpenGL applications without modifications.

CR Categories: I.3.1 [Computer Graphics]: Hardware Architecture—Raster Display Devices; H.5.2 [Information Interfaces and Presentation (I.7)]: User Interfaces (D.2.2, H.1.2, I.3.6)—Graphical User Interfaces (GUI)

Keywords: tiled display, multi-resolution display, foveal display, interaction, projector calibration, camera calibration

1 INTRODUCTION

Large display environments have become increasingly important over the past decade and are used frequently for displaying high-resolution data resulting from imaging applications and simulations. The size and complexity of such data sets increases steadily, and the resolution of single-projector displays is no longer sufficient to reveal details without zooming in and, thus, loosing important context information. One possible solution to this problem is the use of tiled displays that use multiple projectors to increase the total resolution of the system. Even though high-quality projectors are now available at reasonable cost, increasing the number of tiles by adding more rows and columns increases the cost of the system significantly. For example, adding one row and one column to a 4×3-tile display increases the number of projectors (and rendering nodes) from 12 to 20.

We have developed a positional foveal inset mechanism for tiled displays. This is a novel method for interacting with large displays. A high-resolution projector and a mirror mounted on a pan–tilt unit (PTU) are used to move the foveal inset on a tiled display. This provides a method for examination of areas of interest in very high detail without the expense of adding more tiles to the display. It also allows the system to keep pace with advancements in projector technology. Instead of upgrading a large number of projectors, only a single foveal inset projector needs to be replaced.

2 SYSTEM CONFIGURATION

Hardware Setup Our tiled display wall consists of six tiles arranged in a 3×2 grid. Each tile is $6' \times 4\frac{1}{2}'$ for a total size of $18' \times 9'$. A tile is displayed using two Sanyo PLC-XT16 projectors to support stereoscopic imaging[1]. The projectors are driven by a cluster of Linux machines with 2 GHz AMD Opteron processors

IEEE Virtual Reality 2006
March 25 - 29, Alexandria, Virginia, USA
1-4244-0223-9/06/$20.00 ©2006 IEEE

---

[1] Our foveal inset system currently does not use the stereoscopic capabilities or the tiled display.

and 1 GB of memory. The head node of the cluster is a Linux machine with dual 2 GHz AMD Opteron processors and 8 GB of memory. We are using Point Grey Flea [3] cameras for calibration and interaction. These cameras are capable of capturing 1024×768 pixel color images at 30 fps. We are using a Directed Perception Pan-Tilt Unit PTU-C46 to control the inset position. The PTU has position resolution of 184 arc-seconds. Our unit is configured to move at 1000 positions per second. We have mounted a mirror to the PTU using a gimbal adapter. The projector used to project the foveal inset is a Mitsubishi XD50U.

Software Design The position of the foveal inset is specified using a hand-held laser pointer. The control of the foveal inset position is implemented to interface with a laser pointer interaction system [1]. The foveal inset controller receives information regarding the laser pointer position from the tracking application via a network socket. The foveal inset is then positioned about this location on the display by adjusting the pan and tilt angles of the PTU. This allows the user to run-time specify the position of the foveal inset on the display, letting areas of interest be displayed in higher resolution than the rest of the display.

3 CALIBRATION

Our system performs a coordinate mapping from the foveal inset image plane to the display image plane. Mapping a 2D point in homogeneous coordinates on a plane to another plane can be achieved using a 3×3 homogeneous matrix [4]. The mirror which reflects the foveal inset is in a number of different positions as the PTU moves, effectively changing the image plane of the inset. For this reason, a different homography is required for each PTU position. Due to the PTU's high resolution, pre-computing these homographies for each possible pan–tilt angle pair is impractical. Instead, our system calibrates for a configured subset of the possible positions. This allows the range of foveal inset positions to be configured in such a way that all desired areas of the display are covered and minimizes the amount of calibration time and system memory needed to use the system.

4 INTERACTION

We use a laser pointer to interact with our system [1]. The center points of each calibrated foveal inset position are used to determine which of the positions is most appropriate for the selected location. When the client is started and has read the calibration data, the center of each foveal inset is projected into the display image plane and inserted into a nearest-neighbor search structure When the inset controller needs to position the foveal inset, it uses this search structure to find the closest center point to the laser pointer position. The PTU position is then set to the corresponding pan-tilt angles and the corresponding homography is used to pre-warp the foveal inset image.

The controller receives messages indicating both the laser pointer position and when the laser pointer is no longer visible. When a positional message is received, the coordinates are internally recorded. When the laser pointer is no longer visible (i. e., has been turned off) the client moves the PTU and applies the pre-warp matrix.

5 INTEGRATION

In order to modify the OpenGL pipeline without modifying application code, we have implemented the inset controller as a Chromium SPU [2]. This SPU is a combination of the Chromium RenderSPU and PassthroughSPU. Our SPU inherits from the RenderSPU, and also implements PassthroughSPU functionality. This allows it to render the inset locally and pass rendering information down the pipeline.

6 EXPERIMENTAL RESULTS

To evaluate the impact of higher-resolution foveal insets on applications visualizing high-resolution data, we used a prototype image viewer application developed within our research group. This application, allows a user to interactively pan and zoom very large image files using out-of-core rendering methods. The application uses a quadtree-based multiresolution representation which is created in a pre-processing step, and uses OpenGL to render an image as a set of texture-mapped square tiles at several different levels of resolution. The first example image shown in the figures is a stained slice from a cryosection of a monkey brain. The image has a resolution of $5,000 \times 5,800$ pixels, and the multiresolution representation occupies 113 MB on disk. The second example image is generated from aerial photography, with a resolution of about $22,000 \times 16,500$ pixels and an on-disk size of 1.38 GB. Since the foveal inset provides a locally increased resolution, and the appropriate level-of-detail for image rendering is chosen based on pixel size, the application automatically renders the image at a higher resolution in the area covered by the inset.

7 FUTURE WORK

We plan to develop an extension of this work which supports stereographic applications to take advantage of the stereo projection capabilities of our tiled display wall. We plan to support stereo calibration and integration into the toolkit which currently provides support for stereoscopic rendering on the tiled display. In addition to providing stereoscopic viewing, this will allow one to control the foveal inset using techniques commonly used in virtual reality.

Furthermore, we plan to improve further the accuracy the system calibration and the overall performance. Since the PTU supports high-velocity movement, we would like to investigate the possibility of controlling the position of the foveal inset based on the information obtained from a head tracker.

8 ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation under contracts ACI 9624034 (CAREER Award) and ACI 0222909, through the LSSDSV program under contract ACI 9982251, and a large ITR grant. We gratefully acknowledge the support of the W.M. Keck Foundation provided to the UC Davis Center for Active Visualization in the Earth Sciences (KeckCAVES). We thank the members of the Visualization and Computer Graphics Research Group at IDAV at the University of California, Davis.

REFERENCES

[1] B. A. Ahlborn, D. Thompson, O. Kreylos, B. Hamann, and O. G. Staadt. A practical system for laser pointer interaction on tiled displays. In *Proceedings of ACM Virtual Reality Software and Technology 2005*, pages 106–109. ACM, ACM Press, 2005.

[2] G. Humphreys, M. Houston, R. Ng, R. Frank, S. Ahern, P. D. Kirchner, and J. T. Klosowski. Chromium: A stream-processing framework for interactive rendering on clusters. In *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 693–702, New York, NY, USA, 2002. ACM Press.

[3] Point Grey Research, Inc. http://www.ptgrey.com/.

[4] J. Walker. Pan tilt unit and tiled displays. Master's thesis, University of California, Davis, 2004.