

Please find attached the following documents.

- 1) CV
- 2) Advocacy Statement
- 3) Following sample publications
  - a) IEEE Computer 2013 (J23 in CV)
  - b) Siggraph 2012 (J21 in CV)
  - c) Siggraph 2011 (J16 in CV)
  - d) IEEE Visualization 2010 (J13 in CV)
  - e) Best Paper Award in IEEE Visualization 2009 (J10 in CV)
  - f) Best Paper Award in IEEE VR 2010 (C26 in CV)
  - g) Best Paper Award in Procams 2010 (C28 in CV)

## ADITI MAJUMDER

University of California, Irvine, Phone: 949-824-8877  
Department of Computer Science, Fax: 949-824-4056  
4056 Donald Bren Hall, Email: majumder@ics.uci.edu  
Irvine, CA 92697-3425 URL: <http://www.cs.unc.edu/~majumder>

- Areas Of Interest** Multi-Projector Displays, Visualization, Computer Graphics, Computer Vision, Image Processing, Human Perception, Virtual Reality, Human Computer Interaction, Visualization Tools.
- Education** University of North Carolina at Chapel Hill  
**PhD in Computer Science** (1996- 2003).  
Thesis: *A Practical Framework to Achieve Perceptually Seamless Multi-Projector Displays*
- Jadavpur University, Calcutta, India  
**B.E in Computer Science and Engineering** (1992-1996)  
*Silver Medalist in the Engineering Faculty for Overall Academic Performance in Senior Years (1994-1996)*  
*Gold Medalist in the Engineering Faculty for Overall Academic Performance in Junior Years (1992-1994)*
- Experience** **Associate Professor** (2008-Present)  
Department of Computer Science, UC-Irvine
- Assistant Professor** (2003-2008)  
Department of Computer Science, UC Irvine
- Student Research Fellow** (2001-2003)  
Mathematics and Computer Science Division, Argonne National Laboratory
- Givens Associate** (Summer 2001)  
Mathematics and Computer Science Division, Argonne National Laboratory
- Graduate Student Fellow** (Summer 1999)  
Bell Laboratories, Holmdel, NJ
- Graduate Student Researcher** (1998-2001)  
Department of Computer Science, UNC Chapel Hill
- Courses/Tutorials/  
Panels** Aditi Majumder and Behzad Sajadi  
*Building your own Projection Based VR Display System,*  
**Course Organizer, IEEE Virtual Reality, March 2010**
- Invited Author/Speaker on Large Format Displays**  
*Projectors for Graphics,*  
**ACM SIGGRAPH, 2008** (Course Organizer: Ramesh Raskar, Oliver Bimber)
- Invited Panelist**  
*The Future of Projector-Camera Systems*  
**IEEE International Workshop on Projector-Camera Systems (PROCAMS), 2008**
- Aditi Majumder and Michael S. Brown,  
*Camera-Based Vision Techniques to Build Large-Area Multi-Projector Displays*  
**Course Organizer, IEEE International Conference on Computer Vision and Pattern Recognition, June 2005**
- Aditi Majumder and Michael S. Brown,  
*Camera-Based Techniques for Building Large-Area Multi-Projector Displays*  
**Course Organizer, ECCV (European Conference on Computer Vision), May 2004**
- Aditi Majumder and Michael S. Brown,  
*Building Large Area Displays*  
**Course Organizer, Eurographics, September 2003**
- Aditi Majumder and Michael S. Brown,  
*Large-Scale Displays for the Masses: Techniques for Building Affordable and Flexible Multi-Projector Displays*  
**Course Organizer, ACM SIGGRAPH, July 2003**

## Publications

### Book

- B1. Aditi Majumder, Michael S. Brown, " Practical Multi-Projector Display Design", A. K. Peters, August, 2007

### Book Chapters

- E2. Aditi Majumder, "Ubiquitous Displays", *Distributed Video Sensor Networks*, Edited By B. Bhanu, C.V. Ravishankar, A. K. Roy Chowdhury, H. Aghajan, D. Terzopolous, 2011
- E1. Aditi Majumder, "Computer Graphics Optique: Optical Superposition of Projected Computer Graphics", *Energy Simulation-Training, Ocean Engineering and Instrumentation, Research Papers of Link Foundation Fellows, Volume 2*, Brian J. Thompson (Ed.).

### Refereed Journal Articles

- J23. Aditi Majumder, Behzad Sajadi, "Advances in Large Area Displays: The Changing Face of Visualization", *IEEE Computer*, May, 2013.
- J22. Behzad Sajadi, Aditi Majumder, Rosalia G. Schneider, Manuel Menezes De Oliveira Neto, Ramesh Raskar, "Using Patterns to Encode Color Information in Dichromats", *IEEE Transactions in Visualization and Computer Graphics*, 2012, **Presented in IEEE Information Visualization 2012** [26% acceptance].
- J21. Behzad Sajadi, M. Gopi, Aditi Majumder, "Edge-Guided Resolution Enhancement in Projectors Using Optical Pixel Sharing", *ACM Transactions on Graphics*, 31(4), article 79, 2012, **Presented in ACM Siggraph, 2012** [21% acceptance].
- J20. Dan Aliaga, Yu Hong Yeung, Alvin J. Law, Behzad Sajadi, Aditi Majumder, "Fast High-Resolution Appearance Editing Using Superimposed Projections", *ACM Transactions on Graphics*, 31(2), article 13, 2012, **Presented in ACM Siggraph 2012** [23% acceptance].
- J19. Koel Das, Monica Siegenthaler, Aditi Majumder, Hans Keirstead, M. Gopi, "Automated Cell Classification and Visualization for Analyzing Remyelination Therapy", *The Visual Computer*, 27(12), pp. 1055-1069, 2011 [40% acceptance].
- J18. Behzad Sajadi, Aditi Majumder, "Auto-Calibration of Multi-Projector CAVE-like Immersive Environments", *IEEE Transactions on Visualization and Computer Graphics*, 18(3), pp. 381-393, 2011 [26% acceptance].
- J17. Alvin J. Law, Daniel Aliaga, Zygmunt Pizlo, Behzad Sajadi, Aditi Majumder, "Perceptually Based Appearance Modification for Complaint Appearance Editing", *Computer Graphics Forum*, 30(8), pp. 2288-2300, 2011, **Presented in IEEE Scientific Visualization 2011** [29% acceptance].
- J16. Behzad Sajadi, Aditi Majumder, Kazuhiro Hiwada, Atsuto Maki, Ramesh Raskar, "Switchable Primaries Using Shiftable Layers of Color Filter Arrays", *ACM Transactions on Graphics*, 30(4), article 65, 2011, **Presented in ACM Siggraph 2011** [18% acceptance].
- J15. Behzad Sajadi, Aditi Majumder, "Automatic Registration of Multi-Projector Domes Using a Single Uncalibrated Camera", *Computer Graphics Forum*, 30(3), pp. 1161-1170, 2011, **Presented in EUROVIS 2011** [30% acceptance].
- J14. Behzad Sajadi, Aditi Majumder, "Auto-Calibrating Projectors for Tiled Displays On Piecewise Smooth Vertically Extruded Surfaces", *IEEE Transactions on Visualization and Computer Graphics*, 17(9), pp. 1209-1222, 2011 [26% acceptance].
- J13. Pablo Roman, Maxim Lazarov, Aditi Majumder, "A Scalable Distributed Paradigm for Multi-User Interaction with Tiled Rear Projection Display Walls", *IEEE Transactions on Visualization and Computer Graphics*, 16 (6), 2010, **Presented in IEEE Scientific Visualization 2010** [26% acceptance].
- J12. Alvin J. Law, Daniel Aliaga, Aditi Majumder, "Projector Placement Planning for High Quality Visualizations on Real World Colored Objects", *IEEE Transactions on Visualization and Computer Graphics*, 16(6), 2010, **Presented in IEEE Scientific Visualization 2010** [26% acceptance]

- J11. Behzad Sajadi, Aditi Majumder, "Scalable Multi-View Registration for Multi-Projector Displays on Vertically Extruded Surfaces", *Computer Graphics Forum*, 29(3), pp.1063-1072, 2010, **Presented at EUROVIS 2010** [30.5% acceptance].
- J10. Behzad Sajadi, Aditi Majumder, "Markerless View-Independent Registration of Multiple Distorted Projectors on Extruded Surfaces Using an Uncalibrated Camera", *IEEE Transactions on Visualization and Computer Graphics*, 15(6), pp. 1307-1316, 2009, **Presented and Received the Second Best Paper Award in IEEE Scientific Visualization 2009** [26% acceptance]
- J9. Behzad Sajadi, Maxim Lazarov, M. Gopi, Aditi Majumder, "Color Seamlessness in Multi-Projector Displays Using Constrained Gamut Morphing", *IEEE Transactions on Visualization and Computer Graphics*, 15(6), pp. 1317-1326, 2009, **Presented at IEEE Scientific Visualization 2009** [26% acceptance]
- J8. Aditi Majumder, Ezekiel Bhasker, Ray Juang, "Advances towards high-resolution pack-and-go displays: A survey", *Journal for the Society of Information Display (JSID), Special Issue for Selected papers from SID Symposium*, 16(3), pp. 481-491, 2007. [7% acceptance]
- J7. Ezekiel Bhasker, Ray Juang, Aditi Majumder, "Registration Techniques for Using Imperfect and Partially Calibrated Devices in Planar Multi-Projector Displays", *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 13(6), pp. 1368-1375, 2007, **Presented in IEEE Visualization, 2007**.
- J6. Aditi Majumder, Sandra Irani, "Perception Based Contrast Enhancement of Images", *ACM Transactions on Applied Perception*, Vol. 4, No. 3, Article 17, November 2007 .
- J5. Ezekiel Bhasker, Pinaki Sinha, Aditi Majumder, "Asynchronous Distributed Calibration for Scalable and Reconfigurable Multi-Projector Displays", *IEEE Transactions on Visualization and Computer Graphics*, 12(5), pp 1101-1108, 2006, **Presented in IEEE Visualization 2006**.
- J4. Aditi Majumder, M. Gopi, "Modeling Color Properties of Tiled Displays", *Computer Graphics Forum* Vol. 24, No. 2, pp 149-163, June, 2005.
- J3. Michael Brown, Aditi Majumder, Ruigang Yang, "Camera Based Calibration Techniques for Seamless Multi-Projector Displays", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 11, No. 2, pp 193-206, March-April, 2005
- J2. Aditi Majumder, Rick Stevens, "Perceptual Photometric Seamlessness in Projection-Based Tiled Displays", *ACM Transactions on Graphics*, Vol. 24, No. 1, pp 118-139, January 2005.
- J1. Aditi Majumder, Rick Stevens, "Color Non-Uniformity in Projection Based Displays: Analysis and Solutions", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 10, No. 2, pp 177-188, March-April, 2004.

#### **Refereed Conference/Workshop Publications**

- C37. Mahdi Abbaspour Tehrani, Aditi Majumder, M. Gopi, "Undistorting Foreground Objects in Wide Angle Images", *International Symposium on Multimedia*, 2013 [25% acceptance].
- C36. Il-Seok Oh, Jinseon Lee, Aditi Majumder, "Multi-scale Image Segmentation Using MSER", *Computer Analysis of Images and Patterns (CAIP)*, August 2013.
- C35. Behzad Sajadi, Duy-Quoc Lai, Alexander Iher, M. Gopi, Aditi Majumder, "Image Enhancement in Projectors Via Optical Pixel Shift and Overlay", *International Conference on Computational Photography (ICCP)*, April, 2013.
- C34. Sangwon Chen, M. Gopi, Aditi Majumder, "HD-GraphViz: Highly Distributed Graph Visualization on Tiled Displays", *Indian Conference on Vision, Graphics and Image Processing (ICVGIP)*, December 2012.
- C33. Kiarash Amiri, Shih-Hsien Yang, Fadi Kurdahi, Magda El Zarki, Aditi Majumder, "Collaborative Video Playback on a Federation of Tiled Mobile Projectors enabled by Visual Feedback", *ACM Multimedia Systems*, 2012 [30% acceptance].
- C32. Kiarash Amiri, Shih-Hsien Yang, Fadi Kurdahi, Magda El-Zarki, Aditi Majumder, Camera Based Video Synchronization for a Federation of Mobile Projectors, *IEEE/ACM Workshop on*

*Projector Camera Systems, June 2011* [30% acceptance].

- C31. Koel Das, Monica Siegenthaler, Aditi Majumder, Hans Keirstead, M. Gopi, "Automated Analysis of Remyelination Therapy for Spinal Cord Injury", *International Conference on Computer Vision, Graphics and Image Processing, Chennai, India, December 2010* [27% acceptance].
- C30. Behzad Sajadi, Aditi Majumder, "Automatic Registration of Multiple Projectors on Swept Surfaces", *ACM Virtual Reality and Software Technology, Hong Kong, November 2010* [25% acceptance].
- C29. Behzad Sajadi, Maxim Lazarov, Aditi Majumder, "ADICT: Accurate Direct and Inverse Color Transformation", *European Conference on Computer Vision (ECCV), Crete, Greece, September 2010* [27% acceptance].
- C28. Aditi Majumder, Robert G. Brown, Hussein S. El-Ghoroury, "Display Gamut Reshaping for Color Emulation and Balancing", *IEEE/ACM Workshop on Projector-Camera Systems, San Francisco, June 2010, Best Paper Award* [33% acceptance]
- C27. Maxim Lazarov, Aditi Majumder, "Device-Independent Representation of Photometric Properties of a Camera", *IEEE/ACM Workshop on Projector-Camera Systems, San Francisco, June 2010* [33% acceptance].
- C26. Behzad Sajadi, Aditi Majumder, "Auto-Calibration of Cylindrical Multi-Projector Systems", *Proceedings of IEEE Virtual Reality, Waltham, MA, March 2010, Best Paper Award* [25% acceptance].
- C25. Maxim Lazarov, Hamed Pirsiavash, Behzad Sajadi, Uddipan Mukherjee, Aditi Majumder, "Data Handling Displays", *IEEE/ACM Workshop on Projector Camera Systems, Miami, FL, June, 2009, [33% acceptance]*.
- C24. Behzad Sajadi, Aditi Majumder, "P-35: Maintaining Color Consistency Across Non-Linear Devices," *SID Symposium Digest of Technical Papers, San Antonio, TX, May 2009*.
- C23. Hamed Pirsiavash, Vivek Singh, Aditi Majumder, Ramesh Jain, "Shared Visualization Spaces for Environment to Environment Communication", *Workshop on Media Arts, Science, and Technology (MAST), Santa Barbara, CA, Jan 2009*.
- C22. Ezekiel Bhasker, Ray Juang, Aditi Majumder, "Advances Towards Next Generation Flexible Multi-Projector Displays", *ACM Siggraph Workshop on Emerging Display Technologies, San Diego, August, 2007*.
- C21. Ezekiel Bhasker, Aditi Majumder, "Geometric Modeling and Calibration of Planar Multi-Projector Displays Using Rational Bezier Patches", *IEEE CVPR Workshop on Projector Camera Systems, Minneapolis, Minnesota, June 2007* [40% acceptance].
- C20. Ray Juang, Aditi Majumder, "Photometric Self-Calibration of a Projector-Camera System", *IEEE CVPR Workshop on Projector Camera Systems, Minneapolis, Minnesota, June 2007* [40% acceptance].
- C19. Ezekiel Bhasker, Aditi Majumder, "Self Calibrating Tiled Displays", *SID Symposium Digest of Technical Papers, Long Beach, May, 2007* [13% acceptance]
- C18. Pinaki Sinha, Ezekiel Bhasker, Aditi Majumder, "Mobile Display Via Distributed Networked Projector-Camera Systems", *IEEE CVPR Workshop on Projector Camera Systems, New York, June 2006* [30% acceptance].
- C17. Aditi Majumder, Sandra Irani "Contrast Enhancement of Images Using Human Contrast Sensitivity" *ACM Symposium on Applied Perception in Graphics and Visualization, pp 69-76, Boston, July 2006* [35% acceptance].
- C16. Aditi Majumder, "Luminance Management for Seamless Multi-Projector Displays", *SID Symposium Digest of Technical Papers, Volume 36, Issue 1, pages 1506-1509, Boston, May 2005* [10% acceptance].
- C15. Kartic Subr, Aditi Majumder, Sandra Irani, "Greedy Algorithm for Local Contrast Enhancement of Images", *International Conference on Image Analysis and Processing (ICIAP), pp 171-179, Cagliari, Italy, September, 2005* [15% acceptance].
- C14. Aditi Majumder, "Contrast Enhancement of Multi-Displays Using Human Contrast Sensitivity",

*IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 377-382, San Diego, June, 2005. [16% acceptance]

- C13. Aditi Majumder, "Is Spatial Super-Resolution Feasible with Multiple Overlapping Projectors?", *Proceedings of IEEE International Conference on Audio, Speech and Signal Processing (ICASSP)*, Vol. 4 pp 209-212, Philadelphia, 2005. [27% acceptance]
- C12. Aditi Majumder, "Camera Based Evaluation of Photometric Compensation Methods on Multi-Projector Displays", *Proceedings of IEEE International Conference on Image Processing*, pp 3527-3530, Singapore, 2004 [25% acceptance].
- C11. Michael S. Brown, Aditi Majumder, Ruigang Yang, "Camera Based Calibration Techniques for Seamless Multi-Projector Displays", *Applications of Computer Vision Workshop, Proceedings of European Conference of Computer Vision*, 2004. [40% acceptance]
- C10. Aditi Majumder, David Jones, Matthew McCrory, Michael E Papka, Rick Stevens, "Using a Camera to Capture and Correct Spatial Photometric Variation in Multi-Projector Displays", *IEEE International Workshop on Projector-Camera Systems*, 2003 [33% acceptance].
- C9. Andrew Raij, Gennette Gill, Aditi Majumder, Herman Towles, Henry Fuchs, "PixelFlex2: A Comprehensive, Automatic, Casually-Aligned Multi-Projector Display" *IEEE International Workshop on Projector-Camera Systems*, 2003 [33% acceptance].
- C8. Aditi Majumder and Rick Stevens, "Photometrically Continuous Imagery in Reconfigurable Large Area Displays", *3rd Annual High Information Content Display System Symposium*, 2003 [25% acceptance].
- C7. Aditi Majumder and Rick Stevens, "LAM: Luminance Attenuation Map for Photometric Uniformity Across Projection Based Displays", *ACM Virtual Reality Software and Technology*, pp 147-154, 2002 [25% acceptance]
- C6. Justin Binns, Gennette Gill, Mark Hereld, David Jones, Ivan Judson, Ti Leggett, Aditi Majumder, Matthew McCroy, Michael E. Papka, and Rick Stevens, "Applying Geometry and Color Correction to Tiled Display Walls" (Poster), *IEEE Visualization*, 2002 [40% acceptance].
- C5. Aditi Majumder, "Properties of Color Variation Across Multi-Projector Display", *SID Eurodisplay Digest*, pp 807-810, Nice, France, July, 2002 [30% acceptance].
- C4. Aditi Majumder and M. Gopi, "Real Time Charcoal Rendering Using Contrast Enhancement Operators", *Symposium of Non Photorealistic Animation and Rendering*, pp 59-66, Annecy, France, June, 2002 [45% acceptance].
- C3. Aditi Majumder and Greg Welch, "Computer Graphics Optique: Optical Superposition of Projected Computer Graphics", *Eurographics Workshop on Virtual Environments/ Immersive Projection Technology Stuttgart, Germany*, May 2001. [40% acceptance]
- C2. Aditi Majumder, Zhu He, Herman Towles and Greg Welch, "Achieving Color Uniformity Across Multi-Projector Displays", *IEEE Visualization*, pp 117-124, Salt Lake City, October, 2000. [34% acceptance]
- C1. Aditi Majumder, M. Gopi, Brent W. Seales and Henry Fuchs, "Geometric Stitching for Real-Time Panoramic Image Generation Using Texture Maps", *ACM Multimedia*, pp 169-178, Orlando, October, 1999 [18% acceptance].

#### Technical Reports

- T3. Ray Juang, Maxim Lazarov, Aditi Majumder, "Efficient Estimation and Device-Independent Representation of Photometric Properties of a Projector-Camera Pair", *Technical Report UCI*, 08-11, 2008
- T2. Behzad Sajadi, Aditi Majumder, "Smoothing Spatial Chrominance Variation Across Tiled Displays", *Technical Report UCI*, 08-04, 2008
- T1. Aditi Majumder, Robert Brown, Hussein El Ghoroury "Display Gamut Reshaping for Color Emulation and Balancing", *Technical Report UCI* 08-04, 2008
- S1. Jakub Segen, Aditi Majumder, Senthil Kumar and Joshua Gluckman, "Virtual Music and Dance Controlled by the Gestures of a Human Conductor", *Emerging Technologies, ACM SIGGRAPH* 1999.

**System  
Demonstration**

- Patents/Artifacts**
- P10. Behzad Sajadi, Aditi Majumder, "A Projector with Enhanced Resolution Via Optical Pixel Sharing", US Patent Provisional (UC Case No. 2012-446-2).
  - P9. Behzad Sajadi, Aditi Majumder, "Augmented Reality Using Projector Camera Enabled Devices" US Patent Provisional (UC Case No. 2012-134-2).
  - P8. Behzad Sajadi, Aditi Majumder, "Automatic Registration of Multiple Projectors on a Dome or Partial Dome", US Patent Provisional (UC Case No. 2012-133-2).
  - P7. Aditi Majumder, Kiarash Amiri, Shih-Hsien Yang, Magda El Zarki, Fadi Kurdahi, "Camera Based Video Synchronization of a Federation of Pico-Projectors", US Patent Provisional (UC Case No. 2011-860-1).
  - P6. Behzad Sajadi, Aditi Majumder, "An Accurate Direct And Inverse Color Transfer Function", US Patent Pending (UC Case No. 2009-607-1).
  - P5. Aditi Majumder, M. Gopi, Behzad Sajadi, "Color Seamlessness in Tiled Displays", US Patent Pending (UC Case No. 2009-608-1).
  - P4. Behzad Sajadi, Aditi Majumder, "Markerless Geometric Registration Of Multiple Projectors (With Or Without Distortions) On Cylindrical And Other Extruded Surfaces Using An Uncalibrated Camera", US Patent Pending (UC Case No. 2009-606-1).
  - P3. Aditi Majumder, Ezekiel Bhasker and Pinaki Sinha, "Apparatus and Method for Self Calibrating Multi-Projector Displays Via Plug and Play Projectors", United States Patent 7,942,530.
  - P2. Aditi Majumder and Rick Stevens, "Method to Smooth Photometric Variation Across Multi-Projector Displays", United States Patent 7,038,727.
  - P1. Aditi Majumder and Rick Stevens, "Optimal Luminance Smoothing to Achieve Color Seamlessness in Tiled Displays", Software artifact, Copyright June 10, 2005 (ANL software report ANL-SF-02-053)
- Awards**
- A12. *nVidia Academic Partner*, 2011.
  - A11. *CUDA Teaching Center* recognition for ICS from nVidia, 2011
  - A10. *Midcareer Research Award*, 2011, ICS, UCI.
  - A9. *Best Paper Award*, June, 2010, IEEE/ACM Workshop on Projector Camera Systems.
  - A8. *Best Paper Award*, March, 2010, IEEE Virtual Reality.
  - A7. *Faculty Research Incentive Award*, 2009, ICS, UCI.
  - A6. *Runner Up for Best Paper Award*, October, 2009, IEEE Visualization.
  - A5. *NSF CAREER Award*, 2009.
  - A4. *Young Scientist of the Year*, 2002, Argonne National Laboratory.
  - A3. *Wallace Givens Fellow*, Summer 2001, Argonne National Laboratory.
  - A2. *Link Fellow*, 2000 – 2001.
  - A1. *Jagdish Bose National Science Talent Scholar (JBNSTS)*, India, 1992 – 1996.
- Publicity Releases**
- A Wave of Hand*, Calit2 Interface Magazine, Spring 2012.
  - Attendance Up at IEEE Computer Society VR Conference*, IEEE Computer Society News, 13 March, 2012.
  - Visualizing the Future*, ACM Tech News, 13 Oct, 2011.
  - Visualizing the Future*, Feature Panorama on UCI main website, October 2011.
  - CAREER Grant to Fund Ubiquitous Displays*, Calit2 Newsroom, October 2009.
  - Coming Together At The Seams: Picture Perfect Projection*, Calit2 Interface Magazine, Winter 2007.
- Invited Talks**
- Keynote Speaker, Brazilian Symposium on Virtual Reality, Rio de Janeiro, May 2012.
  - Google, Mountain View, Oct 2011.
  - HP Research, Palo Alto, Oct 2011.
  - nVidia, Santa Clara, October 2011.
  - Keynote Speaker, International Symposium on Visual Computing, Las Vegas, Dec 2010.
  - Local IEEE Chapter, San Diego, Nov 2010.
  - Qualcomm, San Diego, Nov 2010.
  - Johannes Kepler University, Department of Computer Science, Linz, Austria, May 2010.
  - National University of Singapore, Computer Science Department, Singapore, Sep 2009
  - University of California – Los Angeles, Department of Computer Science, Oct 2008
  - Rochester Institute of Technology, Munsell Color Science Laboratory, Center for Imaging Science, Sep 2008
  - Yale University, Computer Science Department, Sep 2008
  - University of Utah, Scientific Computing and Imaging Institute, Sep 2008

University of California, Davis, SciDAC Institute for Ultrascale Visualization, Jun 2008  
Northwestern University, EECS Department, May 2008  
University of Maryland at College Park, Computer Science Dept, May 2008  
Purdue University, Computer Science Dept, May, 2008  
Jadavpur University, Kolkata, India, Dec 27, 2007.  
MIT Media Labs, Cambridge, May 27, 2005.  
Mitsubishi Electric Research Laboratory (MERL), Cambridge, May 27, 2005.  
MICS Seminar at Department of Energy (DOE), Washington DC, Feb 25, 2004  
University of Maryland at College Park, Computer Science Dept, October 2003.  
Surgical Simulation Center, Washington DC, October 2003.  
Indian Institute of Technology (IIT), Kharagpur, India, October, 2003.

#### Research Grants

G6. Co-Primary Investigator, NSF EAGER: REAQTIVE - Resource Aggregation and Quality Tradeoffs for Integration of Video projector Ensembles, 2010-2011, \$100,000 [Other Co-PIs: Magda El Zarki and Fadi Kurdahi].

G5. **Sole Primary Investigator**, NSF CAREER AWARD: Ubiquitous Displays Via a Distributed Framework, 2009-2013, \$637,000.

G4. **Sole Primary Investigator**, NSF SGER: Analysis of Solution Space for Achieving Ubiquitous Pixels, 2007-2008, \$70,000.

G3. **Sole Primary Investigator**, UCI COLCLR Travel Grant, 2006-2007, \$3000.

G2. **Sole Primary Investigator**, UCI COLCLR Travel Grant, 2005-2006, \$3000.

G1. **Senior Investigator**, "An IT Infrastructure for Responding to the Unexpected", NSF 446617-21857, 2004-2009, \$1,801,590.

#### Equipment Grants

D5. **nVidia donation for CUDA Teaching Center Facility**, 2011, \$26,000 (46 GPUs and TA funds)

D4. **nVidia GPU donation**, 2011, \$16,000

D3. **Canon Projector Donation**, 2009, \$9,000

D2. **Epson Projector Donation**, 2008, \$9,600

D1. **Canon Camera Donation**, 2008, \$2000

#### Collaborations

I6. **Advisory Member**, Vortex Immersion, May 2012 - Present

I5. **Advisory Member**, Allosphere, UC-Santa Barbara, May 2012 - Present

I4. **Advisory Consultant**, Cubic Defense, Oct 2011 - Dec 2011

I3. **Advisory Consultant**, Disney Imagineering, Glendale, May 2009 - May 2010

I2. **Advisory Board Member**, Ostendo Technologies, Carlsbad, 2005 - 2008

I1. **Consultant**, Ostendo Technologies, Carlsbad, 2005 - 2007

*Development of first multi-projector curved screen desktop, coming to market in 2009*

<http://www.youtube.com/watch?v=3ndH8yPHopI>

#### Professional Activities

##### Program Co-Chair

VISTech: Workshop on Visualization Infrastructure Systems and Technology, SuperComputing, 2013.

##### Guest Editor

Special Section of Computer and Graphics on Advanced Displays, 2013.

##### Guest Editor

Special Issue of IEEE Transactions on Visualization and Computer Graphics on Best Papers from IEEE Virtual Reality, 18(7), 2012.

##### General Co-Chair

IEEE Virtual Reality (VR), Orange County, 2012.

##### Program Co-Chair

IEEE Virtual Reality (VR), Singapore, March, 2011.

##### Program Chair

IEEE/ACM Workshop on Projector Camera Systems (PROCAMS), Miami, June 2009.

##### General Co-Chair

ACM Virtual Reality Software and Technology (VRST), Irvine, November 2007.



**Program Co-Chair**

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), San Diego, June, 2005.

**Session Chair**

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), June, 2007.

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), June, 2006.

**Panels**

NSF SBIR/STTR Phase I Panel, Photonic and Optical Systems, August 2013.

NSF CAREER Award Panel, IIS, April, 2008.

**Program Committe**

IEEE Visualization, October, 2011.

IEEE Virtual Reality, March 2011.

IEEE/ACM SIGGRAPH Workshop on Projector Camera Systems (PROCAMS), June, 2010

IEEE/ACM SIGGRAPH Workshop on Projector Camera Systems (PROCAMS), June, 2009

IEEE/ACM SIGGRAPH Workshop on Projector Camera Systems (PROCAMS), August, 2008

ACM Virtual Reality Software and Technology, Irvine, November 2007

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), June, 2007

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), June, 2006

IEEE CVPR Workshop on Projector Camera Systems (PROCAMS), June, 2005

**Reviewer**

ACM Transactions on Graphics (TOG)

IEEE Transactions on Visualization and Computer Graphics (TVCG)

IEEE Computer Graphics and Application

IEEE Computer Vision and Pattern Recognition (CVPR)

Journal of Electronic Imaging

International Journal of Image and Graphics

Eurographics

Computer Graphics Forum

Eurographics Symposium on Rendering

ACM Siggraph

IEEE Visualization

ACM Symposium on User Interface Software and Technology

International Journal of Computer Vision

Computer Aided Geometric Design

IEEE and ACM International Symposium on Mixed and Augmented Reality

**Service to Campus**

**Director** of Visualization Lab, CalIT2, Fall 2010 - Present

**Member** of Visualization Lab, CalIT2, Inception - Fall 2010

**Member** of CalIT2 Divisional Council, 2010 - Present

**Member** of Student Outreach, Access and Retention Committee, ICS, 2010

**Member** of Faculty Recruitment Committee, CS, 2007

**Member** of Graduate Admissions Committee, CS, 2006 – 2008, 2010, 2011

**Demos and Presentations of the Multi-Projector Display**

Candidates Day, ICS, 2006

Video Presentation during Dedication Day of Bren Hall, 2007

Faculty Candidates, 2007

Several academic/industry visitors hosted by Calit2

**New Course Development**

Undergraduate Course on Digital Image Processing (CS 111)

Graduate Course on Visual Computing (CS 211A)

Graduate Course on Visual Perception (CS 213)

**Advisory Activities****Students Graduated**

Behzad Sajadi - (PhD - Summer 2012) - D. E. Shaw and Associates

Duy Quoc-Lai (MS - Spring 2011) – UCI PhD Student

Maxim Lazarov (M.S. - Summer 2010) – Dreamworks Animations

Pablo Roman (M.S. - Spring 2010) – Kyoto University, PhD Student

Mitsunubo Sugimoto (M.S. - Spring 2009)

Anna Diez (M.S. - Spring 2009)

Ray Juang (M.S. - Summer 2007) – Advanced Physics Lab, Maryland

**Current Graduate Student Advisee**

Duy Quoc-Lai (2011-Present) - Past Candidacy  
Mahdi Abbaspour Tehrani (2012-Present)

**Past Graduate Student Mentorship**

Sridevi Maharaj (2011-2012)  
Golnaz Ghasi (2011-2012)  
Uddipan Mukherjee (2008 - 2009)  
Ankit Gupta (2008 - 2009)  
Ezekiel Bhasker (2006 - 2007)  
Pinaki Sinha (2005 - 2006)

**PhD Dissertation Committee**

Kiarash Amiri - (PhD - Spring 2012)  
Shih Hsien Yang - (PhD - Spring 2012)  
Tien Bau - (PhD - Winter 2011)  
Don Black - (PhD - Summer 2009)  
Pablo Diaz Gutierrez (PhD - Spring 2008)

**Undergraduate Student Advisee**

Kitty Ho (2009-2010) – Rockwell Collins  
Christopher Larsen (2009 - 2010) – Western Digital  
Jason Kim (2004 - 2005) – Pixar Animations  
Christopher Larson (SURF-IT fellowship for summer 2006)  
Maxim Lazarov (SURP awardee for summer 2007, 2007- 2008), **Best UROP Project Award for 2008 - MS, UCI**

**PhD Candidacy Committee**

Duy Quoc-Lai  
Mohsen Hejrati  
Benjamin Koehne  
Sangwon Chae  
Kiarash Amiri  
Shan Jiang  
Uddipan Mukherjee  
Tien Bau  
Hamed Pirsiavash  
Ish Rishabh  
Vidya Balasubramanian  
Gautam Chaudhary (EECS)  
Koel Das (EECS)  
Pablo Diaz-Gutierrez (ICS)  
Kartic Subr (ICS)

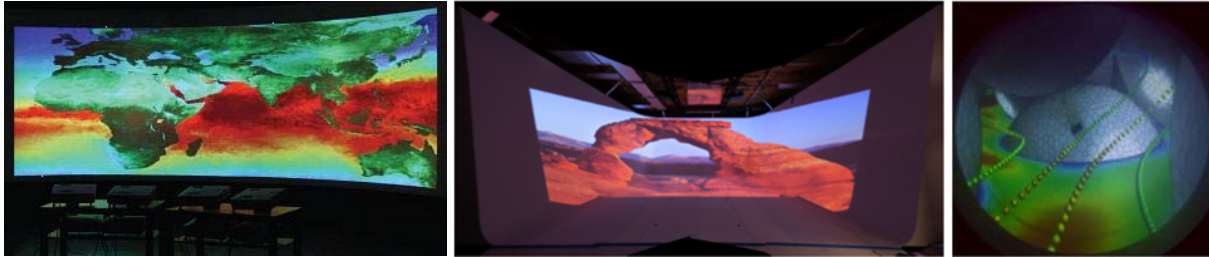
**Masters Thesis Committee**

Mathew Cherian Carathedathu  
Shanaz Mistry  
Ishwar Kulkarni

## Career Highlights

I joined UCI in 2003 as computer graphics and visualization (CG&V) faculty and received tenure in the CS Department in School of Information and Computer Science in 2009. I co-founded our lab in UCI, iGravi: Interactive Graphics and Visualization lab. I am also the Director of the Visualization Lab in Calit2, an institute on campus to foster collaborative research.

## Research



*Figure 1: This figure shows a 4x2 array of 8 projectors on a cylindrical display (left) of 14' radius and angular span of 90 degrees (installation in CalIT building) and a dome of 5ft radius at Bren Hall (right). Both are at UC-Irvine and are registered completely automatically using a single uncalibrated camera by applying the optimization techniques for constrained non-planar surfaces. It also shows a swept surface display at Disney Imagineering (middle) which uses the automatic registration techniques developed at UC-Irvine. The display is 30' wide, 22' deep and 13' high.*

**Existing research area:** My area of research from the inception of my career at UCI has been on novel projection based displays. Technically, this work has focused on automatic camera-based registration and interactive multi-user interfaces on multi-projector based virtual and spatially augmented reality environments making them affordable, accessible and extremely easy to deploy for lay-users as in hospitals, schools, and public spaces. In particular, we have been advancing the frontiers of research in handling large non-planar display surfaces (Figure 1) and in designing distributed scalable methodologies to interact with them (Figure 2).

We have more than 40 publications in this direction (16 journal and 27 conference) in the best venues in graphics and visualization like IEEE Transactions on Visualization and Computer Graphics, ACM Transactions on Graphics, IEEE Visualization and ACM Siggraph. The high-point was in 2009 when we dominated the session on Large Displays in IEEE Visualization with multiple papers and a best paper award. We won two more best paper awards in 2009-2010 including one in the very top venue of IEEE Virtual Reality. All these culminated in a recent invited publication in IEEE Computer, 2013 in the special issue on visualization titled *Advances in Large Area Displays: The Changing Face of Visualization* bringing iGravi to limelight in terms of academic excellence and achievements.

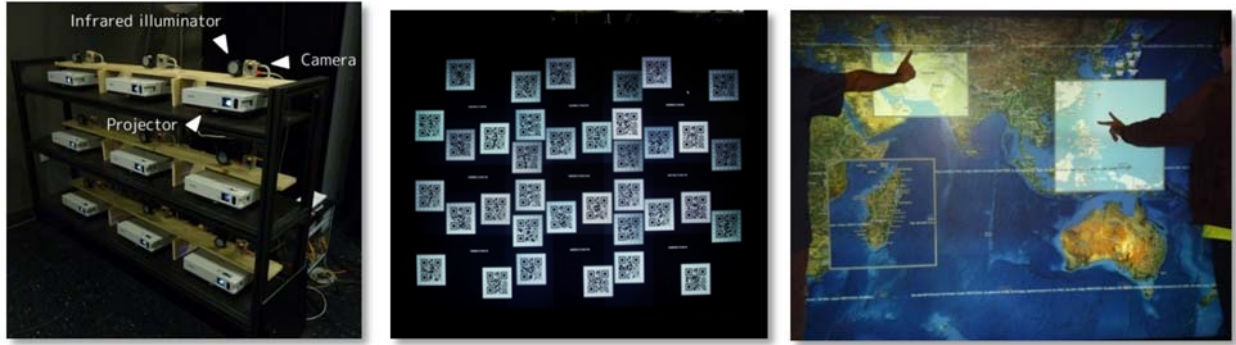


Figure 2: A planar display wall built in UC-Irvine using a distributed network of 9 active displays in a 3x3 array (left). The display is shown from the front undergoing distributed registration using QR codes [10] (middle). The display is then used for map visualization and navigation (right) via the distributed scalable paradigm for simultaneous multi-user real-time interaction [11].

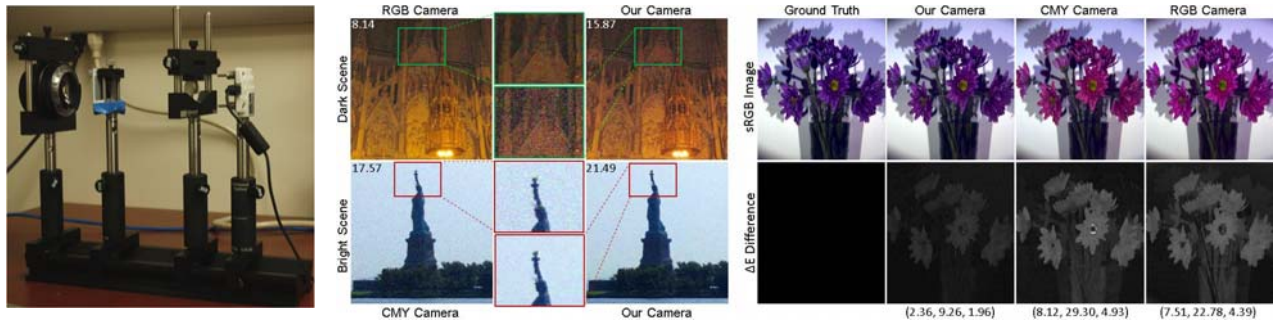


Figure 3: Left: Our prototype flexible camera using shiftable color filter arrays. This allows us to have the SNR quality of RGB camera during the daytime, the SNR quality of a CMY camera in the night time and the higher color fidelity of a 6-primary camera via switchable modes on our single camera. Right: Image quality from our camera matches the image quality of the different kinds of camera, but is available in a single device.

**Establishing a new direction of research:** During the last two years we made inroads in a new area of research in computational cameras/displays. This research is considerably resource heavy requiring grooming students in multiple complementary skill sets (e.g. hardware and optical engineering, prototype building), expensive infrastructure and considerable finances. Hence, the main players in this domain are top ranking places like MIT, Columbia, Stanford and UC Berkeley. When comparing this with the stringent resources available at UCI in these very difficult times, we knew that our journey would be hard. But, we strategically chose important yet low-budget problems to work on and succeeded in building a new camera (Figure 3) and new projector (Figure 4) resulting in 6 high quality publications in the last 2 years at top venues like ACM Siggraph and IEEE Visualization. The high point was in ACM Siggraph 2012 where we made our place as one of the four papers in the only session for computation displays in ACM Siggraph 2012 – the other players being MIT, UC-Berkeley and Stanford. We hope to maintain such good work in this new direction of research in the future.

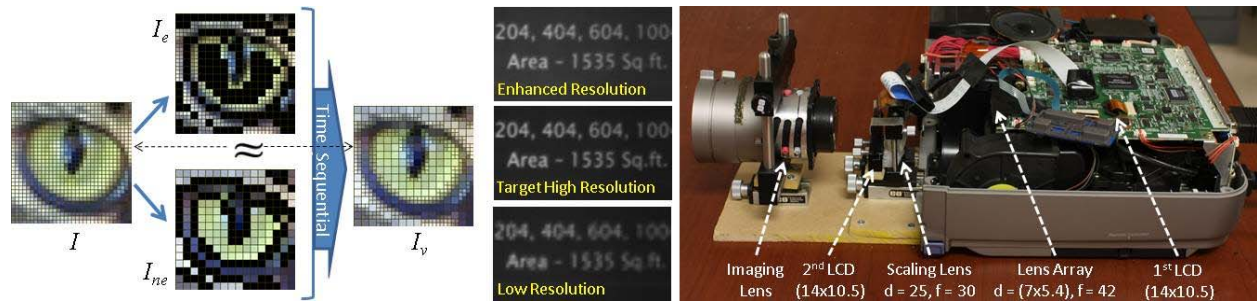


Figure 4: This figure shows our concept and prototype of our variable resolution projector. Left: This illustrates the basic concept of decomposing a target high resolution image,  $I$ , to a high resolution edge image,  $I_e$  and a complementary low resolution non-edge image,  $I_{ne}$ .  $I_e$  and  $I_{ne}$  are then displayed in a time sequential manner to create an image with spatially varying resolution where edges are displayed at a higher pixel density than the rest of the image. We call this the edge-enhanced image,  $I_v$ .  $I_v$  is perceptually close to  $I$ . Middle: The image from the enhanced-resolution projector compared with a target high-resolution image and a low resolution image. Note that the edge-enhanced image is perceptually almost indistinguishable from the target high-resolution image. Right: A side view of our prototype enhanced-resolution projector that achieves 1024x768 edge-enhanced images by cascading two light modulator panels of 512x384 resolution. The numbers in the image are in millimeters.

**Industrial Impact of Research:** My lab is one of the most active labs in the school of ICS in terms of patenting IPs. We have filed more than 5 cases for patents in the past two years and have been granted a couple of them. This has led to industrial collaborations with very visible partners like Disney Imagineering and nVidia. Disney is now using our technology to register multi-projector displays on their immersive virtual reality environment for in-house simulations. nVidia has recognized UCI as an nVidia Academic Partner with equipment donation (of around \$16K) facilitating the application of my research to mobile collaborative displays, a problem of immense interest to nVidia. We have recently worked with a defense company, Cubic Defense, to transfer our IP to a real product of simulation systems for the US military. This is now being considered for licensing by Cubic. We have also helped the Allosphere team at UC-Santa Barbara to use our technology for registering their 16 projector display on the huge double-hemisphere Allosphere facility. Currently, we are also working with Vortex Immersion of Hollywood to explore the application of our technology in the entertainment industry. Such endeavors require a tremendous amount of time and are often not well-recognized within the realm of academics, but nevertheless are very useful in being recognized as an academically excellent research facility.

**Funding:** My main source of funding for supporting my students has been the NSF CAREER award (\$637K) I was granted in 2009. This proposal titled *Ubiquitous Displays Via a Distributed Framework* was one of the select few of the awards chosen to be supported from ARRA (American Recovery and Reinvestment Act) funds. I had also secured two SGER fundings – one as sole PI (\$70K) and another with two Co-PIs (\$100K). However, recently computer graphics and visualization (CG&V) funding in NSF has become very stringent (often

less than 5% success rates), as can be attested by many members of the community. The result of this is the last three of my proposals at NSF has been declined despite having a rating of highly competitive. Funding for graphics has been severely impacted by the recent development in NSF to remove any CG&V representation and merge it with computer vision. I am now in the pursuit of alternate funding sources and am starting collaborations to aid applications of CG&V in domains that impact society directly like health and education.

## **Education**

**Teaching:** I have been instrumental in designing several new courses in ICS. These include a UG course in Digital Image Processing and two graduate courses, Visual Perception and Visual Computing. The course on Visual Computing has been especially successful and has been designated as a core graduate course being the entry point course for students before they can take advanced courses in computer graphics, computer vision and image processing.

**UG Mentoring:** I have been strongly involved in UG mentoring. Initial infrastructure in my lab was set up by UG student Jason Kim in 2005. Jason was placed in Pixar after graduation. UG Maxim Lazarov pursued UG research with me as a SURP (Summer Undergraduate Research Program) Fellow in 2007-2008 and was awarded the *Graduating Senior with Outstanding Research Contribution* from the school in 2008. He works in Dreamworks Entertainment. Finally, more recently I mentored UG Kitty Ho and Christopher Larsen in iGravi who were placed in Rockwell Collins and Western Digital in 2012 respectively.

**Graduate Mentoring:** I have graduated a team of good students in the past few years placed today in top institutions. I have graduated an outstanding PhD student in Behzad Sajadi in summer 2012. Behzad had almost 20 publications when he finished (16 of which were his last two years), all in top venues making him one of the most outstanding students to ever graduate from our department, as many of our colleagues would attest. This helped him to get multiple excellent job offers including a postdoc position in MIT and CMU, a research scientist position at nVidia research to head their displays team, and a research financial analyst position at D.E. Shaw. I also groomed a set of excellent masters students, namely Maxim Lazarov (currently in Dreamworks), Ray Juang (currently in Google), and Pablo Roman (currently a PhD student in Kyoto University). Currently, I have two good PhD students who are working with me for the past year or two. I hope that they will graduate and get placed in similarly great places as my earlier students.

My research requires skill-sets that are not limited to traditional computer science (CS) training. In our lab, we have built cameras from scratch, opened up projector hardware and modified it, and built large VR and spatially augmented reality systems. In addition to complex software/algorithm development, as is common in CS, this needs an innate interest in working with hardware (including cameras, projectors, and optics) and programming them; and willingness to learn topics in domains like electrical engineering, electronics and optical

engineering. This has often been a deterrent in retaining PhD students in my group even after providing them a good start to their PhD career with solid publications. Pinaki Sinha moved to a different group after spending two years in my lab even after having two top quality IEEE Visualization paper. Uddipan Mukherjee moved to a different advisor after spending one year in my lab even after having a good publication in IEEE Workshop in Projector Camera Systems in the very first year of PhD. On the other hand, I did get lucky to get students like Maxim Lazarov, Ray Huang, Ezekiel Bhasker, and Pablo Roman who had the required inclination to work with both software and hardware and spent two extremely productive years in my lab (all have multiple top quality publications in IEEE Transactions on Visualization and Computer Graphics and IEEE Visualization). However, at the end of these two years, these students gathered a set of complementary expertise which is rather unique and rare to find. Hence, exceptionally lucrative offers from the industry made them quit with an MS without staying for PhD program. Maxim Lazarov left with MS to Dreamworks, Ray Juang left with MS to Google, and Ezekiel Bhasker dropped out of the program and joined Qualcomm. The only exception to this has been Pablo Roman who left with MS and is pursuing PhD at Kyoto University instead of UCI due to personal reasons.

***Other Pedagogical Activities:*** I have been active in educating the CG&V community at large via my book titled *Practical Multi-Projector Display Design*. I have been actively offering peer-reviewed courses in premium graphics/vision venues like ACM Siggraph (2003 and 2008), IEEE CVPR(2005), and Eurographics(2004). I have also been invited to be an invited speaker/author on Large Format Displays for the course on Projectors in Graphics in Siggraph 2008 and as a panelist in the expert invited panel on The Future of Projector-Camera Systems at the IEEE/ACMWorkshop on Projector Camera Systems in 2008. I have also been actively involved in speaking assignments across the world to expose and educate professionals and students in the newest research on displays, cameras, virtual reality and augmented reality.

## **Service**

Academic and research excellence alone is not sufficient to bring an institution to limelight or to make a dent in the map of CG&V. To make this happen, one has to serve the community in various leadership roles. I have been serving in such roles, both for ICS on campus and the CG&V community at large for the past few years. When I joined UCI in 2003, we had only two junior CG&V faculty with almost no presence in CG&V. We believe that such service roles have helped us to put UCI in the CG&V map.

***In CG&V community:*** Fortunately, some prominent service opportunities came my way. I was asked to host IEEE Virtual Reality (VR) 2012, the top international virtual reality conference. With the help of my students, I was able to host one of the best IEEE VR conferences in recent years. We took advantage of our location and weather to spearhead a publicity campaign which resulted in 2012 conference being the most attended one in the history of IEEE VR and was featured in the IEEE Computer Society News in March 2012 titled

*Attendance Up at IEEE Computer Society VR Conference.* Further, we used this opportunity to arrange for an Open House at our iGravi Lab which brought in more than 450 attendees from all over the world to UCI. The stage was already set by serving as the Program Co-Chair for IEEE VR the previous year (2011) and we believe that hosting the conference in 2012 provided us the perfect opportunity to showcase our high quality research to the computer graphics and visualization community at large. Hosting VR 2012 provide me with the opportunity to serve as a guest co-editor for IEEE Transactions on Visualization and Computer Graphics (TVCG) Special Issue on Best Papers from IEEE VR 2011. Such publicity always serves the lab and campus positively and brought forth invitations for me to be keynote speaker in various reputed venues like International Conference on Visual Computing 2010 and Brazilian Symposium on Virtual Reality 2012.

***In Campus:*** I have been an active and visible member of the Visualization Lab of Calit2 since its inception running several demos for the many high profile visitors to Calit2. For the past three years, I have been the Director of the Visualization Lab at Calit2. During this time, my lab has received tremendous visibility on campus via the week-long feature panorama on UCI main website on our lab titled *Visualizing the Future*. This feature also made it to ACM Tech News providing a very positive image to UCI in the CG&V community at large. My work was also featured multiple times in the Calit2 Interface magazine, the most recent one being the one titled *A Wave of Hand* in its Spring 2012 issue which focuses on collaborative technologies. Finally, I took the opportunity given by the tremendous amount of GPU programming used in different programs in the school of ICS to give it adequate exposure at nVidia. This resulted in our school of ICS being recognized as one of the few CUDA Teaching Centers around the globe. As part of this process, we were donated around 50 latest nVidia GPUs (\$16K) to equip the CS labs and some funding (\$12K) to train personnel in aiding students with GPU programming.

## **Summary**

I believe that the all-around academic, industrial and public impact of the research and other pedagogical activities conducted in my lab have been well appreciated by the academic community. The tremendous amount of hard work to maintain this excellence has expressed itself positively making UCI a strong name in CG&V. This visibility and reputation was built from scratch, brick-by-brick, through consistent quality research and dedicated service over the past years. We are proud of our achievements and we will continue to make significant impact in academia and to the public.





# Large Area Displays: The Changing Face of Visualization

**Aditi Majumder**, *University of California, Irvine*

**Behzad Sajadi**, *D.E. Shaw & Co*

---

**Although very large immersive displays that can accommodate current data's scale, resolution, and complexity have progressed rapidly in the past decade and are now accessible to lay consumers, installation and maintenance challenges remain.**

---

**V**isualization is a large domain that encompasses the science that deals with massive data processing, display, and interaction. The term usually refers to a cyclic process in which interaction with the displayed visualization should trigger further processing and analyses. Thus, the quality of displays, particularly the large area displays so useful in collaborative projects, is critical to visualization's successful practical application. In that sense, visualization is the body, and the display is the face for expression, communication, and interaction. Without a high-quality display, visualization has no more utility than a body without a face.

We underline this synergistic relationship to make an important point: the tremendous advances in data processing techniques for large-scale visualization would have been futile without the past decade's commensurate progress in large area display technology and its breathtaking promise for the future.

A survey of work from the early 1990s to the present reveals that the display community is at a golden moment in large area display development in which it is enjoying the fruit of recent phenomenal progress and identifying open problems and ambitious visions for the next generation to explore.

## A WIDER VIEW

In the early 1990s, single desktop monitors were the norm—the high end being a 19-inch diagonal with about a 250,000-pixel resolution. Data, in contrast, was entering the realm of petabytes and terabytes to meet the simulation and visualization demands of national projects in various scientific domains. Visualization, in particular, became an essential tool for most projects, which also depended on colocated multiuser collaboration. The data scale and resolution that these projects required along with their reliance on collaboration were way beyond what narrow field-of-view desktop displays could handle.

To meet the growing demand for visualization tools, in 1993 researchers at the Electronic Visualization Laboratory at the University of Illinois at Chicago introduced the first Cave Automatic Virtual Environment (CAVE), a display of three to five walls that resembled a room. As Figure 1a shows, each wall has one or more mono or stereoscopic projectors, enabling an unforeseen degree of immersion. Virtual reality systems incorporate versions of these CAVEs with interactive joysticks and head-tracking devices to facilitate view-dependent navigation and simulation tasks.



**Figure 1.** Early large area displays. (a) University of Illinois at Chicago's CAVE and (b) Princeton University's 18-megapixel display wall, which uses a  $6 \times 4$  array of 24 projectors. (c) UCI's 7-megapixel wall, which uses a  $3 \times 3$  array of nine rear projectors, showing (d) a completely seamless image that looks like a single projector display, although it comes from nine projectors. The UCI projectors had severe color variations, which researchers alleviated by developing camera-based automatic registration software.

**Table 1. Very large planar display walls in the late 1990s.**

Laboratory	Projector array	Display size (ft.)	Resolution (megapixels)
Princeton	$6 \times 4$ array of 24 rear projectors	$18 \times 18$	~18 (6,000 $\times$ 3,000)
Argonne National Lab	$5 \times 3$ array of 15 rear projectors	$16 \times 8$	~10 (5,000 $\times$ 2,000)
Argonne National Lab	$3 \times 2$ array of six rear projectors (mobile)	$4 \times 3$	~3 (2,000 $\times$ 1,500)

By the late 1990s, engineers were synchronously shuttering CAVE projectors with active stereo glasses to provide a 3D stereoscopic experience. However, resolution was still limited to less than 1 megapixel per wall, thus not even coming close to the scale and resolution demands of scientific and information visualization applications. Moreover, CAVEs were focused on single-head tracking, which precluded multiuser applications.

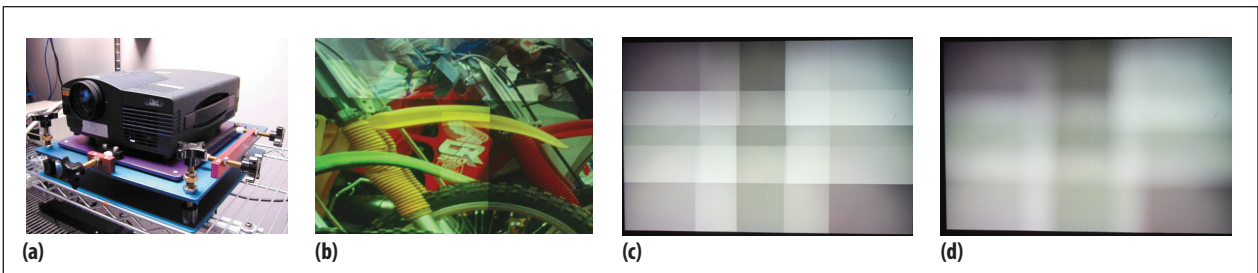
During this same time, very large planar display walls such as those shown in Figures 1b-1d were emerging. In these displays, multiple projectors overlapped their respective boundaries on a planar projection surface to create a large display wall. Table 1 lists the details of some of these displays.

Figures 2a-2c show one of the projectors and some problems on the display walls. Setup was a major drawback to such displays. Overlapping high-gained rear projection screens alleviated the hot spot issue—brighter in the center than on the fringes—but overlapping imagery required

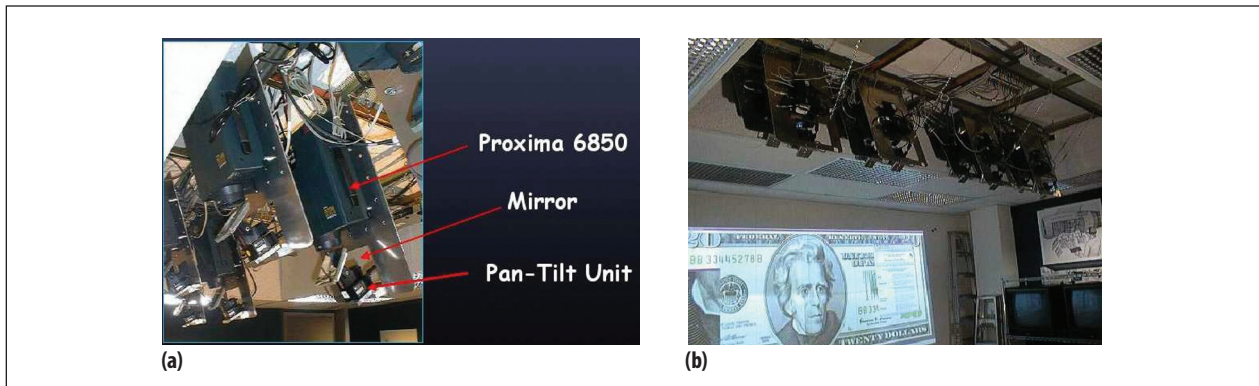
sophisticated mounts with six degrees of freedom that engineers had to manually adjust to align each projector with its neighbors or to align image content across projectors (geometric registration). Such tinkering took many work-hours, even for setups with only two to four projectors, and still did not correct the hot spot issue.

To alleviate this problem, engineers placed metal masks on the optical path between the projector and screen in an attempt to blend the two projectors' contributions to the overlap in a complementary manner. However, as Figures 2c and 2d show, this remedy did not allow a fine per-pixel control, which resulted in a screen door effect—an image viewed as if through a screen door. The large color variations across projectors exacerbated the effect, even when projectors were the same brand.

Traditional semiautomatic color management techniques, such as gamut transformation and matching, did not help because multiprojector displays produced unprecedented degrees of spatial brightness and color



**Figure 2.** Obstacles in image and projection tuning. (a) A projector mount with six degrees of freedom, which required time-consuming manual tinkering to adjust the projector's position to align image content. (b) Error in geometric registration at the overlap of four projectors. (c) Images suffered from color variation and brightness hot spots at overlaps. (d) Correcting hot spots using metal attenuation masks often produced an image that resembled what the viewer might see through a screen door.



**Figure 3.** Evolution of automatic registration techniques. (a) Automatic panning and tilting in UNC's Pixel Flex project and (b) a PTU-mounted mirror that directed each projector's light to different areas of the flat wall, enabling the display to operate in a different scale, form factor, and aspect ratio. Although these systems are no longer operating, their design inspired a new generation of systems that use automatic registration.

variations. Thus, installing and maintaining such a system was labor intensive and expensive, requiring an in-house maintenance crew.<sup>1</sup>

### AUTOMATIC REGISTRATION

The obvious approach to a more sustainable solution was to use a camera to capture the display properties at a very high resolution and then correct the results automatically. By the late 1990s, computer vision research in panoramic image generation was making significant progress, producing techniques similar to the process for registering images from multiple projectors. As a result, the next decade saw a phenomenal amount of work in camera-based automatic geometric registration and color seamlessness—removing perceived spatial variations in color.

In these automated solutions,<sup>2</sup> one or more cameras provided feedback to a server that controlled the multiple projectors and applied the appropriate warps and attenuations at a per-pixel resolution to achieve a seamless display. The color seamlessness problem remained open until 2009, when UCI researchers developed a constrained gamut morphing approach<sup>3</sup> that enabled the completely automatic deployment and maintenance of large planar display walls.

Researchers at the University of North Carolina, Chapel Hill (UNC), subsequently used automatic deployment to create reconfigurable display walls by coupling front projection systems with mirrors mounted on pan-tilt units to direct the light on a planar wall.<sup>2</sup> Figure 3 shows such a projector setup. These flexible units made it possible to change the display's form factor, size, and resolution on demand, enabling the same six projectors to be in a  $6 \times 1$  array for a wide-format display and then in a  $3 \times 2$  array for a display with 4:3 aspect ratio. Automatic registration after each reconfiguration made it possible to create a seamless display in minutes. At the same time, new

distributed rendering paradigms were efficiently rendering large amounts of 3D data in real time, which made these reconfigurable displays highly suitable for any large data visualization application.<sup>4,5</sup>

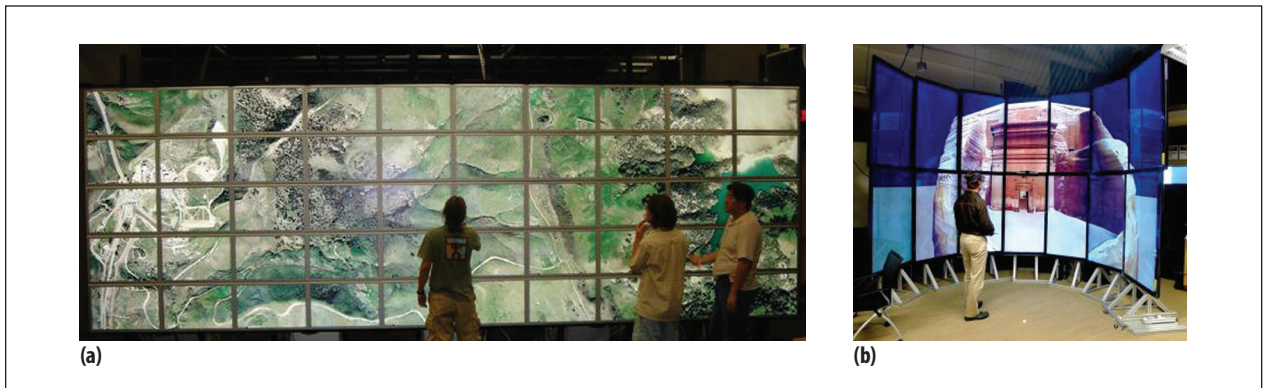
Around the same time, the advent of thin LCD panels enabled researchers to build display walls of multiple LCD panels. Figure 4a shows UCI's HiperWall, the first 55-panel setup. Similarly, a 70-panel HiperSpace was built in the Immersive Visualization Lab at the University of California, San Diego (UCSD).

Because removing the seams in such LCD wall displays and economically mounting them are formidable engineering obstacles, vendors offered these displays, shown in Figure 4b, only as expensive turnkey solutions that lacked the flexibility of reconfiguration. Even so, because distributed rendering architectures like the Scalable, Adaptable Graphics Environment make it easy to port visualization applications on these walls, they are still a popular option for some visualization environments.

### IMMERSIVE DISPLAYS

By 2007, display walls literally changed the face of visualization systems by becoming the most popular option for large-scale visualization. However, it soon became evident that complex multidimensional data demanded immersive environments as well as scale and resolution. For many applications, the user needed a sense of presence and immersion to effectively navigate, modify, and visualize data.

This dream of a completely immersive display was not new. Henry Fuchs of UNC first envisioned such immersion in his 1990s Office of the Future (OOTF),<sup>6</sup> an environment with a sea of cameras for capture and multiple projectors for display. As Figure 5 shows, the idea was to provide a display for visualization within the user's own office environment, as opposed to having the user walk into a special display area.



**Figure 4.** Multilayer LCD panel display walls. (a) UCI's 110-megapixel HiperWall and (b) TourCAVE at UCSD.

Meeting such an ambitious goal required both the 3D reconstruction of large scenes and 3D real-time gesture recognition, which were still open problems at the time. By the early 2000s, however, projects like blue-c<sup>7</sup> had made huge strides in this direction. The resulting novel projection-screen technology used layered LCD glass panels that could turn opaque for projection and transparent for scene acquisition through cameras behind the screen. With this technology, blue-c researchers built the first end-to-end system with multistream videocapture, 3D scene reconstruction, and the projection of the reconstructed scene in an immersive CAVE system with three walls and active stereo.

### Nonplanar geometry

Even this novel technology was not ripe enough to realize OOTF's vision of breaking away from specific display geometry toward an unrestricted environment, but a varied set of nonplanar display geometry seemed the perfect middle ground. Designers could choose surface geometries for immersive displays to suit the users' needs, space, applications, and interaction modalities. Cylindrical, hemispherical, or other uncommon immersive shapes, such as a bowl or an eggshell, became popular for immersive visualization, training, simulation, and edutainment applications. In the mid-2000s, companies such as Elumens, Global Immersion, Fakespace, Christie, and Eon Reality began delivering such systems but only as expensive and rigid turnkey solutions.

Although some immersive displays also used the LCD panels shown in Figure 4, their piecewise-planar rigidity limited their use to a small range of curvature and a compromised sense of presence, as in the TourCAVE display at UCSD, shown in Figure 4b. Projectors offered a much better solution, since they can direct light over any kind of curved surface to achieve a display of any shape and size. However, immersive multiprojector displays had the same issues as the early display walls: registration on nonplanar shapes was much more



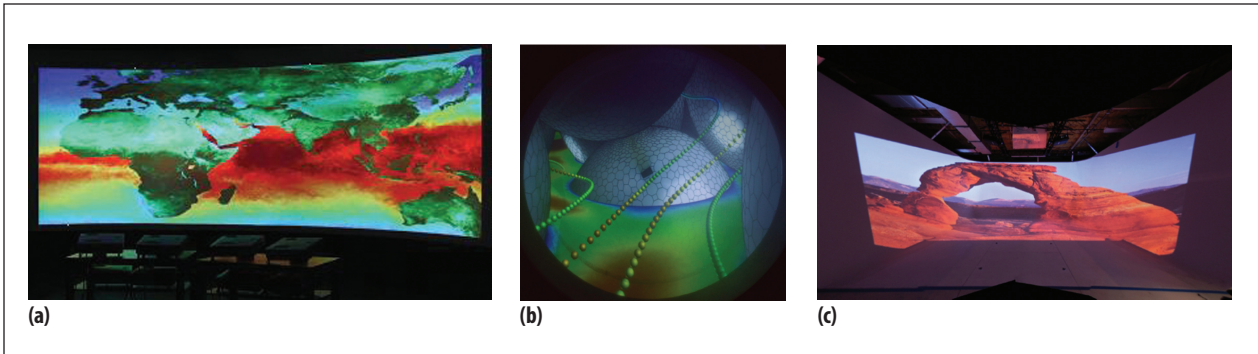
**Figure 5.** UNC's Office of the Future vision in 1998. Nearly a decade before the technology became available, the OOTF imagined displays unfettered by a particular geometry such as a CAVE or other separate walk-in spaces for visualization.

challenging, and once again researchers had to resort to manual or semiautomatic registration techniques, which hindered the displays' mass deployment.

### Automatic registration revisited

Unlike automatic registration in previous years, the main challenge this time around was how to accommodate nonplanar surfaces. Reconstructing very large nonplanar surfaces requires recovering large 3D scenes from multiple stereo rigs, which is still unreliable and problematic.<sup>6</sup> Unlike 3D modeling applications in computer vision, any small error in reconstruction can produce severe errors in the display's geometric registration.

The need for higher accuracy was only one obstacle. Another was the registration problem's highly underconstrained nature: because it was not possible to know device parameters or surface geometry ahead of time, automatic registration techniques were largely impractical. Consequently, most early camera-based



**Figure 6.** Nonplanar displays that use UCI's automatic registration technique: (a) a  $4 \times 2$  array of eight projectors on a cylindrical display with a 14-ft. radius and angular span of 90 degrees, (b) four projectors on a dome with a 5-ft. radius, and (c) a  $30 \times 22 \times 13$ -ft. swept surface display at Disney Imagineering.

registration techniques for nonplanar displays assumed that users would know the surface geometry or device parameters, such as pose, orientation, and focal length. However, most users do not have enough technical savvy to provide such information, and, not surprisingly, the mass deployment of such displays never occurred.

In 2009, UCI researchers began to make significant progress in surmounting these obstacles,<sup>8-10</sup> observing that most immersive displays are not arbitrarily nonplanar, but constrained nonplanar—their boundary curves follow some well-defined constraints, such as a cylinder falling in the general category of an object with a vertically extruded surface. This new perspective led to three broad categories of immersive display surfaces:

- vertically extruded surfaces formed by extruding a planar curve along a vertical line, such as a cylinder;
- swept surfaces formed by sweeping a planar profile curve along a planar path, such as a truncated dome; and
- partial or full spheres.

Figure 6 shows displays that use automatic optimization techniques developed by UCI researchers for constrained nonplanar surfaces. At the heart of these techniques is a novel paradigm of nonlinear optimizations that uses a single uncalibrated camera without any markers instead of calibrated stereo cameras. The paradigm identifies enough prior knowledge about the general properties of boundary curves to sufficiently constrain the problem and recover the unknown camera properties, surface geometry, and projector pose and orientation. For very large scale displays, a single uncalibrated camera on a nodal pan-tilt unit can easily achieve the registration with a higher accuracy than multiple stereo cameras, which tend to be a nonrobust setup. Complete automation and the simplicity of a single camera combine to make deployment easy for lay users.

These techniques brought a new engineering flexibility,

even to CAVEs, which for so long had been treated as segregated planes. With traditional registration methods, engineers not only had to register each plane separately, they also had to manually or semiautomatically register the projectors in each wall with ones in the adjacent wall. This wall-by-wall registration prohibited overlapping projectors across the corners, which in turn made color registration much more difficult.

Designers can use automatic registration for nonplanar walls to treat a three- or five-wall CAVE as a vertically extruded or a swept surface, respectively. Treating the display as a 3D shape rather than as a collection of walls allows more casual projector placement and permits overlaps across corners, both of which make deployment much easier.

With this implementation flexibility and resulting increased sense of presence, automatic registration technology is ripe for use in a variety of constrained nonplanar displays and could greatly impact edutainment, simulation, and training.

## NEW PATHS

As visualization moves from display walls to large immersive displays that provide a stronger sense of presence, researchers are looking ahead to important missing elements, such as the ability to interact naturally using very large displays in collaborative projects. Much work has focused on interaction through touch-sensitive sensors and embedded cameras, but we have yet to see people interacting with one another in a natural way by using face-to-face body language through very large immersive displays. This capability is essential to visionary academic projects like OOTF and the future envisioned in movies like *Minority Report*.

The major roadblock to realizing this dream is the tightly controlled centralized architecture, in which a single server must somehow coordinate a group of disparate devices, such as cameras and projectors. This loosely coupled arrangement of devices limits scalability

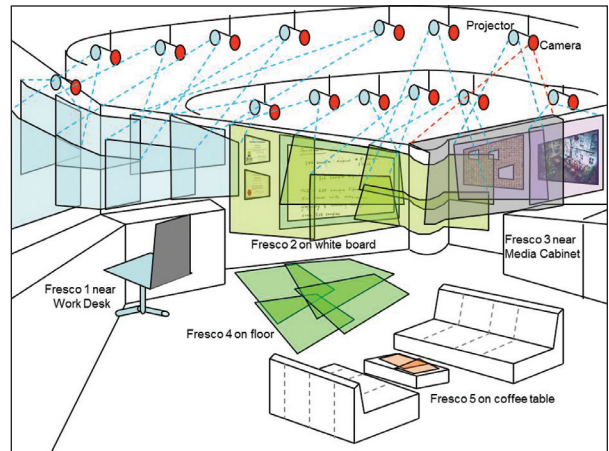
and easy deployment.

UCI researchers working on the ubiquitous displays project<sup>11,12</sup> are looking at an entirely new architecture. As Figure 7 shows, a distributed network of active display units—each with a projector and embedded sensors, including a camera—and computation and communication units populate a workspace. The active displays run identical single-program, multiple-data algorithms that collectively and automatically achieve all the characteristics of a ubiquitous display: scalability, reconfigurability, flexibility, and interactivity. Users can move the displays; define display collections, or *frescos*, to work as a single display; achieve seamlessness on a surface that might not be entirely white or planar; and interact with the data on the display in a natural way.

UCI researchers have begun to explore this distributed paradigm, and our team has already built a planar display wall using active display units, each of which sees its own projection and a small part of its neighbors' projections. Figure 8a shows the projection setup. The research team has also developed distributed registration techniques that use quick response (QR) codes to automatically reconfigure projectors whenever they are moved. Figure 8b shows a front view of self-registration, which such a move triggers. In minutes, the distributed registration process produces a seamless display, such as the one in Figure 8c.<sup>11</sup>

The cameras embedded in the projectors also capture gestures so that users can interact with the display. However, managing gestures and reacting to them is not trivial, since each camera sees only a part of the display, and a user's gesture can cross multiple cameras. A distributed paradigm must ensure that the algorithm running on each active display hands gesture management control from one device to another when appropriate, reacts to gestures that a device has not seen, and maintains data consistency.

Although UCI researchers have taken steps toward realizing these goals by developing the first distributed interaction paradigm for 2D applications on planar display walls,<sup>12</sup> such work is nascent. Capturing the myriad 2D and

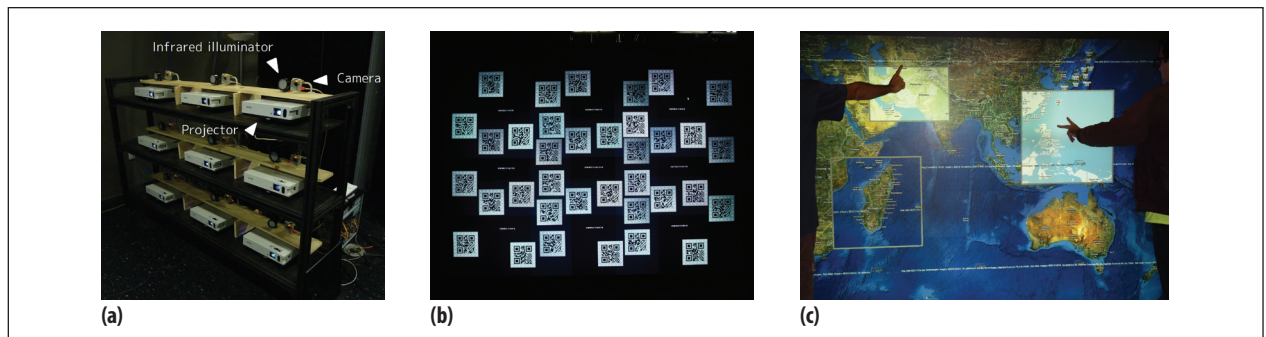


**Figure 7.** How ubiquitous displays might work. Active displays created by augmenting a projector with sensors (including a camera) and a computation and communication unit form a distributed network, or fresco, that users can move and reconfigure as needed.

3D interactions on surfaces that are not flat and white is still a challenge. Realizing that dream requires not only research intensely focused on distributed interaction paradigms but also significant improvement in robust vision techniques to reliably detect 2D and 3D gestures. The fruit of such efforts will drastically alter visualization, empowering users at all levels of expertise with a new efficiency in data exploration, visualization, and modification.

## MERGING REAL AND VIRTUAL

Large, dynamic, complex datasets demand visualization overlaid on the artifact itself rather than through a separate display. In Figure 9, for example, projecting light onto a chipped and faded cultural artifact lets archeologists see its original shape and color. Such in situ visualization has endless possibilities. For example, lighting reliefs of the Colorado River at a very high resolution using projectors reveals river activity in different ages. Placing a physical artifact of a dam in this relief would automatically change the visualization, as projectors adapt and light the entire relief differently to show its effect on the physical



**Figure 8.** UCI's planar display wall, which uses (a) a distributed network of nine active displays in a 3 x 3 x 3 array. (b) Using QR codes, the display undergoes distributed registration. (c) Minutes later, the display is ready for map visualization and navigation involving several users.



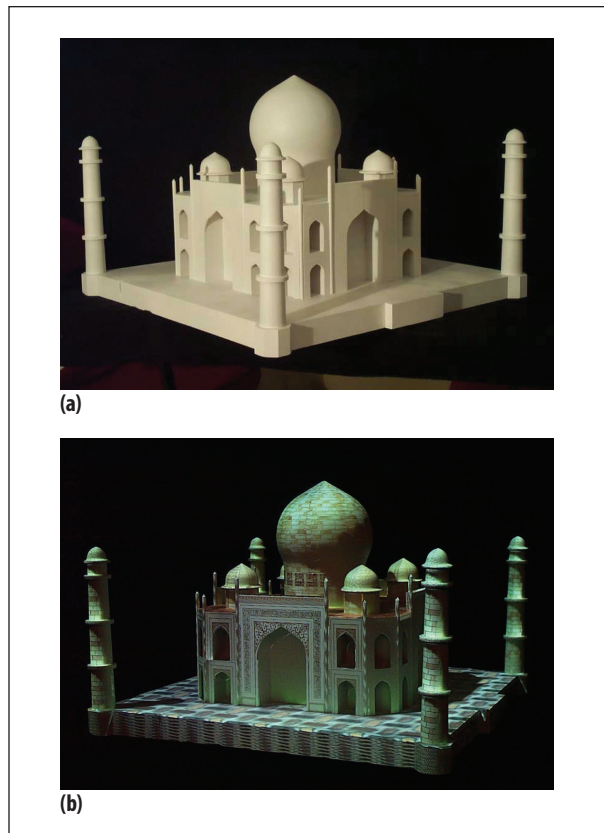
**Figure 9.** Appearance editing for in situ visualization using three superimposed projectors lighting a cultural heritage artifact. (a) The original artifact, (b) visualization of the restored artifact with light projection; and (c) in situ visualization of aging and annotations. Projectors automatically direct light to achieve the desired visualization.

phenomenon.

Historians could annotate the same relief model with facts and figures, or urban planners and emergency management strategists could use it for their scenarios. In the entertainment industry, in situ visualization could add depth to architectural lighting or theatrical lighting.

Ramesh Raskar first demonstrated the power of this visualization in 1999 using four projectors to light a small white table-top model of the Taj Mahal as part of UNC's Shader Lamps project.<sup>13</sup> Raskar applied semiautomatic registration techniques to the projected content in Figure 10a to simulate the appearance of the Taj Mahal on a moonlit night (Figure 10b). Since then, UCI researchers have collaborated with Dan Aliaga and his team at Purdue University to explore the use of superimposed projectors to completely automate the in situ visualization of such objects at a much higher resolution.<sup>14</sup>

Despite progress, in situ visualizations remain limited to small artifacts and setups in which projectors are mounted



**Figure 10.** Using in situ visualization to add depth to architectural lighting. (a) Applying semiautomatic registration techniques to an image of the Taj Mahal (b) simulates its appearance on a moonlit night.

outside looking in. Achieving in situ visualizations on large reliefs and objects requires a distributed network of active displays lighting relatively arbitrary nonplanar surfaces. The registration challenges are many, since unknown devices and surface geometry result in an underconstrained system. Fortunately, multiple overlaps between devices allow them to see parts of the same surface geometry, which provides opportunities for validation across devices and thus offers enough constraints to solve the problem robustly. However, much work remains to be done to make in situ visualizations a commodity system accessible to users at all levels of expertise.


## BEYOND TWO DIMENSIONS

Researchers have recently directed much work toward moving 2D displays to multiview technology, producing autostereoscopic displays that are essentially 2.5D.

About a decade ago, such displays were primarily stereoscopic, either active through synchronous switching of shuttered glasses with the projection-based display, or passive through the selective filtering of differently polarized light from the projectors through glasses. The relatively recent rejuvenation of parallax barrier technology

has led to glasses-free autostereoscopic displays and their extensions to large-scale applications with multiple users.<sup>15</sup> Such advances have come at the cost of resolution, but work on tensor displays is addressing that problem.<sup>16</sup> We have also begun to see small-scale 3D holographic displays from companies like Holografica and QinetiQ.

**T**he ultimate dream of work on very large area displays is to achieve an immersive, truly 3D experience on a massive scale with the highest resolution. Such displays have the potential not only to redefine visualization but also to shape different kinds of education and exploration modalities. When combined with breakthroughs like light field displays,<sup>16,17</sup> the potential of visualization displays is limited only by imagination.

Open problems remain—from developing scalable distributed display architectures to deploying novel display technology and creating unified and scalable interaction to developing a data management and rendering paradigm. However, dedicated effort and resources have already resolved problems that once seemed insurmountable. Continued effort could bring similar results and ensure that the day of very large, truly interactive, immersive, and accessible displays is not far off. 

## References

1. M. Hereld, I. Judson, and R.L. Stevens, "Tutorial: Introduction to Building Projection-Based Tiled Display Systems," *IEEE Computer Graphics and Applications*, vol. 20, no. 4, 2000, pp. 22-28.
2. R. Yang, A. Majumder, and M.S. Brown, "Camera-Based Calibration Techniques for Seamless Multi-Projector Displays," *IEEE Trans. Visualization and Computer Graphics*, vol. 11, no. 2, 2005, pp. 193-206.
3. B. Sajadi et al., "Color Seamlessness in Multi-Projector Displays Using Constrained Gamut Morphing," *IEEE Trans. Visualization and Computer Graphics*, Nov. 2009, pp. 1317-1326.
4. G. Humphreys et al., "WireGL: A Scalable Graphics System for Clusters," *Proc. 28th Ann. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH 01)*, ACM, 2001, pp. 129-140.
5. S. Eilemann, M. Makhinya, and R. Pajarola, "Equalizer: A Scalable Parallel Rendering Framework," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 3, 2009, pp. 436-452.
6. R. Raskar et al., "The Office of the Future: A Unified Approach to Image-Based Modeling and Spatially Immersive Displays," *Proc. 25th Ann. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH 98)*, ACM, 1998, pp. 179-188.
7. M. Gross et al., "Blue-c: A Spatially Immersive Display and 3D Video Portal for Telepresence," *Proc. 30th Ann. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH 03)* ACM, 2003, pp. 819-827.
8. B. Sajadi and A. Majumder, "Markerless View-Independent Geometric Registration of Multiple Distorted Projectors on Vertically Extruded Surfaces Using a Single Uncalibrated Camera," *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 6, 2009, pp. 1307-1316.
9. B. Sajadi and A. Majumder, "Auto-Calibration of Multi-Projector CAVE-like Immersive Environments," *IEEE Trans. Visualization and Computer Graphics*, vol. 18, no. 3, 2012, pp. 381-393.
10. B. Sajadi and A. Majumder, "Automatic Registration of Multi-Projector Domes Using a Single Uncalibrated Camera," *Proc. Eurographics/IEEE Symp. Visualization (EuroVis 11)*, Blackwell, vol. 30, no. 3, 2011, pp. 1161-1170.
11. E. Bhasker, P. Sinha, and A. Majumder, "Asynchronous Distributed Calibration for Scalable Reconfigurable Multi-Projector Displays," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 5, 2006, pp. 1101-1108.
12. P. Roman, M. Lazarov, and A. Majumder, "A Scalable Distributed Paradigm for Multi-User Interaction with Tiled Rear Projection Display Walls," *IEEE Trans. Visualization and Computer Graphics*, vol. 16, no. 6, 2010, pp. 1623-1632.
13. R. Raskar et al., "Shader Lamps: Animating Real Objects with Image-Based Illumination," *Proc. 12th Eurographics Workshop on Rendering (EGWR 01)*, Eurographics, 2001, pp. 89-102.
14. A.J. Law, D. Aliaga, and A. Majumder, "Projector Placement Planning for High Quality Visualizations on Real-World Colored Objects," *IEEE Trans. Visualization and Computer Graphics*, vol. 6, no. 6, 2010, pp. 1633-1641.
15. R. Kooima et al., "A Multi-Viewer Tiled Autostereoscopic Virtual Reality Display," *Proc. 17th ACM Symp. Virtual Reality Software and Technology (VRST 10)*, ACM, 2010, pp. 171-174.
16. G. Wetzstein et al., "Tensor Displays: Compressive Light Field Synthesis Using Multilayer Displays with Directional Backlighting," *Proc. 39th Ann. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH 12)*, ACM, 2012, article no. 80.
17. A. Jones et al., "Rendering for an Interactive 360 Light Field Display," *Proc. 34th Ann. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH 07)*, ACM, 2007, article no. 40.

*Aditi Majumder is an associate professor in the Department of Computer Science at the University of California, Irvine. Her research interests include computer graphics and visualization, with a primary focus on making multiprojector displays truly commodity products that are easily accessible to the common man. Majumder received a PhD in computer science from the University of North Carolina at Chapel Hill. She is a member of IEEE. Contact her at majumder@ics.uic.edu.*

*Behzad Sajadi is a PhD student in computer science at the University of California, Irvine, and a quantitative analyst at D.E. Shaw & Co. His research interests include computer graphics and visualization with a focus on multiprojector displays, computational projectors, and cameras. Contact him at behzad.sadjadi@gmail.com.*

 Selected CS articles and columns are available for free at <http://ComputingNow.computer.org>.

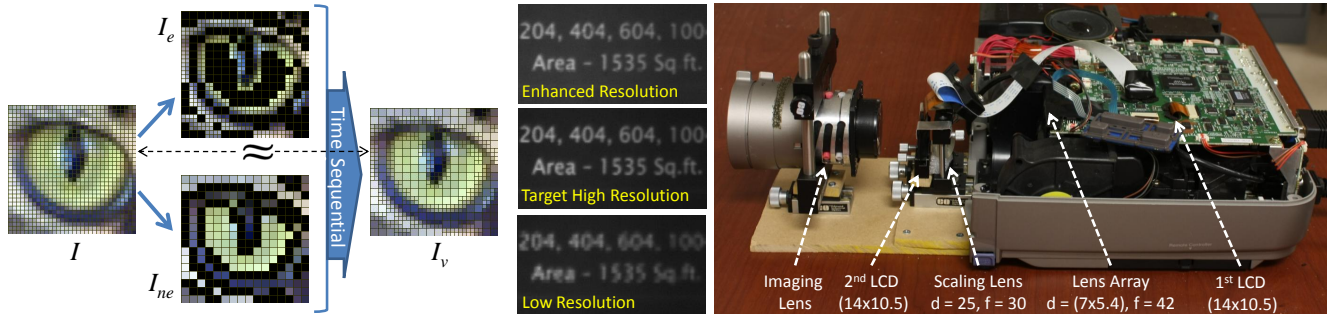


# Edge-Guided Resolution Enhancement in Projectors via Optical Pixel Sharing

Behzad Sajadi\*  
Department of Computer Science  
University of California, Irvine

M. Gopi†  
Department of Computer Science  
University of California, Irvine

Aditi Majumder‡  
Department of Computer Science  
University of California, Irvine



**Figure 1:** Left: This illustrates the basic concept of decomposing a target high resolution image,  $I$ , to a high resolution edge image,  $I_e$  and a complementary low resolution non-edge image,  $I_{ne}$ .  $I_e$  and  $I_{ne}$  are then displayed in a time sequential manner to create an image with spatially varying resolution where edges are displayed at a higher pixel density than the rest of the image. We call this the edge-enhanced image,  $I_v$ .  $I_v$  is perceptually close to  $I$ . Middle: The image from the enhanced-resolution projector compared with a target high-resolution image and a low resolution image. Note that the edge-enhanced image is perceptually almost indistinguishable from the target high-resolution image. Right: A side view of our prototype enhanced-resolution projector that achieves  $1024 \times 768$  edge-enhanced images by cascading two light modulator panels of  $512 \times 384$  resolution (animated illustrations in the video). The numbers in the image are in millimeters.

## Abstract

Digital projection technology has improved significantly in recent years. But, the relationship of cost with respect to available resolution in projectors is still super-linear. In this paper, we present a method that uses projector light modulator panels (e.g. LCD or DMD panels) of resolution  $n \times n$  to create a perceptually close match to a target higher resolution  $cn \times cn$  image, where  $c$  is a small integer greater than 1. This is achieved by enhancing the resolution using smaller pixels at specific regions of interest like edges.

A target high resolution image ( $cn \times cn$ ) is first decomposed into (a) a high resolution ( $cn \times cn$ ) but sparse *edge image*, and (b) a complementary lower resolution ( $n \times n$ ) *non-edge image*. These images are then projected in a time sequential manner at a high frame rate to create an *edge-enhanced image* – an image where the pixel density is not uniform but changes spatially. In 3D ready projectors with readily available refresh rate of 120Hz, such a temporal multiplexing is imperceptible to the user and the edge-enhanced image is perceptually almost identical to the target high resolution image.

To create the higher resolution edge image, we introduce the concept of *optical pixel sharing*. This reduces the projected pixel size by a factor of  $\frac{1}{c^2}$  while increasing the pixel density by  $c^2$  at the edges enabling true higher resolution edges. Due to the sparsity of the edge pixels in an image we are able to choose a sufficiently

large subset of these to be displayed at the higher resolution using perceptual parameters. We present a statistical analysis quantifying the expected number of pixels that will be reproduced at the higher resolution and verify it for different types of images.

**Keywords:** computational photography, computational displays, projection devices, high-resolution displays

## 1 Introduction

The recent years have seen a tremendous development in projector image quality, in terms of brightness, contrast and resolution. But, the relationship of cost with respect to available resolution in projectors is still super-linear. While a typical 2 Megapixel projector (e.g. Epson Home Cinema 8500 UB HD projector) costs around \$2500 today, a 4 Megapixel projector (e.g. Projection Design Cineo 35) costs around \$35,000. Thus, cost increases by more than a factor of 10 while resolution just doubles.

In this paper, we show that we can use two lower resolution light modulator panels (e.g. LCD or DMD panels) of resolution  $n \times n$  (assuming square panels without loss of generality) to design a projector that can display images that provide a close perceptual match to a higher resolution display of resolution  $cn \times cn$  where  $c$  is a small integer greater than 1. Our design uses a novel optical unit that can provide spatially variable pixel density across an image by realizing higher pixel density at specific areas of the image (Section 3). Since edges are known to be perceptually important [Valois and Valois 1990], we enhance the resolution of the image at the *edges* to create an *edge-enhanced* image (Figure 1).

We observe that a display with higher resolution targeted only at pixels where the high frequency content is concentrated is sufficient to create an image that is perceptually almost identical to one produced by a true higher resolution display (Figure 2). Typically, the higher frequency content in an image is given by a small set of pixels at the edges. We decompose our target high resolution ( $cn \times cn$ ) image into two images. We identify a set of edge pixels to create a higher resolution ( $cn \times cn$ ) *edge image* (Section 3.1). The rest of the image, complementary to the edge image, is called the *non-edge* image and is reproduced at the lower resolution ( $n \times n$ ). These two

\*e-mail: bsajadi@uci.edu

†e-mail: gopi@ics.uci.edu

‡e-mail: majumder@ics.uci.edu



**Figure 2:** Columns 1 and 2 show the results on a BUILDING image, 3 and 4 show the results on a PEOPLE image ( $c = 2$ ). Top row: target high resolution image ( $I$ ) and the chosen edge pixels. Bottom row: low resolution image  $I_l$ ; and its CIELAB  $\Delta E$  difference from  $I$ . Middle row: edge-enhanced image,  $I_v$ , i.e. the output of our system, created by superimposing  $I_e$  and  $I_{ne}$ ; and the CIELAB  $\Delta E$  of  $I_v$  from  $I$ . Compare the differences in the text 'PARK' and the lampposts in BUILDING and faces in PEOPLE in  $I_v$  and  $I_l$ . Please zoom in to see the differences.

are then projected in a time sequential manner to create the edge-enhanced image. With current 3D-ready projectors with available refresh rate of 120Hz or higher, such a temporal multiplexing is imperceptible to the user. To generate the edge and non-edge images at interactive rates, we design algorithms that can exploit the parallelism offered by the GPUs.

To display the edge image at a higher resolution, we present a novel concept which we call *optical pixel sharing* (Section 3.2). This reduces the area of each projected pixel by a factor of  $\frac{1}{c^2}$  while increasing their density by  $c^2$  at the edges. Due to the sparsity of edges in most images, our optical pixel sharing unit can choose appropriate regions to be displayed at higher resolution that would minimize visible artifacts while maximizing the number of pixels reproduced at the higher resolution (Section 4). We present a statistical analysis to quantify the edge pixels rendered in higher resolution and validate the results using a large number of images of various categories. Subsequently, we find that a relatively simple optical unit consisting of an array of  $c \times c$  lenses can implement optical pixel sharing effectively.

We demonstrate our enhanced resolution display in a prototype system and analyze the quality of the resulting images. Though theoretically  $c$  can be large, there are practical limits to the enhancement possible from our design – typically  $c = 2$  or  $c = 3$  resulting in 4 or 9 times higher resolution. Finally, this enhanced resolution is achieved by sacrificing the frame rate and the display brightness by a factor of 2. This is independent of the value of  $c$  and is due to the use of interleaved frames. However, our display can still be used in the standard low-resolution form without such a change in frame rate and brightness. We analyze several such issues in a detailed cost-benefit analysis in Section 6.

**Main Contributions:** We explore, for the first time, the concept of *variable spatial resolution display* – a display that does not provide uniform density of pixels everywhere but allocates higher densities at specific regions of interest based on the content. At a conceptual level, we seek to explore the fundamental question of content-specific spatial non-uniformity of resolution across a display.

Further, achieving *higher resolution images using lower resolution light modulator panels* is critically dependent on the ability to create smaller pixels. Smaller pixels when displayed at a higher density achieve true higher resolution, i.e. ability to represent higher frequency signals. Our work presents a system that uses computational optics designs to reduce the projected size of a pixel by a factor of  $\frac{1}{c^2}$  while increasing its density by  $c^2$  thereby enabling high resolution imagery from low resolution display modules.

Finally, we demonstrate the aforementioned concepts by building a laboratory prototype of enhanced-resolution gray scale projector that uses two LCD panels of resolution  $512 \times 384$  to create the perception of a display of resolution  $1024 \times 768$  (Figure 1). In order to generate content-specific edge and non-edge images at interactive rates we design parallel methodologies that can be easily implemented in GPUs for real-time performance.

## 2 Related Work

Our work builds upon a large body of literature in different domains and fills an important gap in multiple directions. Achieving higher resolution images using lower resolution light modulator panels, more commonly known as *display super-resolution*, has been the dream that has illuded the display community so far [Damera-Venkata and Chang 2009; Allen and Ulichney 2005; Aliaga et al. 2011; Jaynes and Ramakrishnan 2003; Majumder 2005]. This problem may seem analogous to that of generating a higher resolution image from multiple jittered lower resolution images, a well studied problem in the computer vision domain (detailed survey in [Babu and Murthy 2011]). However, a deeper analysis reveals them to be significantly different [Majumder 2005]. The information at a larger pixel captured at lower resolution can be thought of as encoding of multiplexed data from multiple smaller pixels that are captured at higher resolution. Hence, generating a higher resolution image from multiple jittered lower resolution images maps to a demultiplexing problem. Demultiplexing involves subtraction which in the context of displays, unlike in cameras, indicates negative light – a practical impossibility. Therefore, in displays, achieving higher resolution is dependent on generating smaller sized pixels – a hardware limitation that is impossible to overcome by software manip-

ulations alone. Hence, achieving true higher resolution, i.e. ability to display higher frequency signals, is not possible without changing the pixel size of the display [Majumder 2005]. Such a reduction of pixel size becomes available in a very limited manner on multi-projector planar displays due to projector key-stoning. The reduction of pixel size is much more significant when projecting on non-planar surfaces. Hence, recent work have proposed super-imposing pixels from multiple projectors on planar or non-planar displays [Damera-Venkata and Chang 2009; Jaynes and Ramakrishnan 2003; Aliaga et al. 2011]. However, since the reduction is very limited, dependent entirely on relative position and orientation of projectors with respect to the surface, and cannot be controlled precisely, the amount of super-resolution does not scale well with the number of projectors. Even for non-planar surfaces, [Aliaga et al. 2011] report a super-resolution of 1.33 when using 3 to 4 projectors. On the other hand, wobulation based techniques use temporal multiplexing of multiple low-resolution frames, each shifted by a fraction of a pixel (usually half a pixel) [Allen and Ulichney 2005]. In all these aforementioned techniques, since the pixel size cannot be reduced much, they cannot display higher frequencies but can only achieve a high frequency boost filter during image reconstruction leading to lesser attenuation of the higher frequencies and hence a more pleasing appearance (Figures 8 and 9). In contrast, we can reduce the pixel size by a factor of  $c$ , typically  $c = 2$  or  $c = 3$ , and hence can produce 4 or 9 times higher resolution at the edges.

The current industry has been on a rampant fervor to create more and more pixels. But the more fundamental question of how much resolution (pixel density) is needed at any spatial location is yet to be explored. Human perception studies have demonstrated our sensitivities to edges and insensitivity to very high or low spatial frequencies [Goldstein 2001; Valois and Valois 1990]. Consequently, edge sharpness, that depends both on the edge contrast and resolution [Dijk et al. 2003; Winkler 2001; Lin et al. 2006], has been used as a non-content-related parameter for object detection [Ran and Farvardin 1995] (content-related parameters being people, actions, facial expressions, etc). This has been exploited in many domains of image processing before. Edge preserving image denoising [C. Tomasi and Manduchi 1998; Durand and Dorsey 2002] compensates the loss of resolution during image filtering by retaining contrast near the edges. In image compression (e.g. JPEG compression) edge resolution is preserved by compressing those frequencies humans are less sensitive to more than the others. Similarly, in image upsampling [Kopf et al. 2007; Fattal 2007], edges have been upsampled more faithfully to retain the overall appearance. In rendering, analytically computed edges are used in combination with sparse sampling of the shading effects to generate high-quality images at interactive rates [Bala et al. 2003]. Finally, many works in non-photorealistic rendering, including silhouette and informative edges, and suggestive contours [Raskar and Cohen 1999; DeCarlo et al. 2004; Cole et al. 2009; Cole and Finkelstein 2010], have used the perceptual importance of the edges to achieve the required rendering effect. We supplement these by exploring the importance of edges in the context of displays by introducing the notion of *variable spatial resolution display* – a display that does not provide uniform pixel density everywhere but produces higher pixel densities at specific regions of the edges. While selecting these regions, our work exploits the sparsity of edges which supplements earlier works in compressive sensing [Wakin et al. 2006; Veeraraghavan et al. 2010; Sun and Kelly 2009].

The past decade has seen a tremendous activity in computational optics for capture devices [Levin et al. 2007; Liang et al. 2008; Raskar et al. 2006; Agrawal and Raskar 2007] that use optically coded images followed by computational decoding to capture additional information beyond just pixels (such as edges [Raskar et al. 2004], global and local illumination [Nayar et al. 2006], light fields

[Wilburn et al. 2005; Baker and Nayar 1999; Kuthirummal and Nayar 2006; Veeraraghavan et al. 2007], motion [Levin et al. 2008], and high dynamic range [Debevec and Malik 1997]). In the context of displays, computational optics has been explored in different contexts such as parallax barrier displays using stacked LCD panels [Lanman et al. 2010; Lanman et al. 2011; Wetzstein et al. 2011] and capture-cum-display devices [Hirsch et al. 2009]. We add to the domain of *computational displays* by using computational optics to impart additional capabilities to traditional displays.

### 3 Enhancement of Resolution

In this section, we describe our method to achieve an enhanced-resolution projector by selectively increasing the pixel density at the edges. First, we describe how to create the *edge-enhanced* image,  $I_v$ , that has higher resolution only at the edges (Section 3.1).  $I_v$  is formed by combining a high resolution ( $cn \times cn$ ) *edge* and a low resolution ( $n \times n$ ) *non-edge* image,  $I_e$  and  $I_{ne}$ , respectively. Thus,  $I_v = I_e + I_{ne}$  (Figure 1) and  $I_e$  and  $I_{ne}$  are temporally multiplexed to achieve this. The selected edge pixels are displayed at a higher resolution using  $c^2$  pixels whose size is  $c^2$  times smaller than the regular pixels. More importantly, these images are created using spatial light modulator panels with only  $n \times n$  pixels. To achieve this, we present the new computational optics model of *optical pixel sharing* (Section 3.2).

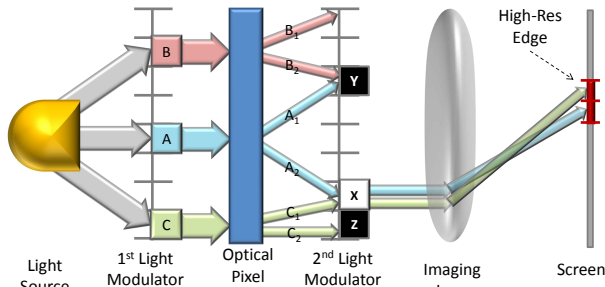
Subsequently, we show that a relatively simple optical design consisting of a grid of  $c \times c$  lenses provides a practical implementation option for the optical pixel sharing unit and also follows a set of optimality criteria that we derive for this unit (Section 3.2.2). However, there are other designs for the optical pixel sharing unit which can be further investigated for different applications. Hence, we first present the general concept of optical pixel sharing and then follow it with the exact optical design that we used in our prototype (using a grid of lenses). Further, we propose an alternate optical design that can provide more freedom in the design but requires more engineering efforts to implement.

#### 3.1 Edge-Guided Variable Resolution

Let us consider a target high-resolution  $cn \times cn$  image  $I$  to be displayed by a projector. Let the high resolution coordinate system be  $(s, t) \in \{1, \dots, cn\} \times \{1, \dots, cn\}$ , where  $\times$  denotes the cartesian product of two sets. Consider  $I_l$ , an  $n \times n$  image, created by low pass filtering  $I$  using a filter of size  $c$  in each direction. Let the low-resolution coordinate system be  $(i, j) \in \{1, \dots, n\} \times \{1, \dots, n\}$ . Also, each pixel  $(i, j)$  corresponds to  $c^2$  higher resolution pixels given by  $C_{ij} = \{c(i-1) + 1, \dots, ci\} \times \{c(j-1) + 1, \dots, cj\}$ . Let  $E$  be the set of edge pixels in  $I_l$  such that at every pixel  $e \in E$  we would like to display  $c \times c$  smaller pixels. Let  $\bar{E}$  be complement of  $E$  and hence the set of non-edge pixels. The edge image  $I_e$  consists of all smaller pixels  $\cup_{(i,j) \in E} C_{ij}$  and the non-edge image  $I_{ne}$  consists of the set of larger pixels  $\bar{E}$ . The sum of these two images provides us the *edge-enhanced* image  $I_v = I_e + I_{ne}$  (Figure 1).  $I_e$  and  $I_{ne}$  are complementary to each other in terms of the pixels they display.

In order to find  $E$ , we consider the maximum CIELAB  $\Delta E$  difference  $h(i, j)$  between  $I_l(i, j)$  and the corresponding  $c^2$  pixels in  $C_{ij}$  which is a measure of the *local variance* within pixels  $C_{ij}$  in  $I$ . All pixels  $(i, j) \in I_l$  such that  $h(i, j) > T$  denote a significant perceptual difference between  $I_l$  and  $I$ . Hence, they are added to  $E$ . Clearly, the number of pixels in  $E$ , depends on the value of  $T$ . The value of  $T$  can be set to 1, 2 or 3 units of just noticeable difference (JND), where each unit of JND equals 2.3 units of  $\Delta E$  difference. In the context of the displays, up to 3 JND difference is imperceptible to the human eye [Stupp and Brennesholtz 1999].

Table 1 provides a statistical analysis of the percentage of pixels in  $I_l$  that differ by more than a few JNDs from  $I$  for different categories



**Figure 3:** Displaying a perceptually high resolution image with low-resolution display modules.  $A$ ,  $B$ , and  $C$  are each copied to smaller pixel pairs  $(A_1, A_2)$ ,  $(B_1, B_2)$ , and  $(C_1, C_2)$  respectively.  $X$  and  $Y$  are in conflict due to  $A$ . Thus,  $Y$  is blocked (indicated by black) because  $X$  is passed (indicated by white) by the second light modulator. The simple jumbling function demonstrated here creates two copies of each pixel separated by half the size of the panel. We show in Section 3.2.2 how such a jumbling function can be implemented using a  $c \times c$  grid of lenses.

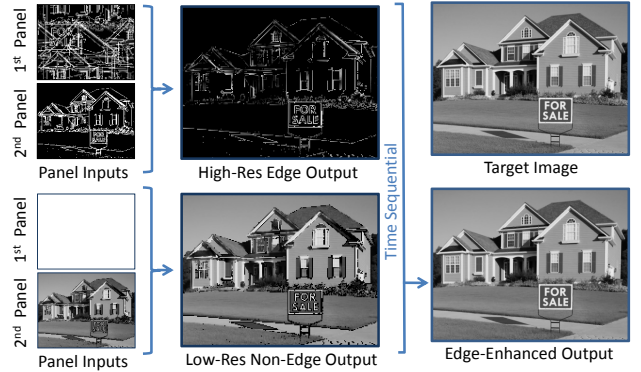
of images. We observe that for  $T = 3 JND$  this is a small number for most images. Thus, if only a small number of pixels are displayed at higher resolution (usually less than 15%), the resulting  $I_v$  would provide a close perceptual match to  $I$  (Figure 2).

### 3.2 Optical Pixel Sharing

In a traditional projector, white light from the lamp illuminates a light modulator panel that spatially modulates the light to produce different pixels. A light modulator panel (either a transmissive LCD or a reflective DMD) is a regular 2D array of modulation elements or pixels, where each element can block or attenuate the light to achieve a brightness between 0 (black) and 1 (white). When assuming a linear transfer function the relationship between the input and the modulated output is linear. For single chip projectors, three channels of colors are produced by temporally multiplexing R, G and B filters placed in front of the lamp. For three-chip projectors, multiple optical paths are used for the three channels that are then combined to create the final image. We first explain our design for a grayscale digital projector, which is then extended to multiple channels in Section 3.2.3.

In order to display an edge-enhanced image, we use two cascaded and aligned low-resolution ( $n \times n$ ) light modulator panels (e.g. LCD or DMD panels) with the optical pixel sharing unit in between them (refer to Figure 3).

**Creating Smaller Pixels:** The first aspect of creating higher resolution image involves creating pixels that are downsized by a factor of  $c$  in each direction. However, if  $n^2$  pixels from the first light modulator panel are made smaller by  $c^2$  (e.g. using a lens), there are still only  $n^2$  pixels available at the second modulator panel filling only  $\frac{1}{c^2}$  part of it. The second aspect of creating a high-resolution image thus involves creating enough smaller pixels to fill the second light modulator panel, i.e.  $c^2 n^2$  smaller pixels (e.g. using  $c^2$  lenses instead of one). Thus, every pixel will have  $c^2$  smaller copies. However, note that if all the  $c^2$  copies land on the same pixel of the second light modulator panel, they will not create higher resolution when passed through it. So, the key to achieve higher resolution is to have non-adjacent  $c^2$  smaller copies of each pixel. We define the mapping between these non-adjacent copies on the second panel to the pixels they are originating from on the first panel as the jumbling function  $F$ . In our particular design, the optical pixel sharing unit is achieved by a grid of  $c \times c$  appropriately designed lenses (Section 3.2.2). This implements a specific jumbling function where each pixel has  $c^2$  smaller copies separated by  $\frac{1}{c}$  distance across the panel



**Figure 4:** This figure shows the different images in the different stages of the designed system. Note that the input images are at half the resolution of the high resolution image so that they can be input to the low resolution display modules.

in each direction. This design is motivated by a set of optimality criteria discussed in Section 3.2.1. However, several other optical design can exist (Figure 10) to define different jumbling functions.

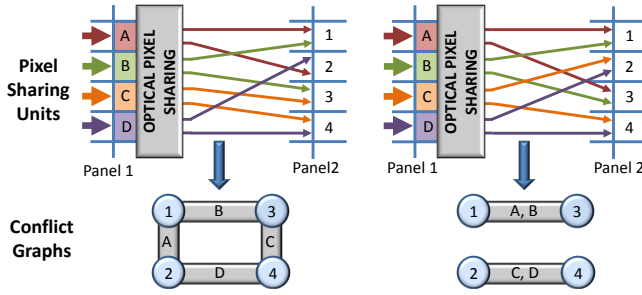
**Shared Control:** Since copies of  $c^2$  pixel from the first panel fall on each pixel of the second panel, they can be only attenuated together, but not individually, using the second panel. Conceptually, each pixel in the first panel creates  $c^2$  smaller spatially non-adjacent copies of the pixel with the same content ( $A$  controls  $A_1$  and  $A_2$  in Figure 3) while each pixel in the second light modulator panel attenuates  $c^2$  spatially adjacent smaller pixels of different content ( $X$  controls  $A_2$  and  $C_2$ ). The smaller pixels from the first panel can be used to focus a high density of pixels at the desired locations on the second panel. However, there will be  $c^2 - 1$  unwanted copies of each of these smaller pixels which are then blocked using the second light modulator panel. This shared control is used effectively to create the non-edge and edge image.

**Creating the Non-Edge Image:** Displaying the lower resolution non-edge image,  $I_{ne}$ , is relatively simple. For this, the first panel is turned ON completely and passes the light from the lamp. In this situation, the optical pixel sharing module does not have any effect and  $I_{ne}$  is used as the input to the second panel to create the non-edge pixels at lower resolution.

**Creating Edge Image:** In order to display the higher resolution edge image,  $I_e$ , the second panel blocks the pixels in  $\bar{E}$  while passing those in  $E$ . Hence, the input to the second panel is a low resolution binary edge-mask where only the pixels in  $E$  are ON.

For each edge pixel of the second panel, the first panel and the pixel sharing unit together create  $c^2$  adjacent smaller pixels. However, these smaller pixels get their input from  $c^2$  different pixels in the first panel. Let pixel  $(i, j)$  in the first panel be routed to the smaller pixel  $(s, t)$  in the second panel. Then the jumbling function  $F$  is defined as  $F(s, t) = (i, j)$  ( $F(A_2) = A$  in Figure 3). Note that  $F$  only depends on how the hardware for the optical pixel sharing is designed and not the image content. To display pixel  $(i', j')$  in the second panel at high resolution, we consider the corresponding  $c^2$  smaller pixels in  $C_{i'j'}$ . For each  $(s, t) \in C_{i'j'}$ , we input the value of  $I(s, t)$  at the location  $F(s, t)$  in the first panel. Note that, due to the jumbling, adjacent pixels in the first panel can create non-adjacent pixels in the displayed image. Figure 4 illustrates the input image to the first and second panels for generating  $I_e$  and  $I_{ne}$  respectively when using a  $c \times c$  grid of lenses that provide a jumbling function of  $F(s, t) = (s \bmod n, t \bmod n)$ .

**Conflicts for Edge Image:** The jumbling function  $F$  is a many-to-one function. In other words, a lower resolution pixel  $(i, j)$  in the



**Figure 5:** Two different jumbling functions and the corresponding conflict graphs. Each node denotes a low resolution pixel in the second panel. An edge between two nodes denotes a conflict. The edge is labeled with the pixel in the first panel due to which the conflict occurs. For the jumbling function on top, every pixel is in conflict with another. Hence, only one of the 4 pixels can be ON in the edge image. The jumbling function in the bottom is achieved by our optical design. This graph is a set of cliques and hence 2 of the 4 pixels can be ON in the edge image.

first modulator panel will feed many higher resolution pixels  $(s, t)$  in the displayed image. Let us assume  $(i, j)$  feeds two such higher resolution pixels:  $F^{-1}(i, j) = \{(s_1, t_1), (s_2, t_2)\}$  (A feeds  $A_1$  and  $A_2$  in Figure 3). Since only one input can be given to the pixel  $(i, j)$  in the first panel, it can either take the value of  $(s_1, t_1)$  or  $(s_2, t_2)$ , but not both. In other words, only one of these two higher resolution pixels can be allowed to pass through the second panel to create the final image, and the other pixel has to be blocked, e.g. in Figure 3,  $A_2$  is passed while  $A_1$  is blocked. Let these higher resolution pixels pass through two lower resolution pixels in the second panel  $(i'_1, j'_1)$  and  $(i'_2, j'_2)$ . In other words,  $(s_1, t_1) \in C_{i'_1, j'_1}$  and  $(s_2, t_2) \in C_{i'_2, j'_2}$ . Because of the above described scenario, only one of  $(i'_1, j'_1)$  or  $(i'_2, j'_2)$  can be kept ON. This is what we call a *conflict* ( $X$  and  $Y$  are in conflict due to  $A$  and  $C$  in Figure 3). Consequently, considering the conflicts among the pixels, only a subset of edge pixels  $E_M \subset E$  can be displayed at a higher resolution. Fortunately, due to the sparsity of the edge pixels  $E - E_M$  is typically a small set. Also, to assure that a pixel is not in conflict with itself,  $F$  is defined such that for any two smaller pixels  $(s, t), (s', t') \in C_{i, j}$ ,  $F(s, t) \neq F(s', t')$ .

### 3.2.1 Conflict Resolution

In this section, we find the maximal  $E_M$  such that there is no conflict between the pixels in  $E_M$ . This fundamentally depends on the jumbling function  $F$  and the content of  $I$  that dictates the position of the edges. We define a conflict graph,  $G$ , that has  $n \times n$  vertices, each representing a pixel in the second light modulator panel. Let us consider two vertices  $u$  and  $v$  in  $G$ . Let us define the jumbling function for a set of pixels  $Q$  as  $F(Q) = \bigcup_{q \in Q} F(q)$ .  $u$  and  $v$  are connected if and only if  $F(C_u) \cap F(C_v) \neq \emptyset$ . Thus, the connectivity of  $G$  is not content-specific and depends only on  $F$  (Figure 5).

For each vertex  $v$  of  $G$  such that  $v \in E$ , we assign as weight, the local variance  $h(v)$  (Section 3.1). For all  $v \in \bar{E}$ , we assign weight 0. Our goal is to resolve the conflicts while retaining the visual quality as much as possible. Achieving this amounts to finding maximum independent set or the maximum weighted independent set of  $G$ .

Since  $G$  is completely dependent on the jumbling function  $F$ , we seek to design  $F$  such that it provides us with a conducive independent set for our purpose. For this, let us consider a vertex  $u$  in  $G$ . Note that the maximum cardinality of  $F(C_u)$  is  $c^2$ . Further, for each pixel  $w \in F(C_u)$ , there are  $c^2 - 1$  other pixels  $v$  such that  $w \in F(C_v)$ . Thus, each pixel  $u$  in  $G$  can be in conflict with at most  $c^2(c^2 - 1)$  pixels. Hence, the maximum possible degree in  $G$  is  $c^2(c^2 - 1)$ .

Let us now consider a  $G$  where for any two vertices  $u$  and  $v$ , either  $F(C_u) \cap F(C_v) = \emptyset$  or  $F(C_u) = F(C_v)$ , i.e. the pixels in the

first light modulator panel from which pixel  $u$  and  $v$  in the second light modulator panel get their smaller pixels, overlap completely or none at all (Figure 5). In such a  $G$ , every pixel  $u$  will be connected to exactly  $c^2 - 1$  vertices which are all connected to each other forming a clique of size  $c^2$ . Thus, assuming  $n$  is divisible by  $c$ ,  $G$  would have  $\frac{n^2}{c^2}$  cliques which is also the size of its maximum independent set (assuming one pixel being chosen from each clique). Such a graph  $G$  is optimal for our purpose since (a) it minimizes the number of edges in  $G$  and hence the number of conflicts; (b) it maximizes the size of the independent set and hence the number of edge pixels that can be displayed at a higher resolution; and (c) it drastically simplifies the computation of the maximum independent set which is given by picking from each clique the vertex with the maximum  $h$  if at least one vertex of the clique belongs to  $E$ . In the following section, we demonstrate that such an optimal  $G$  and the corresponding  $F$  can be achieved using a simple optical design.

### 3.2.2 Optical Design

The previous section provides us the optimal connectivity or topological property for  $G$ . A desired geometric property of two conflicting pixels  $u$  and  $v$  in  $G$  is to be spatially far from each other to minimize scenarios where adjacent pixels in an edge are displayed at different resolutions. However, since this situation cannot be entirely avoided we discuss ways to alleviate the artifacts resulting from this in Section 4.2. In this section, we show that an optical design using a grid of lenses can achieve the optimal  $F$ , and thus  $G$ , respecting both the topological and geometric constraints.

Consider a  $c \times c$  grid of lenses, each of focal length  $f$ . If placed at distance  $d = f(1 + c)$  from the first panel, they create  $c^2$  copies of the image of this panel focused on the second panel, each scaled by a factor of  $\frac{1}{c}$  in each direction. Thus, the jumbling function is  $F(s, t) = (s \bmod n, t \bmod n)$  (Figure 5) and the resulting  $G$  satisfies both the geometric and topological constraints.

The  $c^2$  copies of the first light modulator panel should be placed contiguously without any gap or overlap. Let us assume the distance between the optical axes of two vertically adjacent lenses is  $r$ . Let us consider a pixel at vertical distances  $y$  and  $r + y$  from the optical axis of the top and bottom lenses respectively. The two copies of this pixel will be at distance  $\frac{y}{c}$  and  $\frac{r+y}{c}$  from the optical axis of the top and bottom lenses respectively. Therefore, considering the distance  $r$  between the two axis, the distance between these two copies is given by  $r + \frac{r}{c}$ . Assuming  $a$  to be the height of the light modulator panels, this distance should be same as  $\frac{a}{c}$ . Therefore,  $r = \frac{a}{1+c}$ . Similarly, we can find the required horizontal distance between the optical axis of the lenses. Note that  $f$  can be chosen arbitrarily but needs to be positive to focus the image of the first light modulator panel on the second one. More information on the elements we used in practice is provided in Section 5.2.

### 3.2.3 Extension to Multiple Color Channels

To extend the aforementioned design to three channels, the algorithm to generate  $I_e$  and  $I_{ne}$  remains identical since we use  $\Delta E$  thresholding which is valid for RGB color. However, the optical design can be extended in two different ways depending on the existing projector design architecture. For single chip architecture, our optical design can be ported as it is, by introducing a color wheel right before the first light modulator panel.

For 3-chip projector architecture [Stupp and Brennessoltz 1999], mirrors and dichroic filters are used to divide the white light into red, green and blue with three different optical paths. One panel is used for each path which are then recombined via a color bi-prism to be directed to the projection lens. The naive way to extend our design to this architecture is to use two panels with the optical pixel sharing unit in between for each of the three optical paths. This requires six panels. However, we observe that the separation be-

tween the edge and non-edge images can happen after combining the red, green, and blue images by adding the optical pixel sharing unit and a fourth panel to a traditional projector. However, to be able to project the low resolution frame, we need to bypass the optical pixel sharing unit (e.g. using moving or rotating mirrors). Since the bypassing happens at a very high frame rate, certain engineering efforts are indispensable in manufacturing such a projector.

## 4 Conflict Analysis

The fact that we can only display  $E_M \subset E$  in high-resolution due to conflicts has consequences in terms of image quality. First we analyze the percentage of edge pixels that gets dropped due to conflict (Section 4.1). Next, even if this percentage is relatively low, the location of the dropped edge pixels ( $E - E_M$ ) is important. If these dropped edge pixels, which are displayed at a lower resolution due to a conflict, are adjacent to a pixel in  $E_M$  which is displayed at a higher resolution, then it results in a visual artifact. We design a smoothing algorithm to alleviate such artifacts (Section 4.2).

### 4.1 Probability of Conflicts

Let us assume that  $|E|$  constitutes a fraction  $f$  of the total pixels in  $I_l$ , i.e.  $f = \frac{|E|}{n^2}$ . Since the measure of local variance,  $h$  depends on  $c$ ,  $f$  also depends on  $c$ . Table 1 provides the mean percentage  $f$  for a set of test images of different types for different values of  $c$ . These images were downloaded from the internet, their category assigned based on the search term used to find them. The threshold  $T$  is chosen to be between 1 and 4 JNDs. We experiment with values of  $c$  between 2 and 4 and use  $I_l$  to be of size  $3072 \times 2304$ .

Let  $f_M$  denote  $\frac{|E-E_M|}{|E|}$ , the fraction of the total pixels that cannot be displayed at a higher resolution due to conflicts. Our goal in this section is to estimate the expected value of  $f_M$ . For simplicity, we assume the edge pixels to be distributed spatially uniformly over the image. Let us consider the conflict graph  $G$ , a set of cliques each of size  $c^2$ , resulting from our design described in Section 3. Let us consider a vertex  $v \in G$ . The probability of  $v$  to be an edge pixel, i.e.  $v \in E$ , independent of the connectivity of  $G$  is  $f$ .

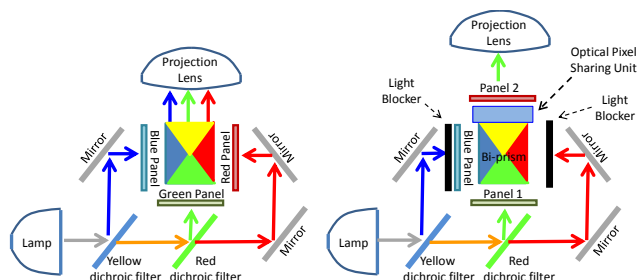
In practice, image sizes are almost always more than 1 megapixel. So, it is reasonable to assume  $c^2 \ll |E|$ . Now, let us consider the clique to which  $v$  belongs. The probability of having exactly  $k$  edge pixels in the  $c^2 - 1$  neighbors of  $v$  in this clique is  $c_{c^2-1} f^k (1-f)^{c^2-1-k}$ . Further, given that  $v$  has  $k$  neighbors, the probability of it being chosen to be displayed at higher resolution after conflict resolution, i.e.  $v \in E_M$ , is  $\frac{k}{k+1}$ . Thus, the probability of  $v$  to be displayed at high resolution when it is an edge pixel is given by

$$P(v \in E_M | v \in E) = \sum_{k=1}^{c^2-1} \frac{c_{c^2-1} f^k (1-f)^{c^2-1-k} k}{k+1} \quad (1)$$

In order to verify this result in practice, we analyze a set of images that have different values of  $f$  and different distribution of edges. The estimated value predicted by Equation 1 matches with  $f_M$  when considering a  $G$  generated by our optical design. This is shown in Table 1. Further, to evaluate how this value changes with changes in  $G$ , we do the same experiment by considering a  $G$  where random set of pixels form the cliques instead of a structured pattern of pixels forming them as in our design. Interestingly,  $f_M$  resulting in such a  $G$  is consistently higher than that of the  $G$  provided by our optical design. We attribute this to the large spatial distance between the vertices in each clique which minimizes the possibility of having more than one  $v \in E$  in the same clique.

### 4.2 Smoothing to Alleviate Artifacts

Due to conflicts, two adjacent edge pixels might be displayed in different resolutions. This drastic change in resolution can become perceptible, creating a visual artifact. In order to alleviate this, we



**Figure 7:** Design of our prototype (right) by changing a standard 3-chip projector (left). The green channel is used as the first panel. The red panel is moved between the bi-prism and the projection lens to be used as the second panel. The optical pixel sharing unit is placed between these two. The blue and red optical paths are blocked using light blockers. Note that the images generated by the prototype are green. Please check the video for illustrations.

propose a smoothing algorithm. For every pixel  $(i, j) \in E$ , we assign a binary weight  $W(i, j)$ . A weight of 1 indicates that the pixel will be displayed at high resolution and 0 indicates that the pixel will be displayed at low resolution due to a conflict i.e.  $\forall (i, j) \in E_M, W(i, j) = 1$  and  $\forall (i, j) \in E - E_M, W(i, j) = 0$ .  $\forall (i, j) \in \bar{E}$ ,  $W(i, j)$  is not relevant and hence are initialized to a don't care term. The smoothing procedure smoothes the discontinuities in the relevant portions of  $W$  to create a non-binary  $W'$ ,  $0 \leq W'(i, j) \leq 1$  and is described in the subsequent paragraph.  $W'(i, j)$  provides a weight for combining the high and low resolution content at pixel  $(i, j)$  to avoid visual artifacts. Hence, to create the final  $I_e$ , we weight the contribution from the higher resolution  $I_e$  by  $W'(i, j)$  and get the rest of the contribution from  $(1 - W'(i, j))I_l$ . Note that for  $(i, j)$  where  $W(i, j) = 0$ ,  $W'(i, j)$  has to be kept at 0 since these pixels cannot be reproduced any better than the low resolution. Further, for pixels where  $W(i, j) = 1$  the goal is to reduce the weight minimally so that high resolution content is retained maximally.

Such a smoothing cannot be achieved by a linear low pass filtering (LPF) operation since LPF both lowers hills and lifts valleys. In this case, the valleys ( $W(i, j) = 0$ ) cannot be lifted. Interestingly, the scenario is similar to smoothing of brightness in a multi-projector display as faced in [Majumder and Stevens 2005] where the brighter pixels have to be attenuated, but the dimmest pixels cannot be brightened. Hence, we use a similar method where for each pixel  $(i, j)$  we apply  $W'(i, j) = \min(W(i, j), W(i+1, j) + \epsilon, W(i-1, j) + \epsilon, W(i, j+1) + \epsilon, W(i, j-1) + \epsilon, W(i+1, j-1) + \sqrt{2}\epsilon, W(i+1, j+1) + \sqrt{2}\epsilon, W(i-1, j-1) + \sqrt{2}\epsilon, W(i-1, j+1) + \sqrt{2}\epsilon)$ . Multiple iterations, where all pixels perform the aforementioned operation in each iteration, provides the same result as presented by the sequential dynamic programming approach in [Majumder and Stevens 2005] to achieve an optimal  $W'$ . However, since in each iteration all pixels can be processed in parallel, this version of the algorithm can be easily implemented on the GPU (Section 5.3). The maximum number of steps required for convergence would be  $\frac{1}{\epsilon}$ . Typical  $\epsilon$  used in our results is 0.125 which indicates 8 steps to convergence. Figure 6 illustrates  $W$  and  $W'$  and how it alleviates the visual artifacts due to conflicts.

## 5 Implementation and Results

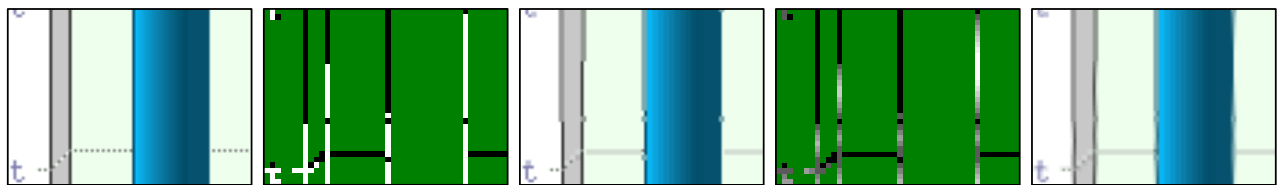
We have performed two experiments to validate the feasibility of our design. We first demonstrate an enhanced-resolution projector using a 3D ready projector (Section 5.1). Next, we build a grayscale prototype by modifying an existing LCD projector (Section 5.2).

### 5.1 Simulation using a 3D Ready Projector

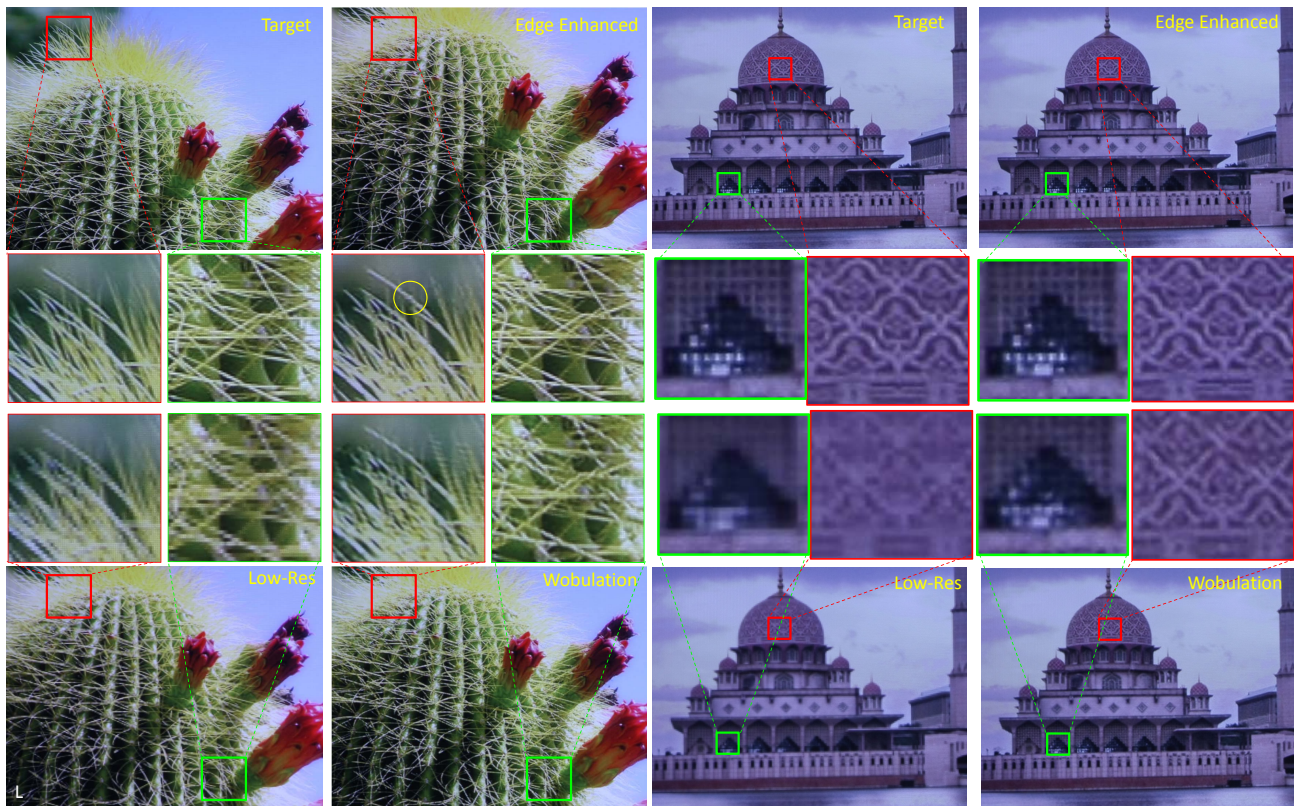
We use a BenQ MX660 3D ready single-chip DLP projector that provides 120Hz refresh rate. We consider a test image,  $I$ , at the

c	T	Architecture				People				Nature				Oil Paintings				Line Drawings			
		f	$f_M^L$	$f_M^R$	$f_M^P$	f	$f_M^L$	$f_M^R$	$f_M^P$	f	$f_M^L$	$f_M^R$	$f_M^P$	f	$f_M^L$	$f_M^R$	$f_M^P$	f	$f_M^L$	$f_M^R$	$f_M^P$
2	1	31	35	36	36	25	30	30	30	41	41	42	42	25	29	29	29	10	15	13	13
	2	14	17	18	18	9	11	12	12	20	23	24	24	7	9	9	9	7	12	10	10
	3	7	10	10	10	4	5	6	6	11	13	14	14	3	4	4	4	6	11	8	8
	4	4	6	6	6	2	3	3	3	6	8	9	9	1	2	2	2	5	9	7	7
3	1	56	77	78	78	54	77	77	77	66	80	80	80	59	78	78	78	15	38	36	36
	2	30	58	59	59	24	53	53	53	41	62	63	63	24	50	50	50	12	34	31	31
	3	19	44	45	45	13	35	35	35	27	49	50	50	12	29	30	30	10	31	28	28
	4	13	34	35	35	8	24	24	24	19	40	40	40	6	18	18	18	9	30	26	27
4	1	65	89	89	89	64	89	89	89	74	91	91	91	71	90	90	90	17	56	54	54
	2	40	80	80	80	35	79	79	79	51	82	82	82	36	75	75	75	14	52	50	50
	3	28	70	71	71	21	65	65	65	38	72	72	72	20	57	58	58	13	50	48	48
	4	20	62	62	62	14	52	52	52	29	63	63	63	12	42	42	42	12	49	46	46

**Table 1:** We collect statistics from 5 different categories of images (100 images each): Architecture, People, Nature, Oil Paintings and Line Drawings for  $c = 1, 2, 3$  and  $T = 1, 2, 3, 4$ JNDs. The resolution of each image is  $3072 \times 2304$ . We present the mean percentage (%) of edge pixels ( $f$ ), mean percentage (%) of pixels dropped due to conflicts for the lens array design ( $f_M^L$ ) and a random jumbling function ( $f_M^R$ ) and the predicted value of  $f_M^P$  using Equation 1 ( $f_M^P$ ). Note that  $f_M$  matches closely to the predicted  $f_M^P$  and, except for the Line Drawings, the number of pixels dropped is consistently higher when using  $f_M^R$  than using  $f_M^L$ , illustrating the superior quality of the lens array design.



**Figure 6:** This illustrates the effect of our smoothing algorithm. From left to right: target high-resolution image; edge-mask before smoothing ( $W$ ) contains only binary values – black denoting the pixels dropped due to conflicts, white denoting the passed pixels and green denoting the non-edge pixels that are irrelevant; edge-enhanced image before smoothing; edge mask after smoothing ( $W'$ ) – and the hence the gray values; and the edge enhanced image after smoothing.



**Figure 8:** Two sets of images, on left and right, captured from our simulation on a 3D ready projector. From top left in a scanline order: true high resolution, edge-guided enhanced resolution, low resolution, and Wobulated image. The zoomed-in views in particular show the close match of our variable resolution image with the true high-resolution image while the low-resolution and Wobulated images fail to show the fine details. The yellow circle shows a spot where our method cannot reproduce the edge at a higher resolution due to a conflict. Please zoom in to compare different regions of the image. The images are cropped to account for the reduction from their projected size.

native resolution of the projector of  $1024 \times 768$  and create the edge image,  $I_e$ , at the same resolution. We assume  $c = 2$  and create the non-edge image,  $I_{ne}$  resolution  $512 \times 384$ . Using this we simulate a projector with resolution  $512 \times 384$  creating an enhanced resolution of  $1024 \times 768$ . The non-edge image is then upsampled to the resolution of the edge image using the nearest neighbor method. These images are then projected in a time sequential manner to simulate  $I_v$ . Simulating  $512 \times 384$  resolution panels allowed us to compare the result of our method with a target high-resolution image. This was also instrumental in comparing our work with wobulation [Allen and Ulichney 2005] and other similar techniques using superimposing multiple projectors [Damera-Venkata and Chang 2009; Jaynes and Ramakrishnan 2003]. Wobulation uses multiple temporally multiplexed low-resolution frames, each mechanically shifted by a fraction of a pixel (usually 0.5) to super-sample each low-resolution pixel more densely. More recently, the e-shift technology introduced by JVC achieves a fractional shift of pixels similar to wobulation using electronic means instead of mechanical ones<sup>4</sup>. We simulate the half pixel shift of wobulation using a shift of 1 physical pixel. We use two frames for wobulation – the usual practical choice, though 4 or 8 is possible theoretically – to avoid significant loss in frame rate and to provide a fair comparison with our method which needs to multiplex only two frames.

We capture  $I_v$  with a Canon Rebel XSi camera ( $4272 \times 2848$  pixels) at  $\frac{1}{30}$  second exposure and  $I_e$ ,  $I_l$  and  $I_{ne}$  at  $\frac{1}{60}$  exposure. The difference in the exposure compensates for the varying brightness of these images in our design. We also capture the wobulated image for comparison. As discussed in Section 2, we demonstrate that wobulation cannot display higher frequencies (Figure 8). In contrast, our technique can produce true higher resolution (via smaller pixels) to create the closest match to a target higher resolution image (More results in the supplemental video and powerpoint slides).

## 5.2 Our Prototype

We also modified an Epson EMP-74 3-chip LCD projector of resolution  $1024 \times 768$  to create a prototype grayscale enhanced-resolution projector for  $c = 2$  (Figure 7). For this, we use the LCD panel for green channel as our first light modulator panel. Then we remove the projector lens and move the panel for the red channel outside the projector in front of the green panel to create our second light modulator panel. This panel is then connected to the input of the red panel of the projector using a ribbon cable. The blue and red light paths are then blocked and the imaging lens is placed after the second panel. This provides us with the two cascaded light modulator panels. We control the image of the first and second LCD panels by using the green and red channels of the projector respectively assuring that the two images are synchronized. An optical pixel sharing unit based on our lens array design is then placed in between the two panels. Finally, we place the projector lens in front of the second LCD panel. The labeled prototype is shown in Figure 1 and illustrated with animation in the supplementary video.

For the optical pixel sharing unit, we had to slightly modify our design since we were limited by the availability of COTS components. First, we needed a  $2 \times 2$  lens array to make  $c^2 = 4$  copies while commodity arrays usually provide larger grids. Therefore, we blocked all the lenslets except for a  $2 \times 2$  grid in the middle. This considerably lowered the light efficiency of our prototype. Second, commodity lens arrays are not available in all different focal lengths and sizes. We used a lens array<sup>1</sup> with focal length of  $f_1 = 41.9mm$  and lenslet size  $7mm \times 5.4mm$  that was the closest to what we need. Therefore, the position of the lenslet array is guided by the requirement that the multiple copies of the image of the first LCD panel

(of size  $14mm \times 10.5mm$ ) formed by this array should not have any gap between them. In this position, the lenslet array provides a magnification factor of 1.2 instead of the desired 0.5. Therefore, we had to use an extra lens between the lens array and the second LCD panel to scale down the pixel size appropriately. This is done using an aspheric lens with focal length of  $f_2 = 30mm$ <sup>2</sup>. Then the second LCD panel is placed where the output from this lens is focused. This is achieved when the lenslet array is placed 76.8mm in front of the first LCD panel, the aspheric lens is placed 50.2mm in front of the lenslet array, the second LCD panel is placed 17.5mm in front of the aspheric lens, and finally the projection lens is placed 27mm in front of the second LCD panel. The commodity lens array (less than \$100) is designed to create a uniform illumination and not for imaging purposes. Therefore, it does not provide a uniform focus across the entire image. In order to alleviate the effect of the spatially varying defocus in our experiments, we grouped every  $2 \times 2$  pixels of the projector in both of the LCD panels to provide larger pixels to alleviate the focusing issues. Thus, our projector had two panels with effective resolution of  $512 \times 384$  and we created an enhanced-resolution projector of  $1024 \times 768$  using these. Note that the prototype generates a green image which is converted to grayscale to generate the results in Figure 9.

In order to make sure all the elements are in focus, we use precision metric stages<sup>3</sup> that allow us to move the elements with micron level accuracy. Due to some small radial distortion from the inexpensive lens array and rotational and translational misalignments between the panels, each pixel in the second panel could not be perfectly aligned with four smaller pixels coming from the optical pixel sharing unit. Therefore, we used standard camera-based registration procedures used in planar multi-projector displays [Chen et al. 2002; Bhasker et al. 2007] to achieve the alignment by warping the input images appropriately. Note that this registration procedure is due to the limitations common to any inexpensive lab setup. The radial distortion can be virtually removed using pairs of aspheric lenses that provide a nearly aberration free system [Stupp and Brensholtz 1999]. Precision alignment, already standard in any 3-chip projector during combination of light from the three channels, can also be used for aligning pixels between the first and second panels.

Similar to our experiments with the 3D-ready projector, we can simulate the results of a true high resolution projector and wobulation. For this, we allow the first modulator panel to pass all the white light and input appropriate images to the second panel. Since the lens array precedes the second modulator panel, we do not face any focusing issues while projecting these images (More results in the supplemental video and powerpoint slides).

## 5.3 GPU Implementation

Our method to generate  $I_e$  and  $I_{ne}$  can be parallelized for efficient GPU implementation. Evaluating  $h$  at every pixel can be done in parallel. In order to compute the jumbled image, all the cliques in the conflict graph  $G$  can be processed in parallel. These generate the inputs for the two light modulator panels for  $I_e$ . To compute  $I_{ne}$ , we first low-pass-filter the image and then turn OFF the pixels belonging to  $E$ . Both these computations can be run in parallel in the GPU. The smoothing method can also be implemented in the GPU as explained in Section 4.2. We used such a GPU implementation on nVidia GeForce GTX 560 Ti to generate  $I_e$  and  $I_{ne}$  at 120fps.

## 6 Discussion

In this section we discuss several analysis and cost-benefit issues related to our design.

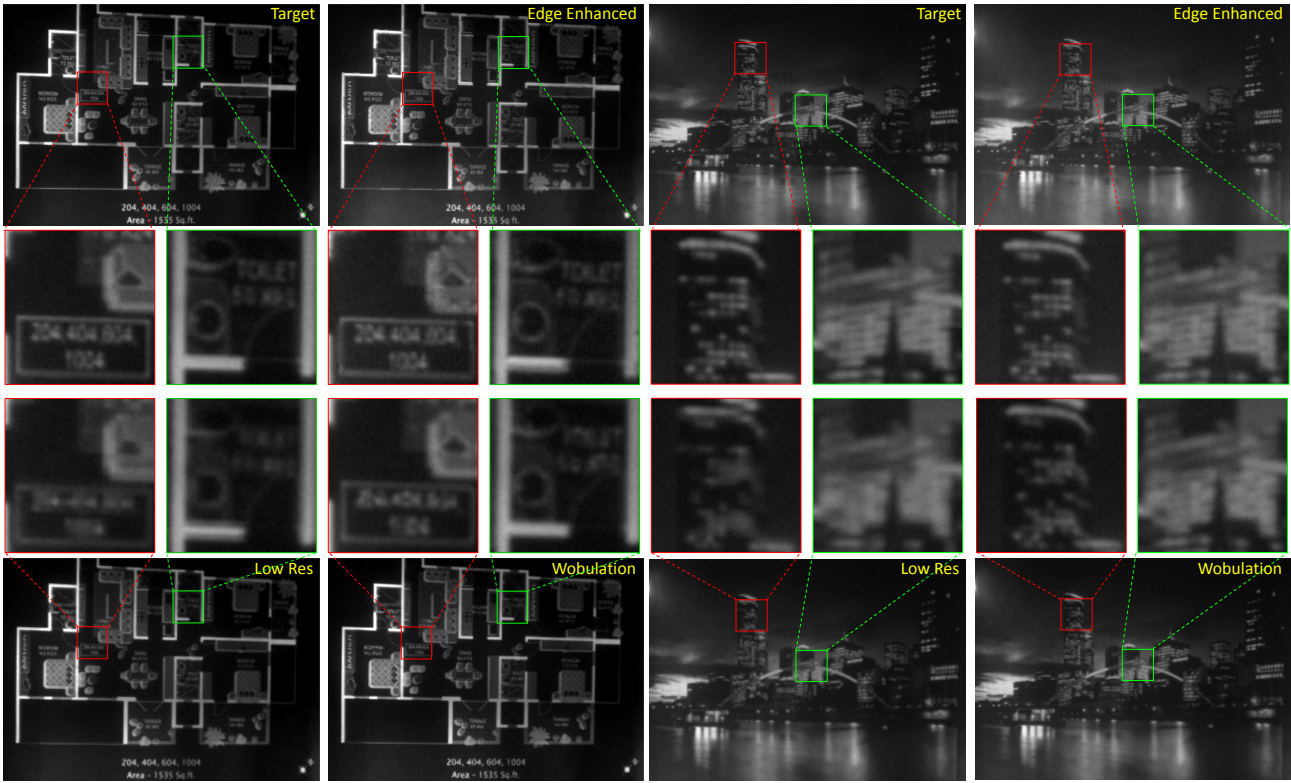
<sup>4</sup><http://www.engadget.com/2011/09/08/jvc-shows-off-projectors-with-4k-precision-but-not-quite-4k-pix/>

<sup>1</sup><http://www.edmundoptics.com/products/displayproduct.cfm?productid=3092>

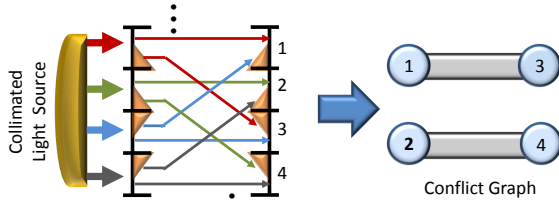
<sup>2</sup><http://www.edmundoptics.com/products/displayproduct.cfm?productid=2953>

<sup>3</sup><http://www.edmundoptics.com/products/displayproduct.cfm?productid=1577>

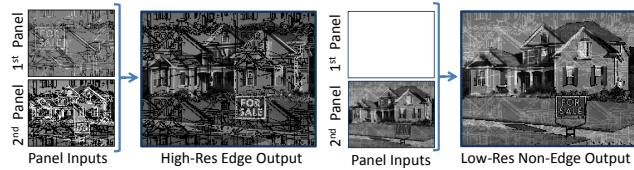




**Figure 9:** Two sets of grayscale images on left and right captured from our optical prototype from comparison. From left to right: true high-resolution; edge-guided enhanced resolution; low-resolution; Wobulation technique. The zoomed-in views in particular show the close match of our variable resolution image with the true high-resolution image while the low-resolution image and the Wobulation technique fail to show the fine details of the image. The images are cropped to account for the reduction from their actual projected size.



**Figure 10:** The picture shows the optical pixel sharing unit using the prism array design. Half of each pixel is covered by a prism. On the right we show the conflict graph for this jumbling function.



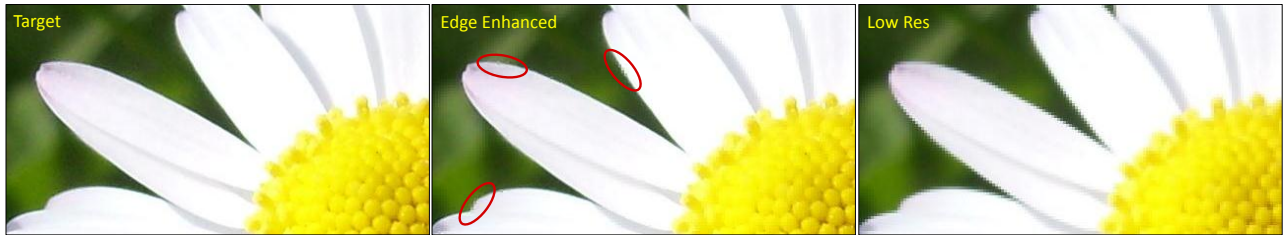
**Figure 11:** Modified input to the first and second light modulator panel and the resulting different  $I_e$  and  $I_{ne}$  to reduce flicker.

**Alternative Optical Design:** Unlike our optical design (Section 3.2.2) that provides a specific jumbling function  $F$ , here we present an alternative optical design to implement a general  $F$ . Assuming a collimated uniform light source, we use two prism arrays, one after the first light modulator panel and another before the second one. These arrays have a grid of  $c \times c$  small prisms covering each pixel, each with a different angular orientation. The former prism array splits the pixels by applying different angular changes to the incoming rays. The latter provides an inverse angular change making all the light rays parallel again (Figure 10). To implement a general  $F$ , the prisms should be custom cut. Figure 10 shows a simple yet different jumbling function using this design. With this jumbling

function, pixels  $(s, t)$ ,  $(s + 2, t)$ ,  $(s, t + 2)$  and  $(s + 2, t + 2)$  will be copies of the same pixel in the first light modulator panel. Though the conflict graph  $G$  is topologically similar to the one achieved by our optical design (creating cliques), the locations of the copies are different. In our lens array based design, pixels  $(s, t)$ ,  $(s + \frac{n}{c}, t)$ ,  $(s, t + \frac{n}{c})$  and  $(s + \frac{n}{c}, t + \frac{n}{c})$  will be copies of each other.

However, this design is harder and more expensive to implement for two reasons. (a) It is very difficult (and expensive) to create a perfectly collimated broadband light source while collimated narrowband light sources (e.g. LED light sources used in pico-projectors) usually come at the cost of low light efficiency; and (b) it is hard to custom manufacture all the small prisms and therefore, the prisms should follow a regular pattern, as showed in Figure 10.

**Extending to Video:** In a projector of temporal frequency 120 fps, our method produces an acceptable video rate of 60 fps. Our first prototype using a 3D ready projector (Section 5.1) however, showed a barely perceptible flicker when viewed from close quarters. This is due to a temporal discontinuity caused by the significant difference in the global brightness of the alternating edge and non-edge images. To alleviate this, we used the pixels in the edge image which are not used to project any edges. Note that very few of the pixels (often even less than 10%) in the second light modulator panel are ON for the edge image. Since every pixel passed by the second light modulator panel results in three blocked pixels, even after considering three more conflicting pixels for each of these, more than 60% of pixels are not used. At these pixels, we divide the low resolution image between the edge and non-edge images (Figure 11). This increases the global brightness of the edge image while reducing the same for the non-edge image. Hence, the discontinuity in brightness is reduced thus removing the flicker.



**Figure 12:** This is the result of our method when enhancing resolution with  $c = 3$  providing 9 times more pixels. For this we have to use a higher  $T = 4$  JND to assure a relatively sparse edge image. This results in slight blurriness in the center of the flower. Further, since conflicts are greater in this scenario, we have marked a few of them with red. However, though our method could not reproduce all the desired edges at high resolution, the quality of the edge enhanced image is still far superior to the low-resolution image. Please zoom in to compare.

Further, the locations of the conflicting pixels (pixels in  $E - E_M$ ) can be temporally incoherent creating visual artifacts. Hence, an extension of our smoothing algorithm (Section 4.2) to three dimensions (two spatial and one time) is needed. Our smoothing algorithm is based on the brightness smoothing algorithm presented in [Majumder and Stevens 2005] for multi-projector displays, a variant of which has been used for contrast enhancement of images and video [Majumder and Irani 2007]. We plan to explore the possibility of adapting this technique for our purpose in the future.

**Image Quality Issues:** In our enhanced-resolution projector, the edge image displayed at 0.5 duty cycle, is sparse and hence reduced in brightness. This leads to an overall reduction in brightness of the image. Since the black offset due to the images do not change, this reduction in the overall brightness results in a small reduction of contrast in the display. For current projectors with contrast as high as 50,000 : 1, this is a small price to pay for the increased resolution.

**Limits on Enhancement of Resolution:**  $c$  is a measure of the resolution enhancement achieved. Though theoretically any  $c$  is possible, the practical value of  $c$  depends on the content and the threshold  $T$  (Table 1). For almost all types of images, we find  $f$  and  $f_M^L$  to be reasonably small for  $c = 2$  and  $T = 3$  JND assuring a close perceptual match between a target high resolution image and our edge-enhanced image. For  $c = 3$ , we need to raise the threshold to be  $T = 4$  JND or higher to get an acceptable value for  $f$  and  $f_M^L$ . We observe that  $T$  higher than 3 JNDs can also provide an adequate perceptual match to the target high resolution image (Figure 12). Finally,  $f$  and  $f_M$  are too high for  $c = 4$  indicating that an enhancement of more than  $c = 2$  or  $c = 3$  is impractical.

**Light Efficiency:** Our design would incur some light loss due to the use of two modulators. This is a minor issue for DMDs which are very light efficient. For LCD panels too, light efficiency has improved considerably and more efforts are in progress [Lazarev and Palto 2009]. Our prototype was designed using LCD based projectors purely due to lesser difficulty in opening them up in a non-production laboratory setting. Further, the loss of light due to the alternation between the edge and non-edge frames can be alleviated by assigning unequal time slots to these two frames at the cost of a small contrast reduction for the very high contrast edges (as in [Majumder et al. 2010] for color balancing).

**Diffraction:** A variable resolution projector using a lens array, reduces the effective aperture size by a factor of  $c$ . However, since depth of focus is not as critical in projection as in photography most projectors use lenses with relatively high f-numbers ( $F/2$  is typical in projection lenses). Further, most lenses, especially in high-resolution projectors (e.g. 8K Sony and Projection Design Cineo 35 projectors) are relatively large, resulting in a large effective aperture. Thus, the resolution limit from diffraction is often considerably higher than the projector resolution. However, this can become a limiting factor of our design with the increasing resolution and decreasing size in projectors. In particular, it can limit the

use of our technique to increase the resolution of pico projectors.

## 7 Conclusion

In conclusion, we have presented the first projector that can use  $n \times n$  resolution light modulator panels to enhance the resolution at selected regions providing a perceptually close match to a target high resolution image of  $cn \times cn$  where  $c$  is a small integer greater than 1. This is made possible by the novel concept of optical pixel sharing that allows selected regions of the image at the edges to be reproduced using smaller pixels with size scaled by  $\frac{1}{c^2}$  at a density scaled by  $c^2$ . Thus, we explore, for the first time, the concept of a display where pixel density changes spatially based on the content. Our prototype provides a convincing proof of concept and shows the superior perceived resolution when compared to images from standard projector with same-sized light modulator panels.

Achieving higher than HD resolution in commodity projectors today is limited by the density of pixels that can be packed in small light modulator panels. Hence, larger panels used for 4K or 8K projectors justify their cost. Though our projector allows super HD resolution in commodity projectors today, the technique is also scalable to higher resolutions of 4K and 8K projectors. More interestingly, this work opens up the concept of “resolution at demand” where resolution can be targeted at “important pixels”, like faces or humans or boundaries of foreground and background. The possibilities are numerous and can even be application specific. Thus, our work can trigger new directions for application-specific resolution retargeting in projection based displays.

## Acknowledgements

We would like to acknowledge our funding source NSF CAREER IIS-0846144. We also thank our co-student at the Interactive Graphics and Visualization (iGravi) laboratory at UC, Irvine, Duy-Quoc Lai, for helping with the GPU implementation.

## References

- AGRAWAL, A., AND RASKAR, R. 2007. Resolving objects at higher resolution using a single motion blurred image. *IEEE CVPR*.
- ALIAGA, D., YEUNG, Y. H., LAW, A. J., SAJADI, B., AND MAJUMDER, A. 2011. Fast high-resolution appearance editing using superimposed projections. *ACM TOG*.
- ALLEN, W., AND ULICHNEY, R. 2005. Wobulation: Doubling the addressed resolution of projection displays. *SID*.
- BABU, R. S., AND MURTHY, K. E. S. 2011. A survey on the methods of super-resolution image reconstruction. *IJCV 15*, 2.
- BAKER, S., AND NAYAR, S. K. 1999. A theory of single-viewpoint catadioptric image formation. *IJCV 35*, 2.

- BALA, K., WALTER, B., AND GREENBERG, D. P. 2003. Combining edges and points for interactive high-quality rendering. *ACM Transactions of Graphics (Siggraph)*.
- BHASKER, E., JUANG, R., AND MAJUMDER, A. 2007. Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. *IEEE TVCG 13*, 6.
- CHEN, H., SUKTHANKAR, R., WALLACE, G., AND LI, K. 2002. Scalable alignment of large-format multi-projector displays using camera homography trees. *IEEE Vis*.
- COLE, F., AND FINKELSTEIN, A. 2010. Two fast methods for high-quality line visibility. *IEEE TVCG 16*, 5.
- COLE, F., SANIK, K., DECARLO, D., FINKELSTEIN, A., FUNKHOUSER, T., RUSINKIEWICZ, S., AND SINGH, M. 2009. How well do line drawings depict shape? *ACM TOG 28*, 3.
- C. TOMASI, AND MANDUCHI, R. 1998. Bilateral filtering for gray and color images. *ICCV*.
- DAMERA-VENKATA, N., AND CHANG, N. L. 2009. Display supersampling. *ACM TOG 28*, 1.
- DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. *ACM Siggraph*.
- DECARLO, D., FINKELSTEIN, A., AND RUSINKIEWICZ, S. 2004. Interactive rendering of suggestive contours with temporal coherence. *NPAR*.
- DIJK, J., VAN GRINKEL, M., VAN ASSELT, R. J., VAN VLIET, L., AND VERBEEK, P. W. 2003. A new sharpness measure based on gaussian lines and edges. *CAIP*.
- DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. *ACM TOG 21*, 3.
- FATTAL, R. 2007. Image upsampling via imposed edges statistics. *ACM TOG 26*, 3.
- GOLDSTEIN, E. B. 2001. *Sensation and Perception*. Wadsworth Publishing Company.
- HIRSCH, M., LANMAN, D., HOLTZMAN, H., AND RASKAR, R. 2009. BiDi screen: A thin, depth-sensing LCD for 3D interaction using lights fields. *ACM TOG 28*, 5.
- JAYNES, C., AND RAMAKRISHNAN, D. 2003. Super-resolution composition in multi-projector displays. *PROCAMS*.
- KOPF, J., COHEN, M. F., LISCHINSKI, D., AND UYTENDAELE, M. 2007. Joint bilateral upsampling. *ACM TOG*.
- KUTHIRUMMAL, S., AND NAYAR, S. K. 2006. Multiview radial catadioptric imaging for scene capture. *ACM TOG*.
- LANMAN, D., HIRSCH, M., KIM, Y., AND RASKAR, R. 2010. Content-adaptive parallax barriers: optimizing dual-layer 3d displays using low-rank light field factorization. *ACM TOG*.
- LANMAN, D., WETZSTEIN, G., HIRSCH, M., HEIDRICH, W., AND RASKAR, R. 2011. Polarization fields: Dynamic light field display using multi-layer LCDs. *ACM TOG 30*, 6.
- LAZAREV, A., AND PALTO, S. 2009. Materials for light efficient LCD. *Society for Information Display (SID)*.
- LEVIN, A., FERGUS, R., DURAND, F., AND FREEMAN, B. 2007. Image and depth from a conventional camera with a coded aperture. *ACM TOG 26*, 3.
- LEVIN, A., SAND, P., CHO, T. S., DURAND, F., AND FREEMAN, W. T. 2008. Motion-invariant photography. *ACM TOG 27*, 3.
- LIANG, C., LIN, T. H., WONG, B. Y., LIU, C., AND CHEN, H. H. 2008. Programmable aperture photography: multiplexed light field acquisition. *ACM TOG 27*, 3.
- LIN, W. S., GAI, Y. L., AND KASSIM, A. A. 2006. Perceptual impact of edge sharpness in images. *IEEE Proceedings on Vision, Image and Signal Processing*.
- MAJUMDER, A., AND IRANI, S. 2007. Perception based contrast enhancement of images. *ACM Transactions on Applied Perception 4*, 3.
- MAJUMDER, A., AND STEVENS, R. 2005. Perceptual photometric seamlessness in tiled projection-based displays. *ACM TOG 24*.
- MAJUMDER, A., BROWN, R., AND GHOROURY, H. E. 2010. Display gamut reshaping for color emulation and balancing. *IEEE CVPR Workshop on Projector Camera Systems (PROCAMS)*.
- MAJUMDER, A. 2005. Is spatial super-resolution possible with multiple overlapping projectors? *ICASSP*.
- NAYAR, S. K., KRISHNAN, G., GROSSBERG, M. D., AND RASKAR, R. 2006. Fast separation of direct and global components of a scene using high frequency illumination. *ACM TOG 25*.
- RAN, X., AND FARVARDIN, N. 1995. A perceptually motivated three component image model: Part i. *IEEE TIP 4*, 4.
- RASKAR, R., AND COHEN, M. F. 1999. Image precision silhouette edges. *ACM 13D*.
- RASKAR, R., H TAN, K., FERIS, R., YU, J., AND TURK, M. 2004. Non-photorealistic camera: Depth edge detection and stylized rendering using multi-flash imaging. *ACM TOG 23*, 3.
- RASKAR, R., AGRAWAL, A., AND TUMBLIN, J. 2006. Coded exposure photography: Motion deblurring using fluttered shutter. *ACM TOG 25*, 3.
- STUPP, E. H., AND BRENNESHOLTZ, M. S. 1999. *Projection Displays*. Wiley.
- SUN, T., AND KELLY, K. 2009. Compressive sensing hyperspectral imager. *Computational Optical Sensing and Imaging, Optical Society of America*.
- VALOIS, R. L. D., AND VALOIS, K. K. D. 1990. *Spatial Vision*. Oxford University Press.
- VEERARAGHAVAN, A., RASKAR, R., AGRAWAL, A., MOHAN, A., AND TUMBLIN, J. 2007. Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. *ACM TOG 26*, 3.
- VEERARAGHAVAN, A., REDDY, D., AND RASKAR, R. 2010. Coded strobing photography: Compressive sensing of high-speed periodic events. *IEEE PAMI*.
- WAKIN, M., LASKA, J., DUARTE, M., BARON, D., SARVOTHAM, S., TAKHAR, D., KELLY, K., AND BARANIUK, R. 2006. An architecture for compressive imaging. *ICIP*.
- WETZSTEIN, G., LANMAN, D., HEIDRICH, W., AND RASKAR, R. 2011. Layered 3D: Tomographic image synthesis for attenuation-based light field and high dynamic range displays. *ACM TOG 30*, 4.
- WILBURN, B., JOSHI, N., VAISH, V., TALVALA, E. V., ANTUNEZ, E., BARTH, A., ADAMS, A., HOROWITZ, M., AND LEVOY, M. 2005. High performance imaging using large camera arrays. *ACM TOG 24*, 3.

WINKLER, S. 2001. Visual fidelity and perceived quality: towards comprehensive metrics. *Proceedings of SPIE 4299*.

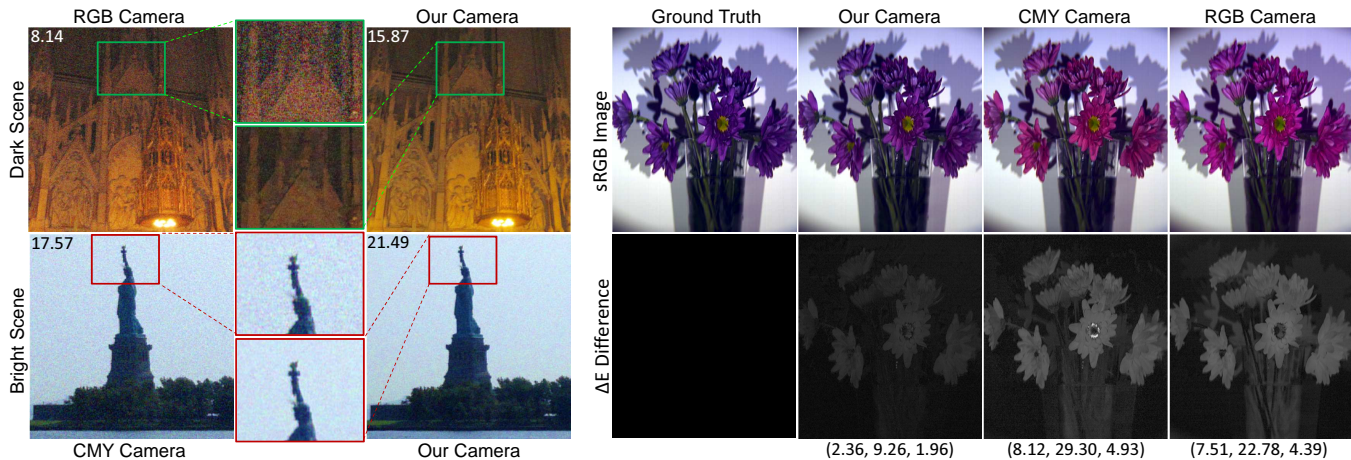
# Switchable Primaries Using Shiftable Layers of Color Filter Arrays

Behzad Sajadi\*  
Aditi Majumder†  
University of California, Irvine

Kazuhiro Hiwada‡  
Toshiba Corporation

Atsuto Maki§  
Toshiba Research Europe  
Cambridge Laboratory

Ramesh Raskar¶  
Camera Culture Group  
MIT Media Lab



**Figure 1:** Left: The CMY mode of our camera provides a superior SNR over a RGB camera when capturing a dark scene (top) and the RGB mode provides superior SNR over CMY camera when capturing a lighted scene. To demonstrate this, each image is marked with its quantitative SNR on the top left. Right: The RGBCY mode of our camera provides better color fidelity than a RGB or CMY camera for colorful scene (top). The  $\Delta E$  deviation in CIELAB space of each of these images from a ground truth (captured using SOC-730 hyperspectral camera) is encoded as grayscale images with error statistics (mean, maximum and standard deviation) provided at the bottom of each image. Note the close match between the image captured with our camera and the ground truth.

## Abstract

We present a camera with switchable primaries using shiftable layers of color filter arrays (CFAs). By layering a pair of CMY CFAs in this novel manner we can switch between multiple sets of color primaries (namely RGB, CMY and RGBCY) in the same camera. In contrast to fixed color primaries (e.g. RGB or CMY), which cannot provide optimal image quality for all scene conditions, our camera with switchable primaries provides optimal *color fidelity* and *signal to noise ratio* for multiple scene conditions.

Next, we show that the same concept can be used to layer two RGB CFAs to design a camera with switchable low dynamic range (LDR) and high dynamic range (HDR) modes. Further, we show that such layering can be generalized as a constrained satisfaction problem (CSP) allowing to constrain a large number of parameters (e.g. different operational modes, amount and direction of the shifts, place-

ment of the primaries in the CFA) to provide an optimal solution.

We investigate practical design options for shiftable layering of the CFAs. We demonstrate these by building prototype cameras for both switchable primaries and switchable LDR/HDR modes.

To the best of our knowledge, we present, for the first time, the concept of shiftable layers of CFAs that provides a new degree of freedom in photography where multiple operational modes are available to the user in a single camera for optimizing the picture quality based on the nature of the scene geometry, color and illumination.

**Keywords:** computational photography, color filters, capture noise

## 1 Introduction

Camera consumers are forced to live with several trade-offs originating from conflicting demands on the quality. For example, broad-band filters (e.g. CMY), being more light efficient than narrow-band filters (e.g. RGB), are desired for low-illumination scenes (e.g. night/dark scenes). But, they have lower color fidelity. Further, demultiplexing RGB values from the captured CMY values can result in more noise in brighter scenes. Hence, narrow-band filters are desired for high-illumination scenes (e.g. daylight/bright scenes). However, since current cameras come with fixed RGB or CMY CFAs, users have to accept sub-optimal image quality either for dark or bright scenes. Similarly, faithful capture of colorful scenes demand more than three primaries that trades off the spatial resolution making it not suitable for architectural scenes with detailed patterns and facades. However, since current cameras come with a fixed number of primaries, users cannot change the spatial and spectral resolution as demanded by the scene conditions.

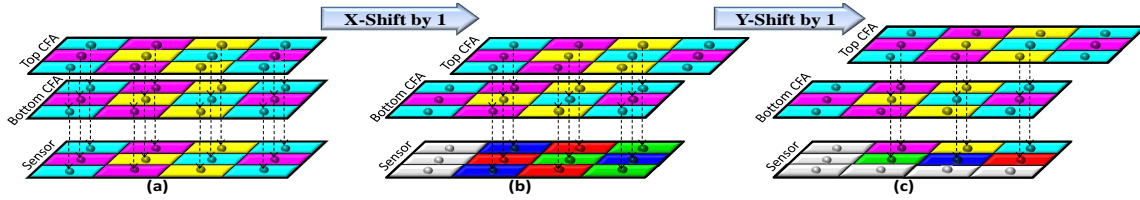
\*e-mail: bsajadi@uci.edu

†e-mail: majumder@ics.uci.edu

‡e-mail: kazuhiro.hiwada@toshiba.co.jp

§e-mail: atsuto.maki@crl.toshiba.co.uk

¶e-mail: raskar@media.mit.edu

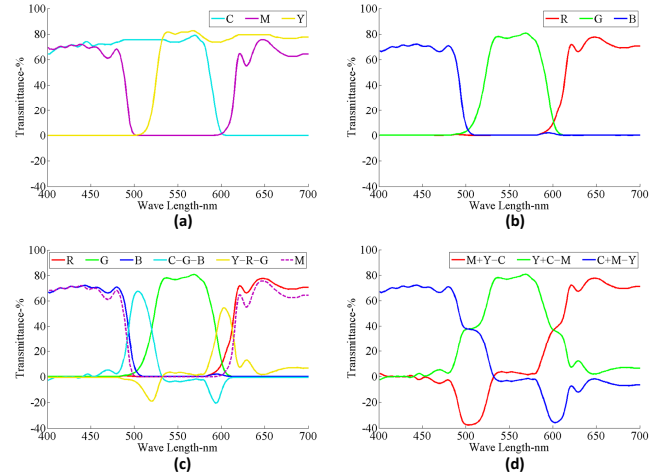


**Figure 2:** Two CMY CFAs before shifting(a), after shifting the top layer one tile to the right(b), and after shifting the top layer by another tile in the vertical direction. The combinations of the layers, shown in the bottom, result in CMY(a), RGB(b), and RGBCMY(c) modes.

**Main Contributions:** We present a technique of layering of a pair of CFAs with precise relative shifts between them to achieve cameras with multiple operational modes where both the number and transmittance of the primaries can be changed. The user will thus have the liberty to cater the primaries towards specific scene conditions. Following are our main contributions.

1. We present the first camera that can switch to three sets of color primaries on demand. Using different relative shifts during the layering of the pair of CFAs, both the number and transmittance of the primaries can be changed (Figure 2) to provide a camera with three different capture modes: RGB, CMY and RGBCY (Section 2).
2. We extend the concept of shiftable layers of CFAs beyond switchable primaries showing that when applied to a different kind of CFA, it provides a camera that can switch between low dynamic range (LDR) and high dynamic range (HDR) modes (Section 3).
3. We show that the problem of finding the desired patterns and shifts of the CFAs to achieve switchable modes can be posed as a constraint satisfaction problem (CSP)(Section 4). We show the utility of this general framework to design an add-on device for existing LDR cameras that provides an additional HDR mode.
4. We present a quantitative analysis to show the benefits of a camera with switchable primaries: (a) significantly superior color fidelity than RGB or CMY cameras when operated in the RGBCY mode (Section 5.1); (b) optimal SNR, for both dark and bright scenes by switching between the CMY and RGB modes (Section 5.2). Though using two CFAs marginally trades off the transmittance of each primary, the benefits far outweigh this shortcoming.
5. Finally, we propose several practical design options to embed such shiftable layers of CFAs in real cameras for multiple switchable operational modes (Section 6). We demonstrate the feasibility of such designs via rudimentary prototypes.

**Related Work:** Many different types of fixed CFAs have been invented and manufactured for photography [Lukac 2008], the most popular being the Bayer CFA [Bayer]. [Yamagami et al.; Gindale and Gallagher; Susanu et al. 2009; Kumar et al. 2009] use RGBW CFAs with white filter elements to sense more light than cameras with traditional Bayer CFAs. [Hirakawa and Wolfe 2008] considered the more general case of a custom designed CFA based on linear combinations of conventional RGB filters to achieve optimal spatial-spectral reconstruction using a sensor with a fixed number of pixels. [Gu et al. 2010] introduced a universal demosaicing framework that can be used to reconstruct the image for any general CFA. Fixed CFAs with more than three colors have been proposed to capture multispectral images [Shogenji et al. 2004; Baone and Qi 2006] sacrificing the spatial resolution for higher spectral resolution. These provide much higher color fidelity, but are still less accurate than an order of magnitude more expensive hyper-spectral cameras. In another line of work multi-spectral images with a low spatial resolution are combined with high resolution lightness images to achieve a high-resolution multi-spectral imaging system [Imai et al. 2000]. This is done using a priori spectral analysis, linear modeling techniques, and using the spatial properties of the human visual system. In contrast to all these works on fixed color primaries, our work is the first one that presents switchable color primaries by shiftable layers of CFAs.



**Figure 3:** Spectral transmittance of our primaries in (a)CMY mode, (b) RGB mode, and (c) RGBCY mode. In (c), the narrow band cyan and yellow are computed from the broad band CMY filters in (a) and the narrow band RGB filters in (b). (d) Spectral transmittance of the RGB channels demultiplexed from the CMY mode.

On the other hand, our work supplements an earlier set of work on computational color in photography. Dynamic modification of spectral transmittance has been proposed in agile-spectrum imaging [Mohan et al. 2008] by using of diffraction grating. In a completely orthogonal domain, limited flexibility in color primaries has been explored via tunable sensors [Langfelder et al. 2009]. These sensors do not require CFAs to capture color images. Instead, each wavelength is captured at a different depth of the sensor. The absorption depth can be changed by applying an electrical voltage to the sensor. Therefore, the spectral-bands that are sensed at each depth can be tuned slightly. This allows for limited flexibility in the amount of overlap between the spectral response of the eye (CIE primaries for the standard observer) and that of the sensors, leading to a little higher color fidelity. However, this only allows a small shift in the spectral transmittance of the narrow band primaries, but cannot achieve a completely different number of primaries with entirely different spectral transmissivity as is possible in our camera.

## 2 Camera with Switchable Primaries

We achieve switchable color primaries by layering a pair of CFAs that can be shifted precisely relative to each other. We use a pair of CMY CFAs (Figure 2(a)) where each row repeats the C, M, and Y tiles. But odd rows start with C while even rows with M. This results in the repetition of a  $3 \times 2$  pattern of CMY tiles (Figure 2(a)).

When two such CMY CFAs are superimposed with no shift, tiles with similar spectral transmittance coincide and the combined effect is that of a CMY CFA, whose spectral transmittance is shown in 3(a). However, if the top layer is shifted by one tile horizontally, each C tile of the top layer superimposes a M tile of the bottom layer resulting in a B tile. Similarly, M and Y tiles of the top layer

superimpose Y and C tiles of the bottom layer resulting in G and R tiles respectively (Figure 3(b)). Therefore, with such a horizontal shift, this layered CFA is similar to an RGB CFA except for the first and last columns (Figure 2(b)). Finally, if the top layer is shifted by another tile vertically, in the odd rows the C tiles superimpose Y tiles, M with C, and Y with M, resulting in RGB tiles as before. But, in the even rows the M tiles from the top layer superimpose with M tiles from the bottom layer, Y with Y and C with C resulting in broad-band CMY tiles (Figure 2(c)). Using these, we can compute narrow-band cyan and yellow primaries,  $C_n = C - B - G$  and  $Y_n = Y - R - G$  (Figure 3(c)). But, since  $M$  is very close to  $R + B$ , we cannot similarly extract a sixth non-overlapping primary. This results in a capture mode with five almost non-overlapping primaries, namely  $R, G, B, C_n$  and  $Y_n$ , leading to a five primary mode – RGBCY. Thus, we achieve three different sets of color primaries in the same camera: (a) RGB, (b) CMY, and (c) RGBCY.

Our camera with switchable color primaries has several advantages over cameras with fixed RGB or CMY CFAs. Narrow-band fixed RGB CFAs mimic the human eye but do not have the desired light efficiency to provide a good signal-to-noise-ratio (SNR) for dark scenes. Wide band CMY CFAs (Figure 3(a)), on the other hand, provide better SNR for dark scenes. However, images need to be converted to the more common RGB format using demultiplexing computations of  $R = M + Y - C$ ,  $G = Y + C - M$ , and  $B = C + M - Y$ . These computations introduce higher noise for bright scenes. Further, the effective spectral transmittance profiles of the  $R, G, B$  channels following this computation (Figure 3(d)) can be negative leading to lower color fidelity due to clamping artifacts [Cao and Kot 2008]. Thus, while CMY CFAs are better for dark scenes, RGB CFAs are preferred for bright scenes. In summary, our camera can provide optimal SNR by capturing dark scenes in the CMY and bright scenes in the RGB mode; and can also provide significantly higher color fidelity for colorful scenes in the RGBCY mode.

We have demonstrated and evaluated the superior color fidelity and SNR of our camera using empirical results (Section 5) obtained from multiple prototypes designed and built in our lab (Section 6).

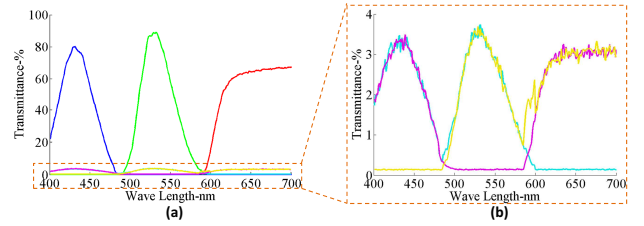
### 3 Camera with Switchable Dynamic Range

The concept of shiftable CFAs can be used to create different operational modes, beyond just switchable primaries. When creating switchable primaries, we considered layers of CMY CFAs. Now, let us consider RGB filters that have a small transmittance over the entire spectrum (Figure 4a) except for peaks in the R, G, and B regions respectively. In this case, superimposition of unlike filters – i.e. B and G, R and B, or R and G – result in very low transmittance cyan, magenta and yellow filters,  $C_h, M_h$  and  $Y_h$ , respectively.

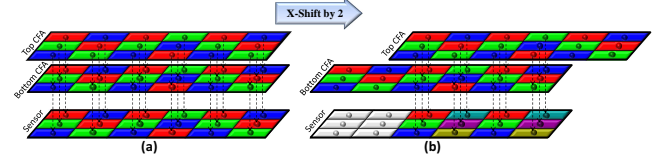
Let us now consider two layers of RGB CFAs (Figure 5). Before shifting, similar tiles superimpose (Figure 5a) resulting in a low dynamic range (LDR) capture mode. But, with a relative horizontal shift of 2 tiles (Figure 5b) we get a column of RGB filters and another column of CMY filters with very low transmittance that are sensitive to a higher range of brightness. Hence, in this mode, we can capture high dynamic range (HDR) image while trading off the spatial resolution. Thus, we now get a camera which can switch between LDR and HDR capture modes. We describe prototypes for such a camera and results thereof in Section 6 and 5.

### 4 A General Framework

In general, we can pose the problem of designing appropriate CFA patterns and their relative shifts as a *constraint satisfaction problem* (CSP). We impose constraints on the combinations of the primaries



**Figure 4:** (a) Spectral transmittance of the  $R, G, B, C_h, M_h, Y_h$  channels. (b) Zoomed-in view of the spectral transmittance of the  $C_h, M_h,$  and  $Y_h$  channels. The zoomed-in view shows that the RGB channels extracted from  $C_h, M_h,$  and  $Y_h$  are similar to the LDR RGB channels but are considerably less sensitive to light.



**Figure 5:** Left: Two Layers of RGB CFA superimposed on each other. Right: The top layer is shifted 2 tiles to the right. After the shift the tiles that overlap with similar tiles work as RGB filters and the rest work as low transmittance CMY filters.

and their proportions in each capture mode which are then solved by a CSP solver to return the patterns for both the CFAs.

Let us assume  $p$  different tiles/filters,  $F_k, 1 \leq k \leq p$ . For example, in the context of Figure 2, there are 6 different tiles,  $(C, M, Y, R, G, B)$ . First, we define the set of valid combinations of the tiles that can be used in the design. This is a set,  $V$ , of 3-tuples that define the tile in the top layer, bottom layer, and their combination. For figure 2,  $V = \{(M, Y, R), (Y, C, G), (C, M, B), (C, C, C), (M, M, M), (Y, Y, Y)\}$ . In all the examples in this paper, switching the first two elements of the 3-tuple also result in valid combinations, but we omit those 3-tuples for compact representation. Next, for each capture mode, we define the desired proportion of each primary in the final combination. We assume  $m$  capture modes. For each mode  $l, 1 \leq l \leq m$ , we define as a  $p$ -tuple,  $M_l$ , which specifies the proportions of tile  $F_k$  in mode  $l$ . For Figure 2,  $M_1 = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0, 0)$ , and defines the CMY mode;  $M_2 = (0, 0, 0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$  and defines the RGB mode; and finally  $M_3 = (\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6})$  and defines the RGBCMY mode.

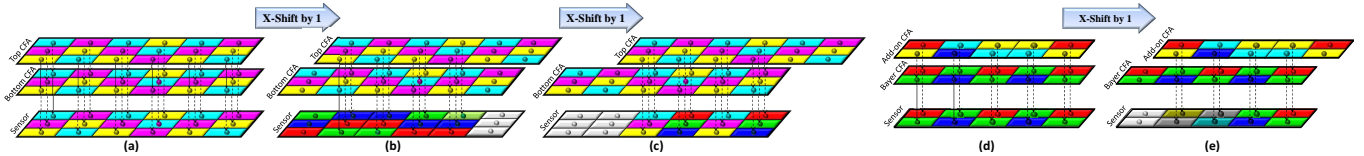
In order to find the CFA patterns, the CSP solver starts the search from the smallest possible number of tiles that can fit all the desired proportions defined by  $M_l$ s. Among different sizes with the same number of tiles, it starts the search from the one which is closest to a square in shape. Let us assume the size of the pattern is  $(n_x, n_y)$ . Let us define the tiles of the top and bottom layers as  $T(i, j)$ , and  $B(i, j)$  respectively, where  $0 \leq i < n_x, 0 \leq j < n_y$ . The combination of the layers, however, depends on the additional parameters of the direction and magnitude of the relative shift between the two layers. Therefore the solver also iterates on the possible shifts starting from the smallest one. Let us assume for mode  $l$  the shift is defined by  $(x_l, y_l)$  and the superposition of the two layers as  $S_l(i, j)$ . Consequently, we enforce the following combination constraints:

$$(T((i+x_l) \bmod n_x, (j+y_l) \bmod n_y), B(i, j), S_l(i, j)) \in V \quad (1)$$

Further, we also impose proportion constraints for each filter  $F_k$  assuring that its total number in the combined layer for mode  $l$  confirms to  $M_l$ . This constraint is as follows.

$$\sum_{ij} (S_l(i, j) == F_k) = M_l(k)n_x n_y \quad (2)$$

Since each of the above constraints only affects a few variables, they can be efficiently solved by standard CSP solvers. Further, we can impose constraints on the amount and direction of the shift. For



**Figure 6:** Results from CSP solver. (a),(b),(c): Layering of two CMY layers to create a camera with switchable primaries with the shift constrained to be in one direction – CMY before shifting (a), RGB after shifting the top layers one tile to right (b), and RGBCMY mode after shifting 2 tiles to right. (d),(e): Layering of an add-on CFA by constraining one layer to be a Bayer CFA to create a camera with switchable LDR/HDR modes – the add-on pattern does not considerably affect the transmittance when superimposed with a Bayer CFA without shifting giving the LDR mode (d), when shifted to the right, some of the tiles are similar to RGB filters and the rest become low transmittance ICY filters that capture HDR values (e). Note that unlike other CFAs in the paper, this has C, Y, R and B filters – not just CMY or RGB.

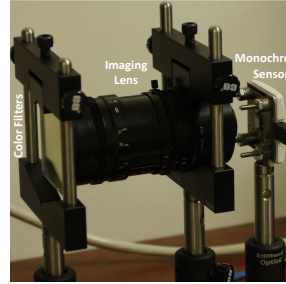
example, for a switchable CMY/RGB/RGBCMY camera, if we impose an additional constraint to limit the shift only in the horizontal direction, the CSP solver fails to find a pattern with only 6 tiles. However, after increasing the size of the pattern, it finds the  $4 \times 3$  pattern in Figure 6 where the CMY, RGB and RGBCY modes are achieved by 0, 1 and 2 tiles shift respectively.

Further, we can impose constraints on one of the layers to have a specific pattern. For example, if we desire to build a switchable LDR/HDR camera using a commodity camera with an existing Bayer CFA, we can specify  $B(i, j)$  to form a Bayer pattern and let the solver find  $T(i, j)$ . In this case we have 6 tiles  $(R, G, B, C_h, M_h, Y_h)$  and the valid combinations are  $V = \{(R, R, R), (G, G, G), (B, B, B), (G, B, C_h), (B, R, M_h), (R, G, Y_d)\}$ . There are two capture modes. In the LDR mode, the Bayer pattern dictates  $M_1 = (\frac{1}{4}, \frac{1}{2}, \frac{1}{4}, 0, 0, 0)$ . However, note that it is difficult to define specific proportions for the low transmittance tiles of  $C_h, M_h$  and  $Y_h$  since multiple combinations may all produce acceptable results. But we can define a range of proportions instead of a specific one. For example, we can define  $M_2 = (\frac{1}{8}, \frac{1}{4}, \frac{1}{8}, [\frac{1}{8}, \frac{1}{4}], [\frac{1}{8}, \frac{1}{4}], [\frac{1}{8}, \frac{1}{4}])$ . Finally, one can impose constraints on the patterns to enforce certain desired properties such as non-adjacency of similar filters, or equal number of other filters in the neighborhood of each filter.

However, note that a CSP solver may not always return a solution. For example, this is the case for the above set of constraints defined for the switchable LDR/HDR camera. One way to alleviate the situation is to provide more sets of valid combinations. For example, we can add constraints to denote that  $R, G$  and  $B$  can be generated differently than just superimposing two layers of  $R, G$  and  $B$ . This can be achieved by adding  $\{(Y, R, R), (C, G, G), (M, B, B)\}$  to  $V$ . Further, we can also experiment with different filters. For example, instead of having  $C_h, M_h$  and  $Y_h$  as the low transmittance filters, we can have an equivalent set of  $C_h, I_h$ , and  $Y_h$  where  $I_h$  is an intensity filter and replaces  $M_h$ . Thus, in this case, we have a set of six different filters  $(R, G, B, C_h, I_h, Y_h)$  where the valid superpositions for achieving  $I_h$  are given by  $\{(C, R, I_h), (M, G, I_h), (Y, B, I_h)\}$ . By doing these changes, the CSP solver can provide a solution for an add-on CFA to the Bayer CFA to achieve switchable LDR/HDR modes as shown in Figure 6. Note that the top layer consists of C, Y, R and B tiles, instead of having just CMY or RGB tiles. We build a sample prototype for this, as explained in Section 6. However, note that in the LDR mode, R can be formed both by superimposing two R tiles or a R and a Y. Similarly, G and B can also be generated in two ways resulting in varying spectral transmittance of the same primary in this mode. However, we find in our prototype that this still produces acceptable results (Figure 14).

Another way to assure a solution from the CSP solver is to weigh some constraints to be more important than the others. For example uni-directional shift can be an important design constraint, while non-adjacency of similar filter may not be as critical. Allowing such weights in the CSP solver results in a Markov Random Field that can be solved efficiently using AI techniques for bounded search.

## 5 Results



**Figure 7:** Our first prototype where color filters are temporally multiplexed.

For the proof of concept of our camera with switchable operational modes, we used a time sequential capture of images using different layers of color filters in front of a monochrome camera to simulate the shiftable layers of CFAs (Figure 7). To demonstrate switchable primaries, we captured the images by superimposing pairs of CMY filters, both like (C and C, M and M, and Y and Y) and unlike (C and M, M and Y, and C and Y). To demonstrate switchable LDR/HDR modes, we captured images by superimposing RGB filters. Next, to simulate the effect of capturing all these in a single shot, we pick the pixels from the appropriate images in this temporally multiplexed sequence. The image thus created, records only one primary at every pixel simulating the effect of the layered CFAs. We demosaic the image in software (Section 7) to achieve the final full-resolution image. The setup of Figure 7 provides us with high quality and high resolution results to prove the concept of shiftable layers of CFAs. However, practical designs for such a camera without time multiplexing are described in Section 6.

For the setup in Figure 7, we used a monochrome  $2560 \times 1920$  sensor (EO-5012BL<sup>1</sup>) and dichroic filters from *EdmundOptics*<sup>2</sup>. The spectral transmittance of the filters (Figure 3) are obtained from the manufacturers website<sup>2</sup>. For the LDR/HDR camera, we create RGB filters by exposing 35mm Kodak films to appropriate lighting. To allow some amount of light (at least 4%) to pass through in the HDR mode after the superposition of the shifted layers, we did not fully expose the films. Figure 4 shows the transmittance profiles of these filters captured using a SOC-730 hyperspectral camera.

Figure 8 shows the results for the switchable LDR/HDR modes. We use an adaptive logarithmic tone mapping operator [Drago et al. 2003] to show the HDR image. Figures 9, 10 and 11 show the results for the camera with switchable primaries. In the rest of the section, we quantify the advantages of our switchable cameras.

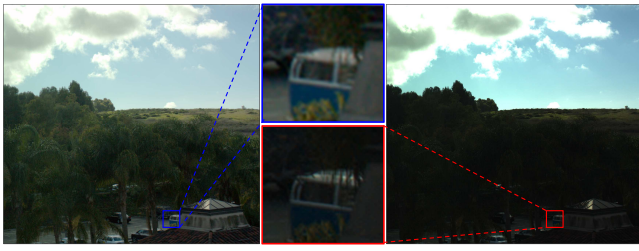
### 5.1 Superior Color Fidelity

First, we show the superior color fidelity of our camera with switchable primaries in the RGBCY mode compared to the RGB or CMY modes. We compared the images captured by each mode of our prototype camera against those captured by a SOC-730 hyperspectral

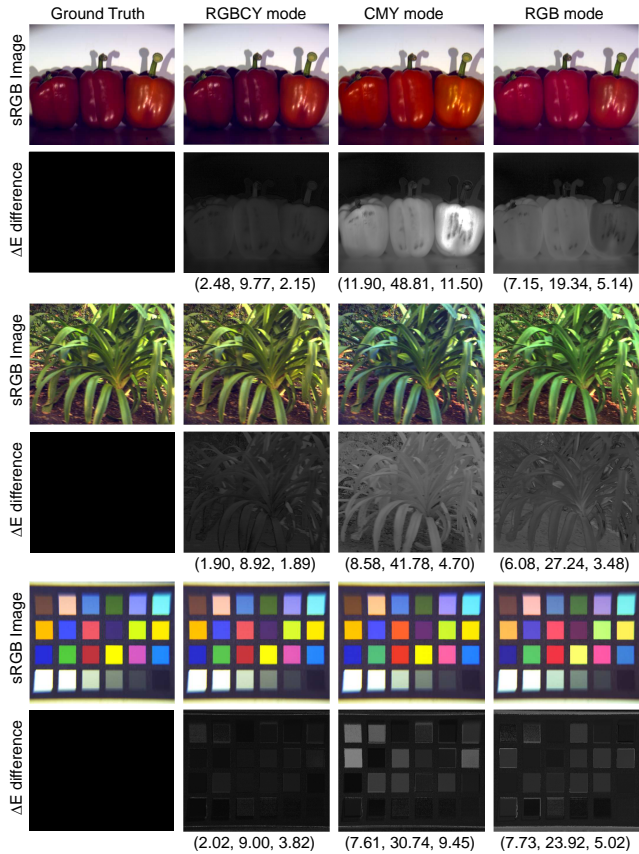
<sup>1</sup><http://www.edmundoptics.com/onlinecatalog/displayproduct.cfm?productID=1734>

<sup>2</sup><http://www.edmundoptics.com/onlinecatalog/displayproduct.cfm?productID=2947>





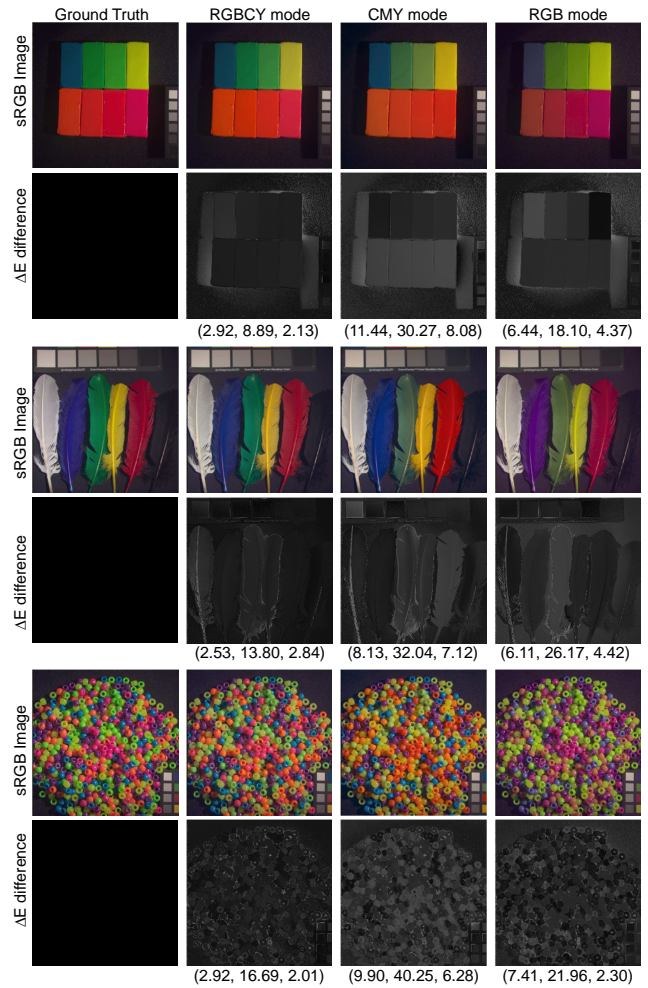
**Figure 8:** Left: A scene captured with the HDR mode of our switchable LDR/HDR camera. Right: The same scene captured with the LDR mode (saturated sky and dark trees). In the zoomed-in view the resolution of the LDR image is higher than the HDR one, emphasizing the need for flexibility based on the scene and application.



**Figure 9:** Three examples of comparison between the ground truth images captured with a SOC-730 hyperspectral camera and images captured with our prototype with RGB, CMY, and RGBCY modes. The gray images show the CIELAB  $\Delta E$  difference along with the error statistics (mean, maximum and standard deviation). Note the better color fidelity of the RGBCY mode especially in the red-purple and cyan-green colors. Also, note that in general the color fidelity of CMY mode is much lower than the RGB mode.

camera at a spatial resolution of  $1024 \times 1024$  and spectral resolution of around  $9nm$  in the range of visible wavelengths from  $420nm$  to  $700nm$ . We used a data set of 35 such images.

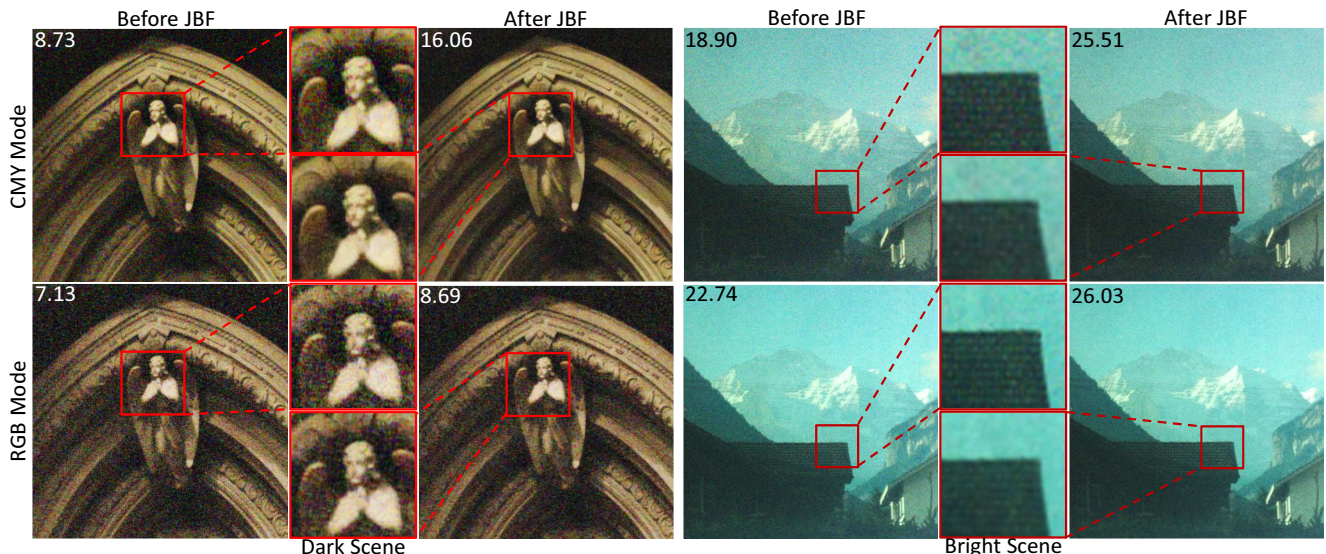
For comparison, we generate four images in the CIE XYZ space. First, we compute a ground truth image from the captured hyperspectral image by finding the CIE XYZ values at each pixel via a scalar dot product of the spectral response at that pixel,  $P(\lambda)$ , with the standard human observer's sensitivity,  $x(\lambda)$ ,  $y(\lambda)$  and  $z(\lambda)$  re-



**Figure 10:** Three examples of comparison between the ground truth and simulated images for RGB, CMY, and RGBCY modes using the CAVE database. The gray images show the CIELAB  $\Delta E$  difference along with the error statistics (mean, maximum and standard deviation). Note the superior color fidelity of the RGBCY mode especially in near-saturated shades of blue, green, and red.

spectively. Next, we convert the images captured by the three different modes of our prototype to CIE XYZ space. The XYZ values corresponding to the captured color are computed by a weighted sum of the captured values, where the weights for  $X$ ,  $Y$  and  $Z$  are computed by finding the correlation of the known spectral transmittance profiles of the primaries (Figure 3) with  $x(\lambda)$ ,  $y(\lambda)$  and  $z(\lambda)$ . To quantify the perceptual difference of each of these camera captured images from the ground truth, we compute their  $\Delta E$  differences in the CIELAB space. Further, to provide a feel of how these images would look on a standard sRGB display, we convert them to the sRGB space. Since the  $\Delta E$  images do not involve errors due to clamping, they are better indicators of the differences. To align the images captured by our camera and those from the hyperspectral camera, we use standard rectification techniques.

Figure 9 shows a few examples from this set of 35 images along with the statistics (mean, maximum, and standard deviation from the mean) of the per-pixel  $\Delta E$  error for each of these images. The average  $\Delta E$  difference, over all the 35 images, for RGBCY mode was 1.95 units and 6.2 and 7.5 units for the RGB and CMY modes respectively. This is a perceptible difference of more than 1JND (3 units of  $\Delta E = 1$  JND). Further, note that the RGBCY mode reduces



**Figure 11:** Scenes captured with CMY (top) and RGB (bottom) modes of our camera is shown before (left) and after (right) applying the JBF for a dark and a bright scene. The SNR are embedded on the top left of each image. For the dark scene, the CMY mode provides superior SNR, particularly after applying the JBF. For the bright scene, the RGB mode provides superior SNR. The JBF reduces the noise but degrades the overall perceptual quality of the image because of the reduced sharpness of the edges.

the maximum deviation from the ground truth tremendously, when compared to the RGB and CMY modes – but some deviation still remains since five primaries are not sufficient to achieve the color fidelity of a hyperspectral camera with 30 spectral bands.

In order to confirm the same result for an existing database, we use the CAVE multi-spectral image database [Yasuma et al. ] that includes 31 pictures sampling the range of the visible wavelengths from 400nm to 700nm at 10nm increments at each pixel. We simulate the images captured by the camera in different modes using the spectral transmittance profile of the primaries (Figure 3) of that mode. Then, we compute the same  $\Delta E$  difference as mentioned above for the simulated camera images in different modes.

Figure 10 shows a few examples along with the error statistics of the  $\Delta E$  difference. The results are similar to the first set of experiments with an average  $\Delta E$  difference of 2.12, 6.5 and 7.6 units for the RGBCY, RGB and CMY modes respectively confirming a significantly improved color fidelity in the RGBCY mode.

## 5.2 Optimal Signal to Noise Ratio

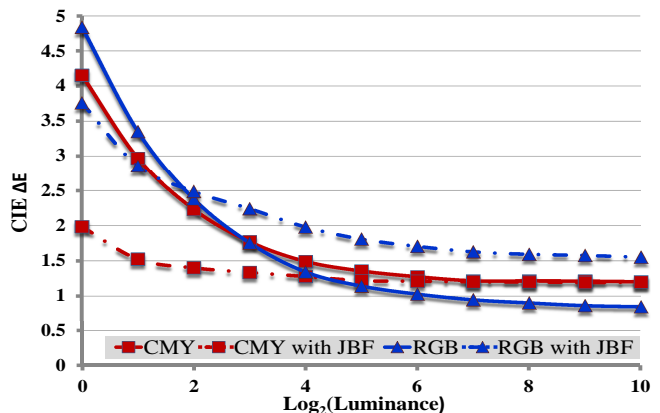
The signal-to-noise-ratio (SNR) of an image is strongly related to the spectral properties of the color filters and the overall brightness of the scene. CMY CFAs are known to have higher SNR compared to RGB CFAs in dark scenes due to their higher spectral transmittance; but result in lower SNR for brighter scenes since the noise adds up when demultiplexing the RGB values from the captured CMY values. Our camera offers the best of both worlds by switching between RGB and CMY modes.

To demonstrate this, we present in the appendix a computational method to analyze the SNR of our camera. We compute two ratios,  $\frac{SNR_{CMY}}{SNR_{RGB}}$  and  $\frac{SNR_{CMY}}{SNR_{RGBCY}}$  (Equation 3), for both bright and dark scenes (Table 1) for captured color vectors  $C$  and for the intensity value  $g$ , obtained by summing the captured values across the channels.

To validate the model in practice, we measure the same ratios for a set of images captured by our prototype (Section 6) and compare them with those predicted using our SNR model in Equation 3. We use images of 20 different scenes for each of the dark and bright conditions, and we capture each scene 25 times under the same

	$\frac{SNR_{CMY}(C)}{SNR_{RGB}(C)}$		$\frac{SNR_{CMY}(C)}{SNR_{RGBCY}(C)}$		$\frac{SNR_{CMY}(g)}{SNR_{RGB}(g)}$		$\frac{SNR_{CMY}(g)}{SNR_{RGBCY}(g)}$	
	M	P	M	P	M	P	M	P
Dark	1.25	1.22	1.94	1.99	2.08	2.12	0.94	0.96
Bright	0.85	0.84	1.60	1.57	1.39	1.37	0.82	0.82

**Table 1:** Comparison of SNR ratios for  $C$  and  $g$  across CMY, RGB, and RGBCY capture modes.  $M$  denotes measured and  $P$  denotes predicted. Note that for all conditions, the measured ratios conform closely to the predicted ones validating our SNR model.



**Figure 12:**  $\Delta E$  difference from ground truth for RGB and CMY modes before and after the JBF. Longer exposure time is used for lower luminance levels such that the amount of light that reaches the sensor remains constant. The longest exposure is 2 seconds and the shortest is 1 millisecond. We used ISO 400. The graph demonstrates that for low luminance levels CMY capture mode is superior particularly after applying the JBF. On the other hand, for high luminance levels RGB capture mode without JBF results in superior fidelity while the JBF degrades the quality of the image.

illumination. To achieve such a controlled illumination, for this experiment, we use projector based illumination over a printed scene to vary the scene conditions from dark to bright. We vary the exposure time inversely proportional to the illumination intensity.

To find the SNR for each scene, we first find the mean and variance

estimators of the captured  $C$  and  $g$  at each pixel using the 25 images captured under the same illumination. From this, we can compute the per pixel noise-to-signal ratio which are then averaged across the pixels and inverted to find the average SNR.

Table 1 shows the predicted and measured SNR ratios for both  $C$  and  $g$ . The closeness of the predicted and measured values in this table validates the accuracy of our noise model and shows that for dark scenes the SNR is more than 20% higher in the CMY mode than the RGB mode. But, for bright scenes, the RGB capture mode shows similar SNR advantage over the CMY mode. Also, when compared to the RGBCY mode, the CMY mode has almost double the SNR for dark scenes. This is due to the very narrow band  $C_n$  and  $Y_n$  primaries in the RGBCY mode. Thus, the greater color fidelity of the 5 color mode comes at the cost of reduced SNR.

**Joint Bilateral Filtering:** Table 1 shows that the SNR for  $g$  is much superior than the SNR for  $C$ , especially in the CMY mode (almost twice), for both dark and bright scenes. Hence, we propose using the intensity image  $g$ , as a guidance image to apply joint bilateral filtering (JBF) on each channel of the image to improve the SNR. However, JBF can also degrade the image fidelity by blurring the high-frequency details. Hence, there is a trade off involved in the improvement in the SNR and the degradation in image fidelity.

To evaluate this, we find the SNR of a scene after applying JBF for a particular mode using the aforementioned SNR analysis using the same set of 20 scenes after applying JBF. We found that for *dark scenes*, JBF improves the SNR of the CMY mode dramatically but does not affect the SNR of the RGB mode as much. Hence, after JBF, the CMY mode provides almost 70% better SNR than RGB mode (as opposed to 20% improvement without JBF).

For *bright scenes* also, JBF improves the SNR. But this comes at the cost of degraded image fidelity. We measure this degradation using  $\Delta E$  difference of the captured image, before and after applying the JBF, from a ground truth image. To find the ground truth for each scene, we average the 25 images captured under the same illumination. Finally, we average the  $\Delta E$  difference over all the pixels for each mode. From this metric, we find that the degradation in the image fidelity due to the JBF, offsets the improvement in the SNR in RGB mode much more than the CMY mode (Figure 12). Hence, for bright scenes, the highest image fidelity is achieved in the RGB mode without applying the JBF.

## 6 Design Options and Prototypes

In this section, we provide design options for embedding shiftable layers of CFAs in a real camera. We build some prototypes based on these designs and show some preliminary results from them.

### 6.1 Mechanical Shift

The easiest way to achieve shiftable layers of CFA is to layer two CFAs on the CCD sensor during manufacturing. However, one of them should be equipped with a shift mechanism. This can be achieved using inexpensive (less than \$175) linear staging devices (e.g. EdmundOptics Part Number NT56-416<sup>3</sup>) some of which allow linear shifts with 1  $\mu m$  accuracy.

To demonstrate the feasibility of this design, we used it to build a rudimentary prototype of our camera with switchable primaries. We opened up a monochrome 2560  $\times$  1920 camera (*EO-5012BL* from EdmundOptics) to expose its sensor. We used printed 35mm

<sup>3</sup><http://www.edmundoptics.com/onlinecatalog/displayproduct.cfm?productID=1844>

digital slides for the CFAs. Such slides can be printed in professional photo labs such as Swan Photo Lab<sup>4</sup> and cost about \$4 for each slide. To implement the shifting, we used a Metric Bar-Type Lens Holder<sup>5</sup> (price: \$79). One of the CFA layers is mounted on the static part of the holder and the other one on the moving part (Figure 13). The screw on the moving part has 20 teeth each of height 0.5mm. Therefore, one turn of this screw results in 0.5mm shift of the moving CFA. Hence, by rotating the head of the screw by one degree we can move the CFA about 1.39 $\mu m$ .

However, this setup has a tremendous scope of improvement. Our cheap CFAs has neither the resolution nor the light efficacy of the CFAs of standard cameras. The pixel size of our printed CFAs is 8.8 $\mu m \times 8.8\mu m$  resulting in 4 times bigger pixel size in each dimension than our sensor pixels (2.2 $\mu m \times 2.2\mu m$ ). Further, the CFAs are printed using light beams that do not produce rectangular pixels but gaussian blurs. Therefore, we printed a pattern with 2 times larger tiles and one black line between every two adjacent tiles to reduce the color bleeding. Consequently, a CFA tile becomes 12 times bigger compared to a sensor pixel. To alleviate this mismatch, we separate the CFAs from the sensor. The image is focussed on the CFAs and refocused on the sensor using an achromatic lens (25mm diameter and 30mm effective focal length) that downsizes the CFA tiles by a factor of 3 making the resolution mismatch 4 in each direction. Even when considering the 4  $\times$  4 pixels on the sensor that are considered as one pixel of the prototype, we observe considerable color bleeding between the adjacent pixels. This is due to the glass cover of the sensor that acts as a diffuser. We could not remove it due to the fragility of the sensor. Hence, to nullify its effect we only consider the 2  $\times$  2 center pixels of the 4  $\times$  4 groups of pixels on the sensor and average their values to get the captured values. All these result in degradation of the image quality and resolution (640  $\times$  480 pixels). Figure 13 shows the picture of this prototype and some images captured with it. Further, in terms of size, note that 16cm length of our 19cm long prototype contributes to refocus the image from the CFA to the sensor that is unnecessary when the CFAs are mounted on the sensor. Finally, in terms of cost, the off-the-shelf devices used in our setup are not custom tailored for our application (for e.g. the Metric Bar-Type Holder can hold much heavier weight than is required by a camera). Devices designed specifically and mass produced for cameras can be considerably cheaper.

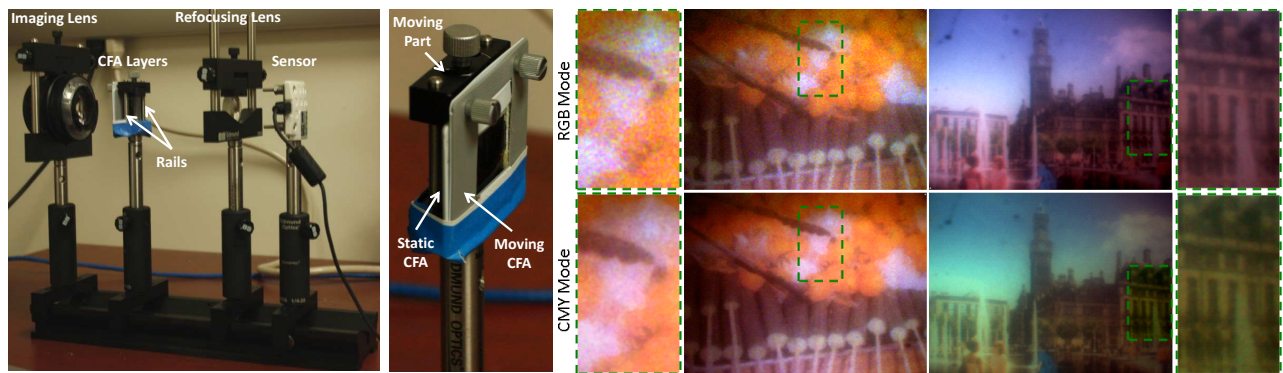
### 6.2 Optical Shift

We also designed an add-on device for DSLR cameras to achieve switchable modes. In this setup, the image is formed on the first CFA and then refocused on the second CFA, attached to the sensor, using two lenses of the same power. However, by making one lens slightly off-axis we can shift the image in the off-axis direction. A precise shift can be obtained by controlling the placement of the lenses between the CFA and the sensor (Figure 14).

Let us assume the first lens is  $\alpha$  units off-axis and the desired shift is  $\beta$  units. The magnification of the two lenses are  $s_1$  and  $s_2$  respectively, where  $s_1 s_2 = 1$ . Assuming the second lens is axis aligned, the total shifting of the image is  $s_2 \alpha = \frac{\alpha}{s_1}$ . Hence,  $s_1 = \frac{\alpha}{\beta}$ . Using the standard thin lens equation, we find that in order to achieve this the first lens should be placed at distance  $d_1 = \frac{f(s_1+1)}{s_1}$  from the CFA. The resulting image will be at distance  $\frac{f(s_1+1)^2}{s_1}$  from the CFA. In order to make  $s_2 = \frac{1}{s_1}$ , using thin lens equation, we find that the second lens should be at distance  $f(s_1 - 1)$  behind the image of the first lens. Therefore the second lens should be placed at

<sup>4</sup><http://www.swanphotolabs.com/swan08/>

<sup>5</sup><http://www.edmundoptics.com/onlinecatalog/displayproduct.cfm?productID=2190/>



**Figure 13:** Left: Picture taken from our sample preliminary prototype. Middle: Zoomed-in view of the shifting mechanism from a different angle. Right: Images taken with our prototype with RGB and CMY modes in different lighting conditions. Please note the better SNR of the CMY mode for the dark scene (left) and the better SNR of the RGB mode for the lighted scene (right) in the zoomed-in views.



**Figure 14:** Left: This figure shows the design of the add-on device that be added to a regular Bayer LDR camera to achieve a HDR mode. A pair of lens, separated by a fixed distance, is put between the Bayer CFA on the camera sensor and the printed CFA. The lens which is close to the CFA is slightly off-axis. The amount of shift is controlled moving this two-lens ensemble on a rail. Right: We show the LDR and HDR images captured by this prototype. Note the saturated sky in the LDR mode is better captured in the HDR mode.

$d_2 = \frac{f(s_1+1)^2}{s_1} - f(s_1 - 1) = 2f + x_1$  and the image of this lens will be formed at  $x_2 + \frac{f(s_1-1)}{s_1} = 4f$ . Thus, irrespective of  $s_1$  and  $s_2$  the distance between the CFA and sensor should be  $4f$  and the distance between the lenses should be  $2f$ .  $\beta$  can be changed by moving the lenses to different positions between the CFA and sensor.

The main advantage of this setup is that the shifting can be precisely controlled by a few orders of magnitude larger movement of the lenses. For example, in our setup we used two lenses with  $f = 3cm$ . The image is not shifted when placing the first lens at distance  $f$  from the first CFA. In the shifted state, we chose  $\alpha = 10\beta = 264\mu m$ . Therefore,  $x_1 = \frac{f(s_1+1)}{s_1} = 3.3cm$ . Thus, we can achieve  $26.4\mu m$  shift by moving the lenses  $3mm$  away from the CFA. Further, the setup can added to any camera with a wide-band CFA without changing any of the internal parts. However, the setup is relatively large since the image is focused twice on the two CFAs.

We used this to design a prototype switchable LDR/HDR camera using a Canon Rebel Xsi camera with a Bayer CFA on its sensor. We use the CFA pattern in Figure 6 for the second layer. Finally, we used two lenses with  $25mm$  diameter and  $30mm$  effective focal length with the first lens moved  $264\mu m$  off-axis to achieve the shift. For CFA, we used  $35mm$  digital slides. All the resolution and quality issues that exists in the previous prototype also exists here and are handled similarly. However, since at least one CFA layer is a high quality one, we achieve better results (Figure 14).

## 7 Discussion

**Demosaicing:** Our camera with switchable modes has novel CFA

	Bilinear	Gunturk	Li	Lu & Tan	Minimum
Fig 2-RGB	1.07	1.03	1.01	<b>0.92</b>	<b>0.92</b>
Fig 2-CMY	1.11	0.98	<b>0.92</b>	1.00	<b>0.92</b>
Fig 2-RGBCY	1.40	1.28	1.18	<b>1.16</b>	<b>1.16</b>
Fig 6-RGB	0.80	0.78	0.76	<b>0.71</b>	<b>0.71</b>
Fig 6-CMY	1.08	0.98	0.89	<b>0.88</b>	<b>0.88</b>
Fig 6-RGBCY	1.39	1.28	1.17	<b>1.16</b>	<b>1.16</b>
Fig 5-LDR	0.84	0.83	0.75	<b>0.70</b>	<b>0.70</b>
Fig 5-HDR	1.63	1.48	<b>1.39</b>	1.41	<b>1.39</b>
Bayer	0.82	0.79	0.74	<b>0.73</b>	<b>0.73</b>

**Table 2:** Comparison of the performance of several demosaicing methods for different capture modes of our camera. We used CIE Lab difference from a non-mosaics image as the error metric.

patterns whose behavior to demosaicing is studied here. There are several demosaicing methods in the literature, often suitable for particular CFA patterns. Freeman [1988] uses a median filter to process the inter-channel differences of demosaiced images obtained by bilinear interpolation. Some other methods investigate the spatial and frequency characteristics of the image to achieve better demosaicing. For example, edge classifiers are often used to identify the best directions for interpolating the missing color values [Li 2005; Hamilton and Adams 1997]. [Gunturk et al. 2002] uses a scheme to exploit spectral correlation by alternately projecting the estimates of the missing colors onto constraint sets based on original CFA samples and prior knowledge of spectral correlation.

In addition to bilinear interpolation, we experimented with several more recent demosaicing methods [Gunturk et al. 2002; Li 2005; Lu and Tan 2003] to evaluate their suitability for our particular

CFA patterns in Figures 2, and 6. To quantify this, we find the average  $\Delta E$  difference of the demosaiced image from the original non-mosaiced image in the CIELAB space (Table 2). We also compare this to the error due to demosaicing for a Bayer pattern.

Table 2 shows that though most methods work well for the different modes, each mode favors some demosaicing methods over others. Most importantly, demosaicing artifacts from the RGB mode of the pattern in Figure 2 is comparable to the Bayer pattern and even slightly better when considering the minimum error. However, the pattern in Figure 6 shows higher error in the same mode primarily due to adjacent tiling of similarly colored filters. Also, the CMY mode of both our patterns show more error than the RGB mode. Finally, the RGBCY mode shows more error than the RGB or CMY modes. This emphasizes the need for switchable primaries where lesser noise and demosaicing artifacts can be traded over color fidelity when it is not of critical importance. Further, like any single shot HDR camera, our switchable LDR/HDR camera compromises spatial resolution in HDR mode (Figure 8). This manifests itself as larger demosaicing errors for the HDR mode than the LDR mode.

**Effects on Light Efficiency:** Usually RGB CFAs are built using layered combinations of CMY dyes [Gunturk et al. 2005] in a fashion equivalent to our RGB mode. Hence, layering CFAs does not compromise the spectral transmittance in the RGB mode of the switchable camera. Since the current filters have light efficiency close to 90%, even in the CMY mode, there is only a small loss in the light efficiency (around 10%) that is outweighed by the 70% improvement in the SNR in this mode.

In order to confirm this in practice we compared the performance of our camera with the raw images (to avoid post-processing) from a standard RGB camera with similar pixel size, *Canon PowerShot S3 IS* on the same set of test images used in Table 1 in similar lighting conditions. We found the SNR of this camera to be about 0.95 of the RGB mode of our camera for both dark and bright scenes. This can be attributed to the lower transmittance of the pigments in the Canon camera compared to the dichroic filters used in our prototype and also the slightly smaller pixel size of the Canon camera.

**Practicality of the layered CFAs:** Spectral bleeding due to the CFA misalignment is the main obstacle of our layered CFA design. This can be alleviated by the use of microlenses. Proper design of microlenses and photo-detectors, that consider the filter thickness, will be the key. Further, use of high precision actuators can reduce misalignment significantly. Certain recent SLR cameras already have actuators to shift the sensors for anti-blur or dust-removal. Also, in Sinar photography<sup>6</sup> the CCD sensor is shifted three times laterally or vertically by exactly one pixel width from one exposure to the next, so that every pixel is covered by every primary color. Similar mechanism can be used for shiftable CFAs. Finally, since CFAs are printed using precise machinery, some of the issues can be alleviated during printing, for e.g. compensating for lens aberration in the second CFA layer in the optical shift setup. The micron level shifts achieved from inexpensive COTS components in our lab setting provides ample encouragement that manufacturers can do much better with the facilities in their disposal.

## 8 Conclusion

In summary, we present the concept of shiftable layering of CFAs to achieve multiple switchable operational modes within the same camera. We demonstrate two different cameras using this concept: a camera with switchable primaries that can operate in the RGB, CMY and 5-color RGBCY modes; and a camera with switchable

LDR and HDR capture modes. The camera with switchable primaries can provide superior color fidelity for colorful scenes and the optimal SNR for both dark and bright scenes. The camera with LDR and HDR modes can trade off resolution to capture a higher dynamic range. Further, we show that the general idea of CFA layering can be posed as a constraint satisfaction problem to find CFA patterns based on the design constraints. Finally, we propose some simple designs to explore the practical feasibility of embedding such shifted layering of CFAs in real cameras in the future.

## Acknowledgements

This project is supported by Majumder's NSF CAREER award IIS-0846144. We would like to specially thank Alex Ihler, our colleague in UC Irvine (UCI), for useful discussions on formalizing the constraint satisfaction problem. We would also like to thank ex-student from our lab at UCI, Maxim Lazarov of Dreamworks for lending us his voice for the supplementary video. Raskar at MIT is supported by Alfred P. Sloan Research Fellowship and Darpa Young Faculty Award.

## References

- ALTER, F., MATSUSHITA, Y., AND TANG, X. 2006. An intensity similarity measure in low-light conditions. In *ECCV*, 267–280.
- BAONE, G. A., AND QI, H. 2006. Demosaicking methods for multispectral cameras using mosaic focal plane array technology. *Proc. SPIE 6062*.
- BAYER, B. Color imaging array. *US Patent 3,971,065*.
- CAO, H., AND KOT, A. C. 2008. A generalized model for detection of demosaicing characteristics. In *ICME*, 1513–1516.
- DRAGO, F., MYSZKOWSKI, K., ANNEN, T., AND CHIBA, N. 2003. Adaptive logarithmic mapping for displaying high contrast scenes. *Computer Graphics Forum 22*, 419–426.
- FREEMAN, T. W. 1988. Median filter for reconstructing missing color samples. *US Patent 4,724,395*.
- GINDELE, E., AND GALLAGHER, A. Sparsely sampled image sensing device with color and luminance photosites. *US Patent 6,476,865*.
- GU, J., WOLFE, P. J., AND HIRAKAWA, K. 2010. Filterbank-based universal demosaicking. 1981–1984.
- GUNTURK, B., MEMBER, S., ALTUNBASAK, Y., MEMBER, S., AND MERSEREAU, R. M. 2002. Color plane interpolation using alternating projections. *IEEE Trans. Image Processing 11*, 997–1013.
- GUNTURK, B., GLOTZBACH, J., ALTUNBASAK, Y., SCHAFER, R., AND MERSEREAU, R. 2005. Demosaicking: color filter array interpolation. *IEEE Signal Processing Magazine*.
- HAMILTON, J., AND ADAMS, J. 1997. Adaptive color plane interpolation in single sensor color electronic camera. *US Patent 5,629,734*.
- HIRAKAWA, K., AND WOLFE, P. 2008. Spatio-spectral color filter spatio-spectral color filter array design for optimal image recovery. *IEEE TIP 17*, 10.
- IMAI, F. H., BERNS, R. S., AND TZENG, D.-Y. 2000. A comparative analysis of spectral reflectance estimated in various spaces using a trichromatic camera system. *Journal of Imaging Science and Technology 44*, 280–287.

<sup>6</sup><http://www.sinar.ch/en/products/digital-backs/241-sinarback-evolution-86-h>

- KUMAR, M., MORALES, E., ADAMS, J., AND HAO, W. 2009. New digital camera sensor architecture for low light imaging. *IEEE ICIP*, 2681–2684.
- LANGFELDER, G., ZARAGA, F., AND LONGONI, A. 2009. Tunable spectral responses in a color-sensitive cmos pixel for imaging applications. *Electron Devices, IEEE Transactions on*.
- LI, X. 2005. Demosaicing by successive approximation. *IEEE Trans. Image Process.* 14, 3, 370–379.
- LU, W., AND TAN, Y.-P. 2003. Color filter array demosaicking: new method and performance measures. *Image Processing, IEEE Transactions on*.
- LUKAC, R. 2008. *Single-Sensor Imaging: Methods and Applications for Digital Cameras*. CRC Press.
- MOHAN, A., RASKAR, R., AND TUMBLIN, J. 2008. Agile spectrum imaging: Programmable wavelength modulation for cameras and projectors. *Computer Graphics Forum* 27, 2, 709–717.
- RATNER, N., AND SCHECHNER, Y. Y. 2007. Illumination multiplexing within fundamental limits. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1–8.
- SCHECHNER, Y. Y., NAYAR, S. K., AND BELHUMEUR, P. N. 2007. Multiplexing for Optimal Lighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 8, 1339–1354.
- SHOGENJI, R., KITAMURA, Y., YAMADA, K., MIYATAKE, S., AND TANIDA, J. 2004. Multispectral imaging using compact compound optics. *Opt. Exp.*, 16431655.
- SUSANU, G., PETERESCU, S., NANU, F., CAPATA, A., AND CORCORAN, P. 2009. Rgbw sensor array. *US Patent 2009/0,167,893*.
- YAMAGAMI, T., SASAKI, T., AND SUGA, A. Image signal processing apparatus having a color filter with offset luminance filter elements. *US Patent 5,323,233*.
- YASUMA, F., MITSUNAGA, T., ISO, D., AND NAYAR, S. Generalized Assorted Pixel Camera: Post-Capture Control of Resolution, Dynamic Range and Spectrum. Tech. rep.

## Appendix

Our SNR analysis is inspired by prior work on illumination multiplexing [Schechner et al. 2007]. To capture the effect of illumination from a single light source in a scene lit by multiple lights, images can be captured by illuminating the scene with one source at a time. However, this leads to considerable noise due to the low illumination, especially in the shadow regions. [Schechner et al. 2007] shows that acquiring images with multiplexed sources reduces the noise. The effect of each light source can then be recovered by demultiplexing the captured values. The scenario with cameras is analogous. The primaries of a narrow band camera are designed to capture each of the red, green or blue channels. Whereas, the primaries of a broad band camera multiplex these bands to improve the light efficiency. Hence, we propose a similar paradigm for analyzing the SNR of the multiplexed or non-multiplexed capture modes.

**Modeling SNR:** Let us consider a color basis with  $n$  channels whose spectral transmittances overlap minimally (e.g. RGB). Let the total number of photons reaching the camera from a spatial point before being filtered by the primaries be  $\alpha$ . Hence,  $\alpha$  changes spatially with the scene content and also with the change in aperture or shutter speed of the camera. For a general camera, let us assume  $m$  physical color filters that multiplex these  $n$  channels by transmitting or blocking each channel completely (e.g. a cyan primary transmits

B and G but blocks R). Let the transmittance of these  $m$  primaries be  $T = (t_1, t_2, \dots, t_m)^T$ . If we assume that the light is evenly distributed across all wavelengths, then the expected value of the amount of light passing through any primary is given by  $\alpha T$ . Let us consider a  $m \times n$  multiplexing matrix  $M$  such that  $M(i, j)$  is 1 if channel  $i$ ,  $1 \leq i \leq n$ , is passed and 0 otherwise. Hence, the expected values computed for each channel  $i$ ,  $c_i$ , is given by  $E(c_i) = \alpha M_i^{-1} T$ , where  $M_i^{-1}$  is the  $i$ th row of  $M^{-1}$ . We define the expected value  $E(C)$  of  $C = (c_1, \dots, c_n)$  to be a vector given by  $E(C) = (E(c_1), \dots, E(c_n))$ .

For the sake of simplicity we assume the noise level is always computed for the same sensor gain, i.e. ISO number. The sources of noise in an imaging pipeline can be categorized into signal-dependent or signal-independent noise [Schechner et al. 2007; Alter et al. 2006; Ratner and Schechner 2007]. The signal-dependent noise can be expressed as a Poisson distribution of the photons that reach the sensor, i.e. each pixel. Since this is dependent on the number of photons, it is the dominant noise when the number of photons is high, i.e. for lighted scenes. The variance of the signal-dependent noise for each primary  $j$  is therefore proportional to the expected captured values  $\alpha t_j$ . We assume the variance of this signal-independent noise is the same across all the primaries,  $S$ .

Hence, the total variance for channel  $i$  is given by  $\sigma_i^2 = \sum_{j=1}^m (M_{ij}^{-1})^2 (\alpha t_j + S)$ . For dark scenes, the signal-independent noise dominates and the above equation becomes  $\sigma_i^2 = \sum_{j=1}^m (M_{ij}^{-1})^2 S$ . For bright scenes, the signal-dependent noise dominates and the above equation becomes  $\sigma_i^2 = \sum_{j=1}^m (M_{ij}^{-1})^2 \alpha t_j$ . Now, we define the total variance for  $C$  as a vector  $\sigma_C = (\sigma_1, \dots, \sigma_n)$ . Hence, the signal to noise ratio for  $C$  is given by

$$SNR(C) = \frac{|E(C)|}{|\sigma_C|} \quad (3)$$

However, note that defining the SNR for the intensity image  $g$  is much simpler. In this case,  $E(g) = \alpha \sum_{i=1}^m t_i$  and the  $\sigma_g = \sqrt{\sum_{i=1}^m S + \alpha t_i}$ . For dark scenes,  $\sigma_g = \sqrt{\sum_{i=1}^m S}$ , and for bright scenes,  $\sigma_g = \sqrt{\sum_{i=1}^m \alpha t_i}$ . Hence, the  $SNR(g) = \frac{E(g)}{\sigma_g}$ .

For any camera, we usually know the matrix  $M^{-1}$ . For example, the matrix  $M$  for an RGB camera is a  $3 \times 3$  identity matrix since the channels and the filters are identical. Hence,  $M^{-1}$  is also identity. But, for CMY cameras with that capture multiplexed RGB channels, the matrix  $M$  and  $M^{-1}$  are as follows.

$$M_{CMY} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}, M_{CMY}^{-1} = \frac{1}{2} \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix} \quad (4)$$

Or, when considering the 5-primary mode of our camera,  $n = 5$  since we can capture 5 almost non-overlapping color channels as shown in Figure 3(d). However,  $m = 6$ . This means that  $M$  is not a square matrix, but a  $6 \times 5$  matrix and  $M^{-1}$  is a non-unique pseudo-inverse.  $M$  and one such pseudo inverse are shown below.

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}, M^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (5)$$

Further, note that when computing the ratios of the SNRs (e.g. RGB vs. CMY) for dark or bright scenes, we do not need to know  $\alpha$  or  $S$  since they cancel out. Hence, as long as we know the transmittance of the primaries, Figure 3, we can predict the relative improvement or degradation of SNR. Since we know the transmittance of the primaries in our camera, we use this to predict two ratios,  $\frac{SNR_{CMY}}{SNR_{RGB}}$  and  $\frac{SNR_{CMY}}{SNR_{RGBCY}}$ , for both bright and dark scenes (Table 1).

# A Scalable Distributed Paradigm for Multi-User Interaction with Tiled Rear Projection Display Walls

Pablo Roman and Maxim Lazarov and Aditi Majumder, *Member, IEEE*



Fig. 1. This figure shows some of our applications in action. From left to right: Our collaborative map visualization application with two users visualizing different parts of the map at the same time on our  $3 \times 3$  array of nine projectors; Our collaborative emergency management application with two users trying to draw a path to hazardous location and dispatching teams of first responders on our  $3 \times 3$  array of nine projectors; Digital graffiti drawn using our collaborative graffiti application on only six of the projectors. We deliberately did not edge blend the projectors to show the six projectors clearly; Four children working together on our digital graffiti application on a  $3 \times 3$  array of nine projectors.

**Abstract**— We present the first distributed paradigm for multiple users to interact simultaneously with large tiled rear projection display walls. Unlike earlier works, our paradigm allows *easy scalability across different applications, interaction modalities, displays and users*. The novelty of the design lies in its distributed nature allowing well-compartmented, application independent, and application specific modules. This enables adapting to different 2D applications and interaction modalities easily by changing a few application specific modules. We demonstrate four challenging 2D applications on a nine projector display to demonstrate the application scalability of our method: map visualization, virtual graffiti, virtual bulletin board and an emergency management system. We demonstrate the scalability of our method to multiple interaction modalities by showing both gesture-based and laser-based user interfaces. Finally, we improve earlier distributed methods to register multiple projectors. Previous works need multiple patterns to identify the neighbors, the configuration of the display and the registration across multiple projectors in logarithmic time with respect to the number of projectors in the display. We propose a new approach that achieves this using a single pattern based on specially augmented *QR codes* in constant time. Further, previous distributed registration algorithms are prone to large misregistrations. We propose a novel *radially cascading geometric registration technique* that yields significantly better accuracy. Thus, our improvements allow a significantly more efficient and accurate technique for distributed self-registration of multi-projector display walls.

**Index Terms**—Tiled Displays, Human-Computer Interaction, Gesture-Based Interaction, Multi-user interaction, Distributed algorithms

## 1 INTRODUCTION

Large multi-projector planar display walls are common in many visualization applications. We have seen a large amount of work on camera-based registration of multiple projectors in such displays, both for geometry and color [25, 8, 4, 21, 22, 2, 3, 28]. This has enabled easy deployment and maintenance of such displays. However, a suitable interaction paradigm for these displays that can be scaled to multiple users for large number of display modules across different applications and interaction modalities is still not available. This has brought in a belief in the visualization community that limited display space and interactivity makes it difficult for application users to solve issues of interactively understanding domain problems. This paper focuses on providing a new approach of scalable interactive multi-user interaction for tiled display walls. The final roadblock in the adoption of any technology is the ease with which users can interact with it. Our scalable interaction paradigm brings in the hitherto unknown ease in user interactivity and deployment of it on commodity projectors. Hence, this work can have a significant impact on wider adoption of

the seamless multi-projector display technology across the visualization community.

Most work in the human computer interaction domain [6, 31, 30, 32, 36, 29, 20, 19, 17, 16, 37, 9, 7] is difficult to scale to multiple interaction modalities, applications, users and displays. Central to this problem is the fact that almost all earlier works in the domain of interaction with tiled displays have explored application specific centralized algorithms and architectures which inherently cannot scale with respect to the number of users and displays due to critical dependency on a single server. Further, scalability to multiple applications and interaction modalities demand careful algorithm design to compartmentalize the application/interface specific modules from the application/interface independent ones and has not been explored before.

This paper makes the first effort to design a scalable interaction paradigm for rear-projected tiled displays that can scale with multiple projectors, users, applications and even interaction modalities. We observe that such a general paradigm is only possible with a distributed architecture that inherently provides mechanisms for scalability. Such a distributed architecture for multi-projector display walls is presented in [3] where the display is built by a distributed network of plug-and-play projectors (PPPs). Each PPP consists of a projector, a camera and a communication and computation unit, simulated by a computer. The display is created by a rectangular array of these PPPs on a planar screen. Each PPP runs an SPMD (single program multiple data) algorithm presented in [3] that starts by believing that it is the only display in the environment. It can communicate with its neighbor using its camera which sees parts of its neighboring PPPs. Using such visual communication via the cameras and a distributed configuration

• Pablo Roman and Maxim Lazarov and Aditi Majumder are in Computer Science Department of University of California, Irvine, E-mail: {proman,mlazarov,majumder}@uci.edu.

Manuscript received 31 March 2009; accepted 27 July 2009; posted online 11 October 2009; mailed on 5 October 2009.

For information on obtaining reprints of this article, please send email to: tvcg@computer.org.

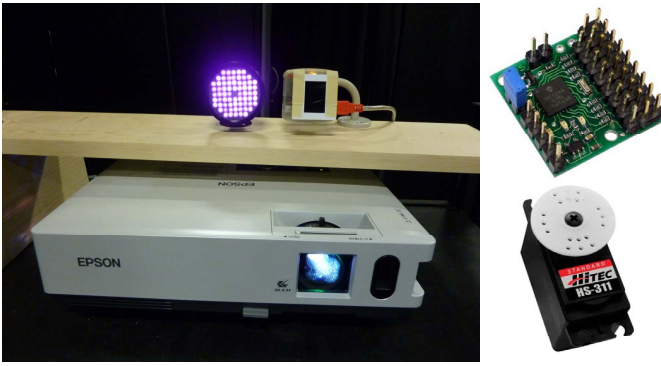


Fig. 2. Left: Our prototype PPP with a projector, a camera and a computer; Right bottom: The inexpensive RC servo that can be used to switch the IR filters back and forth. Right top: The RS-232 8 servo controller.

discovery algorithm, the PPPs discover the total number of PPPs creating the tiled display, their array configuration (total number of rows and columns) and its own coordinates (its own row and column) in this array. Following this, the PPPs can align themselves to create a seamless display using a distributed self-registration algorithm.

### 1.1 Main Contributions

We use the same distributed architecture based on PPPs presented in [3] and build a new distributed registration algorithm and a distributed interaction paradigm on top of it. For interaction, we design a *SPMD distributed interaction algorithm* that runs on each PPP following the registration to allow multiple users to interact with the display using any kind of interaction modality. The highlights of our distributed interaction algorithm are as follows.

1. Since we design an SPMD algorithm, it can easily *scale to multiple projectors*. Hence, adding and removing PPPs to reconfigure the display does not necessitate any change in the interaction algorithm.
2. Most modules of our algorithm are application independent. Hence, to adapt to different 2D applications, only a few application specific modules need to be modified. This allows our algorithm to *scale to many 2D applications*.
3. Similarly, changing the interaction modality requires modifying a small number of interface dependent modules. This allows our algorithm to *scale to different interaction modalities* as well (e.g. laser pointers, gesture-based interface).
4. Unlike a centralized system where all the interaction from multiple users is handled by a single centralized server, a distributed algorithm distributes the load of handling multiple users to multiple PPPs. Hence, our algorithm can easily *scale to multiple users*.

We also propose a new distributed registration technique that achieves much greater accuracy and is more efficient in terms of performance and bandwidth load than the method presented in [3]. Below are the highlights of our new registration technique as compared to [3].

1. First, while discovering the configuration of the PPP array, multiple rounds of visual communication were used via the cameras in [3]. This required processing multiple patterns for each PPP and converged in logarithmic time with respect to the number of projectors in the display. The performance was also compromised due to computationally intensive image processing. In contrast, we design a novel method in which each PPP uses a single pattern made of specially augmented *QR (Quick Response) codes* to discover the display configuration and self-register simultaneously in constant time. More importantly, we achieve this without increasing the network communication load across the PPPs.
2. Second, [3] uses a distributed homography tree algorithm for self-registration of the PPPs. This can lead to large misregistrations (even as large as 10-20 pixels), especially when the PPPs are further away from the reference PPP. This impacts the scalability of the self-registration algorithm to a large number of projectors. We present

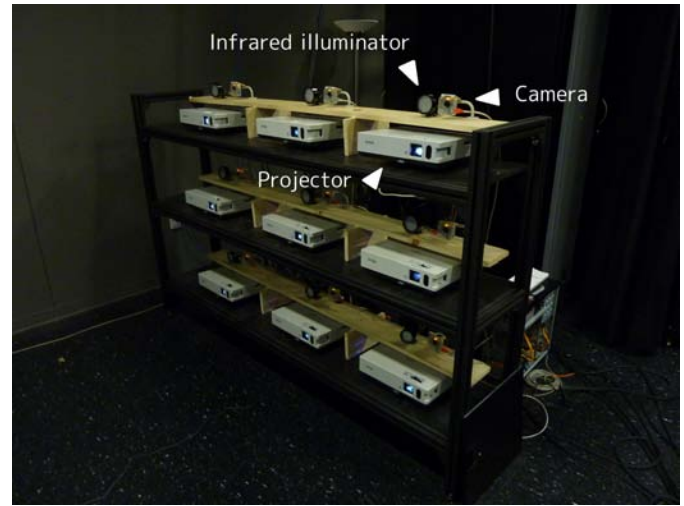


Fig. 3. Our setup of a network of PPPs augmented by the IR illuminators and the IR camera filters.

a novel *radially cascading geometric registration* method that can achieve a much superior accuracy.

In summary, our work, for the first time, *introduces* an entirely *distributed framework for user interaction* with tiled displays. In addition, we *improve* the existing *distributed framework for registering* the many PPPs in the display. We first discuss our system in detail in Section 2, followed by the distributed interaction paradigm and the improved distributed registration in Section 3 and Section 4 respectively, concluding with future directions in Section 5.

## 2 SYSTEM OVERVIEW

Our system consists of  $N$  PPPs, each made of a projector, and a camera connected to a computer. We assume that the projectors and cameras are linear devices. The PPPs are arranged casually in a rectangular array (Figure 3) and overlap with their neighbors (adjacent PPPs). The PPPs are initially unaware of the configuration of the array that they are arranged in. Using visual communication via the cameras, a PPP starts detecting its neighbors whenever its associated camera perceives some other PPP in its overlapping coverage area with the adjacent PPP. Using our distributed registration technique (Section 4) each PPP can discover its neighbor, the total number of projectors in the display and their array configuration, its own coordinates in the array of PPPs and finally self-register itself with other PPPs to create a seamless display. The PPPs use an NTP (Network Time Protocol) clock synchronization to achieve a synchronized clock across the multiple PPPs. The importance of such a synchronization will be evident in the subsequent sections when we describe our distributed interaction paradigm. We also assume that the PPPs use a constant IP multicast group to communicate.

Once the PPPs are registered, we desire to interact with the display. We use two kinds of interaction modalities in this paper – 2D hand gestures and laser based interaction. Though the lasers are bright enough to be detected easily in the projected images, when using gestures the camera cannot detect visible light gestures reliably because the screen and projected image obstruct the hand. To handle this situation, as in [12, 19], we augment our PPP with an IR illuminator and mount an IR bandpass filter on the camera. These filters are removed during registration and then put back to resume interaction. The IR illuminator and the IR filter on the camera allow us to detect gestures when we touch the screen. We use a standard rear-projection screen (from Jenmar), which acts as a good diffuser of IR light. In our setup, we use monochrome cameras without IR cut filters, although we only used one of the color channels. Figure 2 shows one of our IR sensitive PPPs and Figure 3 shows our setup. Removing the IR filter during registration can be achieved automatically by inexpensive RC servos



(\$10/unit) and can be controlled with serial (RS-232) servo controllers (\$20 for controlling 8 RC servos), which are also simple and inexpensive. The IR emitter must also be switched off during registration which could be done via a serial/USB-actuated relay. This can prevent the sensor from getting saturated by both IR and projected visible light.

### 3 THE DISTRIBUTED INTERACTION PARADIGM

In this section, we describe our distributed interaction paradigm in detail. We start by describing the related work in the domain of various interaction paradigms for large scale displays and comparing our work with it (Section 3.1). Next we describe our distributed algorithm in detail in Section 3.2. When describing this, we consider 2D gesture-based interaction since restricting to a specific interaction modality allows us to provide a simple explanation. However, we explain ways to scale to different interaction modalities in the end of Section 3.2.1. We present implementation details and results in Section 3.3 and 3.4.

#### 3.1 Related Work

Large displays require interaction modalities that match their scale instead of the more traditional mouse and keyboard. The most natural form of such an interaction is using gestures and several works have explored it [1, 11, 29]. Since detecting a gesture unambiguously is a difficult computer vision problem, touch sensitive surfaces [19, 27, 37, 10, 12] have been explored for better localization of gesture dependent features. New devices that can aid when gestures are ambiguous have also been explored [34]. Parallely, we have seen the development of interaction devices by which users can convey their intentions much more precisely without the ambiguity of gestures. These include devices like simple lasers [24], VisionWand [5], a special touch pad [23], LEDs on tangible devices [20], a remote control [18], objects with simple geometry like blocks or cylinders [33], or even a handheld camera [17]. However, all these works focus on interfaces and hence use a simple single display and single sensor paradigm.

When using multiple projectors and sensors, new issues arise like tracking the interaction across multiple projectors and cameras, deciding on a reaction that is unanimous across the multiple projectors, and reacting in minimal time using minimal network bandwidth. There have been a few works that use multiple projectors, but they use a single camera or a pair of stereo cameras. Hence, the interaction tracking in these systems is centralized and handled by a single server [18, 35, 36, 20]. Further, the same centralized server decides a suitable reaction to the gesture and informs the different projectors on how to react. Though this does not mean that all projectors react similarly, a centralized server decides and communicates the different reaction each projector should produce.

Few recent works address systems with multiple projectors and cameras. [9] uses a laser based interaction paradigm where multiple cameras can detect the location of multiple lasers used by multiple users. [30] uses multiple cameras to detect gestures of multiple people. Although in both these systems the processing of images from the camera is done in a distributed manner by a computer connected to each camera, the processed data is then handed to a server that finds the 2D position of the gesture directly or by triangulation. The same server is responsible for managing the projectors and hence it decides the reaction for each projector and communicates it to them. Thus, these works all have in common the centralized architecture where a single server is responsible for tracking the gesture and then reacting to it. Distributed rendering architectures [13, 14, 15] also follow a similarly centralized architecture where the rendering takes place in a distributed manner in computers attached to each projector, but they are controlled by a centralized server that manages how the rendering should be distributed.

**Comparison with Our Work:** In contrast, our work focuses on a completely distributed paradigm where each PPP acts as self-sufficient module. Unlike previous work, where the projectors and cameras are treated as different devices, we view the display as a distributed network of PPPs. Our goal is to develop a single program multiple data

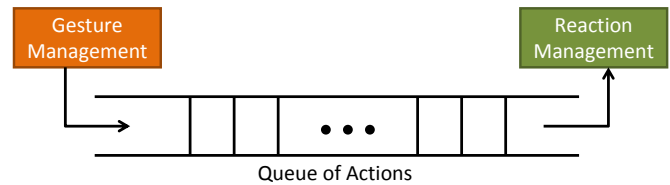


Fig. 4. The Distributed Interaction Paradigm.

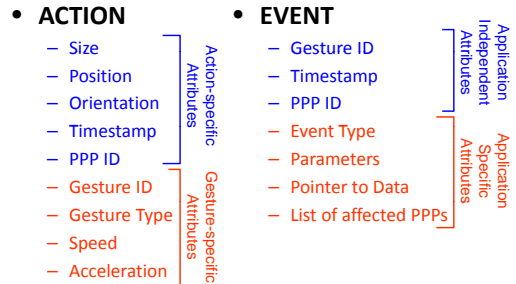


Fig. 5. This figure describes the action data type used in gesture manager and the event data type used in reaction manager.

(SPMD) algorithm to be run on each PPP that would detect and track the user action in a completely distributed manner affecting only the PPPs that see the action. Further, an appropriate reaction should be produced by the relevant PPPs in response to the gesture, even if the gesture does not occur within them. This assures minimal network bandwidth usage since all PPPs do not communicate to a single centralized server and minimal time since the processing is shared by multiple PPPs and is not the responsibility of a single centralized server. Such a distributed paradigm allows easy scalability to multiple displays, applications, interaction modalities and users.

#### 3.2 The Algorithm

We consider interaction to be a set of two operations that occur consecutively: (a) a *2D gesture* made by the user; and (b) a *consequent reaction* provided by the display. We assume that a gesture is a sequence of samples, also called *actions*, detected by the system. These samples can be generated through a multitude of input systems including touch – by placing the palm on the screen, or laser pointers. The meanings of isolated or temporal sequences of actions are predefined by applications for consistent interpretation. Note that since a gesture occurs over an extended period of time, it can span across multiple PPPs moving between the non-overlapping and overlapping areas of the PPPs. Further, it is important that the reaction does not wait for the gesture to complete. For example, if the user is moving his/her hands from left to right, he/she is expecting the underlying image to move from left to right even before he/she completes the gesture. Hence, the goal is to identify the gesture even when it is not complete and start reacting as soon as possible.

Our distributed interaction paradigm consists of two main components: a *distributed gesture management* module (Section 3.2.1) and a *distributed reaction management* module (Section 3.2.2). These are run as two threads in a producer-consumer fashion in each PPP (Figure 4). The distributed gesture management module produces a queue of actions that are then processed (or consumed) by the distributed reaction manager in an FCFS manner. Note that though the user's intentions are interpreted per gesture (which is a series of actions), the processing of these gestures is done per action. This difference in the granularity of interpretation and processing allows the system to respond to a gesture as soon as it commences without waiting for its end. Finally, the distributed gesture management is application independent. The application specific modules occur only during reaction management.



Fig. 6. This figure shows a few different types of hand postures used for gesture-based interaction. Each application can define its own interpretation for each posture. For example, in our map visualization application, touching the screen with two fingers is used to change the displayed layer, touching it with one finger is used to open and close individual working windows, sweeping with an open hand is used to move the map around and twisting a closed hand is used for zooming in and out. On the other hand, in our graffiti application, two fingers are used to bring up a color palette, one finger to select a color from the palette and any other postures to draw lines.

### 3.2.1 Distributed Gesture Management

In a distributed network of PPPs, there is no centralized server that manages the observed actions of the user. Each PPP is responsible for managing the actions that occur within its domain. When the gesture spans across multiple PPPs, we design a mechanism to track it and hand over its management from one PPP to another as it moves across PPPs. This is achieved in a manner transparent to the user. Further, our framework can handle *multiple* users each doing single hand gestures.

The distributed gesture management involves (a) a *shared action management mechanism* to decide which PPP handles which part of the gesture via their reaction managers, and (b) *shared gesture tracking* to follow the path of the gesture as it moves across multiple PPPs and is facilitated via an anticipatory action communication mechanism.

**Action Data-Type:** First we describe the action data structure (Figure 5) filled up by a PPP when it detects an action. This consists of *action specific attributes* like position, orientation and size of the hand, detecting PPP ID and timestamp (synchronized by NTP) in the detecting PPP. The timestamp needs to be included in the attributes to handle race conditions, described in detail in Section 3.2.2. The position is denoted in the global coordinates of the display. Note that since each PPP knows the exact configuration of the entire display and its position in it in the post-registration phase, it can easily calculate the global position of the action. The action also contains some *gesture specific attributes* like gesture ID, gesture type, speed, and acceleration. As soon as the commencement of a new gesture is identified, a new gesture ID is generated. When detecting the  $i$ th gesture in the  $j$ th PPP,  $0 \leq j < N$ , the PPP assigns a gesture ID of  $i * N + j$ . Hence, the identity of the PPP where the gesture commenced can be deciphered from the gesture ID field of the first action in that gesture. Gesture type refers to the type of hand posture which when seen over a period of time constitutes the gesture (Figure 6). The speed and acceleration of the gesture denote its local speed and acceleration at the time when this component action was made in the gesture. The speed and acceleration is computed by finding the differences of the position and speed respectively in two consecutive actions in a gesture.

To detect a gesture, the PPP first recognizes the first action of the gesture in its camera frame. At the commencement of the gesture, the gesture type is set to be *undecided*. To detect the gesture type robustly and reliably, a few of the initial actions are examined. Each of these actions votes for one particular gesture type. The gesture type that receives the maximum votes is selected as the type of the gesture.

**Shared Action Management:** The shared action management scheme enqueues the constituting actions of a gesture as it moves across the display through multiple PPPs. When an action that does not belong to any previous gesture is detected, it indicates the commencement of a *new gesture*. If this new action or part of it occurs in the non-overlapping region of a PPP, since no other PPP can see this action completely, this PPP bears the responsibility to enqueue this action to be processed by its reaction manager. However, when the action occurs in the overlap region of multiple PPPs, it is desirable for only one PPP to enqueue it for processing by the reaction manager. This avoids inconsistent reaction from multiple PPPs. To assure this, when in the overlap region of multiple PPPs, the gesture is only handled by the PPP with the smallest ID. Figure 7 illustrates this process.

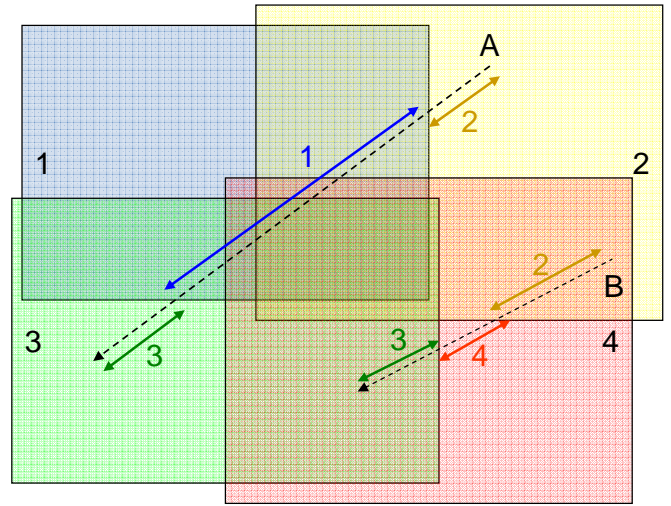


Fig. 7. This figure shows how the gestures made across multiple PPPs are handled in a shared manner by multiple PPPs in their lifespan. A and B denotes two different gestures. The length of the gesture is divided and labeled to show which PPPs handle which part of the gesture.

A and B denote two gestures. A starts in the non-overlapping area of projector 2. As soon as it enters the overlapping region of 1 and 2, 1 picks up the gesture since it has the smaller ID of the two projectors. After this, note that 1 continues to handle the gesture even though it moves through the overlap of 1 and 2, overlap of 1 and 4, overlap of all four projectors, overlap of 1, 3 and 4, and overlap of 1 and 3. Only when the gesture moves to non-overlapping area of 3, it is handled by 3 since no one else can see it. Similarly, in gesture B, when it starts in the overlap of 2 and 3, it is first picked up by 2. Then it is handled by 4 in the non-overlapping area of 4. But as soon as it moves to the overlapping area of 4 and 3, 3 starts to handle the gesture.

**Shared Gesture Tracking:** The gestures are tracked in a shared manner by multiple PPPs when they span multiple PPPs. This is achieved via the temporal and spatial proximity of consecutive actions in a gesture. If an action is close temporally and spatially to another action, it is assigned the same gesture ID and type. If an action is temporally or spatially far away, it is considered the commencement of a new gesture. This can happen when two users are interacting simultaneously with the same PPP. For this purpose, a threshold has been defined that tries to make a compromise between allowing fast gestures and correctly separating different gestures.

When the gesture management migrates from one PPP to another, we use an *anticipatory action handover* mechanism to handle it. When a PPP is tracking the gesture within itself and finds it is to move into the neighborhood of an adjacent PPP, it sends an anticipatory message to notify the neighboring PPP about a gesture coming its way. This message contains all the action data necessary to handle the continuation of a gesture: position, PPP ID, gesture ID, gesture type, speed, acceleration and timestamp. Using the position, speed and timestamp, the receiving PPP can match it against future detected actions by assuming that the gesture continues at a similar speed and acceleration. In a later instant in time, when the adjacent PPP detects an action in the neighborhood of the location predicted by an anticipatory action message, it identifies the action as part of a *continuing gesture* and hence copies its gesture-specific attributes from this anticipation message. Following this, the new PPP starts tracking and managing the actions of this gesture. However, note that between a prediction and actual detection of the action in the adjacent PPP, multiple actions can occur. Hence, the adjacent PPP receives multiple anticipation messages from the same neighbor. When processing them, it only needs to consider the most recent anticipatory message. Also, if a PPP receives anticipation messages from multiple PPPs due to multiple gestures, they can be easily identified by their PPP ID attribute. The end of a gesture is

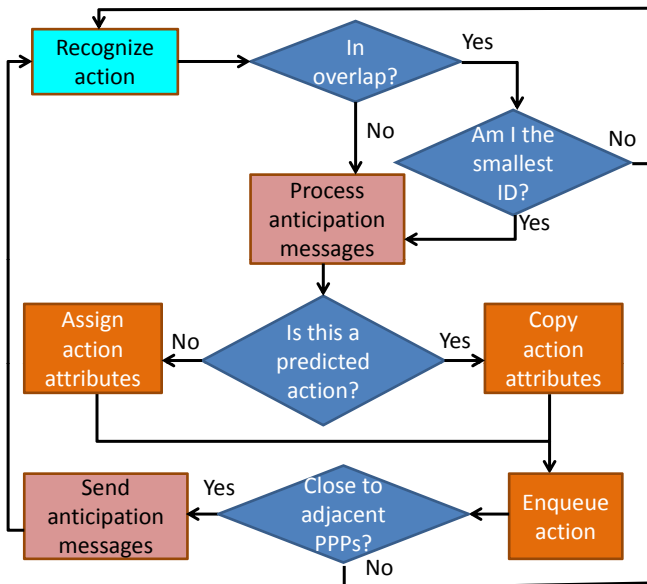


Fig. 8. The Distributed Gesture Management Protocol: The gesture management depends on the 2D application only on interpretation of recognized gestures and hence is mostly an application independent module. The cyan boxes represent the application specific modules and the purple boxes represents modules that are involved in communications.

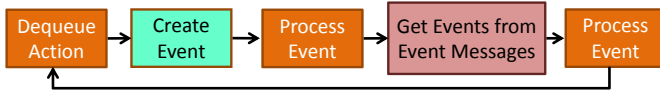


Fig. 9. The Distributed Reaction Management Protocol. The cyan boxes represent the application specific modules and the purple boxes represents modules that are involved in communications.

detected by a timeout mechanism. If the difference in timestamp of two consecutive actions is beyond a certain threshold, the new action is assumed to be the commencement of a new gesture.

**Flow Chart:** The entire gesture management module is summarized in the flowchart in Figure 8. Each PPP starts with detecting an action and deciding to pick up its management using the shared action management protocol. If the gesture continues within itself, the PPP tracks it. If the gesture moves close to an adjacent PPP, it communicates it to the relevant neighbor via the anticipatory action message. And if it receives an anticipatory action message, it picks up the gesture tracking and handling from an adjacent PPP.

**Scaling to Different Interaction Modalities:** To use different interaction modalities, only the cyan box in Figure 8 that recognizes the actions need to change. Instead of gesture based action, this module has to now detect different kinds of actions.

### 3.2.2 Distributed Reaction Management

The distributed reaction mechanism involves processing (consuming) the actions in the queue generated by the distributed gesture manager by reacting with a particular event. Note that the set of PPPs that need to react to a gesture may be larger than the set of PPPs across which the gesture spans. For example, in a map visualization application one can move the map with a sweeping gesture that spans just a few PPPs, but the map across all PPPs must move in response. Further, the event may be associated with creation, movement, or deletion of data across PPPs. Hence, the reaction manager is also responsible for taking steps to assure data consistency across the PPPs. Finally, the job of the event manager also involves informing the PPPs that will be affected by the event so that reaction managers of the PPPs that did not see any gestures can receive events they need to perform from

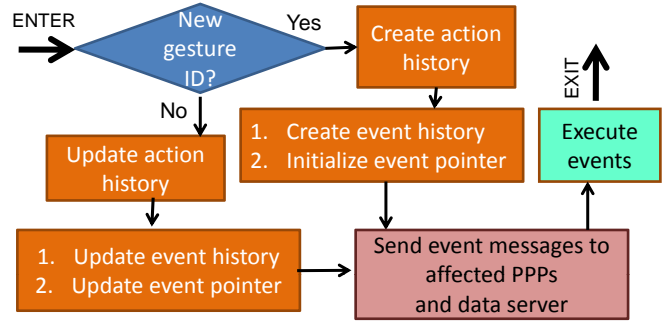


Fig. 10. Details of process events routine in the Reaction Manager in Figure 9. The cyan boxes represent the application specific modules and the purple boxes represents modules that are involved in communications.

other PPPs. The function of the reaction manager is summarized as in Figure 9. It dequeues an action from the queue of actions produced by the gesture manager and creates and processes the corresponding event. Following that, it checks if it has received any events from other PPPs and processes them. The details of the events processing is described later in this section (Figure 10).

**Event Data Type:** Figure 5 shows the event data structure used by the reaction manager. Processing every action invokes an event. Hence, just as every gesture in the gesture manager is associated with a set of actions, every gesture in the reaction manager is associated with a set of events. The event attributes constitute of a gesture ID, a timestamp indicating when the event was invoked, and the PPP ID of where it was invoked. These are all application independent attributes of an event. The application specific attributes of the event are its type and parameters. For example, the event type may be something as simple as move or rotate or as complex as open a window or resize a window. The event parameters can be the horizontal or vertical dimensions for a move, an angle for rotation and so on. Event parameters also contain a pointer to data which the event has created or is handling. For example, for opening a window, the data is the window (that can be defined by a top-left and bottom-right corner in global coordinates of the display). Finally, the event also maintains a list of PPPs it affects (e.g. for a global event like panning the imagery across the display, this will be the set of all PPPs in the display).

**Event and Action History:** As soon as the reaction manager gets an action from the queue, it creates an event. The reaction manager deciphers this predefined application specific event type associated with the action. The reaction manager can also receive events which it has to execute from other PPPs. For processing the events, the reaction manager maintains an *event history* and *event pointer* for every gesture it encounters. The event history is the array of all the events invoked by the actions in a gesture, sorted by their creation timestamp. The *event pointer* marks an index in the event history after which the events have still not been executed. This event history is instrumental in performing the events associated with a gesture in the same order in all the affected PPPs. As detailed later, the PPP may sometimes encounter an event out-of-order after it has executed some of the subsequent events. In such scenarios, the event history will be used to trigger an event 'roll back' procedure. This action history can be as large as the array of all actions comprising a gesture or as small as the previous action in the same gesture. The size of the action history depends entirely on the events that the application needs to perform. Its purpose is to enable the generation of different events depending on a sequence of actions, like detecting hand-drawn letters.

**Event Processing:** In order to process the event (Figure 10), the reaction manager first checks if the action belongs to a gesture that the PPP has not seen before. This indicates a new gesture and the reaction manager creates a new action history, an event history (each of them containing one of the two data types in Figure 5) and initializes the event pointer to the beginning of the event history. Next, the reaction manager computes the event attributes of this new event associated

with this gesture. Note that for some events, like opening a window by drawing a stroke from the top left to the bottom right corner of the window, it may not be possible to find the list of all the affected PPPs with the first action of the gesture. In this case, the PPPs will update the data attributes in a shared manner and inform the relevant PPPs as the gesture proceeds across the different PPPs. This would also mean updating the event history, action history and the event pointer in an appropriate manner. Finally, if some new data has been written, the PPP will also commit this change to the data server so that when other PPPs request the part of the data at a future time, they can see the updated data. Note that since multiple PPPs are accessing and writing data from and to a data server we assume all the different mechanisms are in place to assure consistency in a distributed data management application. Hence, our interaction paradigm is agnostic to the kind of data server being used, either centralized or distributed. Following this the reaction manager proceeds to execute the event. Executing the event involves performing the function associated with the event (moving the imagery or showing a new window and so on) and advancing the event pointer in the event history. Then it sends a message with the event data structure to all the PPPs currently in the list of affected PPPs that should perform a consistent event (e.g. moving the image by an equal amount).

**Race Conditions:** Finally, there may be a situation when an action is being processed by the reaction manager while an event related to a prior action in the same gesture is waiting in the message buffer. Hence, when the PPP gets to process the event message, it arrives out-of-order with respect to the other events in the event history for the particular gesture. Now, for certain events this may not be important since the attributes may be changed in such a manner by the subsequent events that it is not inconsistent with respect to this PPP. But, if this reveals a data inconsistency, then we need to execute an event 'roll back' procedure. In the 'roll back' procedure, we undo the effects of all the events in the event history that have a larger timestamp than the timestamp of the received event and reverse the event pointer appropriately to reflect this. Then the received event is inserted in the event history at the position of the reversed event pointer and executed. Finally, all the subsequent events are executed again and the event pointer is advanced accordingly. If there is more than one gesture affecting the same data in a manner that can cause data inconsistency, all the events with a bigger timestamp in the multiple gestures will have to be rolled back in the same manner – undone in a reverse timestamp order and executed again in the timestamp order. Old events can be removed from the event history when newer packets have been received from all the PPPs (TCP based communication ensure delivery in order) or when the event is older than the connection timeout time. Though the case of out-of-order event does not occur very often, this 'roll back' operation is critical to ensure that the final logical order of event execution is consistent across the PPPs and hence the data. One example of the occurrence of this is when a gesture is right on the division between PPPs and small registration errors result in both PPPs handling the gesture. In this case, the gesture will be treated as rapidly going from one PPP to another. The messages will be received out of order but will be correctly reordered by the 'roll back' procedure. Since our registration is very accurate, this does not produce perceivable effects in the application. Since this procedure modifies actual data, its effects can be sometimes perceived by the users. For example, in the graffiti application described below and shown in the accompanying video, when crossing the border between PPPs a line will sometimes be drawn for a split second in the wrong order (as going back and forth between PPPs) before quickly correcting itself. This will, however, happen rarely and be quick enough to not be a nuisance.

**Scaling to Different Applications:** Note that only a few modules of the reaction manager are application specific (highlighted in cyan boxes in flowcharts in Figure 9 and Figure 10). The design of the event attributes and types depends on the application and hence so does their assignment during event creation. Further, the way events are executed is also application specific. The rest of the reaction manager is common for all kinds of 2D applications and is hence application independent. The application specific modules for our test applications

Table 1. This table represents the network usage in amount of packets and bytes per second for two cases: an application with a localized reaction (graffiti) and an application with a global reaction (map). Calculations of network usage for a possible binary protocol have also been included.

App	Packets/s	ASCII bytes/s	Binary bytes/s
Graffiti	68.5	90	30
Map	26	23	8

are explained in detail in Section 3.4.

### 3.3 Implementation

The distributed interaction framework has been implemented using Java SE 1.6. When the framework starts running, TCP connections are established between all the PPPs. Each PPP either waits for the connection or it establishes it depending on the relation between each pair of PPP IDs. The applications have been written using the JOGL 1.1.1 library for OpenGL. This allows us to perform the geometric and photometric registrations (explained in Section 4) easily. But any library that allows linear transformations and alpha masking would work.

The camera image processing and recognition is performed in Matlab. For prototyping, Matlab is invoked to run the code, but for purposes requiring a higher performance, native code should be used. For the hand-based interaction, we use our home-grown simple Matlab-based software that detects the hand, computes its location, size and orientation, and determines its type by matching it to an existing hand posture library. However any existing 2D gesture recognition software can be used [38] for this purpose. For the laser-based interaction, a simple image thresholding detects the bright laser spot.

### 3.4 Results

We have prototyped four different 2D collaborative applications using this distributed interaction paradigm on our  $8' \times 6'$  display wall made of a  $3 \times 3$  array of nine projectors. The applications are (a) graffiti; (b) map visualization; (c) emergency management; and (d) virtual bulletin board. The graffiti application allows several users to draw at the same time with their hands. Touching the screen with two fingers brings up a palette, and the user can choose a color by tapping on it. That creates a temporary color square that the user can use to start drawing lines with that color. This window can be closed tapping on it again. The map visualization allows individual working windows (that can be moved or resized) for several users to work on different parts of the map simultaneously. The background map and the working windows can be panned and zoomed independently and the map type can be changed. The emergency management application demonstrates a possible interface that could be used to coordinate response teams in which several emergency management officials can coordinate the first responder efforts in different parts of the affected region. Markers can be added to indicate a danger area, two associated numbers indicating present and dispatched personnel can be updated, and routes can be drawn and erased to indicate the safest/fastest routes to reach or avoid danger areas. The virtual bulletin board allows the users to hang digital documents and manipulate them. Bulletins can be moved around, resized, highlighted and removed.

We have tested multi-user interaction successfully with up to five simultaneous users, but a display with more area should easily fit a much larger number of users. To demonstrate the ease of interaction we brought children ranging from ages 7 to 13 years old to draw on the display in a collaborative manner. It took them only a few minutes to get comfortable interacting with the display and its inherent scalability allowed multiple children to simultaneously draw on the screen without introducing additional latency. To demonstrate the scalability of our paradigm to different interaction modalities, we also show multi-user interaction with blue and green laser pointers with our existing interactive applications. We have shown a few static images of our applications in action in Figure 1, but we strongly encourage the readers to check the video for an accurate impression of functionality and performance.

Network usage has been measured during interaction for the cases of a gesture affecting only a few of the PPPs and for a gesture affecting all of them (Table 1). These values represent the traffic for an effective recognition refresh rate of 8.12 times per second, but no time has been spent optimizing the protocol for network usage. We have also included calculation of how much traffic there would be if a binary protocol were to be implemented.

Note that we have not explicitly assured synchronization of event execution across multiple PPPs. However, in practice we found the latency of execution of the same event across multiple PPPs to be small, less than 15ms. However, the main contributor for delay was the gesture recognition code since we used MATLAB for quick prototyping. Though this did not seem to bother our users – even the over-active children – we believe this delay should be greatly reduced using native code.

The application specific modules of the reaction manager, though non-trivial, are relatively simple to design. Adapting an existing application to our distributed interaction system took an average graduate student one to two days. To demonstrate this, we next describe how we designed the application specific modules for the few applications we have prototyped in our lab.

To fill the *Create Event* module, the applications have to be able to decide what kind of event should be generated and define the attributes for each kind of event. For example, the Map application defines an event to pan over the map (containing the moved distance), another event to create a personal working area (containing the position and size of the created area), etc. and the Graffiti application defines an event to draw a line (containing the color and width of the line and the position of the next point), another one to open a color palette (containing the position where it should be opened), etc. In the case the same event can be applied to different objects, those attributes will contain a reference to the affected data. For example, when dragging a bulletin in the Virtual Bulletin Board application, the generated events will contain a reference number to the affected bulletin that is consistent among all the PPPs. In the *Execute Events* module, the applications apply the event depending on its type and attributes. For example, the Virtual Bulletin Board application loads a new bulletin from the data server and displays it when the event to load a bulletin is executed and the Emergency Management application displays a warning sign in the position contained in the event generated when the user does a gesture to mark a danger area.

For ‘roll back’, the applications have to implement a way to undo each of the events to ensure that the application returns to the exact same state as before the execution of the event. When no data is involved, this is simple and can be achieved by performing operations in the reverse order. For example, to roll back an event in the Bulletin Board application that highlighted a bulletin, the application just has to de-highlight it. However, if data is involved, we have to keep the original data before modification, since it may be impossible to get it from the modified data. Hence, all data needs to persist for a while even after it is removed from the application. The clearest example of this is when the event execution removes an object from the display and it has to be restored when undoing it (e.g., when closing a working area in the Map application).

#### 4 DISTRIBUTED REGISTRATION IMPROVEMENTS

In this section, we describe our new distributed registration technique in detail. We discuss related work in Section 4.1 followed by the innovations of our method in Section 4.2 and 4.3. Finally, we discuss the implementation details and results in Section 4.4.

##### 4.1 Related Work

[3] presents a distributed registration method when using a network of  $m \times n$  PPPs on a planar display. The method has three steps. (a) First, in a *neighbor detection* step a pattern made of 4 disjoint grids of  $5 \times 5$  blobs (Figure 12) is used to detect the neighbors of each PPP via their cameras. (b) Next, in the *configuration identification* step binary coded information is encoded in these grids of blobs and propagated in multiple rounds via communication using the cameras to decipher the

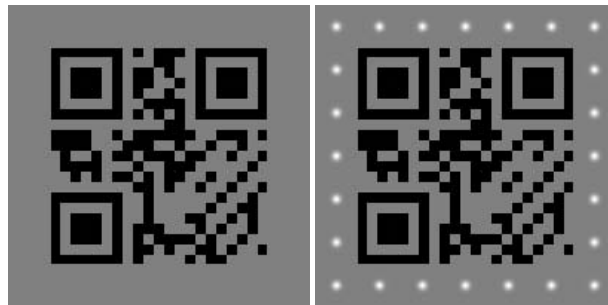


Fig. 11. Left: A standard version 1 QR Code. Right: The same QR Code augmented with our Gaussian blobs used in the registration phase.

total number of PPPs in the display, their configuration (total number of rows and columns) and the coordinates of the PPP in this display. (c) Finally, in the *registration* step, the pattern in the neighbor discovery step is used to register the display using a distributed homography tree technique (See video for the method in action).

This method has several shortcomings. First, the configuration identification step requires  $O(\ln(mn))$  rounds. In each round, each PPP projects its own coded pattern to communicate its belief about the display configuration (total rows and columns in the display and its own 2D coordinates in it), sees the projected coded pattern of the adjacent PPPs and changes its belief based on some rules. This continues iteratively across the PPPs until all converge to the correct configuration. However, multiple rounds of such camera based communication need considerable image processing and hence impacts performance. This also limits the scalability of the method across a larger number of PPPs. Finally, since colored patterns are used, the image processing is not robust when using commodity cameras with lower color fidelity.

Second, the homography tree technique [8] is inherently a centralized technique. A homography *graph* considers each projector as a node and places an edge between adjacent projectors  $i$  and  $j$ . Each of these edges is associated with the *local homography* between the adjacent projectors  $H_{i \rightarrow j}$  recovered using the cameras.  $H_{i \rightarrow j}$  is the transformation that takes pixels in the coordinate system of projector  $i$  to that of projector  $j$ . The *homography tree* technique identifies a projector  $P_R$  as the reference and finds for each projector  $P_i$ , a path to  $P_R$ . This results in a spanning tree in the homography graph, called *homography tree*, whose root is the reference projector (Figure 14). The homography that relates the projector  $P_i$  to  $P_R$  is given by the concatenation (multiplication) of the local homographies on the path from  $P_i$  to  $P_R$  in the homography tree.

The homography graph should ideally have some invariants: (a) the concatenation of homographies across a cycle in the homography graph should be identity; (b)  $H_{i \rightarrow j}$  should be the inverse of  $H_{j \rightarrow i}$ . But this is seldom the case due to several estimation errors and small projector non-linearities resulting in significant misregistrations, especially along the edges which are not part of the homography tree. To alleviate this problem, the homography tree is usually followed by a global alignment method via non-linear optimization techniques like bundle adjustments [8]. Since any global optimization technique is hard to achieve in a distributed manner, [3] avoids this step when designing a distributed homography tree method to achieve the registration. In this distributed variant, the tree is formed in a distributed manner by communicating the homography to the reference to neighbors who choose one of the many communicated homographies and multiply it with the local homography to the neighbor to find a PPP’s own homography to the reference. The process starts from the reference whose homography to itself is identity.

**Comparison with our work:** In our new registration technique, we introduce the following innovations. We use a single pattern based on specially augmented QR Codes to simultaneously achieve neighbor detection, configuration identification and registration. This allows us to eliminate the  $O(\ln(mn))$  rounds of camera based communication in the configuration identification round in [3] and achieve this in

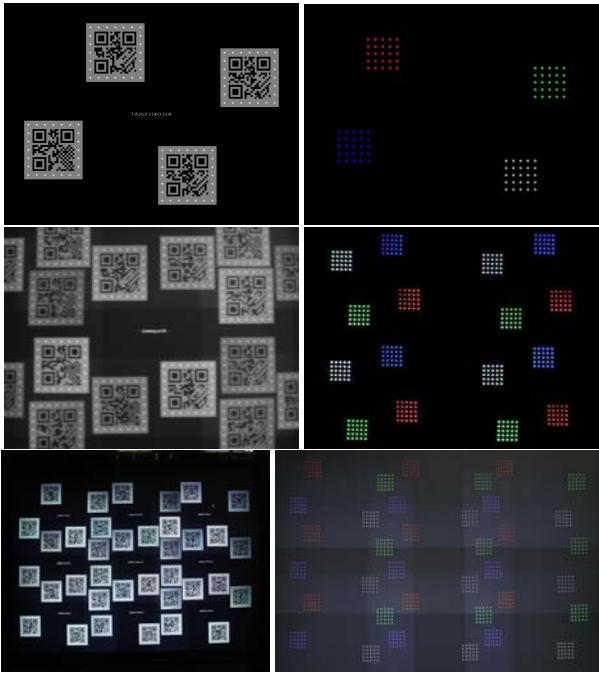


Fig. 12. We show the patterns for our work (left) compared against [3] (right). Top: The pattern projected by each PPP. Middle: The image seen by a PPP when all neighbors and itself are projecting their patterns. Bottom: The image of a  $3 \times 3$  array of nine projectors projecting their patterns.

$O(1)$  time. Also, this significantly reduces visual communication iterations and image processing time thus improving performance. This faster convergence is possible by supplementing the single camera-based communication with network based communications. Hence, as opposed to a multi-second registration of [3], our registration is almost instantaneous. However, the network overhead is still at most the same as [3]. Finally, since we use monochrome patterns instead of color patterns as in [3], our image processing is much more robust and allows inexpensive cameras with much lower color fidelity.

Second, we propose a new radially cascading registration method (Section 4.3) that is amenable to a distributed implementation and achieves much superior registration. This method can also be used for any centralized tiled display and performs better than the homography based global alignment technique proposed by [26]. The homography based global alignment seeks to find a global homography  $G_i$  for each projector  $P_i$  constrained by the fact that when considering any other projector  $P_j$ ,  $i \neq j$ ,  $G_j H_{i \rightarrow j}$  should provide  $G_i$ . Hence  $G_i = G_j H_{i \rightarrow j}$ . These provide a set of over-determined linear equations which when solved provides a transformation  $G_i$  for each projector that aligns all projectors with a common global coordinate system. This method tends to distribute the errors in the local homography across the entire display and hence, cannot remove pixel misregistration entirely. Unlike the homography tree where the misregistrations are concentrated at a few edges not included in the tree, the global alignment technique has small errors across all the overlap regions. However, the worst misregistration is significantly reduced from the homography tree technique. Our radially cascading registration method provides much superior results when compared with this global alignment technique and the distributed homography tree technique (Figure 14). The slight global distortion visible in the results of our method is due to small non-linearities in our commodity cameras.

## 4.2 Minimizing Camera Based Communication

QR (Quick Response) code is a 2D barcode which can embed a certain number of bits of data. The number of bits that can be encoded in the QR Code changes with its size. We encode in a QR Code the IP address and the port of the PPP, the location of the QR Code (2D coordinates of its top left corner in the projector coordinate system),

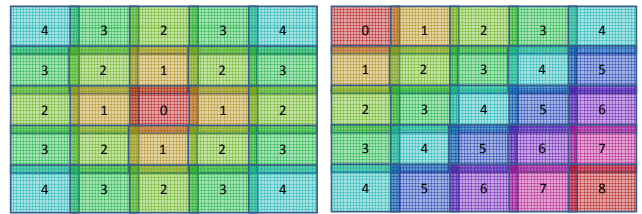


Fig. 13. Explanation of addition of PPPs to the pool of registered PPP for the radially cascading geometric registration method. Left: The middle PPP is the reference. Right: The top-left PPP is the reference.

and the size of the QR Code. The QR Code has a surrounding large 'quiet zone'. We augment the QR Code with some blobs embedded in this quiet zone which are used to detect correspondences across the projectors and the cameras for registration purposes (Figure 11). The blobs are embedded in a manner so that the quiet zone is retained after the binarization of the QR Code in the decoding phase. Hence, we can still use standard QR Code decoders without any change.

The pattern projected by each PPP as they are turned ON has four of these augmented QR Codes placed in an offset manner around the center such that each of the left, right, top and bottom neighbors can see at least one of these completely. The placement of the pattern and the overlaps required to assure no conflicts with neighboring projector are decided as in [3]. Since the camera of each PPP sees more than their own display, they see the neighbors' patterns along with their own. Figure 12 shows the pattern projected by each PPP, the image seen by each PPP and the display when all PPPs are projecting their own QR based patterns.

Each PPP detects the patterns from its neighbors to find out which of the left, right, bottom and top neighbors exist and creates the local connectivity graph of itself with its neighbors. Next, they decode this pattern to find out the exact IP-address of each of their neighbors. Finally they broadcast the location of each of their neighbors (left, right, top or bottom) along with the associated IP-address to all the PPPs in a single UDP packet. When each PPP receives this information, it augments its local connectivity graph using this information. Thus, each PPP now builds the connectivity graph for the entire display and associates a unique IP address with each node. Thus, it knows the total number of projectors in the display and their configuration. Following this, each node performs the same row-major naming convention to detect its own coordinates in the display. Unlike [3] which builds the connectivity over multiple rounds of camera based communication and broadcasts the IP addresses only following configuration identification, we achieve the same result with the same amount of network communication but with no computation overhead.

## 4.3 Radially Cascading Geometric Registration

Once the QR Codes are deciphered, each PPP  $i$  performs a blob detection in the quiet zone of the codes to decipher all the blobs. These blobs provide correspondences between the PPP's own projector and camera and hence allows it to recover the self-homography between its projector and camera. Next it detects the homographies with its adjacent projector  $j$  using the blobs detected in its QR Codes. Finally, it concatenates its self-homography with the homography of its camera with the adjacent projector to create the local homography  $H_{i \rightarrow j}$ .

The radially cascading geometric registration method starts from a reference projector which is considered as the only registered PPP initially. In each subsequent step  $S$ , PPPs with Manhattan distance  $S$  from the reference join the set of registered PPPs by aligning themselves with the PPPs who joined the registered display in step  $S - 1$ . The process stops when all the projectors belong to the set of registered projectors. Figure 13 shows the PPPs that join the set of registered PPPs for different steps  $S$  for two reference projectors in the display, the center one and the top left one respectively. Note that for a rectangular array of PPPs, all the PPPs that join the set of registered PPPs in step  $S$  share at most two boundaries with the set of registered PPPs.

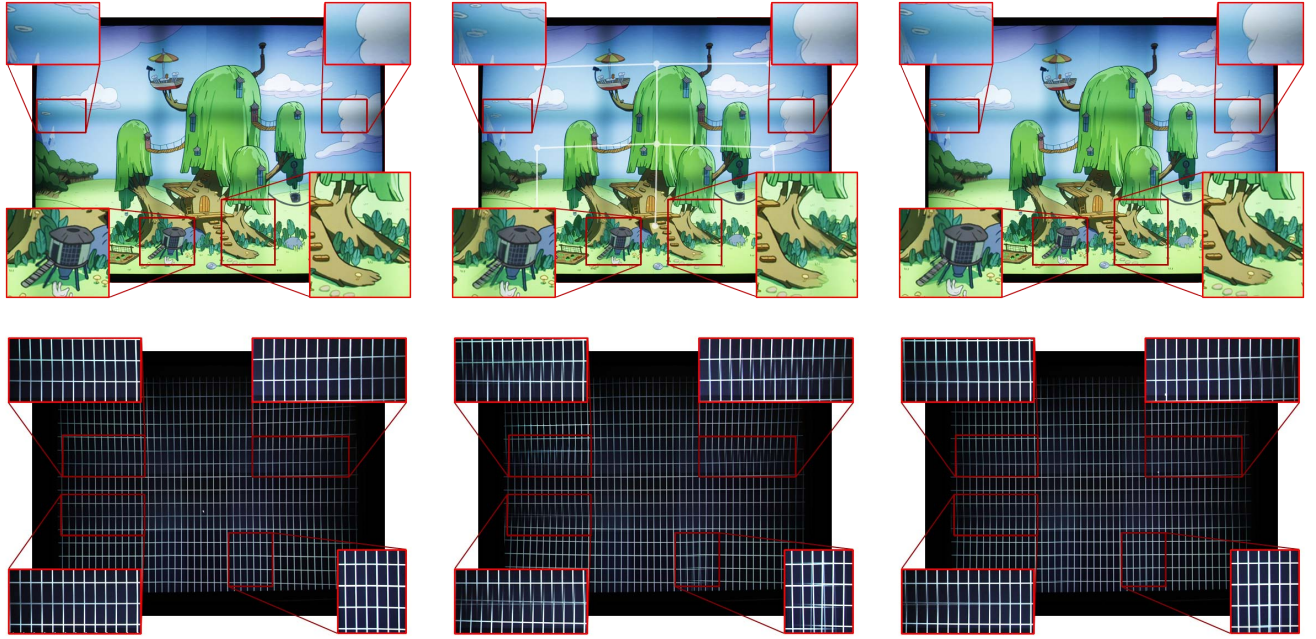


Fig. 14. Here we compare our method (top) with the homography tree (center) and the global optimization (bottom) technique on a  $3 \times 3$  array of nine projectors for two different images on left and right. The homography tree (center) is shown superimposed in white – note that the misregistrations are mostly along the links which are not used in the tree. Please zoom in to see details.

In step  $S$ , all the PPPs that joined the set of registered PPPs in step  $S - 1$  send their homography with respect to the reference PPP to all their neighbors whose Manhattan distance is  $S$  from the reference PPP. Thus, the PPP at a Manhattan distance  $S$  receives a homography from all the registered PPPs with Manhattan distance  $S - 1$  that share a boundary with it. Let us assume a PPP  $i$  in step  $S$  is receiving two such homographies from two neighbors  $j$  and  $k$ , denoted by  $G_j$  and  $G_k$  respectively. This PPP first converts the correspondences in the overlap with  $j$  to the coordinate system of the reference projector using  $G_j$ . Similarly, it converts the correspondences in the overlap with  $k$  using  $G_k$ . This gives PPP  $i$  a set of correspondences with the reference PPP via multiple possible paths through the registered projectors. PPP  $i$  then computes its own homography with the reference,  $G_i$ , using all these correspondences. This method can be summarized by a SPMD algorithm running on each PPP  $i$  as follows.

```

if center PPP {
    Send  $I$  to PPPs with  $d = 1$ ;
else {
     $d =$  Manhattan distance to center;
    forall neighbors  $j$  with  $\text{dist} = d - 1$  {
        Receive  $G_j$  from  $j$ ;
        Multiply all correspondences in overlap with  $j$  using  $G_j$ ;
    }
    Estimate  $G_i$  using all correspondences with all neighbors;
    Send  $G_i$  to all neighbors  $j$  with  $\text{dist} = d + 1$ ;
}

```

The total number of steps required for this algorithm to register will be the maximum Manhattan distance of any PPP from the reference. For a display of  $m \times n$  PPPs, if the top left PPP is chosen as the reference, the PPP with the maximum Manhattan distance of  $(m - 1) + (n - 1)$  from the reference is the bottom right. If the central PPP is chosen as the reference, the number of steps will be  $\frac{m-1}{2} + \frac{n-1}{2}$ . Figure 14 compares our superior registration with that achieved by the homography tree and the global alignment technique.

#### 4.4 Implementation and Practical Improvements

We demonstrate our distributed calibration system on a grid of nine PPPs in a  $3 \times 3$  array. Since this method does not rely on color patterns, each PPP is equipped with a monochrome VGA webcam. Due to noisy and low-resolution cameras we use the lowest resolution QR

Code ( $203 \times 203$  pixels embedding a  $29 \times 29$  grid) which can embed at most 152 bits of information. We embed a 32-bit IP address, 16-bit port, the top left corner of the code in the projector coordinate represented as two 16-bit numbers, and the 8-bit size of the code (using only 88 of the 152 bits). We embed 24 Gaussian blobs in the quiet zones of the QR Codes. Gaussian blobs allow us to robustly determine blob positions with subpixel accuracy and improve the quality of homography estimation. We use the ZBar barcode library to quickly decode QR Codes seen by our cameras as well as provide a rough estimate of the QR Code corners.

Since each PPP independently builds the complete graph of the display, our radially cascading geometric registration technique can be performed either via message passing (Section 4.3) or independently on each PPP after it forms the adjacency graph for the entire display reducing the network overhead significantly. Hence, we include the PPP's local homographies with all its neighbors in the message broadcast during the configuration identification stage. Each PPP  $i$  sees only one of the four QR Codes for its neighbor  $j$  completely. The four corners of the QR Code are used to estimate a coarse local homography, which is used to initialize the blob detector. The detected blob positions in this QR Code are then used to produce a more accurate local homography estimate. For each projector  $i$ , we find the two homographies with each of its neighboring PPP  $j$ ,  $H_{i \rightarrow j}^i$  and  $H_{j \rightarrow i}^i$ , where the superscript denotes the PPP which computes these homographies. Note that the same homographies can be computed by  $j$  as well,  $H_{i \rightarrow j}^j$  and  $H_{j \rightarrow i}^j$  respectively. Ideally,  $H_{i \rightarrow j}^i = H_{i \rightarrow j}^j$ . But, due to the distribution of the blobs only around the QR Codes instead of the entire overlap, especially in the face of mild non-linearities in either the projector or camera, this results in a situation where there can be a slight deviation from this constraint. So, we design a method to compute a more accurate homography  $H_{i \rightarrow j}$  from  $i$  to  $j$ , as follows. We generate a set of points uniformly over the overlap of  $i$  with  $j$  and find their corresponding points in  $j$  using  $H_{i \rightarrow j}^i$ . Similarly, we generate a set of points uniformly over the overlap of  $j$  with  $i$  and find their corresponding points in  $i$  using  $H_{j \rightarrow i}^j$ . Then we use this combined set of correspondences to generate a more robust estimate of  $H_{i \rightarrow j}$  using a linear least squares technique. In our system this computation of the radially

cascading registration on each PPP did not exceed the network latency and time to sequentially compute and propagate this information.

To achieve photometric seamlessness, we use the recovered homography of each PPP  $i$  with its neighbor  $j$  to detect the exact shape of the overlap. Finally, each PPP independently applies a blending function in each of its overlap.

## 5 CONCLUSION

In conclusion, we present the first distributed interaction paradigm for large rear projected display walls. We demonstrate the scalability of our method to multiple displays, users, applications and interaction modalities by showing a working prototype of multiple 2D applications using both gestures and laser based interaction modality. We also propose a new distributed registration technique that is more accurate and efficient than prior methods. This technique is deterministic, can be easily implemented for a centralized system, and does not involve time-consuming global optimizations.

In the future we would like to extend our work for front projection systems where occlusion is an issue. We would like to extend it beyond 2D applications. We believe that our paradigm can extend to 3D applications, however we would like to explore the different issues in detail. We would also explore designing distributed versions of the more rigorous photometric calibration methods like [22, 28].

## REFERENCES

- [1] S. C. Ahn, T.-S. Lee, I.-J. Kim, Y.-M. Kwon, and H.-G. Kim. Large display interaction using video avatar and hand gesture recognition. *International Conference on Image Analysis and Recognition*, pages 261–268, 2004.
- [2] E. Bhasker, R. Juang, and A. Majumder. Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. *IEEE TVCG*, pages 1368–1375, 2007.
- [3] E. Bhasker, P. Sinha, and A. Majumder. Asynchronous distributed calibration for scalable reconfigurable multi-projector displays. *IEEE Transactions on Visualization and Computer Graphics (Visualization)*, 2006.
- [4] M. S. Brown and B. W. Seales. A practical and flexible tiled display system. *IEEE Pacific Graphics*, 2002.
- [5] X. Cao and R. Balakrishnan. Visionwand: Interaction techniques for large displays using a passive wand tracked in 3d. *ACM symposium on User interface software and technology*, pages 193–202, 2003.
- [6] X. Cao and R. Balakrishnan. Interacting with dynamically defined information spaces using a handheld projector and a pen. *ACM Symposium on User Interface Software and Technology*, pages 225–234, 2006.
- [7] X. Cao, C. Forlines, and R. Balakrishnan. Multi-user interaction using handheld projectors. *ACM Symposium on User Interface Software and Technology*, pages 43–52, 2007.
- [8] H. Chen, R. Sukthankar, G. Wallace, and K. Li. Scalable alignment of large-format multi-projector displays using camera homography trees. *Proc. of IEEE Vis.*, pages 339–346, 2002.
- [9] J. Davis and X. Chen. Lumipoint: multi-user laser-based interaction on large tiled displays. *Displays*, 23:205–211, 2002.
- [10] P. Dietz and D. Leigh. Diamondtouch: A multi-user touch technology. *ACM symposium on User interface software and technology*, pages 219–226, 2001.
- [11] T. Grossman, R. Balakrishnan, G. Kurtenbach, G. W. Fitzmaurice, A. Khan, and W. Buxton. Interaction techniques for 3d modeling on large displays. *ACM Symposium on Interactive 3D Graphics*, pages 17–23, 2001.
- [12] J. Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. *ACM Symposium on User Interface Software and Technology*, pages 115–118, 2005.
- [13] G. Humphreys, I. Buck, M. Eldridge, and P. Hanrahan. Distributed rendering for scalable displays. *Proceedings of IEEE Supercomputing*, pages 129–140, 2000.
- [14] G. Humphreys, M. Eldridge, I. Buck, G. Stoll, M. Everett, and P. Hanrahan. Wiregl: A scalable graphics system for clusters. *Proceedings of ACM SIGGRAPH*, pages 129–140, 2001.
- [15] G. Humphreys, M. Houston, R. Ng, R. Frank, S. Ahem, P. Kirchner, and J. Klosowski. Chromium: A stream processing framework for interactive rendering on clusters. *ACM Transactions on Graphics (SIGGRAPH)*, pages 693–702, 2002.
- [16] S. Izadi, H. Brignull, T. Rodden, Y. Rogers, and M. Underwood. Dynamo: a public interactive surface supporting the cooperative sharing and exchange of media. In *ACM symposium on User interface software and technology*, pages 159–168, 2003.
- [17] H. Jiang, E. Ofek, N. Moraveji, and Y. Shi. Direct pointer: direct manipulation for large-display interaction using handheld cameras. In *SIGCHI conference on Human Factors in computing systems*, pages 1107–1110, 2006.
- [18] A. Khan, G. Fitzmaurice, D. Almeida, N. Burtnyk, and G. Kurtenbach. A remote control interface for large displays. *ACM symposium on User interface software and technology*, pages 127–136, 2004.
- [19] J. Kim, J. Park, H. Kim, and C. Lee. Hci(human computer interaction) using multi-touch tabletop display. *Pacific Rim Conference on Communications, Computers and Signal Processing*, pages 391–394, 2007.
- [20] C. Krumbholz, J. Leigh, A. Johnson, L. Renambot, and R. Kooima. Lambda table: High resolution tiled display table for interacting with large visualizations. *Workshop for Advanced Collaborative Environments (WACE)*, 2005.
- [21] A. Majumder and R. Stevens. Color nonuniformity in projection-based displays: Analysis and solutions. *IEEE Transactions on Vis and Computer Graphics*, 10(2):177–188, March–April 2003.
- [22] A. Majumder and R. Stevens. Perceptual photometric seamlessness in tiled projection-based displays. *ACM TOG*, pages 111–134, 2005.
- [23] S. Malik, A. Ranjan, and R. Balakrishnan. Interacting with large displays from a distance with vision-tracked multi-finger gestural input. *ACM symposium on User interface software and technology*, pages 43–52, 2005.
- [24] D. R. Olsen and T. Nielsen. Laser pointer interaction. *ACM conference on human factors in computing systems*, pages 17–22, 2006.
- [25] R. Raskar. Immersive planar displays using roughly aligned projectors. In *Proc. of IEEE VR*, pages 109–115, 2000.
- [26] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM Transaction on Graphics (SIGGRAPH)*, pages 809–818, 2003.
- [27] J. Rekimoto. Smartskin: An infrastructure for freehand manipulation on interactive surfaces. *ACM conference on human factors in computing systems*, 2002.
- [28] B. Sajadi, M. Lazarov, A. Majumder, and M. Gopi. Color seamlessness in multi-projector displays using constrained gamut morphing. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, pages 1317–1326, 2009.
- [29] C. Shen, F. D. Vernier, C. Forlines, and M. Ringel. Diamondspin: an extensible toolkit for around-the-table interaction. In *SIGCHI conference on Human factors in computing systems*, pages 167–174, 2004.
- [30] D. Stødle, J. M. Bjørndalen, and O. J. Anshus. A system for hybrid vision- and sound-based interaction with distal and proximal targets on wall-sized, high-resolution tiled displays. *IEEE International Workshop on Human Computer Interaction*, pages 59–68, 2007.
- [31] D. Stødle, Tor-Magne, S. Hagen, J. M. Bjørndalen, and O. J. Anshus. Gesture-based, touch-free multi-user gaming on wall-sized, high-resolution tiled displays. *4th International Symposium on Pervasive Gaming Applications, PerGames*, pages 75–83, 2007.
- [32] D. Stødle, O. Troyanskaya, K. Li, and O. J. Anshus. Device-free interaction spaces. *IEEE Symposium on 3D User Interfaces*, pages 39–42, 2009.
- [33] B. Ullmer and H. Ishii. The metadesk: Models and prototypes for tangible user interfaces. *ACM symposium on User interface software and technology*, pages 223–232, 1997.
- [34] D. Vogel and R. Balakrishnan. Distant freehand pointing and clicking on very large high resolution displays. *ACM symposium on User interface software and technology*, pages 33–42, 2005.
- [35] A. D. Wilson. Touchlight: An imaging touch screen and display for gesture-based interaction. *International Conference on Multimodal Interfaces*, pages 69–76, 2004.
- [36] C.-O. Wong, D. Kyoung, and K. Jung. Adaptive context aware attentive interaction in large tiled display. *IEEE International Workshop on Human Computer Interaction*, pages 1016–1025, 2007.
- [37] M. Wu and R. Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. *ACM Symposium on User Interface Software and Technology*, pages 193–202, 2003.
- [38] Y. Wu and T. S. Huang. Vision-based gesture recognition: A review. In *GW '99: Proceedings of the International Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction*, pages 103–115, 1999.



# Markerless View-Independent Registration of Multiple Distorted Projectors on Extruded Surfaces Using Uncalibrated Camera

Behzad Sajadi and Aditi Majumder, *Member, IEEE*

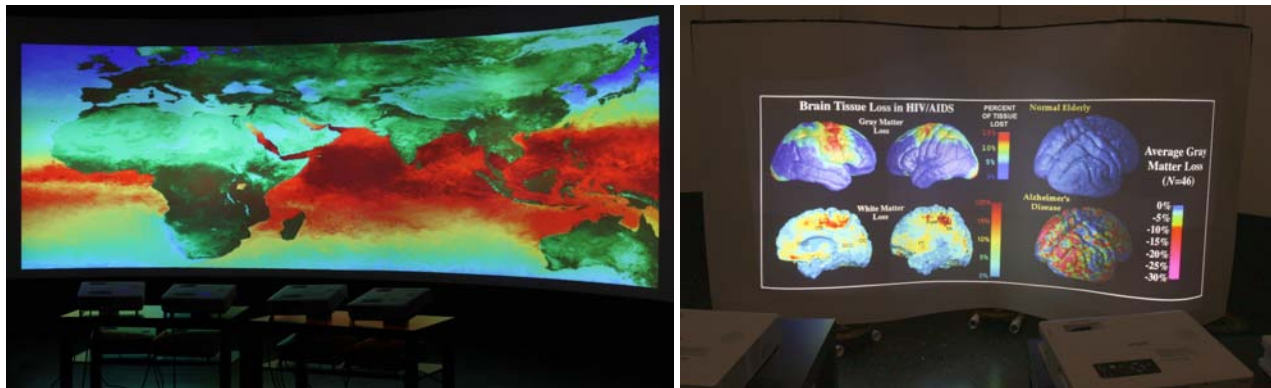


Fig. 1. Left: A  $2 \times 4$  array of eight projectors on a cylindrical display surface showing a weather map visualization; Right: A  $2 \times 3$  array of six projectors on a more general extruded surface showing a medical visualization.

**Abstract**— In this paper, we present the first algorithm to geometrically register multiple projectors in a view-independent manner (i.e. wallpapered) on a common type of curved surface, vertically extruded surface, using an uncalibrated camera without attaching any obtrusive markers to the display screen. Further, it can also tolerate large non-linear geometric distortions in the projectors as is common when mounting short throw lenses to allow a compact set-up. Our registration achieves sub-pixel accuracy on a large number of different vertically extruded surfaces and the image correction to achieve this registration can be run in real time on the GPU. This simple markerless registration has the potential to have a large impact on easy set-up and maintenance of large curved multi-projector displays, common for visualization, edutainment, training and simulation applications.

**Index Terms**—Registration, Calibration, Multi-Projector Displays, Tiled Displays

## 1 INTRODUCTION

Tiled multi-projector displays on curved screens (e.g. cylindrical or spherical screens) are becoming more popular for visualization, education, entertainment, training and simulation applications. Their appeal lies in the greater sense of immersion and presence they can create, and at times, the superior aesthetics they provide. Displays are tools used by these application users who are not expected to be experts in setting them up or maintaining them. Unfortunately, most registration algorithms designed for curved displays expect them to be one. Registering multiple projectors on such a display has been a challenge, primarily due to the fact that recovering the 3D shape of the display quickly almost always requires attaching fiducials (physical markers) on the display screen for providing robust correspondence between the screen and the camera, which is especially obtrusive. Using structured light patterns to achieve the same results is a time consuming process. Finally, both these methods are complex requiring a complex camera calibration, all of which are too difficult for a layman user to execute in a successful manner.

We seek a simple procedure to register multiple projectors on a curved display that can be used even by a layman user like a doctor in a medical facility, teacher in a school or a worker in a theme park. We observe that most of the time, geometrically simple surfaces, like partial cylinders (e.g. pillars or surround immersive environments), are used as the display screen. So we impose two simple priors on the screen. First, the screen is a vertically extruded surface - a surface made by sweeping a 2D curve, called the profile curve, along a direc-

tion perpendicular to it. This covers a large number of shapes that can be built by soft folding of a rectangular sheet in one direction (Figure 2) – a cylinder is an important special case. Second, we assume the aspect ratio of the planar rectangle formed by the four corners of the extruded surface is known. Such a measurement is easy to provide, even for a layman user. Having these priors allows us to prevent the use of any markers on the display screen and still recover the shape of the display using a single image from an uncalibrated camera. This allows easy set-up and maintenance of such multi-projector displays by the user, even in the face of changes in the display surface or projector configurations and severe non-linearities.

### 1.1 Main Contributions

In this paper we present a new efficient algorithm to register images from multiple projectors on a vertically extruded surface. Using the priors of an extruded shape and the known aspect ratio, we use a single image of the display surface from an *uncalibrated* camera to recover both the camera parameters and the 3D shape of the surface. The display surface is then arc length parameterized in both dimensions. Then we capture a few images of patterns from the projectors to relate the projector coordinates with the display surface points, and represent this relationship using a rational Bezier patch. This relationship is then used to segment the appropriate parts of the image for each projector to register them and create a seamlessly wall-papered projection on the display screen.

To the best of our knowledge, this is the first work that can achieve the following many desirable qualities of geometric registration on these non-planar surfaces. All prior work can only address one or a few of these desired qualities, however our work addresses all simultaneously for the first time.

1. *Markerless*: Using some simple priors on the display surface,

• Behzad Sajadi and Aditi Majumder are in Computer Science Department of University of California, Irvine, E-mail: {bsajadi,majumder}@uci.edu.

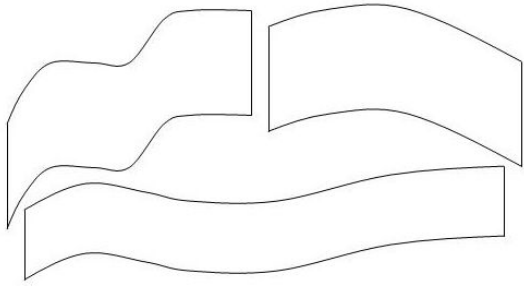


Fig. 2. Some examples of vertically extruded surfaces.

we can register images from multiple projectors on a vertically extruded screen without using any correspondence between the 3D display and the camera.

2. *Uncalibrated Camera:* We show that with some simplifying assumptions on the intrinsic parameters of the camera, we can achieve this registration using an uncalibrated camera.
3. *View Independent:* Unlike registering with respect to a camera where the registration is correct from a single sweet spot, we paste the image like a wallpaper on the display. A wallpapered image does not look perspectively correct from any single viewpoint. Yet, human observers can easily correct for the existing distortions irrespective of their viewpoint since we are used to seeing wallpapered images commonly. Hence, by wallpapering we assure that multiple viewers can observe our display at the same time, making our method view independent.
4. *Allowing Low-Resolution Sensor:* Since we use a rational Bezier patch to relate the projector to the display parameters, we can achieve a good fit even if we sample the function sparsely. As a result, we can use a relatively low-resolution camera (e.g. VGA camera) to register a much higher resolution display.
5. *Allowing Non-Linearities:* Further, since our registration depends on a 2D parametrization of the display generated from the recovered 3D surface rather than auto-calibrating projectors on the 3D surface itself, we can handle severe non-linearities in the projectors (like radial distortion). Thus, we can allow a compact set-up with inexpensive short-throw lenses mounted on the projectors that usually have non-linear distortions. Non-linear distortions have been addressed when using planar displays [4]. When using non-planar displays, non-linear lens distortions has been addressed in a limited manner by using a high-resolution camera to sample the function relating the projector to the display parameters densely [18, 7]. However, we can correct such distortions even with a low-resolution camera using a sparse sampling of the function relating the projector and display parameters.
6. *Allowing Display Imprecision:* The 2D parametrization additionally assures that a moderate deviation of the screen from being a perfectly extruded surface will not affect the accuracy of the geometric registration. Thus, we can handle manufacturing imprecision in the vertically extruded display surface.
7. *Accuracy:* Our method assures subpixel accuracy even in the presence of projector non-linearities.
8. *Efficiency:* Finally, our method can be run in real-time on the GPU making it ideal for interactive video applications.

## 2 PREVIOUS WORK

Camera-based geometric registration of multi-projector displays can be either view-dependent or view-independent. View-dependent registration yields an image on the display that is correct from only one sweet view-point, usually the view of the camera. Deviation of the viewer from this location shows view-dependent distortions. Hence, view-dependent registration is usually appropriate for static single-user applications. On the other hand, view-independent registration pastes or wall-papers the images on the display surface. Since wall-papering is a common way to accommodate multiple viewers, such registration can cater to more than one viewer easily. Such a registration not only requires registering the projectors in a common camera frame but also the (conformal) parameterization of the shape of the display surface.

There has been a large amount of work on registering images on planar multi-projector displays in a view-independent fashion using linear homographies enabled by the planar screen [11, 5, 14, 10, 18, 19, 9, 3]. Such registration can be achieved in the presence of projector non-linearities using rational Bezier patches [4].

View-dependent registration on a non-planar display has been achieved by using special 3D fiducials and a large number of structured light patterns for a complete device (camera and projector) calibration and 3D reconstruction of the display surfaces, which are then used to achieve the registration [13]. Aliaga et al. in [2, 1] use a similar 3D reconstruction method to achieve a similar registration on complex 3D shapes, but without using any physical fiducials. To constrain the system sufficiently, this method uses completely superimposed projectors and validates results from photometric and geometric stereo, resulting in a self-calibrating system. Raskar et al. in [12] use a stereo camera pair to reconstruct special non-planar surfaces called quadric surfaces (spheres, cylinders, ellipsoids and paraboloids) and propose conformal mapping and quadric transfer to minimize pixel stretching of the projected images in a view-dependent registration.

More recently, there have been works on view-independent registration for the special case of a cylindrical surface rather than a general non-planar surface [7, 17]. Using the fact that cylindrical surfaces are developable, they have achieved a ‘wall-paper’ registration on such surfaces. However, these works do not recover the shape of the surface in 3D, but attempt to find its 2D parametrization in the camera space. Hence, they need precise correspondences between the physical display and the observing camera. To achieve this, a precisely calibrated physical pattern is pasted along the top and bottom curves of the cylinder. Using these correspondences, a piecewise linear 2D parametrization of the display is computed and linked to a piecewise linear representation of the projector coordinates via the camera that observes both. This allows segmenting the appropriate parts of the image for each projector using linear/non-linear interpolations to create a wall-papered display. However, to avoid fiducials at a high spatial density, the physical pattern only samples the rims of the display. This insufficient sampling results in distortions or stretching, especially towards the middle of the display surface.

### 2.1 Comparison of Our Method

Unlike earlier methods for view-independent registration of cylindrical displays that assume a piecewise linear representation of the surface to parametrize it in the 2D camera space [7, 17], we recover the 3D geometry of the display. Hence, we can parametrize the display directly in 3D rather than in the camera image space, resulting in a geometric registration of the projected imagery without any stretching or distortions. Use of a perspective projection invariant function, e.g. a rational Bezier function, for interpolation instead of a simple linear interpolation allows us to maintain registration in the presence of severe projector distortions and considerable imprecision in manufacturing of the extruded surface. Further, as shown in [4], unlike a piecewise linear function, a rational Bezier function can be interpolated accurately even from a sparse set of samples. This allows our method to use a low resolution camera while registering a much higher resolution display. Unlike earlier methods for non-planar displays that recover the 3D shape using complex stereo or structured light based procedures

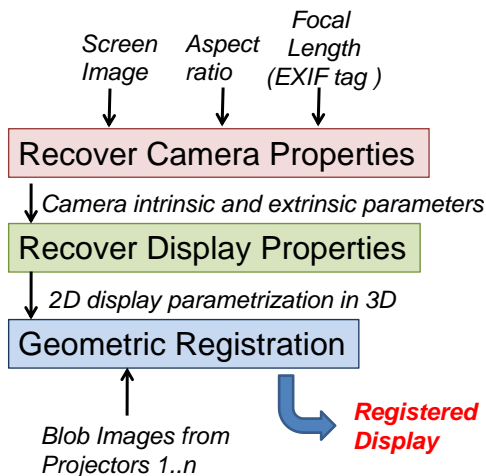


Fig. 3. The flowchart of our algorithm.

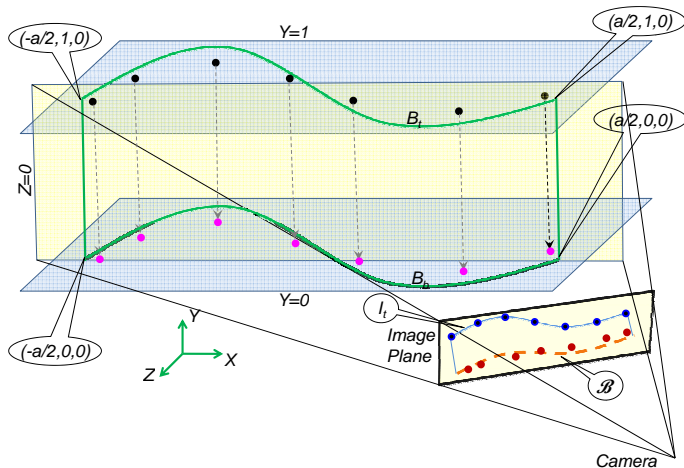


Fig. 4. This illustrates the world coordinate systems and the display surface and camera set-up with respect to it. The sampled points on the 2D top curve in the camera (blue) is reprojected in 3D to estimate the 3D top curve (black), and translated down to estimate of the 3D bottom curve (purple), and finally projected back on the camera (red). The distance between these points and the orange curve on the camera image plane,  $\mathcal{B}$ , is minimized in the extrusion based optimization step.



Fig. 5. The image used for calibrating a cylindrical display with  $2 \times 4$  array of eight projectors. This image of the screen, with no projectors turned on, used for recovering the camera and the display properties.

[13, 12, 2, 1], we simplify the process using a single image from a single camera position by imposing the prior that the surface is vertically extruded. Finally, we avoid calibrating the camera to recover the 3D shape by using some simplifying assumptions on the intrinsic parameters of the camera and the aspect ratio of the display surface that is provided by the user.

### 3 ALGORITHM

Let the display surface, the image planes of the camera and the projector be parametrized by  $(s, t)$ ,  $(u, v)$  and  $(x, y)$  respectively. We denote the 3D coordinates of the point at  $(s, t)$  in the display by  $(X(s, t), Y(s, t), Z(s, t))$ . Since the display is a vertically extruded surface, the four corners of the display lie on a planar rectangle, whose aspect ratio  $a$  is known. We define the world 3D coordinate with  $Z$  axis perpendicular to this plane and  $X$  and  $Y$  defined as the two orthogonal basis of this planar rectangle. We also consider this planar rectangle to be at  $Z = 0$ . Considering these 3D coordinates, the top and bottom curves of the surface lie respectively on  $Y = 1$  and  $Y = 0$  planes in 3D. Hence,  $Y(s, 0) = 0$  and  $Y(s, 1) = 1$ . Further, these two curves are identical except for a translation in the  $Y$  direction. Therefore,  $\forall s, (X(s, 0), Z(s, 0)) = (X(s, 1), Z(s, 1))$ . This is illustrated in Figure 4. We assume that our camera is a linear device without any radial distortion. Note that a distorted camera will still provide good registration but the wallpapering will be imperfect. Limitations that will be imposed by this assumption are discussed in further detail in Section 5.3. However, our projectors need not be linear devices.

A view-independent geometric registration essentially requires us to define a function from  $(x, y)$  projector coordinates to the  $(s, t)$  display coordinates. Our method follows three steps to achieve this (Figure 3). First we use a single image of the display from the uncalibrated camera and the known aspect ratio of the display to *recover the camera properties* (intrinsic and extrinsic parameters) using a non-linear optimization. Using the estimated camera parameters, we next *recover the 3D shape of the display*. Then, we use the profile curves of the vertically extruded surface to define 2D parametrization of the display surface based on the arc length of the profile curves flanking the display. After calibrating the camera and reconstructing the display, in the next phase, we capture an image of a blob-based pattern from each projector and use these to find samples of the mapping from the projector  $(x, y)$  to the display  $(s, t)$ . Then we approximate this mapping from these samples by fitting a rational Bezier patch to the correspondences. Assuming that an image pasted on the display results in the image coordinates being identical to the display coordinates  $(s, t)$ , this automatically achieves the *geometric registration* by defining the part of the image to be projected by each projector so that the resulting display is seamlessly wallpapered. Each of the above four steps are described in detail in the following sections.

#### 3.1 Recovering Camera Properties

In this step, we use a single image of the display surface (Figure 5) to recover the intrinsic and extrinsic parameters of the observing uncalibrated camera using a non-linear optimization. A large number of image formats like jpg or tiff store EXIF tags for images which provide some of the camera parameters used during the capture. One of these is the focal length of the camera, the critical component for the intrinsic parameter matrix of the camera. As in [16], we use this focal length to initialize the intrinsic parameter matrix in our non-linear optimization. To convert the focal length to the unit of pixels, we divide resolution of the camera by the CCD sensor size and multiply it with the focal length specified in the EXIF tags. The sensor size of the camera is available in its specifications.

In most cameras today, it is common to have the principal center at the center of the image, no skew between the image axes and square pixels. Using these assumptions, we express the intrinsic parameter matrix of a camera,  $K_c$ , as

$$K_c = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

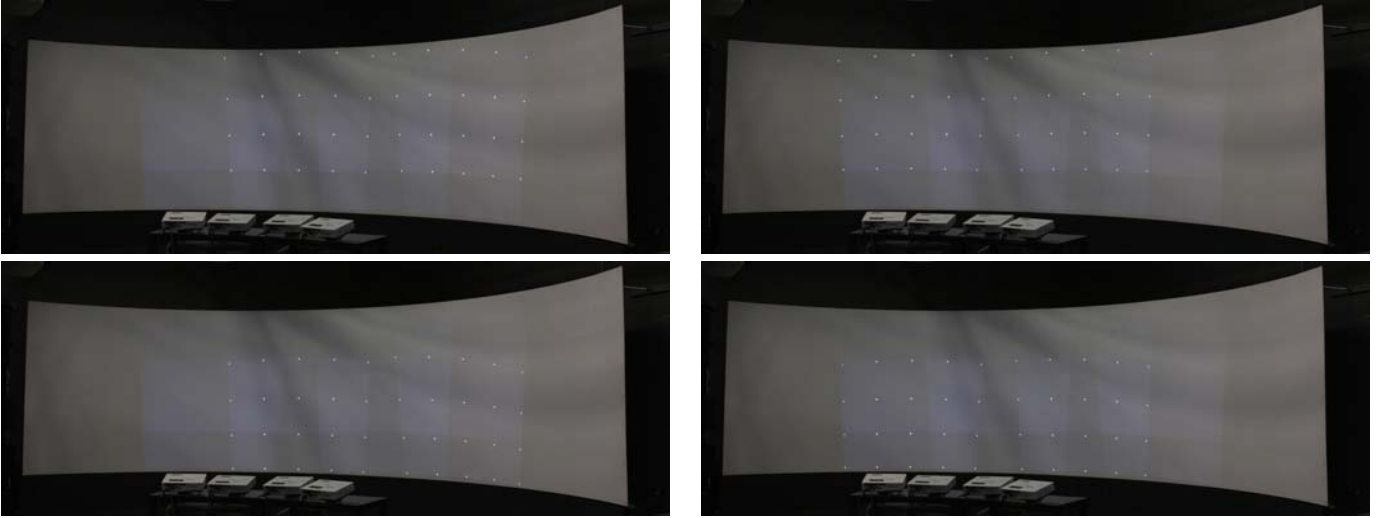


Fig. 6. The images used for calibrating a cylindrical display with  $2 \times 4$  array of eight projectors. Four different images of blobs from non-overlapping projectors used to find the projector to display correspondences. Please zoom in to the image to see blobs.

The camera calibration matrix that relates the 3D coordinates with the 2D camera image coordinates  $(u, v)$  is given by  $M = K_c[R|T]$  where  $R$  and  $T$  are the rotation and translation of the camera with respect to the world coordinate system. In this step, we use the initial estimate of  $f$  and the aspect ratio  $a$  as input and use a non-linear optimization to estimate seven parameters of the camera calibration matrix – these include the focal length  $f$ , the three rotations that comprise  $R$  and the three coordinates of the center of projection of the camera  $T$ .

Our non-linear optimization has two phases. In the first phase, *plane based optimization* (Section 3.1.1), the seven camera parameters are estimated using just the projection of the corners of the display surface on the camera image. These estimates are used to initialize the *extrusion based optimization* (Section 3.1.2) with a more expensive error function to refine the camera parameters.

### 3.1.1 Plane Based Optimization

We estimate the seven parameters in this step based on the image of the plane formed by the four corners of the screen whose 3D coordinates are given by:  $(-\frac{a}{2}, 1, 0)$ ,  $(\frac{a}{2}, 1, 0)$ ,  $(\frac{a}{2}, 0, 0)$  and  $(-\frac{a}{2}, 0, 0)$ . Consequently, the  $(u, v)$  coordinates in the camera of any 3D point  $(X(s, t), Y(s, t), Z(s, t))$  on the display are given by

$$(uw, vw, w)^T = M(X(s, t), Y(s, t), Z(s, t), 1)^T \quad (2)$$

where  $(uw, vw, w)^T$  are the 3D homogeneous coordinates corresponding to the camera coordinate  $(u, v)$  and  $M$  is the  $3 \times 4$  camera calibration matrix defined by the seven camera parameters. We estimate the seven camera parameters in this step by using a non-linear optimization method that minimizes the reprojection error  $E_r$ , (i.e. the sum of the distances of the projection of these 3D corners on the camera image plane from the detected corners). We initialize the angle of rotations about the X, Y and Z axes that comprise  $R$  to be zero and  $T$  to be roughly at the center of the planar rectangle formed by the four corners of the display at a depth of a similar order of magnitude as the size of the display i.e.  $(0, 0, a)$ .

### 3.1.2 Extrusion Based Optimization

The seven estimated camera parameters in the plane based optimization are used to initialize the extrusion based optimization that attempts to refine these parameters further. This also uses a non-linear optimization method that minimizes the error  $E = w_r E_r + w_c E_c$ , where  $E_r$  is the error function from the plane based optimization step, and  $E_c$  is an error function based on the reprojection error in the similarity of the flanking curves of the display as described next, and  $w_r$  and  $w_c$  are the weights to combine them.

The vertically extruded display surface is constrained by the fact that the points on the top curve of the vertically extruded surface when translated by  $Y = -1$  should lie on the bottom curve. We use the deviation from this constraint to define  $E_c$ . Let the image of the top and bottom boundaries of the vertically extruded display in the camera be  $I_t$  and  $I_b$  respectively. We first use image processing to segment the image and sample the curves  $I_t$  and  $I_b$ . We fit a parametric curve to the samples on  $I_b$ . Let us denote it with  $\mathcal{B}$ . We use the current estimate of  $M$  to reproject  $I_t$  in 3D. This is achieved by ray casting through the sampled points on  $I_t$  and intersecting it with  $Y = 1$  plane. The 3D curve thus obtained is  $B_t$ . Then we translate the samples on  $B_t$  along  $Y$  direction by 1 to get the samples on the 3D bottom curve,  $B_b$ . Then we project these samples back on to the camera using  $M$ , denoted by  $M(B_b)$ . Sum of the square of the distances of these samples from the curve  $\mathcal{B}$  provides the reprojection error of the estimated bottom curve from the detected bottom curve. In case of perfect estimation, this error should be zero. Hence, we seek to minimize  $E_c$  in addition to  $E_r$ .

To solve both the plane and extrusion based optimizations, we use standard gradient descent methods. To assure faster convergence we (a) apply a pre-conditioning to the variables so that the range of values that can be assigned to them is normalized; and (b) use decaying step size.

## 3.2 Recovering 3D Display Parameters

After convergence of the optimization process, we use the estimated  $M$  to reproject samples on  $I_t$  and  $I_b$  in 3D and intersect them with  $Y = 1$  and  $Y = 0$  planes to find  $B_t$  and  $B_b$  respectively. Due to accumulated errors,  $B_t$  and  $B_b$  may not be identical. So, we translate both the curves on  $Y = 0$  plane and find their average to define  $B_b$ . This is then translated to  $Y = 1$  to define  $B_t$ . This assures that both  $B_t$  and  $B_b$  are identical except for a translation along Y. We use a polynomial curve fitting to find a parametric representation of  $B_t$  and  $B_b$ .

Next, we seek a 2D parametrization of the display  $D$  with  $(s, t)$ . The profile curve  $B_b$  on the XZ plane is arc length parametrized using the parameter  $s$ . Considering the 3D point  $(X, Y, Z)$  on the display surface,  $X = X(s, t) = X(s)$  and  $Z = Z(s, t) = Z(s)$ . Since extrusion is along the Y direction,  $Y = Y(s, t) = t$ . Using the vertical extrusion assumption we can conclude that  $X$  and  $Z$  are independent of  $t$  and  $Y$  is independent of  $s$ .

## 3.3 Geometric Registration

Geometric registration entails defining, for each projector, a function that maps the projector coordinates  $(x, y)$  to the display coordinates

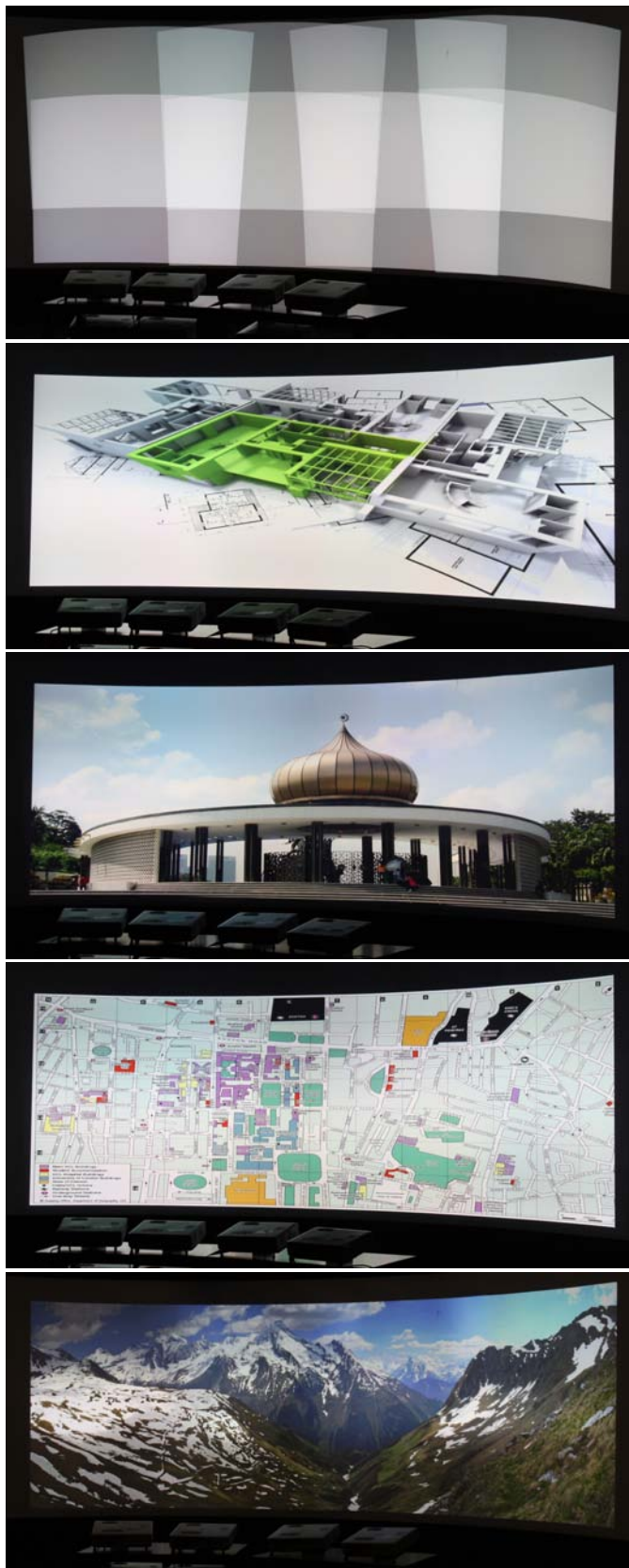


Fig. 7. Our geometric registration on a cylindrical display using  $2 \times 4$  array of 8 projectors. Please zoom in to see registration.

$(s, t)$  via the camera coordinates  $(u, v)$ . Mathematically,

$$(s, t) = M_{D \leftarrow C}(M_{C \leftarrow P}(x, y)) \quad (3)$$

where  $M_{C \leftarrow P}$  maps the  $(x, y)$  to  $(u, v)$  and  $M_{D \leftarrow C}$  maps  $(u, v)$  to  $(s, t)$ . As in [4], we use a rational Bezier patch to define  $M_{C \leftarrow P}$ . To find  $M_{C \leftarrow P}$  we project a number of blobs and use the camera to capture them (Figure 6). The center of these blobs are known in the projector coordinate space  $(x, y)$ . When these centers are detected in the camera space  $(u, v)$ , they provide direct correspondences between  $(x, y)$  and  $(u, v)$ . We fit a rational Bezier patch using a non-linear least squares fitting solved efficiently by the Levenberg-Marquardt gradient descent optimization technique. To compute  $M_{D \leftarrow C}$ , we do the following. For every mapped  $(u, v)$  coordinate in the camera, we cast a ray through this point and find the point of intersection with the recovered 3D display. Then we find the 2D parameter corresponding to this 3D point.

Using a rational Bezier for representing  $M_{C \leftarrow P}$  provides two important capabilities to our algorithm, as in [4]. First, we can achieve accurate registration in the face of severe non-linear distortions like lens distortion (barrel, pin-cushion, tangetial and so on). Such distortions are common when using inexpensive short throw lenses on projectors to allow a compact setup. The rational Bezier in this case can represent the non-linearities both due to the curved nature of the display and projector non-linearities. Second, unlike previous method [18] that uses a piecewise linear function to represent  $M_{C \leftarrow P}$  and hence requires a dense sampling of the correspondences to estimate it, the rational Bezier can be estimated accurately even from a sparse sampling of the correspondences. This allows the use of the low resolution camera to calibrate a much higher resolution display. For example, we can achieve calibration on a  $3000 \times 1500$  display using a VGA camera ( $640 \times 480$ ). Though these two capabilities were demonstrated for planar displays in [4], we demonstrate them for the first time for a class of non-planar displays.

### 3.4 Implementation

We have implemented our method in MATLAB for two types of displays. First, we have used a large rigid cylindrical display - an extruded surface with a radius of about 14 feet and an angle of 90 degrees. Since a cylinder is an extruded surface, our method is applicable. We tiled eight projectors in a casually aligned  $2 \times 4$  array to create the display. Second, in order to demonstrate the success of our method on a large number of vertically extruded shapes, we made a flexible display using a rectangular sheet of flexible white styrene. This was supported by five poles to which the styrene sheet was attached (Figures 9 and 10). The shape of the profile curve of this extruded display can be changed by simply changing the position of the poles. Thus, we can create a large number of extruded shapes. We use six projectors on this display in a casually aligned  $2 \times 3$  array to create the tiled display. For all the setups, we use Epson 1825p projectors (\$600). We show results by using two types of sensors: (a) a high-end high-resolution (13 Megapixel) Canon Rebel Xsi SLR camera (\$800); and (b) a low-end low-resolution (0.25 Megapixel) Unibrain camera (\$200). We achieve color seamlessness by using the constrained gamut morphing method presented in [15].

Figure 5 shows the single image used to recover the camera and display properties. To find the projector to camera correspondences, we display a rectangular grid of Gaussian blobs whose projector coordinates are known. These are then captured by the camera. We use a 2D stepping procedure where the user identifies the top-left blob and its immediate right and bottom neighbors in camera space. Following this, the method (a) estimates the rough position of the next blob in scan-line order, and (b) searches for the correct blob position using the nearest windowed center-of-mass technique [6]. If this is not possible for extreme projector/screen distortions, one can binary-encode the blobs and project them in a time sequential manner to recover the exact ids of the detected blobs and find the correspondences [13, 18] (Figure 6).

Our projectors have relatively large throw-ratios and hence do not reveal major lens distortions. To demonstrate the capability of our method to handle non-linearities, we chose to simulate the distortion

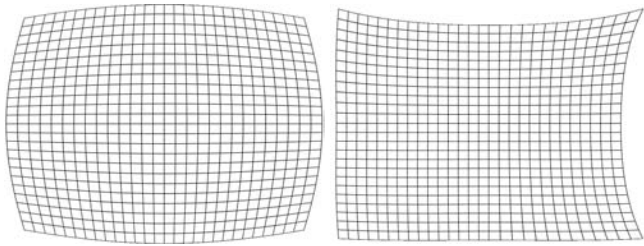


Fig. 8. Top: A barrel distortion of degree 2; Bottom: A combination of a pincushioning of degree two and a first order tangential distortion.

digitally by distorting the input images to the projectors. Such distortions will be common when mounting inexpensive short throw lens on the projector to create a compact setup.

**Real time image correction using GPU:** The registration is done offline and takes about five minutes. This generates the rational Bezier patches,  $(u, v) = B(x, y)$ , for each projector, which are then used for image correction. We have implemented a real-time image correction algorithm using modern GPUs through Chromium - an open-source distributed rendering engine for PC clusters [8]. A module for Chromium is written that first precomputes the coordinate-mappings of all pixels using the rational Bezier parameters. This per-pixel projector to screen lookup table is used by a fragment shader to map pixels from the projector coordinate space to the screen coordinate space during rendering.

## 4 RESULTS

Figures 1, 7 and 9 show the results of our method on different extruded surfaces including the most common case of a cylindrical surface. We demonstrate our results on particularly challenging contents like text, especially common for visualization applications, and show accurate geometric registration. Figure 10 demonstrates that our method can handle severe projector non-linearities enabling mounting inexpensive short throw lens for compact set-up. Figure 8 shows the two distortions we used in our experiments. Our supplementary video demonstrates the interactive rates which we achieve in all these renderings using our GPU implementation.

The degree of the rational Bezier used to achieve geometric registration depends on the amount of non-linearities present due to the curved screen and the distortions in the projectors. In our set-ups, we used a bicubic rational Bezier representation for the cylindrical surface. For our flexible display, we use a rational Bezier of degree 5 and 3 in horizontal and vertical directions respectively. With large projector distortions and larger curvature of the display, higher order rational Beziers will be more appropriate.

In Figure 11 we compare our method with three different methods. Since our work is the only work that can achieve a markerless and view-independent registration, probably the only fair comparison is with using a homography-based registration that assumes a piecewise planar display surface and uses a homography tree to register all the projectors [5]. However, in Figure 11 we also show comparisons with the view-dependent method presented in [18]. View-dependent registration defines a mapping from the projector coordinates  $(x, y)$  to the camera coordinates  $(u, v)$ , as opposed to the display coordinates  $(s, t)$  and Equation 3 becomes

$$(u, v) = (M_{C \leftarrow P}(x, y)). \quad (4)$$

Hence, the distortions of the camera (like the perspective projection) embeds itself in the registered display. Further, this method uses a piecewise linear mapping for  $M_{C \leftarrow P}(x, y)$  that requires a much denser sampling of projector-camera correspondences compared to our method. Hence, in the face of severe distortion, even with an order of magnitude higher number of samples, it cannot achieve the accuracy of registration achieved by our method. Finally, the ability to reconstruct the rational Bezier patches from a sparse sampling of the function allows us to use a very lower resolution camera (e.g. 640x480

Table 1. Percentage Errors of the estimated camera and display parameters over a large number of simulations with different configuration of the devices and the 3D display.

Parameter	Max	Mean	Std
Camera Orientation (deg)	0.494	0.192	0.167
Camera Position (%)	0.432	0.186	0.150
Focal Length (%)	3.82	2.26	0.98
Top and bottom curves (%)	0.547	0.217	0.153

VGA camera) to accurately calibrate a much higher resolution display (e.g. 3500x1200). Figure 12 compares the geometric registration achieved using a high-resolution vs a low resolution camera.

## 5 DISCUSSIONS

In this section, we discuss the dependency of our method on various parameters like the camera position, precision in the display surface, and the degree of the rational Bezier.

### 5.1 Camera Placement

Our method achieves markerless registration on extruded surfaces using an uncalibrated camera. Even in the presence of the priors on the display surface, there is a set of camera positions that will lead to degeneracy for one or both phases of our non-linear optimization. Consider the plane based optimization stage where the goal is to find the focal length  $f$  and the extrinsic parameters. Let us assume the camera calibration matrix  $C$  to be

$$C = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_1 & r_2 & r_3 & t_x \\ r_4 & r_5 & r_6 & t_y \\ r_7 & r_8 & r_9 & t_z \end{pmatrix}. \quad (5)$$

Note that in the plane based optimization we are using four points that have  $Z = 0$  and  $Y = 0$  or  $Y = 1$ . Now, consider the case where  $r_7 = r_8 = 0$  and  $r_9 = 1$ . This is equivalent to placing the camera on the  $Z$ -axis with the normal to the image plane being parallel to the  $Z$ -axis. In this case, the homogeneous coordinates of the images of the four corners of the plane are given by  $(f \times (ar_1 + t_x), f \times (ar_5 + t_y), t_z)^T$  when  $Y = 0$  and  $(f \times (ar_1 + r_2 + t_x), f \times (ar_5 + r_6 + t_y), t_z)^T$  when  $Y = 1$ . Note that these points have a scale factor ambiguity, i.e. multiplying  $t_z$  and  $f$  with the same scale factor would result in the same image coordinates. Intuitively, if the camera is placed with the image plane parallel to the planar rectangle defined by the extruded surface on the  $Z$ -axis, moving the camera on the  $Z$ -axis can create the similar change as scaling its focal length and we cannot find a unique solution to the camera parameters. Hence, this camera placement should be avoided.

Second, let us consider the two 3D curves  $B_t$  and  $B_b$ , where  $B_t = B_b + (0, 1, 0)$ . If the camera placement is such that the images of these two curves,  $I_t$  and  $I_b$  respectively, are related by  $I_t = I_b + (0, k)$  where  $k$  is a translation in the vertical image direction, then the extrusion based optimization will be redundant. This camera placement occurs when the normal to the camera image plane lies on a plane parallel to the  $XZ$  plane i.e. is perpendicular to the  $Y$ -axis. Hence, this camera placement should also be avoided. Note that the former placement that resulted in the scale factor ambiguity is contained in this latter condition since  $Z$ -axis is on the  $XZ$  plane. Hence, as long as a camera placement where the normal to the image plane is parallel to the  $XZ$  plane is avoided, our optimization will yield an accurate solution.

### 5.2 Accuracy and Sensitivity

Our system estimates the camera and display parameters and makes assumptions on the type of the display surface. Hence, it is important to answer two questions: (a) How accurate are the estimated camera and display parameters in the non-linear optimization stage?; and (b) how sensitive is the geometric registration to the inaccuracies of these estimates or the priors imposed on the display surface? It is difficult to analyze all the above issues in real systems, hence we have conducted extensive analysis in simulation and real systems (whenever possible) to answer these questions.

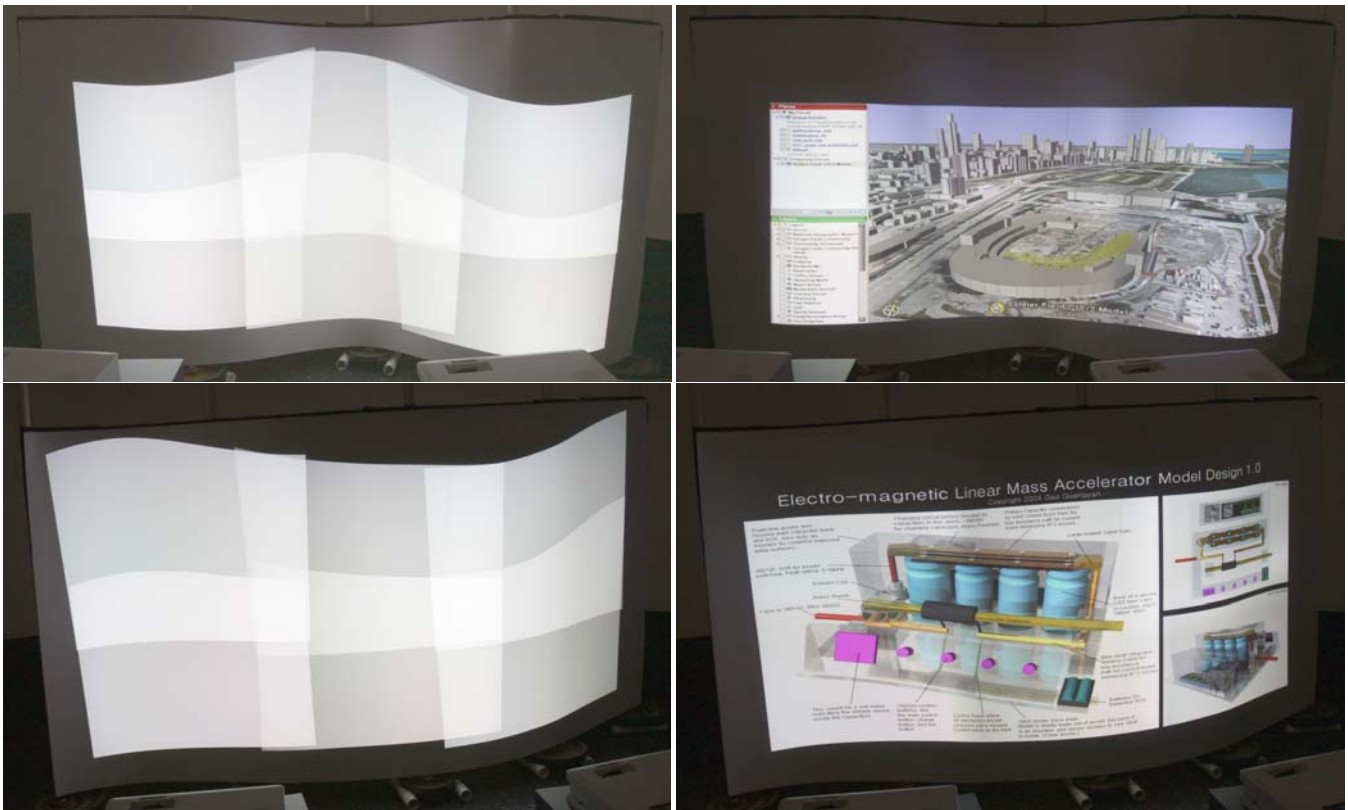


Fig. 9. Our geometric registration on two different extruded surfaces (top and bottom) created using our flexible display set-up when using  $2 \times 3$  array of 6 projectors- left column shows the casually aligned set-up and the right column shows the wall papering of images after our registration is used. Please zoom in to see the quality of results.

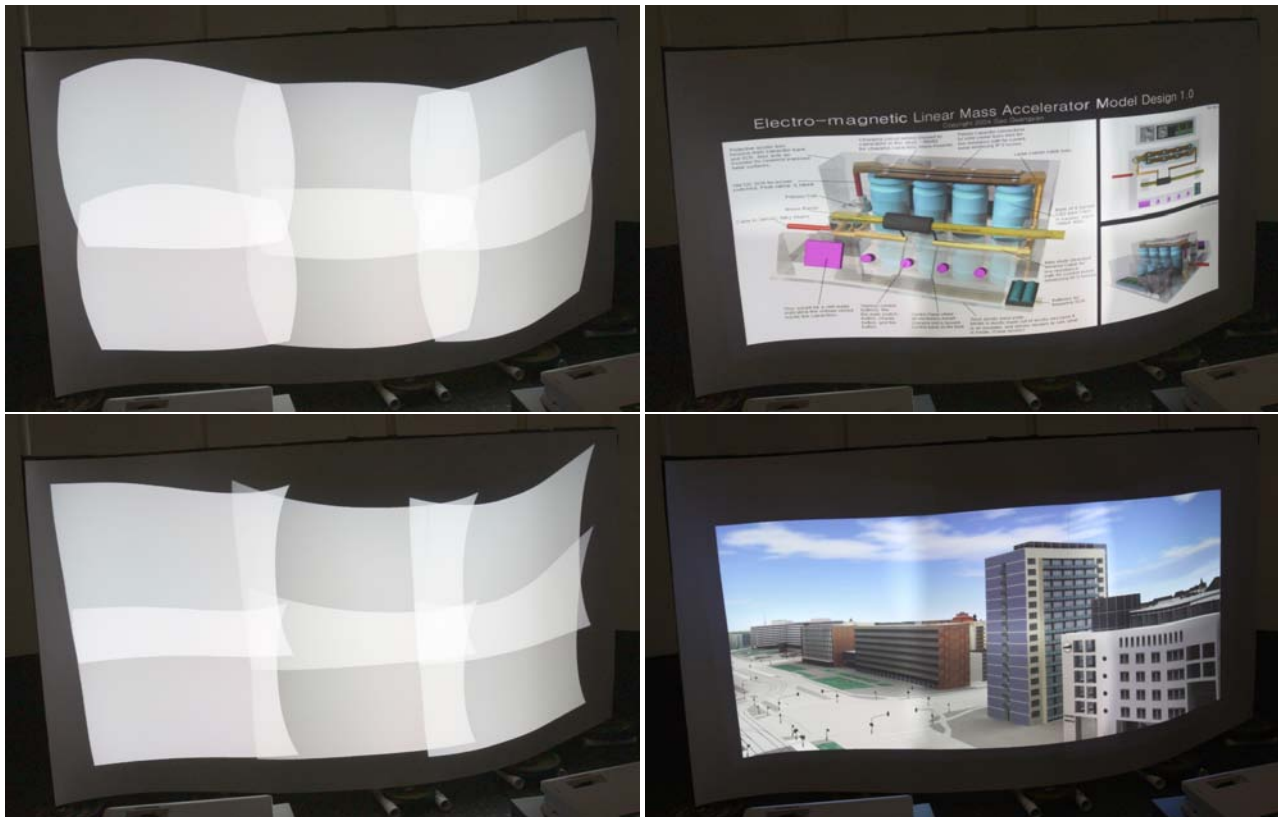


Fig. 10. Our geometric registration handling severe non-linear distortion on an general extruded surface when using  $2 \times 3$  array of 6 projectors - left column shows the casually aligned set-up and the right column shows the wall papering of images after our registration is used. Top: Using severe barrel distortion; Bottom: Using severe pin-cushioning and tangential distortion. Please zoom in to see the quality of results.

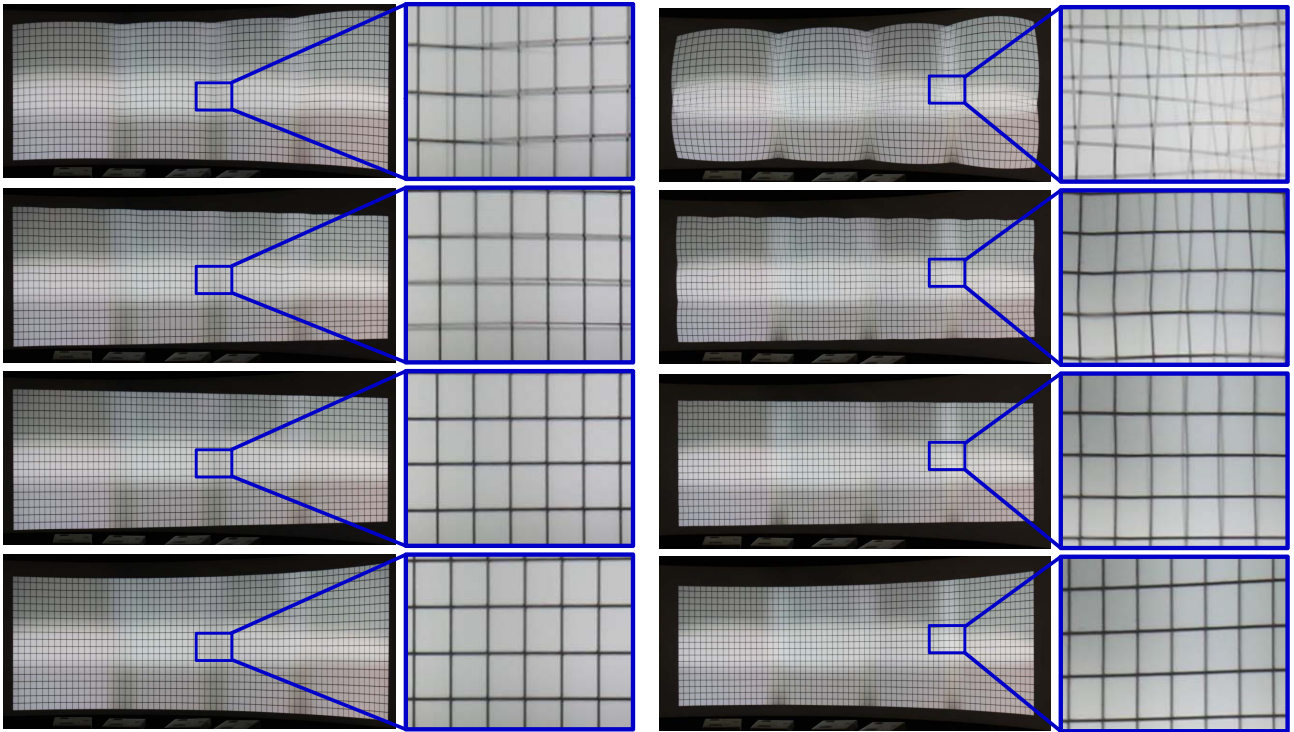


Fig. 11. Comparison of our method with other methods - Without projector non-linearities on the left and with severe radial distortion on the right. From top to bottom: (a) Using simple homography; (b) Using view dependent piecewise linear method with a sparse  $4 \times 5 = 20$  projector to camera correspondences - note the perspective projection of the camera embedded in the display shown by the more stretching on the left side than on the right; (c) Using view dependent piecewise linear method with a dense  $10 \times 12 = 120$  projector to camera correspondences; (d) Our method using a sparse  $4 \times 5 = 20$  projector to camera correspondences - note the perfect wall papering and the equal or higher quality of registration than the piecewise linear method despite the sparse sampling, especially in the presence of projector non-linearities. Please zoom in to check quality of results.

First, we study the accuracy of the estimated camera extrinsic parameters following our non-linear optimization process. Our simulation of many different camera and display parameters shows that when an accurate intrinsic parameter matrix is given, our estimated extrinsic parameter matrix is very accurate. The error analysis of the deviation of the estimated parameters from the actual parameters is provided in Table 1. For the orientation of the camera, we provide the deviation in degrees from the actual orientation. For translation, we provide the ratio of the error in estimation with the distance from the screen. We also study the accuracy of the estimated 3D profile curves of the display in this situation. To compare the estimated curves with the actual ones, we first sample the estimated curves densely. Then, for each sample, we find the minimum distance to the original curve. The ratio of the maximum of these distances to the length of the original curve is considered to be the measure of the accuracy of the display geometric reconstruction and is reported in Table 1.

We analyzed the validity of our simplifying assumption for the camera intrinsic matrix by running some experiments. For each of our camera set-up, we used standard algorithms and toolboxes to accurately estimate the camera’s intrinsic matrix [20]. The skew estimated by Zhang’s method was always zero and the principal center deviated from the center of the image by a percentage error that is within the error tolerance of Zhang’s method. These two confirm the validity of our use of a simpler intrinsic matrix. Further, we compared the estimated focal length from this method to the focal length estimated by our non-linear optimization to analyze the accuracy of the estimated intrinsic camera parameters. We found that when provided with a good initial estimate as is available from the EXIF tags, the focal length estimated by our method is very close to that recovered by Zhang’s method as indicated in Table 1.

We analyze the sensitivity of our registration to imprecision in the display surface or errors in the estimation of the display shape, both of which would result from a deviation of the real surface from a per-

fect extruded surface. However, our rational Bezier function provides a particularly robust framework for handling deviation from extruded surfaces. This is due to the fact that a small deviation from extrusion will lead to an erroneous 2D parametrization of the display surface, but the overlapping pixels from the multiple projectors will still map to the same  $(s, t)$ . Hence, an imprecision in the extrusion can create small image distortions but will not lead to any misregistration. This is one of the strengths of our algorithm and is well-demonstrated by our flexible display which shows considerable imprecision due to its make-shift flexible prototype nature, but almost no misregistration of the projected images is visible even on this display. We quantitatively evaluate the effect of deviation of a surface from an extruded surface on the accuracy of the estimated camera parameters in Figure 13. Deviation from extrusion is measured by the maximum difference of the top and bottom curves with respect to the curve length. This plot shows even in presence of large deviation of the screen from being an extruded surface our method can achieve a reasonable estimation of camera pose and focal length.

### 5.3 Camera Non-Linearity and Resolution

We assume the camera to be a linear device devoid of any non-linear distortion in Section 3. However, even if this is not true when using commodity cameras, our method will not result in any pixel misregistration since the camera non-linearity will be accounted for by the fitted rational Bezier patches. However, the camera non-linearity will affect the accuracy of the reconstruction of the 3D shape of the screen and hence, the final result may not be perfectly wall papered. Fortunately, human visual system can tolerate such minor deviation from wall papering. For verification, we performed our registration using an uncalibrated Unibrain Fire-i webcam with  $640 \times 480$  resolution (one tenth of our display resolution) which had significant non-linear lens distortion (with quadratic coefficient of 0.01 and quartic coefficient of -0.009). We compare the achieved result with the ones achieved by



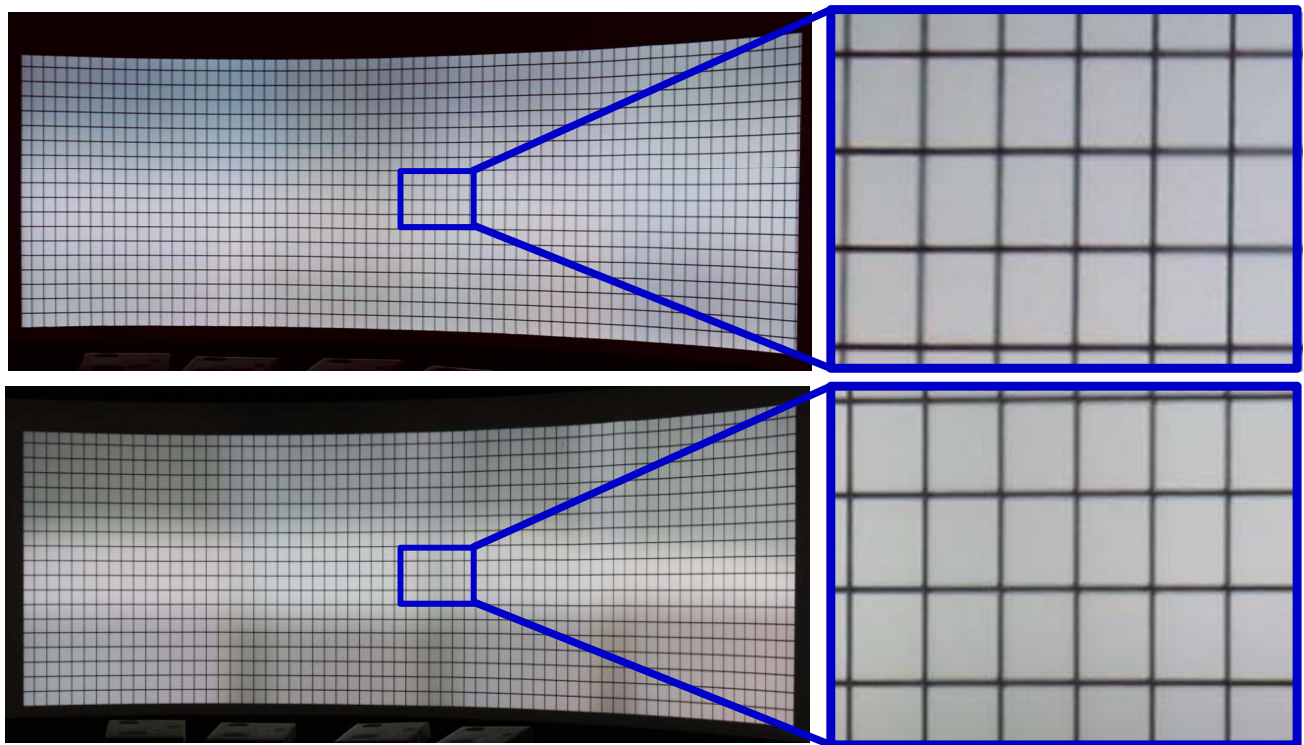


Fig. 12. Geometric registration, Top: with our low resolution webcam (0.25 Megapixels), Bottom: with our high-end SLR camera (13 Megapixels).

our high resolution camera in Figure 12. Note that the deviation from wall-papering is hardly detectable and the registration is comparable. In case of more severe camera non-linearities one can use standard camera calibration techniques to undistort the captured images.

#### 5.4 User Assistance

Our method needs to detect the four corners and the top and bottom curves of the extruded surface. Since the screen is usually the most distinct white object in the environment, segmenting it is relatively easy if the background is of reasonable contrast. Further, more often than not, a display environment is designed to have relatively diffused illumination, which does not affect the segmentation adversely. Even in the worst case of a low contrast between the screen and the background color, one can always use user interaction to improve the segmentation. All other steps of our method are completely automated as long as the projection area of the projectors are entirely within the screen and the screen is entirely within the camera’s field-of-view.

#### 6 CONCLUSION

In summary, we have presented the first work for markerless view-independent registration of tiled projection-based displays on extruded surfaces using an uncalibrated camera. We have shown that by imposing practical priors on the display surface, the registration technique can be simplified to be easily used by layman users. Our method provides a very user-friendly and cost-effective way to sustain such displays in large establishments like visualization centers, museums, theme-parks. Further, our method also offers the ability of recalibration and reconfiguration at a very short notice. These can be especially useful for applications like digital signage and aesthetic projections in malls, airports and other public places.

In the future, we would like to explore the similar concept of practical priors leading to easier registration for a different kind of widely-used non-planar surfaces, the domes. In the recent years, the number of domes have surpassed the number of IMAX theater installations (Figure 14). However, there still does not exist an easy way to calibrate these displays. Our goal is to extend our fundamental concept in

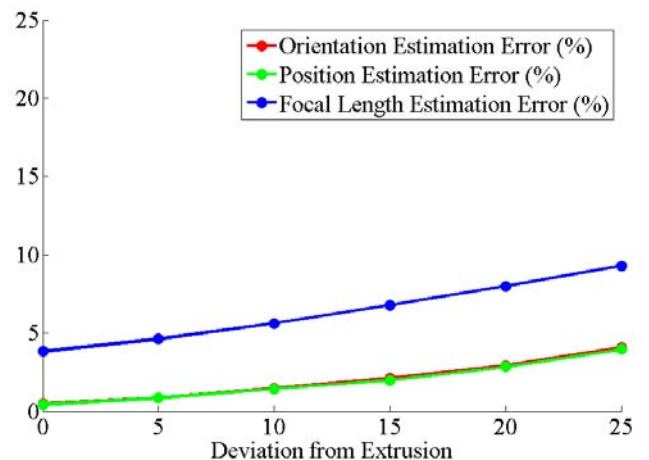


Fig. 13. Accuracy of camera parameter estimation in presence of deviation of the screen from being an extruded surface.

this direction.

#### REFERENCES

- [1] D. Aliaga. Digital inspection: An interactive stage for viewing surface details. *Proc. ACM Symp. on I3D*, 2008.
- [2] D. Aliaga and Y. Xu. Photogeometric structured light: A self-calibrating and multi-viewpoint framework for accurate 3d modeling. *Proc. of IEEE CVPR*, 2008.
- [3] M. Ashdown, M. Flagg, R. Sukthankar, and J. M. Rehg. A flexible projector-camera system for multi-planar displays. *Proc. of IEEE CVPR*, 2004.
- [4] E. Bhasker, R. Juang, and A. Majumder. Registration techniques for using imperfect and partially calibrated devices in planar multi-projector

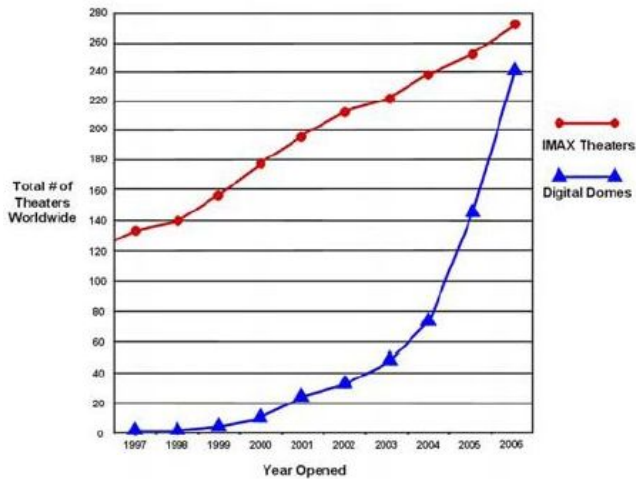


Fig. 14. Domes vs IMAX theaters opened in recent years.

- displays. *IEEE TVCG*, 2007.
- [5] H. Chen, R. Sukthankar, G. Wallace, and K. Li. Scalable alignment of large-format multi-projector displays using camera homography trees. *Proc. of IEEE Vis*, 2002.
- [6] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison Wesley, 1992.
- [7] M. Harville, B. Culbertson, I. Sobel, D. Gelb, A. Fitzhugh, and D. Tanguay. Practical methods for geometric and photometric correction of tiled projector displays on curved surfaces. *IEEE PROCAMS*, 2006.
- [8] G. Humphreys, M. Houston, R. Ng, R. Frank, S. Ahem, P. Kirchner, and J. Klosowski. Chromium : A stream processing framework for interactive rendering on clusters. *ACM Transactions on Graphics (SIGGRAPH)*, 2002.
- [9] A. Raij, G. Gill, A. Majumder, H. Towles, and H. Fuchs. Pixelflex 2: A comprehensive automatic casually aligned multi-projector display. *IEEE PROCAMS*, 2003.
- [10] A. Raij and M. Pollefeys. Auto-calibration of multi-projector display walls. *Proc. of ICPR*, 2004.
- [11] R. Raskar. Immersive planar displays using roughly aligned projectors. In *Proc. of IEEE VR*, 2000.
- [12] R. Raskar, J. V. Baar, T. Willwacher, and S. Rao. Quadric transfer function for immersive curved screen displays. *Eurographics*, 2004.
- [13] R. Raskar, M. Brown, R. Yang, W. Chen, H. Towles, B. Seales, and H. Fuchs. Multi projector displays using camera based registration. *Proc. of IEEE Vis*, 1999.
- [14] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM Transaction on Graphics (SIGGRAPH)*, 2003.
- [15] B. Sajadi, M. Lazarov, A. Majumder, and M. Gopi. Color seamlessness in multi-projector displays using constrained gamut morphing. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2009.
- [16] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. *ACM Transactions on Graphics (SIGGRAPH)*, 2006.
- [17] W. Sun, I. Sobel, B. Culbertson, D. Gelb, and I. Robinson. Calibrating multi-projector cylindrically curved displays for "wallpaper" projection. *IEEE/ACM Workshop on PROCAMS*, 2008.
- [18] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. S. Brown. Pixelflex: A reconfigurable multi-projector display system. *Proc. of IEEE Vis*, 2001.
- [19] R. Yang, A. Majumder, and M. Brown. Camera based calibration techniques for seamless multi-projector displays. *IEEE TVCG*, 2005.
- [20] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. *International Conference on Computer Vision*, 1999.

# Auto-Calibration of Cylindrical Multi-Projector Systems

Behzad Sajadi\*  
Department of Computer Science  
University of California, Irvine

Aditi Majumder†  
Department of Computer Science  
University of California, Irvine

## ABSTRACT

In this paper we present a novel technique to calibrate multiple casually aligned projectors on a fiducial-free cylindrical curved surface using a single camera. We impose two priors to the cylindrical display: (a) cylinder is a vertically extruded surface; and (b) the aspect ratio of the rectangle formed by the four corners of the screen is known. Using these priors, we can estimate the display's 3D surface geometry and camera extrinsic parameters using a single image without any explicit display to camera correspondences. Using the estimated camera and display properties, we design a novel deterministic algorithm to recover the intrinsic and extrinsic parameters of each projector using a single projected pattern seen by the camera which is then used to register the images on the display from any arbitrary viewpoint making it appropriate for virtual reality systems. Finally, our method can be extended easily to handle sharp corners - making it suitable for the common CAVE like VR setup. To the best of our knowledge, this is the first method that can achieve accurate geometric auto-calibration of multiple projectors on a cylindrical display without performing an extensive stereo reconstruction.

**Keywords:** Multi-Projector Displays, Tiled Displays, Calibration, Registration

**Index Terms:** I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality

## 1 INTRODUCTION

Cylindrical virtual reality systems are very common for large number of applications like gaming, entertainment, and 3D visualization. An inexpensive and popular way to increase the resolution of such displays is to tile multiple projectors on the cylindrical display surface. The challenge lies in automatic registration of the multiple projected imagery on the display surface to create one seamless image. The problem is further complicated when this needs to be achieved quickly without involving a time-consuming complete 3D reconstruction via structured light or attaching any special fiducials to the display surface.

Registering images from multiple projectors on non-planar displays requires 3D reconstruction of the display surface which in turn requires multiple cameras. Though there is a large body of literature that addresses such a reconstruction and registration [19, 8, 9, 12, 27, 13, 11, 22], these are complex procedures requiring camera calibration and multiple physical fiducials on the display surface. Hence, many methods try to avoid the complexity of using multiple cameras when using non-planar screens. Brown et al. [25] register multiple projectors with respect to the single point of view of a calibrating camera. This still achieves a seamless registration of the multiple projectors, but avoids the 3D reconstruction of the display surface entirely. Though widely adopted by commercial multi-projector display auto-calibration vendors, this reg-

istration is correct only from one view, the view of the calibrating camera. Hence, when viewed from a different point, distortions reflecting the camera perspective and the display surface geometry are visible. So, this registration is particularly unsuitable for VR applications where the user moves around deviating considerably from the position of the calibrating camera. [11, 22] try to avoid the view-dependency for cylindrical surfaces by relating the camera coordinates to the physical coordinates of the display by pasting special fiducials on the two rims of the display surface. Though this 'wall-papers' the imagery on the display, lack of a 3D display reconstruction does not allow registration from arbitrary viewpoints.

## 1.1 Main Contributions

In this paper, we present a new method that can register images from multiple projectors on a cylindrically non-planar display using a single uncalibrated camera and without using any fiducials in a *view-independent* manner – i.e. the registration does not depend on the view (pose and orientation) of the calibrating camera, hence, we can compute the correct image for any arbitrary view point. Contrary to multi-frame structured light patterns, we avoid using explicit correspondences between the display surface and the observing camera by imposing two priors: (a) a cylindrical display is a vertically extruded surface; and (b) we know the aspect ratio of the planar rectangle formed by the four corners of the display. With these priors, we can use a single image of the cylindrical surface from the camera to recover the camera pose and orientation and the 3D display geometry via a non-linear optimization. We design a deterministic geometric algorithm which uses these recovered properties to auto-calibrate (i.e. find both the intrinsic and extrinsic parameters) each projector from a single pattern captured by the camera. Once auto-calibrated, we achieve geometric registration on the display surface via a ray-casting method.

Unlike any existing method that use a single camera to register multiple projectors on a non-planar display [25, 11, 22], we can reconstruct the shape of the 3D display. This enables us to parametrize the display directly in a 3D coordinate system, rather than in the camera image space, to achieve a view-independent geometric registration. Finally, unlike the above methods which achieve registration without completely calibrating the projectors, we achieve a complete auto-calibration of the projectors. This results in a simple and accurate method to compute correspondences between the display and the projector. Hence, for static display surfaces, once the 3D display geometry is recovered, our auto-calibration method can be used for quickly changing the projector position and orientation to create displays of different scale, resolution and aspect ratio.

Our method can handle any smooth vertically extruded surface which can be interesting for entertainment purposes. More importantly, it can be easily extended to handle extruded surfaces with sharp edges. This opens up the possibility of using our algorithm in a CAVE like setup. Finally, the image correction required to register the images can be achieved in real-time on GPUs making our method especially suitable for real-time VR applications. To the best of our knowledge, this is the first method that can achieve complete auto-calibration and consequently a view-independent registration on specialized non-planar displays - vertically extruded surfaces like cylinders - without using any physical fiducials on the

\*e-mail: bsajadi@ics.uci.edu

†e-mail:majumder@ics.uci.com

display surface.

**Organization:** We present a survey of related work in Section 2. We present our auto-calibration algorithm followed by two different view-independent registration techniques, suited for different applications, in Section 3. Next, we provide a proof of concept that our method can be extended to handle the common CAVE like VR systems in Section 4. We present our results in Section 5. Finally, we conclude the with future work in Section 6.

## 2 RELATED WORK

Our work is related to a large body of literature that deals with various aspects of calibration in projection-based displays. Considering *planar surfaces and single cameras*, Raji and Pollefeys [17] and Raskar et al. [20] describe techniques to automatically calibrate multiple projectors on planar display surfaces. PixelFlex [16, 23] provided a multi-projector display on planar surfaces where each projector image can be easily and quickly repositioned to create new display configurations that can be calibrated within minutes. Bhasker et al. [4] achieve the same in presence of projector non-linearities (i.e. radial and tangential distortions) using rational Bezier patches.

Chen et al. [7] used *multiple cameras on planar displays* to achieve a homography-tree based registration across multiple projectors. Moving away from a centralized architecture where the multiple cameras and projectors are controlled from a central server, Bhasker et al. [5] present a distributed framework where each projector is augmented by a single camera and has the responsibility of registering itself with the rest of the display. An asynchronous distributed calibration algorithm runs on each augmented projector in a SIMD fashion to create a seamless display.

When considering *non-planar displays*, especially arbitrary ones, using *multiple cameras* becomes necessary for 3D reconstruction of the non-planar surface. Raskar et al. in [19] use special 3D fiducials to achieve a complete device (camera and projector) calibration and 3D reconstruction of the display surface using a large number of structured light patterns, which are then used to achieve the geometric registration. Aliaga et al. in [2, 1] also achieve a 3D reconstruction to register multiple images on complex 3D shapes, but without using any physical fiducials. To constrain the system sufficiently, this method uses completely superimposed projectors and cross-validates calibration parameters and display surface estimates using both photometric and geometric stereo, resulting in a self-calibrating system. Raskar et al. in [18] use a stereo camera pair to reconstruct special non-planar surfaces called quadric surfaces (spheres, cylinders, ellipsoids and paraboloids) and propose conformal mapping and quadric transfer to minimize pixel stretching of projected pixels after the geometric registration.

All of the above methods achieve a pre-calibration, sometimes in a few minutes. A complementary set of techniques exist that can focus on continuous image registration during the display time for change in the display shape and movement in the projectors. Yang and Welch [24] use the projected content (as opposed to special patterns) at the display time to automatically estimate the shape of the display surface and account for the changes in its shape over time. Using a projector augmented by two stereo cameras, Cotting et al. [8, 9, 12] estimate the shape of the display surface and the pose of a single projector continuously over time by embedding imperceptible calibration patterns into projected imagery. Zhou et al. [27] achieve the same by tracking displayed image features. Johnson et al. [13] show that multiple such units can be used in a distributed framework to achieve continuous geometric calibration in a multi-projector setup. Zollman et al. [28] present a hybrid technique that can compensate for small changes in display configuration using optical flow, and will resort to active structured light projection when the optical flow becomes unreliable.

Our work belongs to the body of literature that tries to avoid

the complexity of using multiple cameras when using non-planar screens. Brown et al. [25, 6] avoid reconstructing the display geometry by registering multiple projectors with respect to the single point of view of a calibrating camera. More recently, [11, 22] tried to avoid this view-dependency in registration for the special case of cylindrical surfaces by finding a way to relate the 2D parametrization of the cylindrical display with that of the camera image space without reconstructing the 3D display surface. A precisely calibrated physical pattern is pasted along the top and bottom curves of the cylinder to provide a physical 2D display parametrization. By identifying the corresponding images of these fiducials in the observing camera, a piecewise planar representation of the display is achieved in the camera space. The projectors can then be registered directly in the display space rather than the camera space resulting in a 'wall-papered' registration. However, since it is not possible to have fiducials at a high spatial density on a display and the fiducials only samples the rims of the display, these methods result in distortions or stretching, especially towards the middle of the display surface. The more important point to note here is that in both these methods, since the display surface is not reconstructed, *registering images from an arbitrary viewpoint as is required in a virtual reality system, is not possible*. Our work uses a single uncalibrated camera, does not need to use physical fiducials, and can still achieve a calibration from any arbitrary viewpoint.

Technically, our work is close to [17] that achieves a similar goal of auto-calibration of projectors for planar screens, but our method is entirely different catered towards cylindrical screens. In particular, unlike [17] where the projector auto-calibration results from an involved optimization process, our projector auto-calibration is achieved by an efficient and fast deterministic algorithm allowing quick recalibration in the event of change in pose and orientation of the projectors. Further, we do not make restrictive assumptions like square projector pixels and identical vertical shift for all projectors.

## 3 AUTO-CALIBRATION ALGORITHM

Let the display surface, the image planes of the camera, and the projector be parametrized by  $(s, t)$ ,  $(u, v)$ , and  $(x, y)$  respectively. We denote the 3D coordinates of the point at  $(s, t)$  on the display by  $(X(s, t), Y(s, t), Z(s, t))$ . Using the fact that a cylinder is a vertically extruded surface we impose the following constraints on the display surface. The four corners of the display lie on a planar rectangle, whose aspect ratio  $a$  is known. We define the world 3D coordinate system with  $Z$  axis perpendicular to this plane and  $X$  and  $Y$  defined as the two orthogonal basis of this planar rectangle. We also consider this planar rectangle to be at  $Z = 0$  and the top and bottom curves of the cylinder to lie respectively on  $Y = 1$  and  $Y = 0$  planes in this coordinate system. Hence,  $Y(s, 0) = 0$  and  $Y(s, 1) = 1$ . Further, these two curves are identical except for a translation in the  $Y$  direction. Consequently,  $\forall s, (X(s, 0), Z(s, 0)) = (X(s, 1), Z(s, 1))$ . These are illustrated in Figure 3.

We make the following practical assumptions to simplify the problem:

- Our camera and the projectors are linear devices with no radial distortion.
- Projectors are considered dual of a pin-hole camera.
- The camera intrinsic parameters are known, but not its pose and orientation.

For an  $n$  projector system, our auto-calibration takes  $n + 1$  images as input. The first image,  $I_0$ , is that of the display surface with no projectors turned on. Next, for each projector  $i$ ,  $1 \leq i \leq n$ , we take a picture  $I_i$  of the same display surface with projector  $i$  projecting a special line pattern (Figure 1).

Our algorithm consists of three steps:

1. With  $I_0$  as input we estimate the camera and display surface properties using a non-linear optimization (Section 3.1).

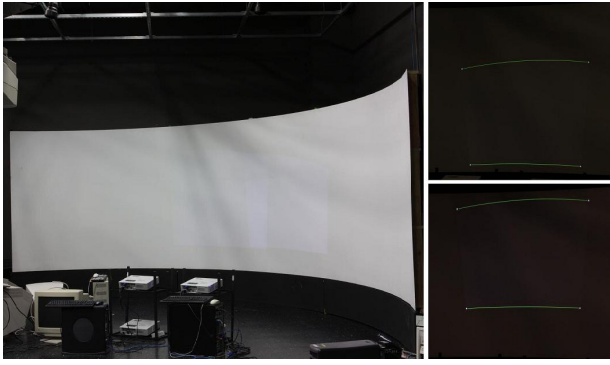


Figure 1: Left: The image  $I_0$  used for estimating camera and display properties. Right: The zoomed in view of  $I_1$  and  $I_2$  for one of our setups, where the projector 1 and 2 are projecting the single pattern used for auto-calibration.

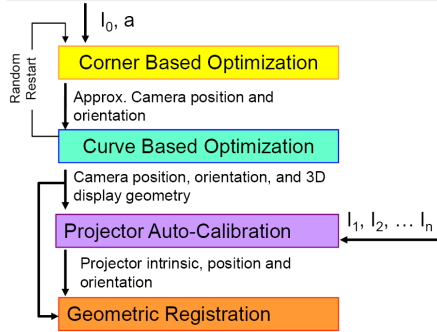


Figure 2: The pipeline of our algorithm.

- Using the recovered camera and display properties and the image  $I_i$ , we find the intrinsic and extrinsic parameters of projector  $I_i$ , thus auto-calibrating the projectors. For this we use a deterministic algorithm which is fast and efficient enabling quick changes in projector properties (position, orientation, and zoom) (Section 3.2).
- We use the recovered projector properties to register images seamlessly on the cylindrical display (Section 3.3). We present two ways to register the images. (a) The first kind of registration provides seamless imagery that looks correct from an arbitrary view point which can change without requiring a recalibration. This is suitable for a VR application with a head-tracked single user. (b) The second type of registration wall-papers the image seamlessly on the display. This is suitable for multi-user visualization applications.

The complete pipeline of our method is illustrated in Figure 2.

### 3.1 Camera and Display Property Estimation

The input to this step is the image  $I_0$ , the  $3 \times 3$  intrinsic parameter matrix of the camera, and the aspect ratio  $a$ . The output is an estimation of the  $3 \times 4$  extrinsic parameter matrix (defining position and orientation) of the camera and the 3D geometry of the display defined by the top and bottom 3D curves. The novelty of this step is to estimate the camera parameters and the 3D display geometry from a single image without using any correspondences. The correspondences are avoided by exploiting the fact that the top and the bottom curves of the display are identical except for being in two different XZ planes in 3D.

To estimate the camera parameters, we use a two phase non-linear optimization method. In the first step we gather a *rough estimate* of the camera extrinsic parameters (pose and orientation) using the projection of just the corners of the display surface on the

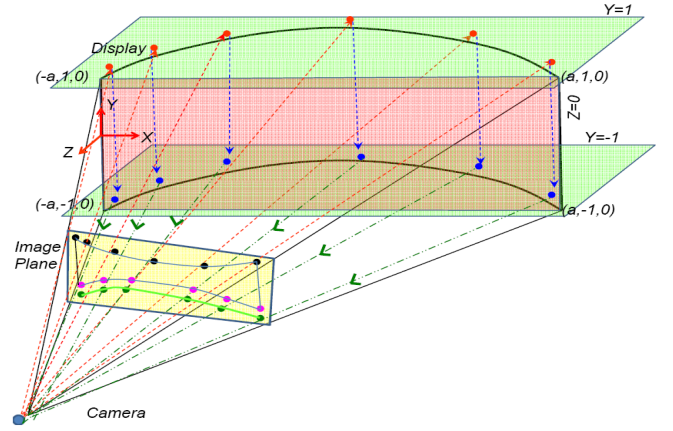


Figure 3: A curve is fitted through the sampled points on the 2D bottom curve (purple line). The sampled points on the 2D top curve in the camera (black) are reprojected in 3D to estimate the 3D top curve (red), and translated down to estimate of the 3D bottom curve (blue), and finally projected back on the camera (green). The distance between these points and the purple curve is minimized in the curve based optimization step.

camera image. These rough estimates are then used to initialize the second optimization step with a more expensive error function that refines these camera extrinsic parameters to provide an *accurate estimate*. The recovered extrinsic camera parameters are then used to estimate the 3D display geometry.

**Rough Estimation of Camera Parameters:** The camera coordinates,  $(u, v)$ , of any 3D point  $(X(s, t), Y(s, t), Z(s, t))$  on the display are given by,

$$(uw, vw, w)^T = M(X(s, t), Y(s, t), Z(s, t), 1)^T \quad (1)$$

where  $(uw, vw, w)^T$  is the 2D homogeneous coordinate corresponding to the camera coordinate  $(u, v)$  and  $M = K(R|RT)$  is the camera calibration matrix comprising of the  $3 \times 3$  intrinsic parameter matrix  $K$  and the  $3 \times 4$  extrinsic parameter matrix  $(R|RT)$ . We assume that  $K$  is known or estimated using standard camera calibration techniques [26]. We estimate the  $(R|RT)$  matrix that provides the pose and orientation of the camera.  $(R|RT)$  comprises of six parameters including three rotations to define the orientation and a 3D center of projection (COP) of the camera to define the position. Given our 3D world coordinate system, the 3D locations of the four corners of the cylindrical display in a counter-clockwise manner starting from top left are given by:  $(-\frac{a}{2}, 1, 0)$ ,  $(\frac{a}{2}, 1, 0)$ ,  $(\frac{a}{2}, 0, 0)$ , and  $(-\frac{a}{2}, 0, 0)$ . In this step, we estimate the six camera extrinsic parameters by minimizing the reprojection error  $E_r$ , (i.e. sum of the distances of the projection of these corners on the image plane from the camera captured positions) of the 3D corners from the detected corners in the image.

To initialize this optimization, we use the following. The angles of rotations about the X, Y, and Z axes that comprise  $R$  are initialized to zero.  $T$ , the COP of the camera is initialized roughly at the center of the planar rectangle formed by the four corners of the display at a depth of a similar order of magnitude as the size of the display. This is achieved by initializing  $T$  to  $(0, 0, a)$ .

**Accurate Estimation of Camera Parameters:** These rough estimates of the camera extrinsic parameters achieved in the previous step are used to initialize a second optimization that attempts to refine these parameters. Here, we augment the error function from the previous optimization step,  $E_r$ , with another error function,  $E_c$ , which is the reprojection error of the estimated 3D top and bottom curves of the cylindrical display (Figure 3). We seek to minimize the weighted combined error,  $w_r E_r + w_c E_c$ .

Let  $C_T$  and  $C_B$  be the 3D top and bottom curves of the display.

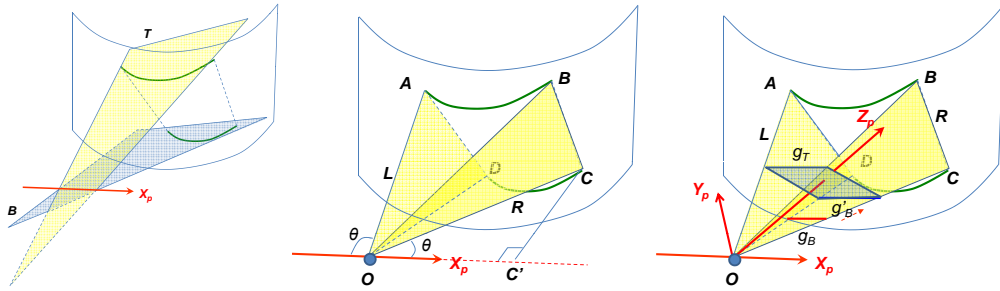


Figure 4: Illustrations demonstrating the methods to find  $X_p$  (left), enter of the camera  $O$  (middle), and  $Y_p$  and  $Z_p$  (right).

We first use segmentation and contour detection techniques to sample the 2D projected coordinates of  $C_T$  and  $C_B$  in the camera space, denoted by the set of samples  $S_1^T, S_2^T, \dots, S_N^T$  and  $S_1^B, S_2^B, \dots, S_N^B$  respectively. Prior to the optimization, we fit a curve  $C_B'$  through  $S_1^B, S_2^B, \dots, S_N^B$ . During the optimization, we reproject  $S_1^T, S_2^T, \dots, S_N^T$  to the 3D coordinate system. To reproject the  $k$ th sample  $S_k^T$ ,  $1 \leq k \leq N$  in 3D, we cast a ray from the center of projection of the camera through  $M^{-1}(S_k^T, 1)^T$ . We intersect this ray with the  $Y = 1$  plane to find the corresponding reprojected 3D coordinate,  $r(S_k^T)$ , where  $r$  denotes this reprojected function. We know that  $r(S_k^T)$  when translated -1 unit in the  $y$  direction, i.e.  $r(S_k^T) + h$ , where  $h = (0, -1, 0)$ , should lie on  $C_B$  since the cylinder is a vertically extruded surface. We then project these translated 3D points,  $r(S_k^T) + h$ , back on to the camera image plane to generate a new set of points  $Q_k^B = M(r(S_k^T) + h)$  where  $M$  is the 3D to 2D perspective projection of the camera. If the estimated camera calibration parameters are accurate, then all the samples  $Q_1^B, Q_2^B, \dots, Q_N^B$  would lie exactly on  $C_B'$ . So,  $E_c$  is defined as the sum of the distances between  $Q_1^B, Q_2^B, \dots, Q_N^B$  and  $C_B'$  in a least squares sense.

To solve both the above optimizations, we use standard gradient descent methods. To assure faster convergence we (a) apply a preconditioning to the variables so that the range of the values that can be assigned to them is normalized; and (b) use decaying step size.

**Estimation of the Display Geometry:** After convergence of the non-linear optimization process, we use the estimated camera calibration parameters to reproject  $S_1^T, S_2^T, \dots, S_N^T$  and  $S_1^B, S_2^B, \dots, S_N^B$  in 3D to find  $C_T$  and  $C_B$  respectively. Due to accumulated errors,  $C_T$  and  $C_B$  may not be identical. So, we project both the curves on  $Y = 0$  plane and find their average to define  $C_B$ . This is then translated to  $Y = 1$  to define  $C_T$ . We then use a polynomial curve fitting to find a parametric representation of  $C_T$  and  $C_B$ .

### 3.2 Projector Calibration

In this step, we project a pattern from each projector comprising of four corner blobs and a top and bottom line. An image  $I_i$  of this pattern is captured by the camera. Using  $I_i$  and the estimated camera calibration parameters and 3D display geometry, we estimate the intrinsic and extrinsic parameters of each projector.

Let the image of the top and bottom lines for the projector in  $I_i$  be denoted by  $l_T$  and  $l_B$  respectively. Let the blobs be denoted by  $b_A, b_B, b_C$  and  $b_D$  from the top left corner in a clockwise manner (Figure 1). Note that though these lines are straight in the projector space, they look curved in 3D due to projection on a curved surface. The auto-calibration of each projector consists of two steps. First, we find the view frustum of the projector defined by its center and five planes (top, bottom, left, right and the image plane) that define the extrinsic parameters of the projector. Next we use this view frustum and the known projector resolution ( $W \times H$ ) to recover its intrinsic parameters. Most projectors have a vertical principle center offset to avoid occlusion with the table or the ceiling where the

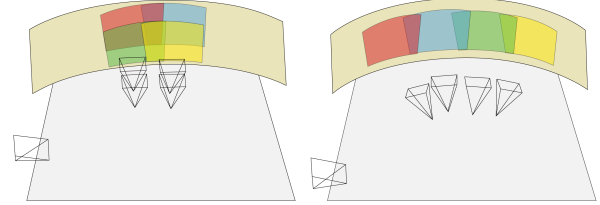


Figure 5: The estimated setup (camera, 3D display and projectors) using our algorithm for the  $2 \times 2$  array (top) and  $1 \times 4$  array of four projectors.

projector is mounted. This results in an offset in the  $y$ -direction for the principle center. We assume that the  $x$ -coordinate of the principle center coincides with the center of the  $x$ -direction. Additionally we do not consider any skew. This results in a simplified intrinsic parameter matrix  $K_p$  for the projectors given by

$$K_p = \begin{pmatrix} f_x & 0 & 0 \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2)$$

Hence, to recover the projector intrinsic parameters, we determine three parameters for each projector: the focal lengths in the two image coordinate directions ( $f_x$  and  $f_y$ ) and the offset in the  $y$ -direction ( $o_y$ ). Our method is absolutely deterministic without using any optimizations and hence is accurate and efficient.

**Estimation of the Extrinsic Parameters:** Let us consider a 3D local coordinate frame for each projector defined by the COP,  $O$ , (position) and axes  $X_p, Y_p$ , and  $Z_p$  (orientation). We use a three step procedure to reconstruct the view-frustum of the projector. (a) First, we find a line that is parallel to  $X_p$  and passes through  $O$ . (b) Second, we find the position of  $O$  on  $X_p$ . (c) Finally, we recover the other two local coordinate axes  $Y_p$  and  $Z_p$ .

**Finding  $X_p$ :** We first sample  $l_T$  and  $l_B$  in the 2D camera space and reproject these samples in 3D using the estimated camera pose and orientation. Each sample defines a 3D ray. We find the intersection of these rays with the display via a line-curve intersection. This gives us the corresponding samples on the 3D curves  $l_T'$  and  $l_B'$ . Note that the samples of  $l_T'$  and  $l_B'$  lie respectively on the top and bottom planes of the projector view frustum,  $T$  and  $B$ . So, first we fit a plane to the samples of  $l_T'$  and  $l_B'$  in a linear least squares sense to estimate  $T$  and  $B$ . Then we find the intersection of  $T$  and  $B$  to find  $X_p$ .

**Finding  $O$ :** The center of projection,  $O$ , is on  $X_p$ . Since the projector view-frustum is symmetric in the horizontal direction, the center  $O$  is a point on the line  $X_p$  constrained by the fact that the two vertical planes formed by the view frustum,  $L$  and  $R$ , should make the same angle with the line  $X_p$  (Figure 4). We first reproject the blobs  $b_A, b_B, b_C$ , and  $b_D$  in 3D using the camera calibration matrix. This generates the four points  $A, B, C$ , and  $D$  where the four rays of the projector view frustum meet the display surface. For

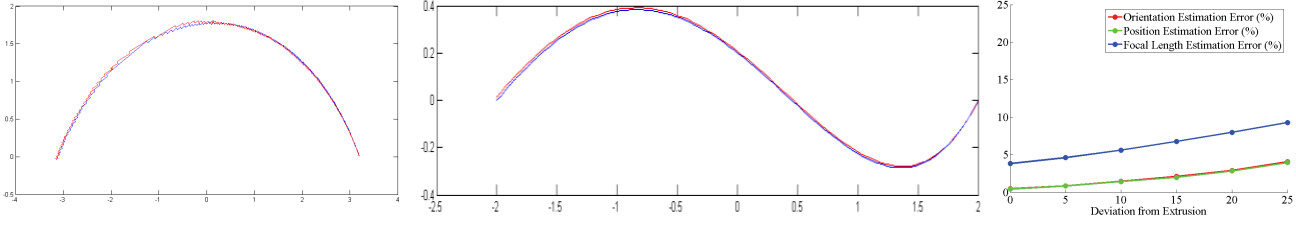


Figure 6: Left: In the real system, red and blue are the estimated 3D top and bottom curves of the display. Middle: In simulation, the blue curve is the original display surface curve and the red curve is the estimation provided by our method. Right: The plot shows the surface registration as the surface deviates from being an extruded surface. This shows that if we assume a roughly tiled configuration and can tolerate 1 or 2 pixel misregistration, we can allow 4% and 6% deviation from an extruded surface respectively.

each of these, let  $A'$ ,  $B'$ ,  $C'$ , and  $D'$  be the projection of  $A$ ,  $B$ ,  $C$  and  $D$  respectively on  $X_p$ . Due to equal angle constraint, the triangles  $OCC'$  and  $ODD'$  will be similar to each other and so will be  $OBB'$  and  $OAA'$ . Thus, the position of  $O$  on  $D'C'$  will be given by the weighted average of  $D'$  and  $C'$  where the weights are inversely proportional to the lengths  $CC'$  and  $DD'$ . This yields to two estimates of  $O$ , one from the similar triangle constraint on  $OCC'$  and  $ODD'$  and another from the similar triangle constraint on  $OBB'$  and  $OAA'$ . A mean of these two estimates results in a robust estimate of  $O$ .

**Finding  $Y_p$  and  $Z_p$ :** We have thus found the four lateral planes of the projector view frustum. Now, we want to find the view direction  $Z_p$ . Note that for any plane  $P$  perpendicular to  $Z_p$ , the length of the intersections of  $P$  with  $OAB$  and  $ODC$  are equal (Figure 4). We use this constraint of equal length to find  $Z_p$ . We first consider two lines on  $OAB$  and  $ODC$ ,  $g_T$  and  $g_B$  respectively. Both are parallel to  $X_p$ .  $g_T$  lies at unit distance from  $O$  on  $OAB$ . Similarly,  $g_B$  lies at unit distance from  $O$  on  $ODC$ . Note that  $g_T$  and  $g_B$  will not have equal length. Assuming  $|g_T| > |g_B|$ , if we move  $g_B$  along  $B$  keeping it parallel to  $X_p$  such that the distance from  $O$  becomes  $\frac{|g_T|}{|g_B|}$ , the new line  $g'_B$  thus formed is equal in length to  $g_T$ . Hence, the plane passing through  $g_T$  and  $g'_B$  satisfies the constraint of equal length (Figure 4).  $Z_p$  is the normal to this plane and  $Y_p = Z_p \times X_p$ .

**Estimation of the Intrinsic Parameters:** Let the resolution of the projector between the four blobs in the pattern be  $P \times Q$ . To find  $f_x$  and  $f_y$ , we first project the 3D points  $A$ ,  $B$ ,  $C$ , and  $D$  on a plane perpendicular to  $Z_p$  and at unit distance from  $O$ . Let these points be  $A''$ ,  $B''$ ,  $C''$ , and  $D''$  respectively. Then,  $f_x$  is given by  $\frac{P}{|A''B''|}$ . Similarly,  $f_y$  is given by  $\frac{P}{|A''C''|}$ . To find  $o_y$ , we consider the center of the 3D line  $AB$ . Since we know the 3D coordinate of this point and  $f_x$  and  $f_y$ , we can find the projector y-coordinate for this point assuming  $o_y = 0$  and subtract  $\frac{Q}{2}$  from it to obtain  $o_y$ .

### 3.3 Geometric Registration

Geometric registration from an arbitrary viewpoint is achieved when an image rendered from the viewpoint is projectively mapped on the 3D display. This kind of registration is especially suitable for virtual reality applications like virtual walkthroughs. However, for other applications like visualization, collaboration, or teleconferencing that tend to have multiple users, correcting for a single viewpoint presents distortions for others. An image wallpapered on a surface has been time tested for multi-viewing purposes (e.g. in museums, airports, and other public places). For such scenarios, the images from the projectors are pasted or seamlessly ‘wall-papered’ on the display surface. In this section, we describe both these registrations.

**Registration from Arbitrary Viewpoint:** After extracting the geometry of the screen we can choose any arbitrary viewpoint and define an image plane for that viewpoint. Afterwards, we can find a mapping between the image plane coordinate system and the screen coordinate system by shooting rays from the viewpoint to the de-

sired image plane coordinates and intersecting these rays with the screen. This mapping can be used then to correct any image for the defined coordinate system. The corrected image will show perspective distortion from other viewpoints and therefore it can be used only for a single user. This is well suited for single-user VR applications which use head tracking to find the proper viewpoint.

**Wallpapered Registration:** Following auto-calibration of the projectors, we use the projector and the display parameters to register the images from the multiple projectors on the display in a ‘wallpaper’ fashion. To wallpaper the image on the display, we seek a 2D length preserving parametrization of the 3D display surface with  $(s, t)$ . As per our setup,  $t$  is automatically parametrized since  $Y = t$ . Also,  $Z = f(X)$ . Hence, we find a curve length based parametrization given by  $s = \int_0^X \sqrt{1 + f'(x)^2} dx$ .

The geometric registration involves finding the function that relates the projector coordinates  $(x, y)$  to the display parameters  $(s, t)$ . Assuming the image to be wall-papered to have the same parametrization of the display, we first cast a ray through each pixel  $(x, y)$  using the auto-calibrated projector coordinates and find the 3D coordinates of its intersection with the cylindrical surface. Then we find the corresponding  $(s, t)$  values and bilinearly interpolate the color in the image to generate the color at the projector pixel  $(x, y)$ .

## 4 EXTENSION TO PIECEWISE PLANAR CAVES

Our algorithm assumes a vertically extruded surface. Since we assume that the top and bottom boundaries of the surface are smooth curves, the algorithm implicitly assumes a smooth vertically extruded surface. However, the basic algorithm remains unchanged even if we have a piecewise linear curve, instead of a smooth one. CAVE like VR setups are built on vertically extruded surfaces with piecewise linear boundaries, and hence our method can be easily extended to such situations. Currently, since it is difficult to calibrate multiple projectors on such surfaces, most CAVE setups use multiple projectors on each of the planar faces, but does not allow overlap of projectors across different faces. This does not allow blending regions for good color calibration [15, 14] and also makes it difficult to achieve automatic geometric calibration across the different planar faces. Our method removes this restriction by allowing the projectors to overlap even across the edges of the planar surfaces (Figure 10).

Our algorithm needs a few small changes to accommodate a CAVE kind of setup. When detecting the top and bottom curves in the camera image, we have to fit a piecewise linear function, instead of a smooth curve. Automatic piecewise linear regression (also referred to as segmented regression) pose ill-conditioned optimization problems for completely unconstrained data sets, but can be solved automatically if constrained sufficiently by providing the breakpoints between the number of segments [10]. Such inputs can easily be provided manually in the image  $I_0$ . For the particular case of displays, where the points on the detected display boundary are extremely structured (very little noise), it is very easy to manually

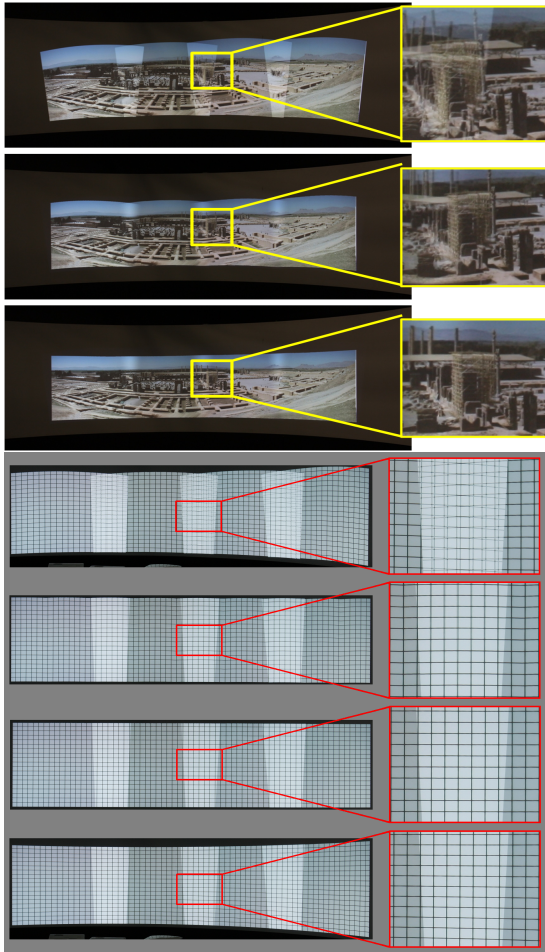


Figure 7: These two sets of images show our geometric registration for the panoramic configuration. The top three images for Persepolis from top to bottom show a naive method, homography, and our method. The bottom four images for a grid from top to bottom show homography, piece-wise linear method with a sparse set of correspondences, piece-wise linear method with a dense set of correspondences, and our method. Please note that the piecewise linear images are only correct from a single view-point and show perspective distortion from any other viewpoint.

identify the piecewise linear line segments in the image  $I_0$ . Hence, we take this route of manual detection of the piecewise linear curve in the image  $I_0$ . Following this manual step, the rest of the process remains unchanged and still automatic.

## 5 RESULTS

We have implemented our method on a cylindrical display using four projectors. We used Epson 1825p LCD projectors (about \$600 each). Our display has a radius of about 14 feet and an angle of 90 degrees. We arranged the projectors in two different configurations: a panoramic configuration where projectors are arranged in a  $1 \times 4$  array (projectors in a row) and a second one where they are arranged in a  $2 \times 2$  array. Our unoptimized matlab implementation of the algorithm takes about 6 minutes. The non-linear optimization for estimating the camera parameters and display geometry takes about 5 minutes. Auto-calibration of the projectors takes about 10 seconds per projector. In the curve-based optimization step, we use  $w_r = 3$  and  $w_c = 1$ .

Figure 5 provides a visualization of the estimated camera, display, and projector locations and orientations in 3D using our algo-

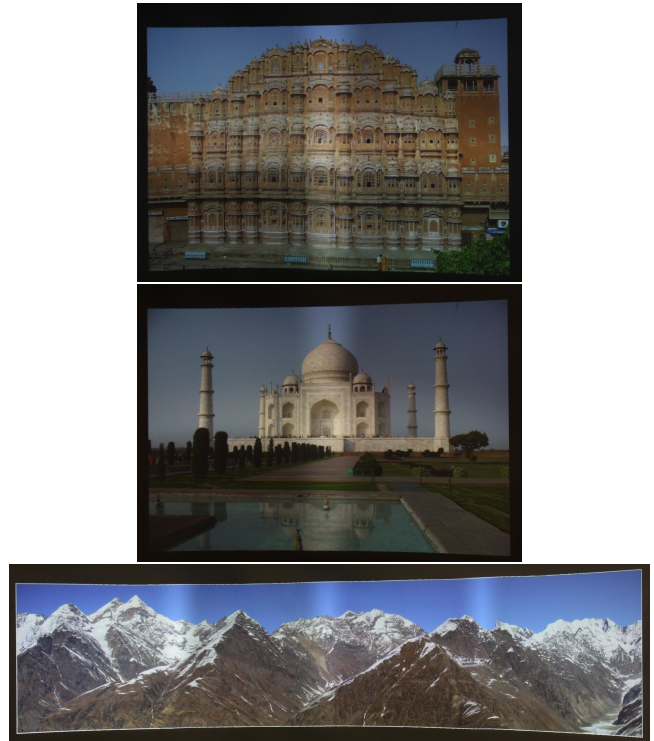


Figure 8: Top two: Geometric registration on our  $2 \times 2$  projector display using our algorithm. Bottom: A registered and wall papered panorama of the Himalayas in the panoramic setup.

rithm for the two different setups. Figure 6 shows the error between the reconstructed top and bottom curves of the display. They coincide demonstrating the accuracy of our method.

The accuracy of our method is demonstrated by a robust geometric registration. Empirically, we have seen a maximum misregistration of less than a pixel. Figure 7, and 8 show the results on our displays. Since all prior methods can achieve geometric registration only with precise physical fiducials or complete 3D reconstruction, it is difficult to find a fair comparison to our method that does not use either of them. In the absence of fiducials, we compare our method with a naive homography-based registration [3, 7] and the registration with respect to a single view-point of the calibrating camera [25]. In addition to the obvious misregistrations, the shape of the display shows that wall-papering cannot be achieved on the curved surface without recovering the 3D geometry of the screen. Please zoom in to see the quality of registration. To reduce the higher brightness in the overlap region, we use a simple cosine blending function [19, 23]. Photometric seams can be further removed by using [15].

To demonstrate that our method is not limited to just cylinders, but can handle any smooth vertically extruded surface, we made an inexpensive flexible display using a rectangular sheet of flexible white styrene. This was supported by five poles to which the styrene sheet was attached (Figure 9). The shape of the profile curve of this extruded display can be changed by simply changing the position of the poles. Figure 9 illustrates the accuracy of our auto-calibration on such a display.

When a projector is moved after auto-calibration, we only need to find the change in the parameters of the moved projector. Since we use a deterministic method, we can achieve this in less than 10 seconds. Thus, we can achieve quick recalibration in the event of a projector movement.

We also demonstrate the extension of our algorithm to CAVES.



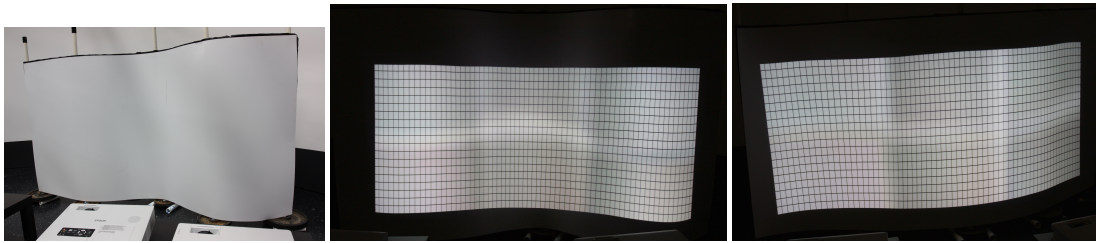


Figure 9: On the left: Our flexible screen. On the right: Geometric registration on our flexible screen with two different shapes.

Table 1: Percentage Errors of the estimated parameters over a large number of simulations with different configuration of the devices and the 3D display.

Parameter	Max	Mean	Std
Camera Orientation (deg)	0.494	0.192	0.167
Camera Position (%)	0.432	0.186	0.150
Top and bottom curves (%)	0.547	0.217	0.153
Projector Position (%)	0.313	0.115	0.972
Projector Orientation (deg)	0.131	0.052	0.050
Projector Focal Length (%)	0.295	0.105	0.895
Projector Offset (%)	1.251	0.486	0.452

Since we do not own a CAVE setup, we tried to create a proof-of-concept demo using our same flexible display setup. Instead of a smooth curve, we rearranged the poles to create a CAVE like setup. Figure 10 shows this setup and the results of our algorithm on it. Note that due to the very flexibility of this display, the curve deviates considerably from a piecewise linear curve. Hence, errors creep in and we cannot achieve a sub-pixel accuracy – but see 2-3 pixels misregistration, as shown in the video. However, in simulation, we can achieve the same sub-pixel accuracy as for our smooth extruded surface.

## 5.1 Evaluation

We have conducted an extensive analysis on the accuracy of the estimated parameters using a simulator. The maximum, mean and standard deviation errors are presented in Table 1. For the orientation of devices (projectors and cameras), we provide deviation in degrees from the actual orientation. For translation, we provide the ratio of the error in estimation with the distance from the screen. For all other parameters, we measure the deviation from the original value of the parameter divided by the original value.

To compare the accuracy of the estimation of the display curves, we sample the estimated curves densely. Then, for each sample, we find the minimum distance to the original curve. The ratio of the maximum of these distances to the length of the original curve is considered to be the accuracy of the display geometry reconstruction and is reported in Table 1. For this, we did not limit the error analysis to cylindrical displays only, but ran experiments with any vertically extruded surface including ones with piecewise linear boundaries like CAVE. To evaluate the accuracy of the geometric registration, we find the deviation of  $(s, t)$  parameter to which a projector pixel will be mapped to in the original setup and compare it with the same from the estimated setup. We find a maximum of 0.3 pixel misregistration from a single projector. Hence, assuming a roughly tiled configuration, the worst case misregistration in any direction will be  $0.3 \times 2 = 0.6$  pixels. This is consistent with our empirical observation of geometric misregistration of less than a pixel. Such accurate registration and calibration for cylindrical tiled displays has never been reported in the literature prior to our work. Finally, we also show the generality of our method for handling any vertically extruded surface (not necessarily a cylinder). The accuracy of reconstructing the display curves are evaluated over extru-

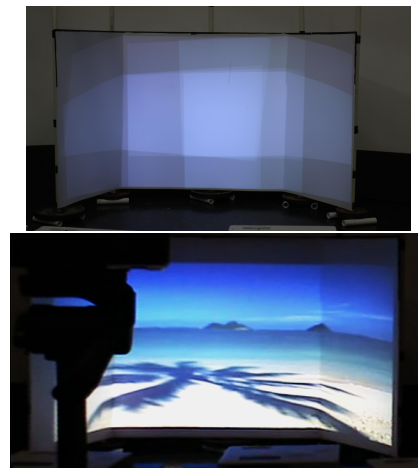


Figure 10: Top: A picture of our CAVE setup with the projectors overlapping on the edges. Bottom: Geometric registration on our CAVE setup. The calibrating camera is visible in the top-left corner of the picture.

sions of different shapes. One example is shown in Figure 6.

## 5.2 Discussion

Most screens designed for commercial purposes are quite rigid infra-structure as is the screen we used for this work. However, we studied the effect of small deviation from extruded surface on the geometric registration in simulation. For this, the deviation is simulated using the same metric as is used to measure the accuracy of estimating the curves. The results in Figure 6 shows that the surface need not be perfectly extruded. 4 to 6% deviation from thereof results in less than 1 to 2 pixel misregistration.

The projectors we used, even if inexpensive, were close to perfect linear devices. However, sometimes they may have small radial distortions. In such a case, a pattern can be used that has more than just two lines. If  $m$  equally placed lines are used,  $X_p$  will be provided by the intersection of the  $m$  planes each containing a planar curve in 3D corresponding to the lines on the projector image plane. When fitting each plane, the eigenvalues of the matrix used for the linear least square fit provides some insights on the 3D curve shape. A small third eigenvalue indicates a degenerate case where the curve is close to a line and one cannot robustly fit a plane. A high fourth eigenvalue indicates a large fitting error, i.e. the curve does not lie on a plane due to the presence of radial distortion in the projectors. Hence, when finding  $X_p$  using linear least squares intersection of the planes, the equations due to each plane can be weighted by a ratio of its third and fourth eigenvalues found during the prior plane fitting step. This assures that curves which indeed lie on a plane are given larger weight than either the degenerate case or when severe radial distortion is present. To avoid an infinite weight resulting from a fourth eigenvalue which is close to 0 (the best case

of a curve robustly lying on a plane), we provide a threshold to the maximum weight. Our simulation shows acceptable registration when using this method in the presence of small radial distortions.

Finally, we found that knowing the intrinsic parameters of the camera is not critical for our method. A large number of image formats like jpg or tiff store EXIF tags for images provide some of the camera parameters. One of these is the focal length, the most important parameter of the intrinsic parameter matrix  $K$  of the camera. To convert the focal length to the unit of pixels, we divide resolution of the camera by the CCD sensor size and multiply it with the focal length specified in the EXIF tags. The sensor size of the camera is available in its specifications. Also, in most cameras today, it is common to have the principal center at the center of the image, no skew between the image axes, and square pixels. Hence, similar to [21], we use these assumptions to initialize the intrinsic parameter matrix of a camera,  $K$ , as

$$K = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (3)$$

Our non-linear optimization can accurately refine the single parameter in the intrinsic matrix and we do not see any degradation in the quality of the registration.

## 6 CONCLUSION

In summary, we have presented the first work to auto-calibrate projectors on vertically extruded surfaces without using display to camera correspondences. Our projector auto-calibration is achieved via a deterministic efficient algorithm that allows interactive changes in the projector position, orientation and zoom factor. Our method can have tremendous applications in auto-calibration of large cylindrical displays commonly used for edutainment purposes. It can also be extended to CAVEs to allow projector overlap across the planar screens of the CAVE.

However, our method is limited to extruded surfaces and cannot handle another kind of commonly used screens, the domes. In future, we would like to extend similar fundamentals of using prior knowledge of the screen to design methods to achieve geometric registration without the use of fiducials on other types of screen. Further, we would like to design a variation of our method that can tolerate greater deviation from extruded surfaces. Reasonable deviation from perfectly extruded surfaces will allow lower precision in the screen manufacturing, making these displays more affordable.

## ACKNOWLEDGEMENTS

We would like to acknowledge our funding agencies NSF IIS-0846144. We would like to acknowledge Maxim Lazarov for helping us creating the video and building the flexible vertically extruded surface. We would like to thank the members of the Creative Technologies Group at Walt Disney Imagineering for helping us to test our algorithm successfully on their virtual reality system.

## REFERENCES

- [1] D. Aliaga. Digital inspection: An interactive stage for viewing surface details. *Proc. ACM Symp. on 13D*, 2008.
- [2] D. Aliaga and Y. Xu. Photogeometric structured light: A self-calibrating and multi-viewpoint framework for accurate 3d modeling. *Proc. of IEEE CVPR*, 2008.
- [3] M. Ashdown, M. Flagg, R. Sukthankar, and J. M. Rehg. A flexible projector-camera system for multi-planar displays. *Proc. of IEEE CVPR*, 2004.
- [4] E. Bhasker, R. Juang, and A. Majumder. Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. *IEEE TVCG*, 13(6):1368–1375, 2007.
- [5] E. Bhasker, P. Sinha, and A. Majumder. Asynchronous distributed calibration for scalable reconfigurable multi-projector displays. *IEEE Transactions on Visualization and Computer Graphics (Visualization) - To Appear*, 2006.
- [6] M. Brown, A. Majumder, and R. Yang. Camera based calibration techniques for seamless multi-projector displays. *IEEE TVCG*, 11(2), March-April 2005.
- [7] H. Chen, R. Sukthankar, G. Wallace, and K. Li. Scalable alignment of large-format multi-projector displays using camera homography trees. *Proc. of IEEE Vis*, 2002.
- [8] D. Cotting, M. Naef, M. Gross, , and H. Fuchs. Embedding imperceptible patterns into projected images for simultaneous acquisition and display. *International Symposium on Mixed and Augmented Reality*, pages 100–109, 2004.
- [9] D. Cotting, R. Ziegler, M. Gross, and H. Fuchs. Adaptive instant displays: Continuously calibrated projections using per-pixel light control. *Proc. of Eurographics*, pages 705–714, 2005.
- [10] G. Ferrari-Trecate and M. Muselli. A new learning method for piecewise linear regression. *ICANN Lecture Notes in Computer Science, Eds. J. D. Borronoro, Berlin: Springer, 2415:444–449*, 2002.
- [11] M. Harville, B. Culbertson, I. Sobel, D. Gelb, A. Fitzhugh, and D. Tanguay. Practical methods for geometric and photometric correction of tiled projector displays on curved surfaces. *IEEE PROCAMS*, 2006.
- [12] T. Johnson and H. Fuchs. Real-time projector tracking on complex geometry using ordinary imagery. *IEEE CVPR Workshop on Projector Camera Systems (PROCAMS)*, 2007.
- [13] T. Johnson, G. Welch, H. Fuchs, E. L. Force, and H. Towles. A distributed cooperative framework for continuous multi-projector pose estimation. *IEEE Virtual Reality Conference*, pages 35–42, 2009.
- [14] A. Majumder and R. Stevens. Color nonuniformity in projection-based displays: Analysis and solutions. *IEEE Transactions on Vis and Computer Graphics*, 10(2), March–April 2003.
- [15] A. Majumder and R. Stevens. Perceptual photometric seamlessness in tiled projection-based displays. *ACM TOG*, 24(1), January 2005.
- [16] A. Raij, G. Gill, A. Majumder, H. Towles, and H. Fuchs. Pixelflex 2: A comprehensive automatic casually aligned multi-projector display. *IEEE PROCAMS*, 2003.
- [17] A. Raij and M. Polleyfeys. Auto-calibration of multi-projector display walls. *Proc. of ICPR*, 2004.
- [18] R. Raskar, J. V. Baar, T. Willwacher, and S. Rao. Quadric transfer function for immersive curved screen displays. *Eurographics*, 2004.
- [19] R. Raskar, M. Brown, R. Yang, W. Chen, H. Towles, B. Seales, and H. Fuchs. Multi projector displays using camera based registration. *Proc. of IEEE Vis*, 1999.
- [20] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM TOG*, 22(3), 2003.
- [21] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. In *SIGGRAPH Conference Proceedings*, pages 835–846, New York, NY, USA, 2006. ACM Press.
- [22] W. Sun, I. Sobel, B. Culbertson, D. Gelb, and I. Robinson. Calibrating multi-projector cylindrically curved displays for "wallpaper" projection. *IEEE/ACM Workshop on PROCAMS*, 2008.
- [23] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. S. Brown. Pixelflex: A reconfigurable multi-projector display system. *Proc. of IEEE Vis*, 2001.
- [24] R. Yang and G. Welch. Automatic projector display surface estimation using every-day imagery. *9th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2001.
- [25] R. Yang, G. Welch, and G. Bishop. Real-time consensus-based scene reconstruction using commodity graphics hardware. *Proceedings of Pacific Graphics*, 2002.
- [26] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. *International Conference on Computer Vision*, 1999.
- [27] J. Zhou, L. Wang, A. Akbarzadeh, , and R. Yang. Multi-projector display with continuous self-calibration. *IEEE/ACM Workshop on Projector-Camera Systems (PROCAMS)*, 2008.
- [28] S. Zollmann, T. Langlotz, and O. Bimber. Passive-active geometric calibration for view-dependent projections onto arbitrary surfaces. *In Workshop on Virtual and Augmented Reality of the GI-Fachgruppe AR/VR*, 2006.

# Display Gamut Reshaping for Color Emulation and Balancing

## Abstract

Emerging next generation digital light projectors are using multiple LED/laser sources instead of one white lamp. This results in a color gamut much larger than any existing display or capture device. Though advantageous in theory, when used to display contents captured/processed at a smaller gamut, a large gamut expansion results in hue-shift artifacts.

We present a hardware-assisted 3D gamut reshaping method that handles the gamut expansion in LED based DLP displays by hierarchical temporal multiplexing of the multiple primaries. This, in turn, results in a color emulation technique by which projectors with such large gamuts can also achieve a standard color gamut and white point – the two most important color properties in terms of display quality, with an additional advantage of increased brightness and dynamic range. The same method can also be used for color balancing across multiple projectors that are often used to create large-scale high resolution displays.

## 1. Introduction

The traditional digital light projection (DLP) technology includes a white light bulb and a color wheel with differently colored filters. The filters are temporally multiplexed at a high speed to selectively pass any one of the multiple primaries at any instant of time on to the digital micromirror device (DMD) array [34]. The number of filters on the color wheel can be three (R,G and B), four (R,G, B and W) or more [30, 13, 24, 1, 25, 6, 15](Figure 1). These are wide band filters creating a gamut smaller than the standard industry-specified gamuts like NTSC, PAL and HDTV (Figure 2). Hence, media in one of these standard gamuts is mapped to the smaller gamut of the display.

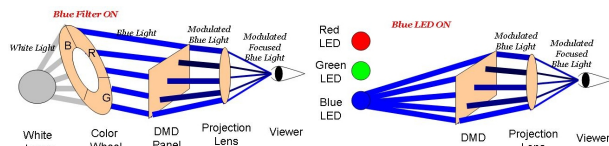


Figure 1. Left to right: Light Path of a traditional DLP projector and a DLP projector with multiple LED sources (Blue filter is ON).

The projection industry has recently introduced projectors where the color wheel is eliminated by using multi-

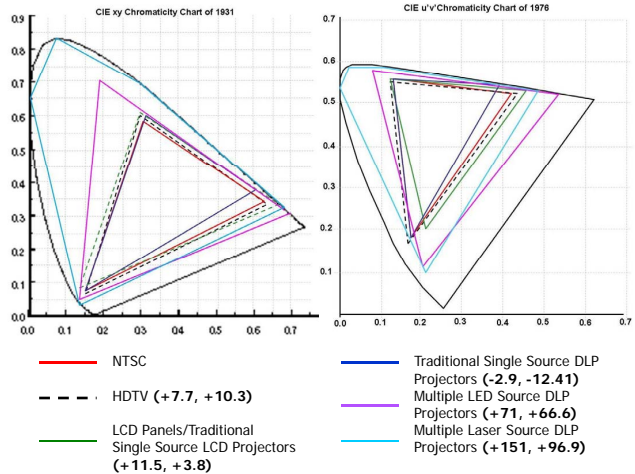


Figure 2. Comparison of the different standard 2D color gamuts with the gamuts provided by the LED or laser based projectors, both in CIE xy and u'v' space. The bold numbers indicate the percentage deviation in the area of the 2D gamut when compared to the NTSC gamut, both in CIE xy and u'v' space respectively

ple light sources, one for each primary, created from one or more light emitting diodes (LEDs) [8, 29, 10]. The primaries are then switched ON and OFF or multiplexed temporally independent of each other (Figure 1).

The LEDs in these projectors provide more saturated colors than the color wheel resulting in a much *larger color gamut* than any traditional projector and standard color gamuts like NTSC, PAL, and even the most recent HDTV (56% larger in the CIE u'v' chromaticity chart). In fact, the emerging laser projectors, due to monochromatic primaries, promise to provide an even larger color gamut, covering almost all the colors visible to the human eye (almost double than that of the NTSC gamut in the CIE u'v' space) [20]. The percentage increase/decrease of different display gamuts when compared to the NTSC gamut both in the CIE xy and u'v' chromaticity charts is quantified in Figure 2.

Though larger color gamut assures reproducibility of a larger range of chrominance, this causes *gamut expansion* creating several problems (e.g. hue-shifts, white-point shift and non-optimal utilization of color resources) when displaying existing media generated in devices with a much smaller gamut(Section 3). As a result, these upcoming projectors cannot be used in applications using projectors and cameras in a tightly coupled feedback loop

[18, 17, 21, 37, 19, 35, 28]. When the projected images are captured by the lower gamut cameras severe gamut clipping artifacts occur.

In this paper, we present an algorithm to address this gamut expansion. Unlike traditional single source projector architecture where at any particular instance of time only one of the primaries can be turned on, in projectors with multiple LED sources more than one primary can be turned on at the same time. Our method takes advantage of this key property of *simultaneous ON times* to design a *hardware assisted scheme of hierarchical temporal multiplexing* of the LED primaries that can emulate a standard color gamut and white point without compromising other properties like brightness, contrast and light efficacy, and enables the following.

1. *Dynamic Color Emulation*: Operability at standard color gamut (like HDTV, NTSC and PAL) and white point (like D85 and D65) is very important for any display. Our method *emulates* many different color gamut and white point standards from the *same* set of LED primaries *dynamically* as demanded by the application, just by changing the parameters of the temporal multiplexing (Section 4). These parameters can be precomputed automatically and then stored in the projector itself.

2. *Robustness to Manufacturing Imprecision*: The only way to achieve a desired color specification in a traditional single source projector is to control the color properties of their color filters via precision manufacturing. Since our method can achieve the same standard color properties from LEDs that have a large variation in color, such strict control in manufacturing can be avoided. This can make the technology more flexible and cost effective.

3. *Color Balancing in Multi-Projector Displays*: Finally, the same hierarchical scheme can also be used to achieve color balancing across multiple projectors, common for building large-area high-resolution displays (Section 4.2).

## 2. Notation

We first briefly describe our color notation. Our algorithm involves only *color matching* and does not deal with color distances. Hence, all computations in our algorithm are carried on in CIE XYZ space. However, when evaluating the display quality, we use a perceptually uniform color space, like CIELAB or CIELUV space.

Let  $(X, Y, Z)$  be the 3D coordinates of a color in the CIE XYZ space, called the tristimulus values. In our algorithmic computations, total tristimulus value (TTV)  $X + Y + Z$  (the indicator of the total energy of the spectrum) plays an important role. Hence, we specify the  $(X, Y, Z)$  color alternatively by its TTV  $I = X + Y + Z$ , and its chromaticity coordinates (the indicator of its *chrominance*),  $(x, y)$ , defined as

$$(x, y) = \left( \frac{X}{X + Y + Z}, \frac{Y}{X + Y + Z} \right). \quad (1)$$

The XYZ coordinates of a color can be derived easily from  $(I, x, y)$  using

$$(X, Y, Z) = (xI, yI, I(1 - x - y)). \quad (2)$$

Further, matching two colors,  $(I_1, x_1, y_1) = (I_2, x_2, y_2)$  assures that they also match in their XYZ coordinates i.e.  $(X_1, Y_1, Z_1) = (X_2, Y_2, Z_2)$ . Finally, the most important point to note is that, for colors of similar chrominance,  $I$  scales proportionally to the luminance  $Y$ . Hence, in displays, for considering each primary or the grays,  $I$  and  $Y$  are both scaled equally when the inputs are scaled.

It can be shown that in the CIE XYZ space, a ray through the origin is the locus of colors with the same chromaticity coordinate  $(x, y)$  but different TTVs  $I$ . The chromaticity coordinates  $(x, y)$  is a 2D projection of these rays on the  $X + Y + Z = d$  plane. The set of all chrominance visible to the human eye creates a horse-shoe shaped plot in the  $xy$  space that represents different chrominance values phasing out the  $I$ . This is called the chromaticity chart (Figure 2). The point  $(0.33, 0.33)$  in this chart indicates a perfect achromatic color with  $X = Y = Z$ . As the colors move away radially from this point towards the periphery of the horse-shoe shape, they change in saturation, while the hue remains constant.

Finally, it can be shown with simple algebra, that adding two colors,  $(I_1, x_1, y_1)$  and  $(I_2, x_2, y_2)$ , result in a color  $(I_3, x_3, y_3)$  where  $I_3$  is the sum of the TTVs of the superimposing colors and chrominance is the weighted convex combination of the chrominance of the superimposing colors in the  $xy$  chromaticity chart, where the weights are given by the proportion of their TTVs. In other words,

$$(I_3, x_3, y_3) = \left( I_1 + I_2, \frac{x_1 I_1 + x_2 I_2}{I_1 + I_2}, \frac{y_1 I_1 + y_2 I_2}{I_1 + I_2} \right). \quad (3)$$

This result can be generalized to  $n$  colors, where the chrominance of the new color lies within the convex hull of the chrominance of the constituting  $n$  colors. Thus, in a projector with three primaries, the reproducible chrominance lies within the triangle spanned by the chrominance of the three primaries (Figure 2). This is called the *2D color gamut* or simply the *color gamut*. The chrominance of the white created by full intensity primaries superimposed from three channels is called the *white point*. This depends on the proportion of the TTVs of the three primaries and need not be the perfect white,  $(0.33, 0.33)$ . The *brightness* of a device is defined by the luminance  $Y$  of full intensity white, and is proportional to its TTV. Note that the white point and 2D color gamut specifies the chrominance capabilities while brightness specifies the dynamic range capabilities. If the brightness of black is constant, a higher brightness indicates a higher dynamic range. When considering brightness and chrominance together, the XYZ values of the primaries

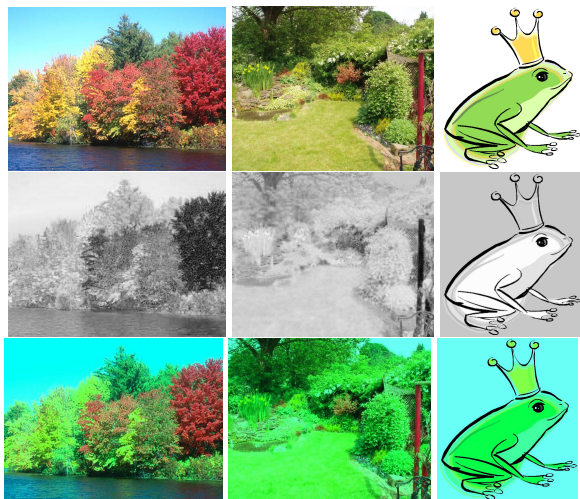


Figure 3. Effects of No Color Management – Row 1: Images in NTSC color gamut; Row 2: The grayscale representation of normalized hue shift in the perceptually uniform CIELAB space due to gamut expansion in LED projectors (Brighter grays indicate more hue-shift) – the normalization is with respect to the maximum hue shift seen in each image; the max and mean hue shift in the three images from left to right are (52.5, 32.8), (55.2, 39.1) and (54.3, 41.8) respectively; Row 3: The images displayed on the LED projector with a larger 2D color gamut is then recaptured by a standard NTSC camera. Note that due to shifting of colors outside of the NTSC space during display, the recaptured image loses many of the colors due to gamut clipping.

span a 3D parallelepiped in the CIE XYZ space. This constitutes the range of all different colors (both chrominance and brightness) that can be reproduced by the device, called the *3D color gamut*.

### 3. Gamut Expansion and Related Work

A smaller gamut media displayed on a larger gamut display without applying any color management techniques (i.e., no modification of content) can show visible *color incoherence* (jarring hue-shifts) due to the increase in the perceptual distance between two perceptually coherent colors. Figure 3 visualizes this hue-shift at every pixel in perceptually uniform CIELAB space as a gray scale image normalized with respect to its maximum hue-shift in the image (since this cannot be captured or printed in a device with a smaller gamut than the LED projector). Since a hue-shift of 3-5 in CIELAB space is easily visible [38], the hue-shifts resulting from the gamut expansion (between 30-55) is very significant. Note that the color shifts are predominantly in the red-yellow (fall image) and green-yellow (garden image) region of the chromaticity chart where most of the gamut expansion occurs (Figure 2). It also causes a significant *white point shift* (e.g. frog image). A NTSC gamut with standard D65 white (chromaticity coordinate of (0.3127, 0.3290)) shifts considerably to the greenish-blue white region (0.272, 0.369) when used for the larger

gamut, display. Hence, in any projector camera application [18, 17, 21, 37, 19, 35, 28, 22] when images of the gamut expanded media on an LED projector is captured by a lower gamut camera, gamut clipping results in severe color blotches (Figure 3).

Gamut expansion is prevalent in a much smaller scale when moving from the smaller color gamut of a printer to a larger gamut of a monitor, but not as pronounced as in the context of projection-based displays, especially when using multiple primaries [15, 20]. However, a similar scenario is prevalent currently in the context of dynamic range of displays [27] and has led to development of inverse tone mapping methods that map LDR content to HDR displays [23, 2]. Gamut being a 2D/3D entity, as opposed to 1D dynamic range, makes gamut expansion a more complex problem. In this section, we briefly visit the relevant works in this direction.

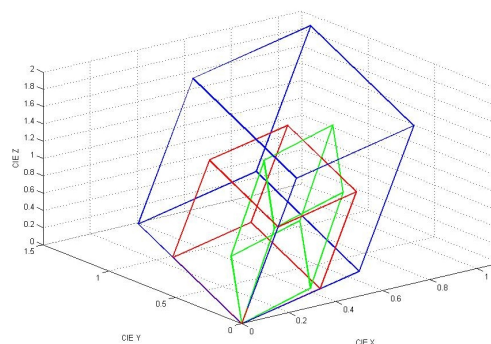


Figure 4. Comparisons of the 3D gamuts in CIE XYZ space of a traditional projector (green) and a LED projector assuming their primaries have the same TTV (red), and 1.5 times the TTV (blue).

**Gamut Clipping:** Standard gamut clipping techniques [33, 7, 14, 5] are currently used when converting between gamuts of similar shape and volume. The source input  $0 \leq (r, g, b) \leq 1$  is mapped to a target color  $(r', g', b')$  using a linear transformation  $M$ , i.e.  $(r', g', b')^T = M(r, g, b)^T$ . The resulting  $(r', g', b')$ , if outside the target 3D gamut is clipped to the boundary of the 3D gamut. However, this can have some adverse effects on gamut expansion. First, a larger 2D color gamut of the LED projector does not necessarily indicate a larger 3D color gamut. If the luminance (and hence the TTVs) of the primaries are similar, the standard NTSC 3D gamuts may be significantly different in shape and not contained within the LED projector's 3D gamut (Figure 4). Hence, clipping of transformed colors still maps multiple source colors to a single target color resulting in *color blotches* (Figure 3 bottom row).

**Gamut Extension:** Instead of clipping only the out-of-gamut colors, other techniques aim at moving all different colors in an optimized fashion from the smaller source gamut to the larger target gamut. [12, 26] apply hue-preserving color extrapolation by changing only the brightness (and hence TTVs) and saturation, and [15, 9] apply

complex non-linear optimizations on perceptually uniform 3D color spaces constraining the movement of colors within an acceptable distance. However, since the increase in gamut volume in LED projectors is much more drastic than what is expected in these methods, they result in perceivable hue shift [15] and may not utilize the entire gamut.

**Comparison of Proposed Method:** Our method is complementary to all the above methods. In contrast to complex gamut extension techniques [12, 5, 14, 15, 26, 9], the biggest advantage of our method lies in its sheer simplicity - both in concept and in computation. Unlike existing gamut extension techniques our method can be directly implemented using the existing DLP hardware. Further, our method can exactly emulate a predefined 2D gamut or white point, without working within the confines of the 3D gamut provided by an optimization. Finally, the hierarchical nature of our method (Section 4.1.3) allows a larger flexibility in terms of the emulated properties and extension to multi-projector displays (Section 4.2).

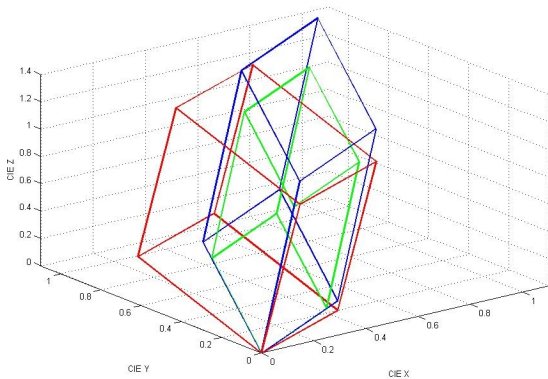


Figure 5. The 3D NTSC gamut (green), 3D gamut of the LED projector before our method is applied (red) and the reshaped 3D gamut of the projector after our method is applied (blue), in CIEXYZ space. The projection of the three basis vectors that span the 3D gamut on a  $X+Y+Z=d$  plane is the 2D color gamut. Note that after applying our method, the basis vectors of the reshaped 3D gamut is coincident with the NTSC gamut but is bigger. This assures that the extra 3D volume is used to increase the brightness (and hence the dynamic range), rather than the 2D gamut.

#### 4. Hardware Assisted Gamut Reshaping

Most of the extra volume of the 3D gamut of LED projectors (Figure 4) is used to reproduce a larger range of chrominance, while the range of reproducible brightness, given by the proportional TTVs, is still similar. We present a content-independent hardware-assisted algorithm that *reshapes* this 3D gamut using hierarchical temporal multiplexing, so that most of this extra volume is instead utilized to increase the brightness, while the 2D color gamut is matched to a standard 2D color gamut (Figure 5).

Our algorithm uses this property of *simultaneous ON time* of the primaries to design a scheme to superimpose

the multiple primaries in a controlled, deterministic (non-iterative) manner so that any specified 2D color gamut and white point that lies within the gamut can be achieved.

#### 4.1. Color Emulation for a Single Projector

Our method takes two sets of *inputs* – (a) the target (desired) color specifications, i.e., chromaticities of the primaries and white point; (b) the chromaticity coordinates and the TTV,  $(I, x, y)$ , of the LED primaries at full intensity, measured by a radiometer. The *output* is the pulse width modulated ON-times for superimposing the primaries to achieve the target specifications. Note that this is applicable even to systems with more than three primaries.

Let the measured color properties of the LED primary  $l$ ,  $l \in \{R, G, B\}$ , be  $(I_l, C_l)$  where  $C_l = (x_l, y_l)$  is the chromaticity coordinate of  $l$ . Let the target 2D color gamut be defined by the new primaries  $R', G'$  and  $B'$  whose chromaticity coordinates are  $C_{l'}$ ,  $l' \in \{R', G', B'\}$  and the white point chrominance is  $C_{W'} = (x_{W'}, y_{W'})$ .

The output of the different intensity levels in LED-based DLP projectors are achieved via temporal multiplexing of the ON times of each LED primaries,  $R, G$  and  $B$  [11]. This forms the Level 0 of our hierarchical scheme. Each of the successive levels run two methods: (a) *temporal-modulation* that determines the relative ON times of the different LEDs which will be turned ON simultaneously to achieve the target specification; (b) *TTV-computation* that determines the TTV of the new primaries thus formed by the temporal modulation step. Figure 6 illustrates the method.

##### 4.1.1 Gamut Emulation in Level 1

The goal of this step is to superimpose the colors from more than one LED to transform the larger 2D gamut to the smaller standard 2D gamut like NTSC (Figure 2).

**Temporal Modulation:** We control the ON times of the three LEDs to realize the proportions of the LED primaries that achieve the target primaries.  $t_{ij}$ , denotes the ON-time for LED primary  $i$ ,  $i \in \{R, G, B\}$ , required to create the target primary  $j$ ,  $j \in \{R', G', B'\}$ . For e.g.  $t_{GR'}$  denotes the ON-time of the green LED primary to create the target red primary.

First, we compute a  $3 \times 3$  matrix that transforms the triangle  $C_R C_G C_B$  to  $C_{R'} C_{G'} C_{B'}$ , given by

$$\begin{pmatrix} C_{R'} \\ C_{G'} \\ C_{B'} \end{pmatrix} = \begin{pmatrix} d_{R'} & e_{R'} & f_{R'} \\ d_{G'} & e_{G'} & f_{G'} \\ d_{B'} & e_{B'} & f_{B'} \end{pmatrix} \begin{pmatrix} C_R \\ C_G \\ C_B \end{pmatrix} \quad (4)$$

where  $d_{l'}$ ,  $e_{l'}$ ,  $f_{l'}$  denote the proportion of  $R, G$  and  $B$  (barycentric coordinates) required to create the target primaries  $l'$ ,  $l' \in \{R', G', B'\}$ . Since  $f_{l'} = 1 - d_{l'} - e_{l'}$ , there are only six unknowns which are provided by the known  $C_l$  and  $C_{l'}$ .

For  $I_R = I_G = I_B$ ,  $d_{l'}$ ,  $e_{l'}$  and  $f_{l'}$  can be directly assigned to  $t_{Rl'}$ ,  $t_{Gl'}$  and  $t_{Bl'}$  respectively. However, since

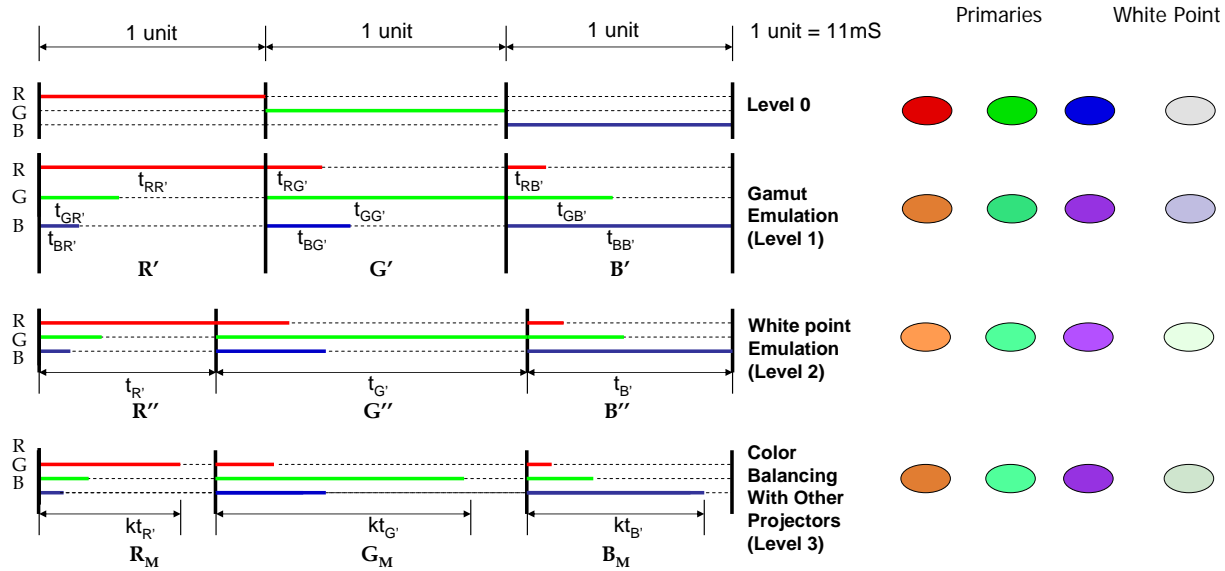


Figure 6. This shows the 33ms time interval for displaying a single frame of a 30fps video. The colored bars show how much time each of the primaries are ON in each interval for each level of the hierarchy. The circles on the right hand side show how the color properties of the display changes with each level. Note that this is just an illustration of the computation and does not correspond to any real data.

$I_R \neq I_G \neq I_B$  in reality, the ON times  $t_{R'}$ ,  $t_{G'}$  and  $t_{B'}$  required to achieve the target chrominance  $C_{l'}$  must take this into consideration. For example, if  $I_R < I_G$ , then  $I_R$  should be kept ON longer to provide the desired TTV proportional to  $d_{l'}$ . Thus, the timing is a function of both the chrominance and TTV of the source LED primaries. For example, the  $t_{lR'}$ ,  $l \in \{R, G, B\}$  is given by

$$t_{RR'} = d_{R'} \times \frac{I_R + I_G + I_B}{I_R} \quad (5)$$

$$t_{GR'} = e_{R'} \times \frac{I_R + I_G + I_B}{I_G} \quad (6)$$

$$t_{BR'} = f_{R'} \times \frac{I_R + I_G + I_B}{I_B} \quad (7)$$

Since,  $R'$  is formed by the superposition of all three of  $R, G$  and  $B$ , these timings are normalized by the *maximum* of  $t_{RR'}$ ,  $t_{GR'}$  and  $t_{BR'}$ . For simplicity, we retain the same notation for the normalized timings. Similar computations are performed to compute the timings for  $G'$  and  $B'$ . For generating  $C_{R'}$ ,  $t_{RR'}$  will usually be much larger than  $t_{GR'}$  and  $t_{BR'}$  since  $C_{R'}$  is much closer to  $C_R$  than to  $C_G$  and  $C_B$ . Hence, for  $C_{R'}$ ,  $t_{RR'} = 1$  after normalization. Similarly, after normalization,  $t_{GG'} = 1$  and  $t_{BB'} = 1$  for  $C_{G'}$  and  $C_{B'}$  respectively.

**TTV Computation:** The temporal modulation creates new primaries  $R', G'$  and  $B'$  whose TTV we compute next. The TTV of the new primary  $l' \in \{R', G', B'\}$ , is  $I_{l'}$  and is given by

$$I_{l'} = t_{R'l'}I_R + t_{G'l'}I_G + t_{B'l'}I_B \quad (8)$$

Since  $t_{R'l'} = 1$ ,  $I_{l'} > I_l$ . Thus, the new primaries are brighter (due to superimposition of additional light from

other primaries) leading to a brighter projector.

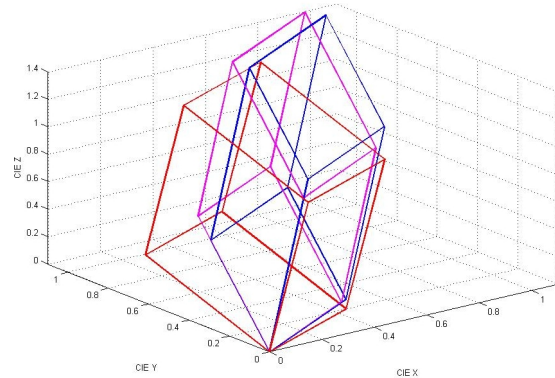


Figure 7. The 3D gamut in CIEXYZ space at Level 0 (red), Level 1 (magenta) and Level 2 (blue) of our method.

#### 4.1.2 White Point Emulation in Level 2

This step realizes the target white point without affecting the new primaries achieved in the previous level using the same steps of *temporal modulation* and *TTV computation* but with the new primaries  $R', G'$  and  $B'$ .

**Temporal Modulation:** We modify the contributing proportions of the new primaries,  $R', G'$  and  $B'$ , in such a manner that the chromaticity of target white point,  $C_{W'}$ , is matched. Let this target proportion be  $p_{R'} : p_{G'} : p_{B'}$  where  $p_{B'} = 1 - p_{R'} - p_{G'}$ . The equation is given by

$$p_{R'}(x_{R'}, y_{R'}) + p_{G'}(x_{G'}, y_{G'}) + p_{B'}(x_{B'}, y_{B'}) = (x_{W'}, y_{W'}) \quad (9)$$

Thus, the ON times for the new primaries to achieve the

Desired Gamut/ White Point	Increase in CIE Y		Decrease in Volume of 3D Gamut	
	Lvl 1	Lvl 2	Lvl 1	Lvl 2
HDTV/D65 (USA)	20.8%	17.1%	3.0%	5.5%
HDTV/D85 (Japan/Korea)	20.8%	19.1%	3.0%	11.1%
NTSC/D65 (USA)	23.4%	20.1%	4.4%	7.2%
NTSC/D85 (Japan/Korea)	23.1%	21.8%	4.4%	13.8%
PAL/D65 (Europe/India)	20.1%	15.9%	2.8%	5.5%

Table 1. Statistics of how the brightness and volume of the 3D gamut of the projector changes in Level 1 and Level 2 from its original state in Level 0.



Figure 8. This figure demonstrates the white point emulation. The original non-standard reddish white (0.35,0.325) in CIE xy space (left) has been changed to a neutral D65 white (middle) and a bluish D85 white (right) using our white emulation.

target white point is

$$t_{I'} = p_{I'} \times \frac{\sum_{I' \in \{R', G', B'\}} I_{I'}}{I_{I'}} \quad (10)$$

In this step, the normalization of the timings is different. Unlike Level 1 where the LED primaries  $R$ ,  $G$  and  $B$  were superimposed, in Level 2 the primaries  $R'$ ,  $G'$  and  $B'$  are lighted sequentially for a total of 3 units of time. Hence,  $t_{I'}$  should be normalized by  $\frac{t_{R'} + t_{G'} + t_{B'}}{3}$ . Here also, we retain the same notation for the normalized timings for simplicity. **TTV Computation:** Hence, the TTV of the new primaries is given by  $t_{I'} I_{I'}$ . The TTV of the white,  $I_{W'}$ , is the sum of the TTVs of these new primaries  $I_{W'} = \sum_{I' \in \{R', G', B'\}} t_{I'} I_{I'}$ .  $I_{W'}$  is proportional to the luminance of the white which is the measure of the display brightness. In Section 4.1.3, we show that  $I_{W'}$  is greater than  $I_W = I_R + I_G + I_B$  in the original projector.

#### 4.1.3 Discussion

This section offers some useful insights to the transformations achieved by our method.

**Reshaping of the 3D Gamut:** Figure 7 shows how the 3D gamut reshaping happens across different levels of our hierarchical scheme. The 3D gamut reduces in volume from Level 1 to 2 due to the constraints imposed by a specific target white point. However, the final 3D gamut is always bigger than the standard industry specified 3D color gamut but smaller than the original LED 3D gamut.



Figure 9. Please zoom in to see our curved screen display made of four projectors (real system, not simulated) running our color balancing algorithm. Top: This shows the whites before (left) and after (right) our color emulation matching to an NTSC gamut. Note that the brightness balancing across projectors described in Section 4.2 is still not applied. Hence, intensity variation and seams are still visible. Middle: This shows a regular content (Venice) before (left) and after (right) our color emulation for each projector to match the NTSC gamut, followed by our TTV (and hence brightness) balancing across projectors and finally removing spatial variation in TTV (and hence in brightness) using existing camera-based calibration methods. Bottom: In this zoomed in views, compare the colors of the buildings near the text Hotel Marconi to see the effect of our color balancing.

Table 1 compares the volume of the 3D gamut and the TTV of the white ( $I_{W'}$ ), finally achieved by the color emulation method. Note that since for the equi-chrominance grays of a display,  $I$  and luminance  $Y$  are scaled similarly with the input (Section 2),  $I_{W'}$  provides a direct measure of the increase in the display brightness. However, the loss in the volume is very small when compared to the gain in brightness, which can be as large as 25%.

**Hierarchical Nature of the Scheme:** Our method is hierarchical in nature where each level of the hierarchy modifies the primaries which are then used as new primaries in the subsequent level. This provides a few nice properties to our algorithm. (a) *Level Independency:* In each step of the hierarchy, only one property of the display is modified *independently* (e.g. 2D color gamut in Level 1 and white point in Level 2). (b) *Preservation of Lower Level Properties:* A color property standardized at a particular level of the hierarchy is preserved through the subsequent levels. (e.g. changing white point in Level 2 does not change the new primaries  $R'$ ,  $G'$  and  $B'$ ). (c) *Module Invariance:* The same computations are used in each level of the hierarchy but with different inputs to impact different properties.

**Flexibility of the Scheme:** Note that we can precompute timings to achieve multiple different specifications (for e.g. HDTV and D65 white or NTSC and D85 white) and store them in a look-up-table (LUT) in the projector itself. The application software can then use this to create different color properties as the desired white point or gamut used in devices changes from country to country (from NTSC





Figure 10. This shows our planar display made of 16 projectors (real system, not simulated) after color balancing using the following three steps in succession: (a) color emulation of each projector to NTSC gamut (Section 4); (b) color balancing across projectors (Section 4.2); (c) removal of spatial vignetting effects by existing camera-based registration methods.

in US to PAL in Europe/India, from D65 in US to D85 in Japan/Korea).

#### 4.1.4 Implementation and Results

We tested our color emulation on over 20 different projectors to realize NTSC gamut and D65 color point. The time multiplexing was realized using the DLP chip hardware. Figure 8 shows the results of changing the white point from the existing white point (shifted towards red) to a D65 (neutral) and D85 (shifted towards blue) white. Since, the saturated primaries of the projectors cannot be reproduced either in print or in existing displays, we provide statistical data on the accuracy of the gamut mapping. Our red, green and blue LEDs had mean dominant wavelength of 617nm, 520nm green and 464nm with a variation about 16nm, 8nm and 6nm respectively. We achieved an NTSC gamut for all the projectors up to an absolute error of  $10^{-4}$  in the CIE xy chromaticity chart. Since we are dealing with color matching and not dealing with perceptual distances, use of CIE xy space is justified. All measurements (before and after color emulation) were from the same spot on the projector using a Photo Research 705 Spectrascan spectroradiometer.

#### 4.2. Color Balancing Multiple Projectors

The advent of low-cost LED projectors bring in the potential of building very high-resolution tiled projection-based displays that are both portable and affordable [22, 4, 3]. Currently, color balancing across multiple projectors is achieved via software in two steps: (a) first, a common 3D color gamut contained within the gamut of all the projectors is computed; (b) next, a linear [31, 32] or piecewise linear [36] transformation is used to convert the gamuts of all the projectors to the common gamut. For LED projectors such a gamut mapping leads to degradation in image brightness and contrast.

We can use our hierarchical temporal multiplexing scheme to achieve color balancing across multiple projec-

tors in two steps. (a) First, we use our color emulation method to match all the different projector to the same standard color gamut and white point (Section 4). (b) Next, to balance the still varying brightness we add an extra level to our hierarchical method (Section 4.1).

Let us assume  $n$  projectors, with projector  $i$  having the TTV for white  $I_{W_i}$  following the first two levels. We seek to match the TTV (and hence brightness) across multiple projectors, such that for any  $i, j \in 0, \dots, n-1$ ,  $I_{W_i} = I_{W_j}$ . First, we choose the minimum TTV of all projectors as the target TTV  $I_M = \min_{i \in 0, \dots, n-1} I_{W_i}$ . Next the ON period of each of the new primaries of the projector  $i$  are scaled by  $k_i = \frac{I_M}{I_{W_i}}$  matching the TTV (and hence brightness) of all projectors to  $I_M$ . The new primaries thus formed are denoted by  $R_M$ ,  $G_M$  and  $B_M$ . However, since the relative proportions of the primaries are not changed, the 2D color gamut and the white point are unaffected by this step.

Our method can only achieve color and brightness balancing across projectors, but cannot handle the intra-projector spatial variation in brightness (commonly called vignetting effect). For this, existing camera-based brightness calibration methods [16] can be used following our color balancing scheme. Since our hierarchical temporal multiplexing method has already modified the larger projector gamut to be close to that of the camera, using a camera in the feedback loop no longer poses a problem.

#### 4.2.1 Implementation and Results

We have implemented our color balancing technique on two displays – (a) a curved display made of four projectors (Figure 9); and (b) a planar display made of 16 projectors (Figure 10). The whites on the calibrated display in the top row of Figure 9 is only after application of our color emulation method, as described in Section 4. Hence, the brightness variation across projectors and the spatial vignetting is still evident. All the other images of calibrated displays are achieved by applying three steps: (a) color emulation, as in Section 4; (b) brightness balancing as in Section 4.2; and (c) removal of spatial vignetting by applying the methods in [16]. Note that these three steps make both our four and sixteen projector displays perceptually seamless.

### 5. Conclusion

In conclusion, we demonstrated that the larger gamut of the current LED projectors lead to several hue-shift artifacts and sub-optimal utilization of color resources when displaying existing lower gamut media due to gamut expansion. We presented a content-independent hardware-assisted method that reshapes the 3D gamut to match a standard 2D color gamut while directing the extra volume towards achieving a higher dynamic range. This method can be used for color emulation of single projectors and also for color balancing multiple projectors.

As is evident, in the future it is critical to explore approaches to develop a sensor with large gamut so that the display resources can be utilized to its fullest. One can also explore better content-dependent approaches in the future that can handle very large gamut expansions so that the larger chrominance gamut offered by these emerging displays are optimally utilized.

## References

- [1] T. Ajito, T. Obi, M. Yamaguchi, and N. Ohshima. Expanding color gamut reproduced by six-primary projection display. *Proceedings of SPIE*, 3954:130–137, 2000. 1
- [2] F. Banterle, P. Ledda, K. Debattista, and A. Chalmers. Inverse tone mapping. pages 349–356, 2006. 3
- [3] E. S. Bhasker, R. Juang, and A. Majumder. Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. *IEEE TVCG*, 13(5), 2007. 7
- [4] E. S. Bhasker, P. Sinha, and A. Majumder. Asynchronous distributed calibration for scalable reconfigurable multi-projector displays. *IEEE TVCG*, 12(5):1101–1108, 2006. 7
- [5] G. Braun. *A Paradigm for Color Gamut Mapping of Pictorial Images*. PhD thesis, RIT, Rochester, NY, 1999. 3, 4
- [6] D. Eliav, S. Roth, and M. B. Chorin. Application driven design of multi-primary displays. *Proc. of IS&T/SID's 14th Color Imaging Conference*, 2006. 1
- [7] E. J. Giorgianni and T. E. Madden. *Digital Color Management : Encoding Solutions*. Addison Wesley, 1998. 3
- [8] G. Harbers and C. Hoelen. High performance lcd backlighting using high intensity red, green and blue light emitting diodes. *SID Digest*, pages 702–705, 2001. 1
- [9] T. Hirokawa, M. Inui, T. Morioka, and Y. Azuma. A psychophysical evaluation of a gamut expansion algorithm based on chroma mapping ii: Expansion within object color data bases. *NIP23*, pages 175–179, 2007. 4
- [10] R. L. Holman and A. Cox. High-density illumination system. *US Patent 7210806*, 2007. 1
- [11] L. J. Hornbeck. Field updated deformable mirror device. *US Patent 5280277*, 1994. 4
- [12] T. Hoshino. A preferred color reproduction method for the hdtv digital still image system. pages 27–32, 1991. 4
- [13] C. Joubert, C. Puech, B. Loiseaux, and J.-P. Huignard. Video image projector with improve luminous efficiency. *US Patent 5526063*, 1996. 1
- [14] B. H. Kang, M. S. Cho, J. Morovic, and M. R. Luo. Gamut extension development based on observer experimental data. *IS&T/SID 9th Color Imaging Conference*, pages 158–162, 2001. 3, 4
- [15] M. C. Kim, Y. C. Shin, Y. R. Song, S. J. Lee, and I. D. Kim. Wide gamut multiprimary display for hdtv. *Second European Conference on Color in Graphics, Imaging and Vision, Aachen, Germany*, page 248253, April 2004. 1, 3, 4
- [16] A. Majumder and R. Stevens. Perceptual photometric seamlessness in tiled projection-based displays. *ACM Transactions on Graphics*, 24(1), January 2005. 7
- [17] V. Masselus, P. Peers, P. Dutre, and Y. D. Willems. Relighting with 4d incident light fields. *ACM SIGGRAPH*, 2003. 2, 3
- [18] S. K. Nayar, G. Krishnan, M. D. Grossberg, and R. Raskar. Fast separation of direct and global components of a scene using high frequency illumination. *ACM TOG*, 25(3), 2003. 2, 3
- [19] S. K. Nayar, H. Peri, M. D. Grossberg, and P. N. Belhumeur. A Projection System with Radiometric Compensation for Screen Imperfections. *IEEE PROCAMS*, 2003. 2, 3
- [20] G. Niven and A. Mooradian. Low cost lasers and laser arrays for projection displays. pages 1904–1907, 2006. 1, 3
- [21] C. Pinhanez, M. Podlaseck, R. Kjeldsen, A. Levas, G. Pingali, and N. Sukaviriya. Ubiquitous interactive displays in a retail environment. *ACM SIGGRAPH Sketches*, 2003. 2, 3
- [22] R. Raskar, J. van Baar, P. Beardsley, T. Willwacher, S. Rao, and C. Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM TOG*, 22(3), 2003. 3, 7
- [23] A. G. Rempel, M. Trentacoste, H. Seetzen, H. D. Young, W. Heidrich, L. Whitehead, and G. Ward. Ldr2hdr: on-the-fly reverse tone mapping of legacy video and photographs. *ACM TOG*, 26, 2007. 3
- [24] J. E. Roddy, R. J. Zolla, N. A. Blish, and L. S. Horvath. Six color display apparatus having increased color gamut. *US Patent 6769772*, 2004. 1
- [25] S. Roth, I. Ben-David, M. Ben-Chorin, D. Eliav, and O. Ben-David. Wide gamut, high brightness multiple primaries single panel projection displays. *SID Digest*, 2003. 1
- [26] J. J. Sara. *The Automated Reproduction of Pictures with Non-reproducible Colors*. PhD thesis, MIT, 1984. 4
- [27] H. Seetzen, W. Heidrich, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs. High dynamic range display systems. *ACM TOG*, 23(3), 2004. 3
- [28] P. Sen, B. Chen, G. Garg, S. R. Marschner, M. Horowitz, M. Levoy, and H. P. A. Lensch. Dual photography. *ACM Transactions on Graphics*, 24, 2005. 2, 3
- [29] M. Shindoh. Projector optics and projector with light source of leds. *US Patent 7101049*, 2006. 1
- [30] P.-L. Song. Projector apparatus. *US Patent 6830343*, 2004. 1
- [31] M. C. Stone. Color balancing experimental projection displays. *9th IS&T/SID Color Imaging Conference*, 2001a. 7
- [32] M. C. Stone. Color and brightness appearance issues in tiled displays. *IEEE CG&A*, 2001b. 7
- [33] M. C. Stone. *A Field Guide to Digital Color*. A.K. Peters, 2003. 3
- [34] E. H. Stupp and M. S. Brennessoltz. *Projection Displays*. John Wiley and Sons Ltd., 1999. 1
- [35] R. Sukthankar, R. Stockton, and M. Mullin. Smarter presentations: Exploiting homography in cameraprojector systems. *ICCV*, 2001. 2, 3
- [36] G. Wallace, H. Chen, and K. Li. Color gamut matching for tiled display walls. *IPT Workshop*, 2003. 7
- [37] R. Yang, A. Majumder, and M. Brown. Camera based calibration techniques for seamless multi-projector displays. *IEEE TVCG*, 11(2), March-April 2005. 2, 3
- [38] Z. Zalevsky and A. Goldman. Spectral modeling and improvement of target detection in a visible cluttered environment. *Optical Engineering*, 41:1358–1364, 2002. 3