Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

Clase 1 : Miercoles Marzo 27

Gabriel Taubin

Brown University

BROWN

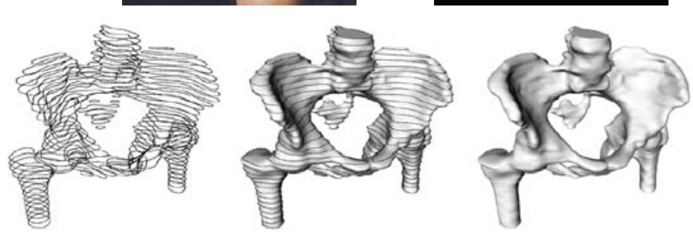
3D Shapes

- Industry
  - Reverse engineering
  - Fast metrology
  - Physical simulations

- Entertainment
  - Animating digital clays for movies or games

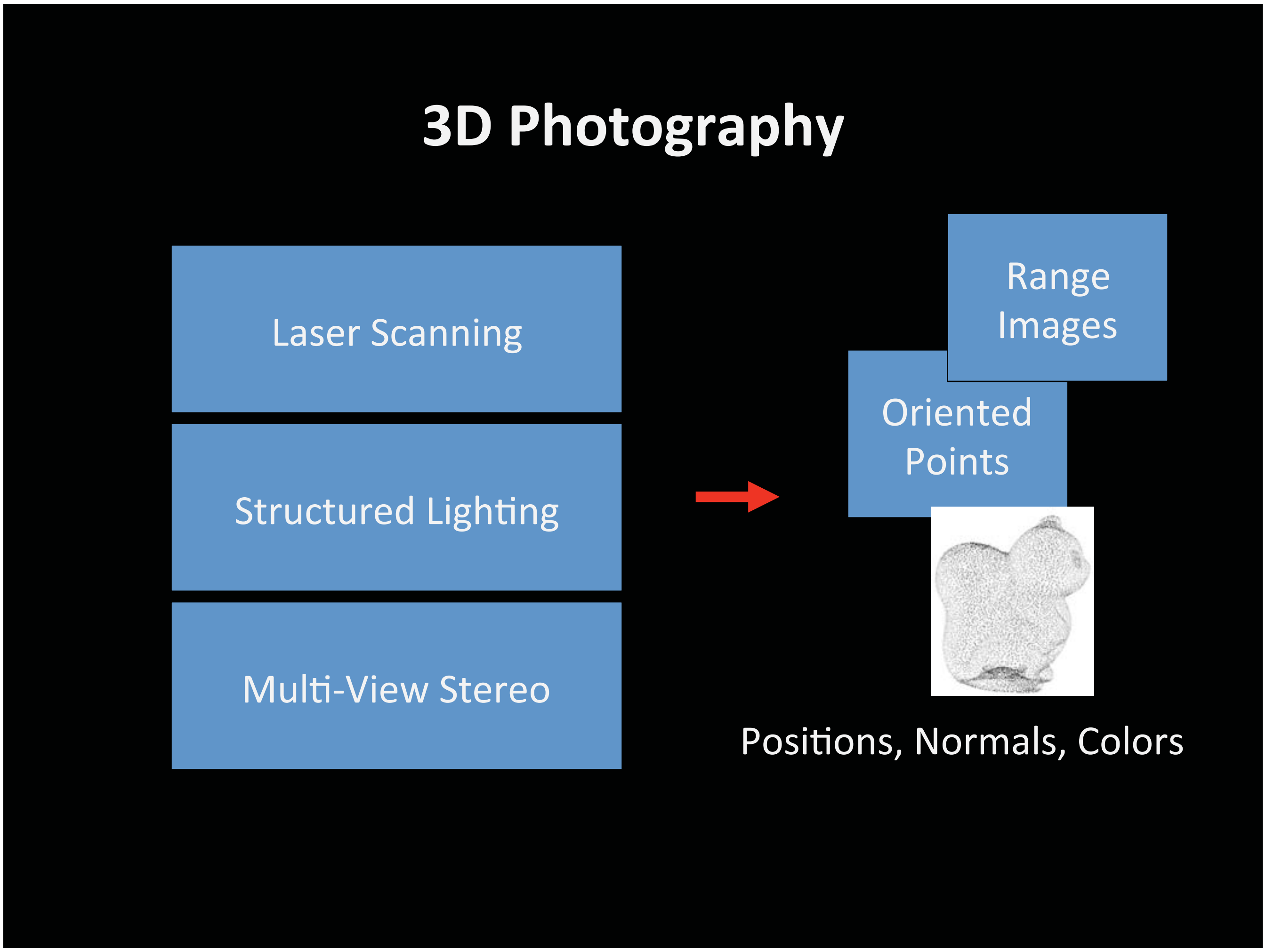
- Archeology and Art
  - Digitization of cultural heritage and artistic works

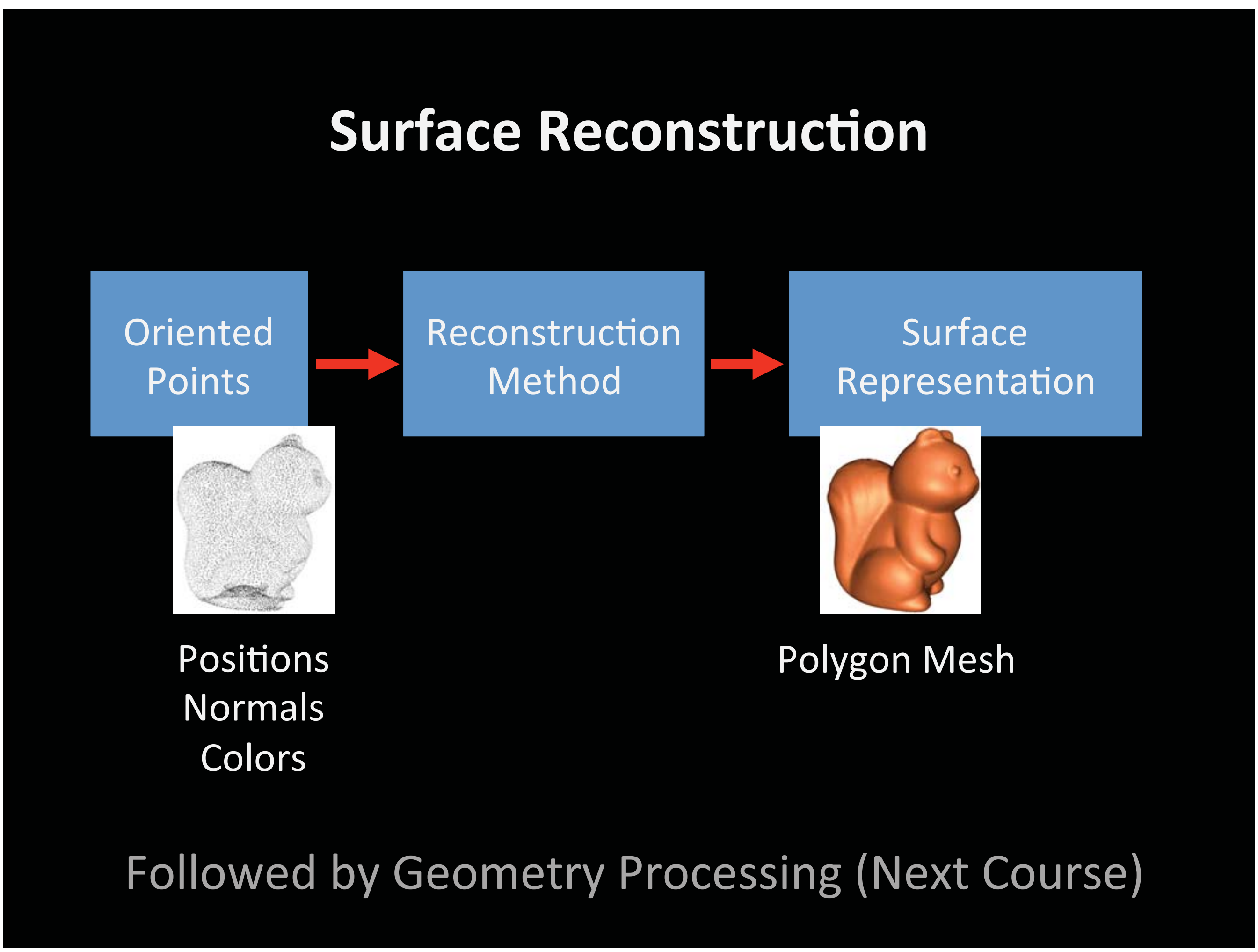
- Medical Imaging
  - Visualization
  - Segmentation



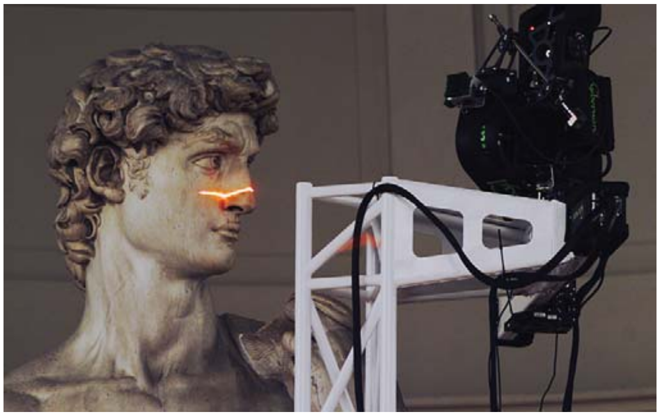
Que es la Fotografia 3D ?

Metodos y sistemas para capturar la geometria y la apariencia de objetos tridimensionales, basados en el uso de camaras y fuentes de luz


3D Photography
Laser Scanning
Structured Lighting
Multi-View Stereo
Range Images
Oriented Points
Positions, Normals, Colors

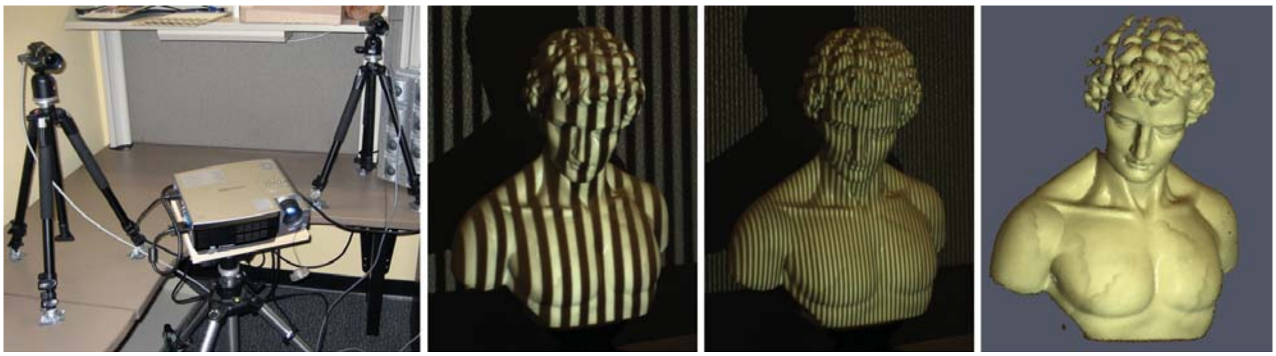

Surface Reconstruction
Oriented Points
Reconstruction Method
Surface Representation
Positions
Normals
Colors
Polygon Mesh
Followed by Geometry Processing (Next Course)

3D Shape and Appearance Capture



Laser range scanning devices

Multi-camera systems



Structured lighting systems

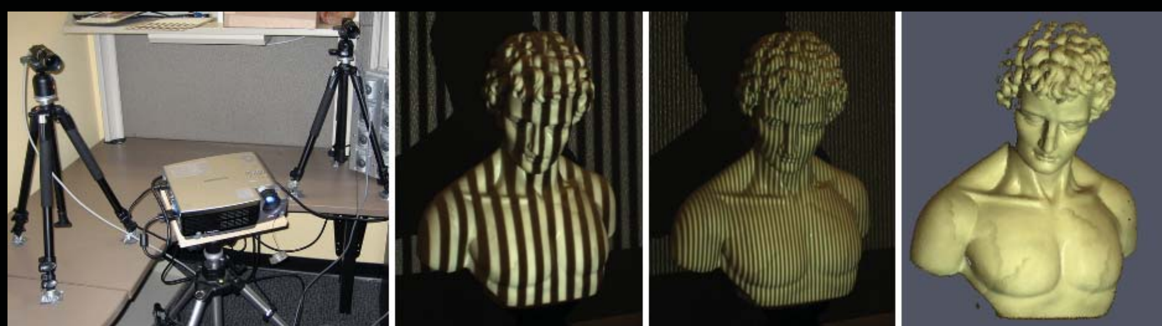
Objetivos

Adquirir conocimientos básicos sobre:

- Métodos y sistemas para la captura, reconstrucción y procesamiento de objetos en 3D
- Fundamentos Matemáticos, en particular de métodos basados en cámaras y proyectores
- Requerimientos para implementar y calibrar sistemas de bajo costo
- Diseñar e implementar, como trabajos practicos, dos metodos para la captura de objetos en 3D
- Identificar ideas para projectos que resulten en publicaciones

Triangulation and Scanning with Swept-Planes

Structured Lighting using Projector-Camera Systems



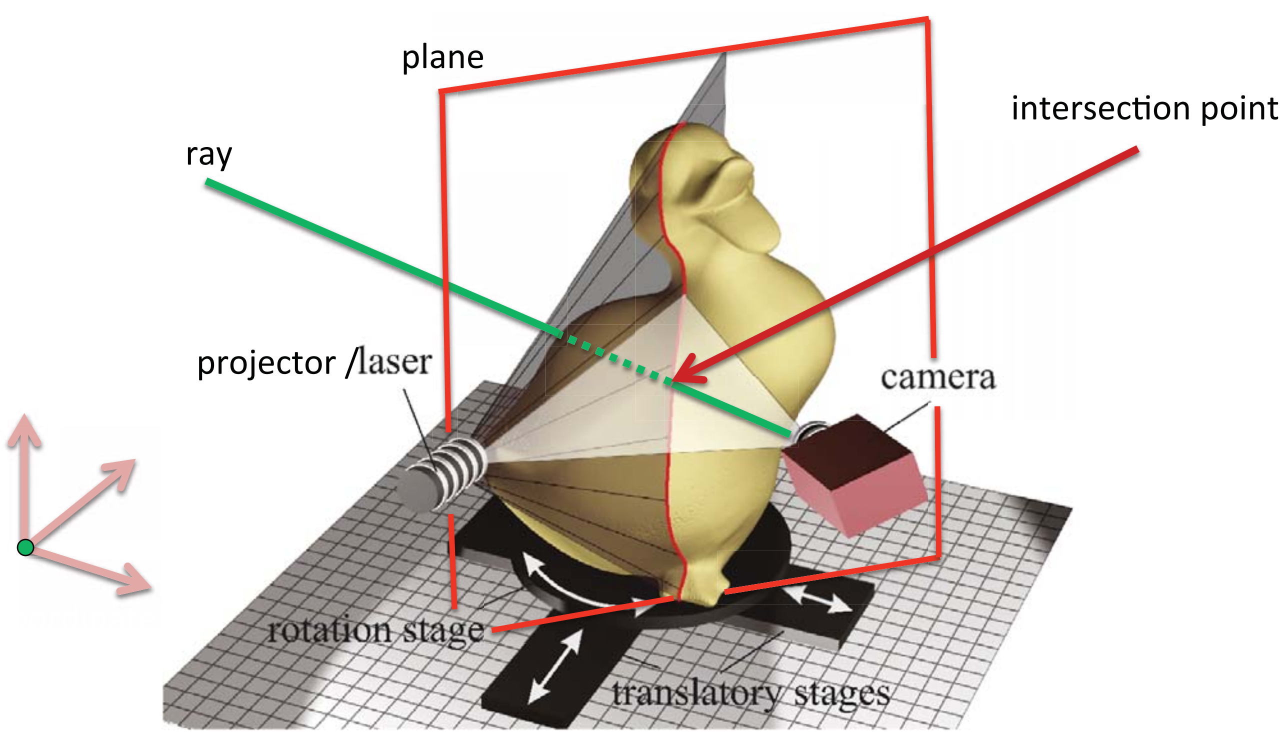
Que hace falta saber?

- Algebra Lineal
- Geometria Analítica en 3D
- Geometria Afin y Projectiva
- Estructuras de Datos / Complejidad
- Metodos Numericos
- Procesamiento de Imagenes
- Programacion en Matlab, C++ y/o Java
- Interes en construir y armar (DIY)

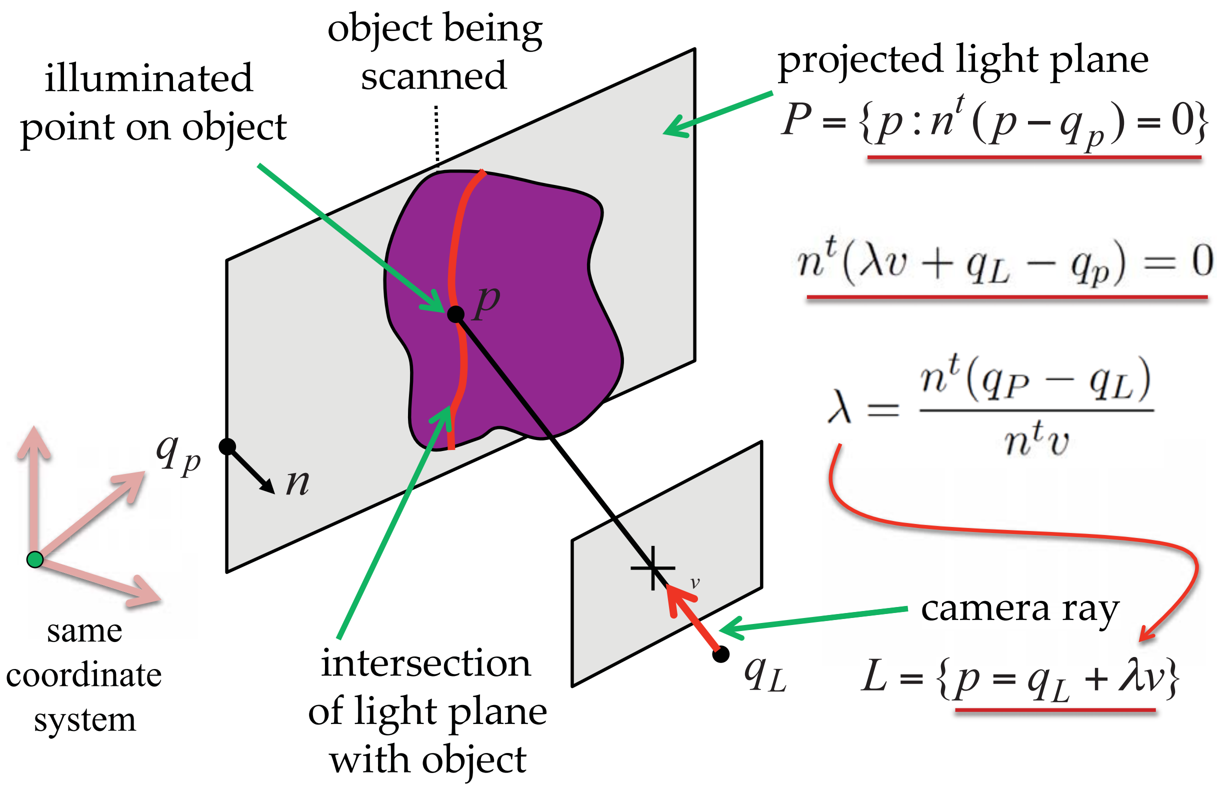
Plan del Curso

1. Introducción
2. Las Matemáticas de la Triangulación en 3D
3. 3D Scanning con Barrido de Planos
4. Calibración de Cameras y Fuentes de Luz
5. Superficies, Nubes de Puntos, Mallas Poligonales
6. Iluminación Estructurada
7. Calibración de Proyectores
8. Combinación de Múltiples Puntos de Vista
9. Reconstrucción de Superficies
10. Procesamiento de Geometria

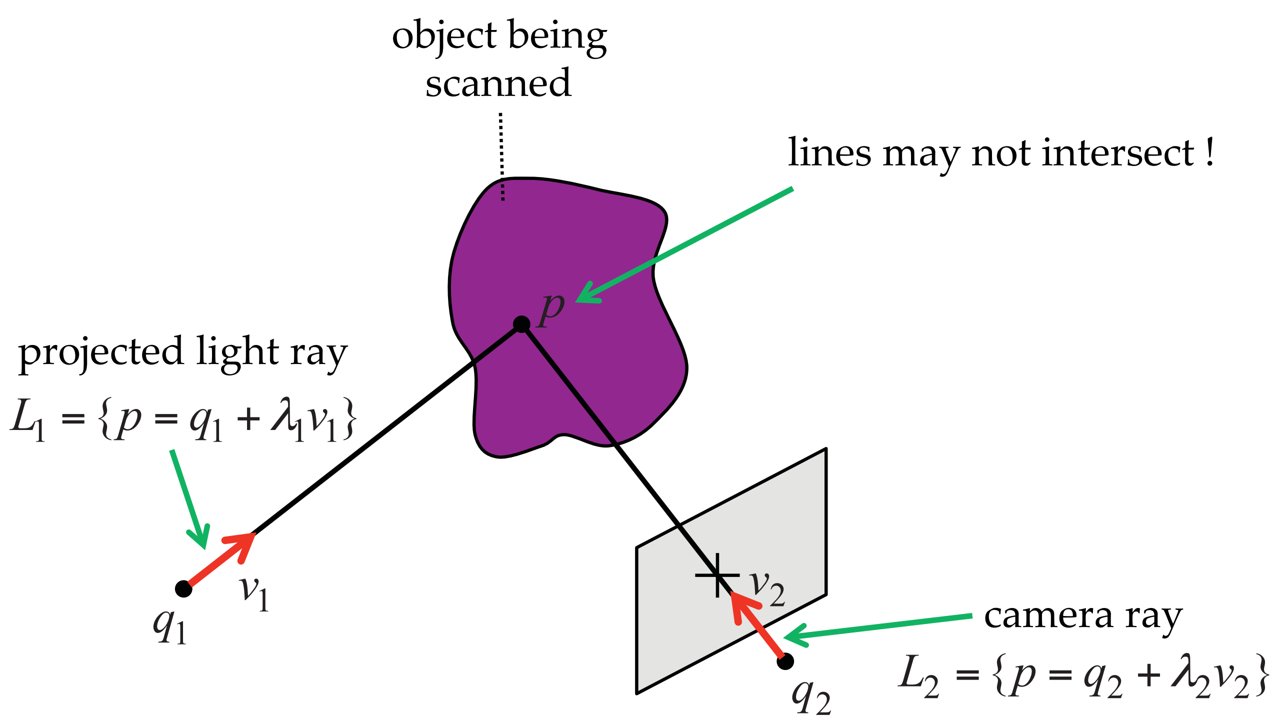
3D Triangulation: Ray-Plane Intersection



Triangulation by Line-Plane Intersection



Triangulation by Line-Line Intersection

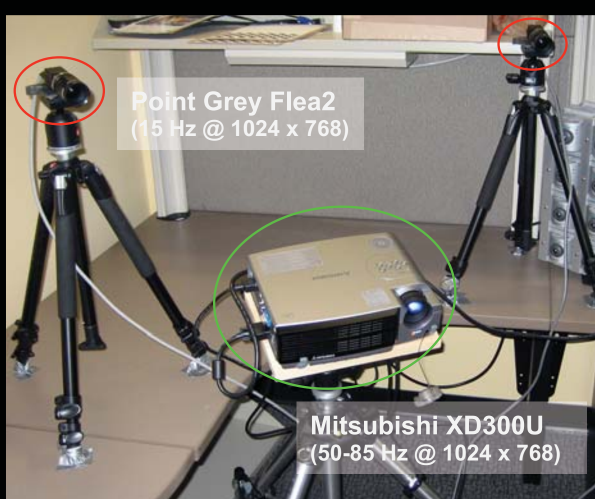


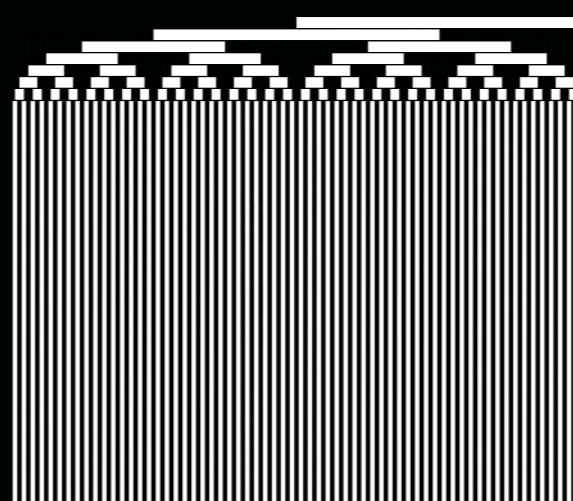
Gray Code Structured Lighting

3D Reconstruction using Structured Light [Inokuchi 1984]

- Recover 3D depth for each pixel using ray-plane intersection
- Determine correspondence between camera pixels and projector planes by projecting a temporally-multiplexed binary image sequence
- Each image is a bit-plane of the Gray code for each projector row/column

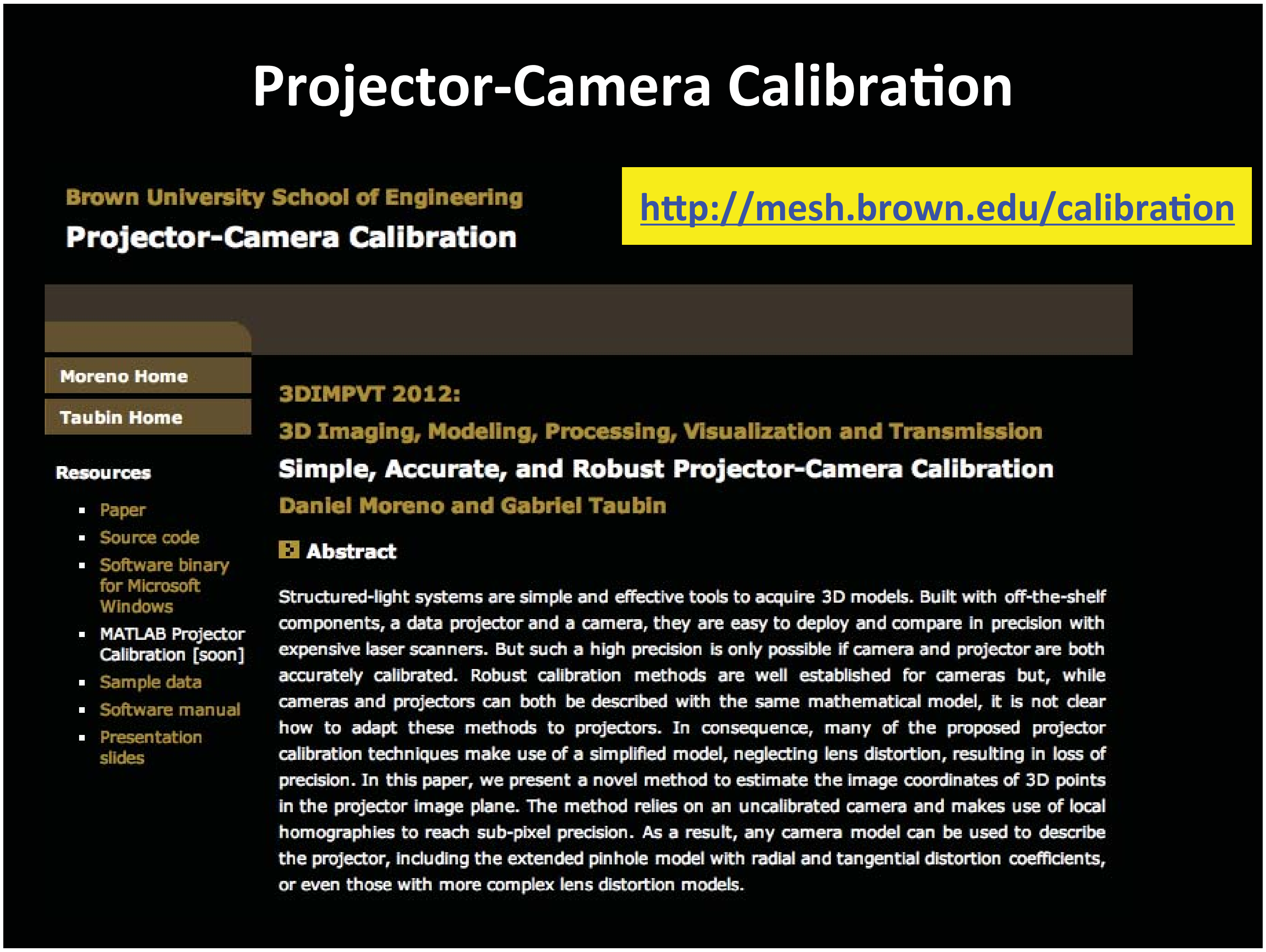
Gray Code Structured Lighting



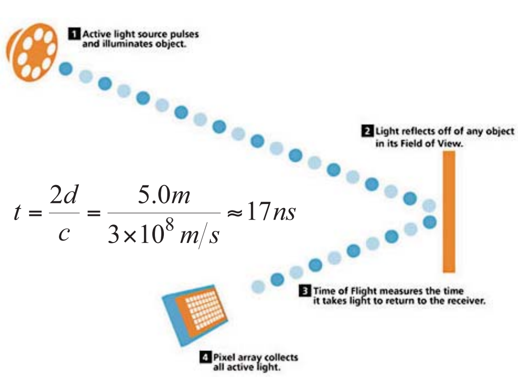


3D Reconstruction using Structured Light [Inokuchi 1984]

- Recover 3D depth for each pixel using ray-plane intersection
- Determine correspondence between camera pixels and projector planes by projecting a temporally-multiplexed binary image sequence
- Each image is a bit-plane of the Gray code for each projector row/column
- Encoding algorithm: integer row/column index → binary code → Gray code



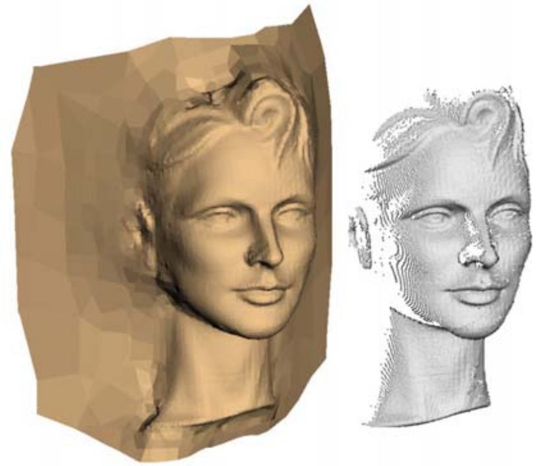
Time of Flight 3D Scanning



Single Shot Structured Lighting: MS Kinect

SSD: Smooth Signed Distance Surface Reconstruction

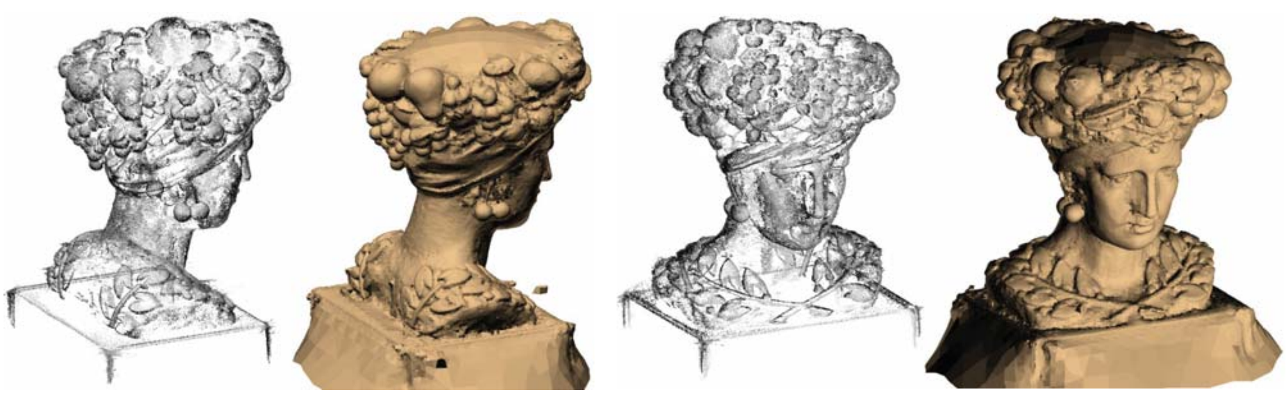
F. Calakli, G. Taubin, Computer Graphics Forum, 2011.

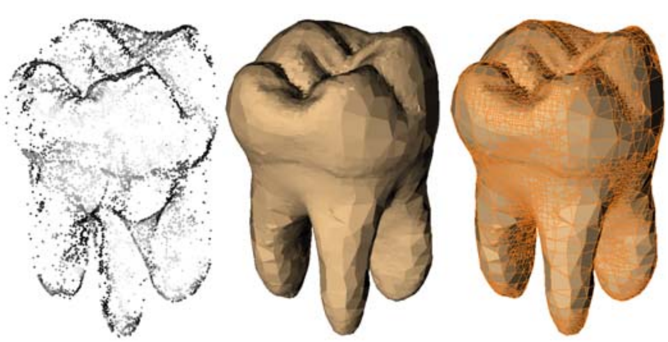


- A new mathematical formulation
- And a particular algorithm
- To reconstruct a watertight surface

  From a static oriented point cloud

Particularly Good at Extrapolating Missing Data





Computer Graphics Forum, Vol. 30, No. 7, 2011 [Pacific Graphics 2011]

Smooth Signed Distance Surface Reconstruction

Fatih Calakli and Gabriel Taubin

- Home
- Paper
- Software

Abstract

We introduce a new variational formulation for the problem of reconstructing a watertight surface defined by an implicit equation, from a finite set of oriented points; a problem which has attracted a lot of attention for more than two decades. As in the Poisson Surface Reconstruction approach, discretizations of the continuous formulation reduce to the solution of sparse linear systems of equations. But rather than forcing the implicit function to approximate the indicator function of the volume bounded by the implicit surface, in our formulation the implicit function is forced to be a smooth approximation of the signed distance function to the surface. Since an indicator function is discontinuous, its gradient does not exist exactly where it needs to be compared with the normal vector data. The smooth signed distance has approximate unit slope in the neighborhood of the data points. As a result, the normal vector data can be incorporated directly into the energy function without implicit function smoothing. In addition, rather than first extending the oriented points to a vector field within the bounding volume, and then approximating the vector field by a gradient field in the least squares sense, here the vector field is constrained to be the gradient of the implicit function, and a single variational problem is solved directly in one step. The formulation allows for a number of different efficient discretizations, reduces to a finite least squares problem for all linearly parameterized families of functions, and does not require boundary conditions. The resulting algorithms are significantly simpler and easier to implement, and produce results of quality comparable with state-of-the-art algorithms. An efficient implementation based on a primal-graph octree-based hybrid finite element-finite difference discretization, and the Dual Marching Cubes isosurface extraction algorithm, is shown to produce high quality crack-free adaptive manifold polygon meshes.

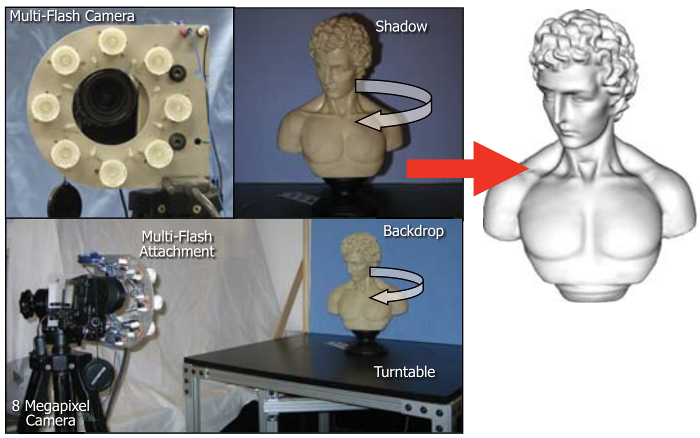
Acknowledgment of Support

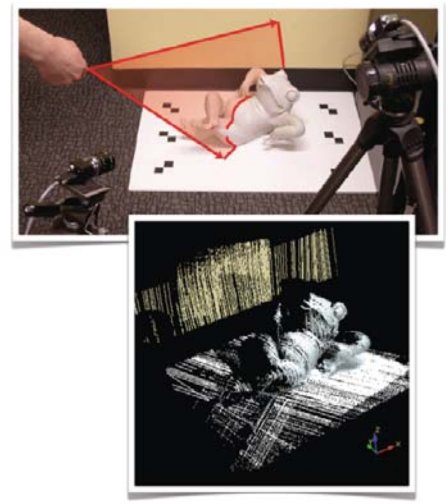
The material presented in this web site describe work supported by the National Science Foundation under Grants No. CCF-0729126, IIS-0808718, and CCF-0915661.

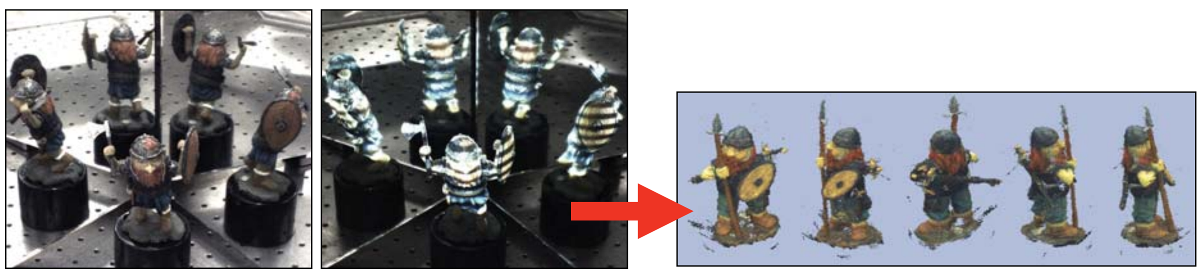
http://mesh.brown.edu/ssd

Final Projects which Resulted in Publications

Some Methods to Capture 3D Point Clouds







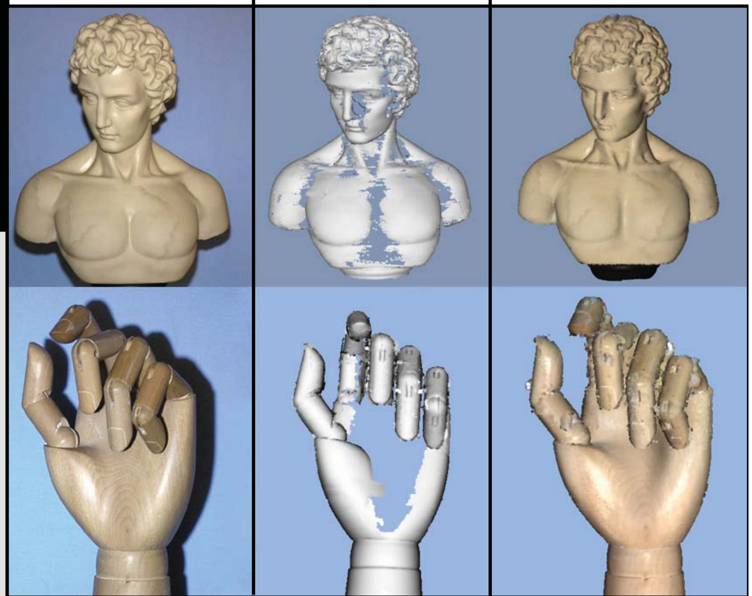
Beyond Silhouettes: Surface Reconstruction using Multi-Flash Photography

D. Crispell, D. Lanman, P. Sibley, Y. Zhao and G. Taubin [3DPVT 2006]

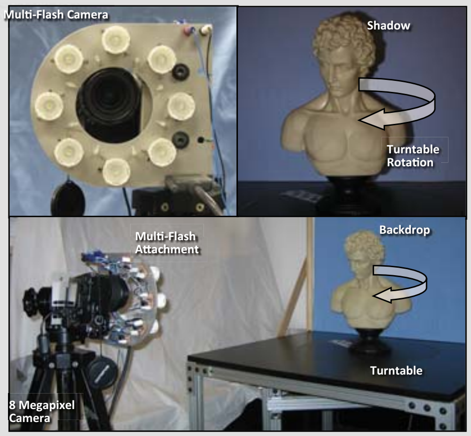
Multi-Flash 3D Photography: Capturing the Shape and Appearance of 3D Objects

A new approach for reconstructing 3D objects using shadows cast by depth discontinuities, as detected by a multi-flash camera. Unlike existing stereo vision algorithms, this method works *even with plain surfaces*, including unpainted ceramics and architecture.

Multi-Flash Turntable Sequence: Input Image	Estimated Shape: 3D Point Cloud	Recovered Appearance: Phong BRDF Model



Data Capture: A turntable and a digital camera are used to acquire data from 670 viewpoints. For each viewpoint, we capture a set of images using illumination from four different flashes. Future embodiments will include a small, inexpensive handheld multi-flash camera.

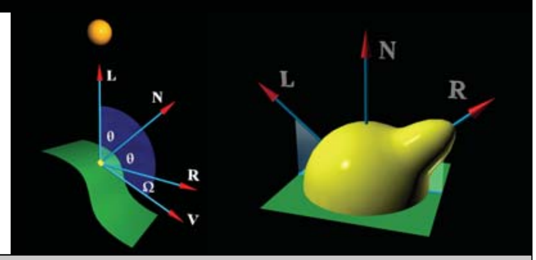


Recovering a Smooth Surface

The reconstructed point cloud can possess errors, including gaps and noise. To minimize these effects, we find an implicit surface which interpolates the 3D points. This method can be applied to any 3D point cloud, including those generated by laser scanners.

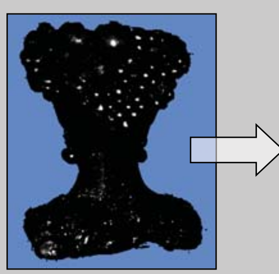
Multi-Flash 3D Photography: Photometric Reconstruction

Using the implicit surface, we can determine which points are visible from each viewpoint. To model the material properties of the surface, we fit a per-point Phong BRDF model to the set of visible reflectance observations (using a total of 67 viewpoints).



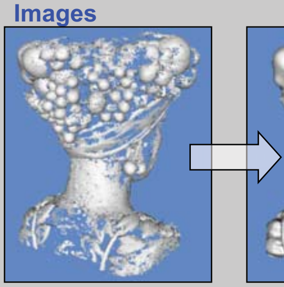
Ambient Diffuse Specular

Multi-Flash Turntable Sequence Images

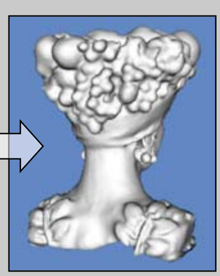


Phong (Specular)

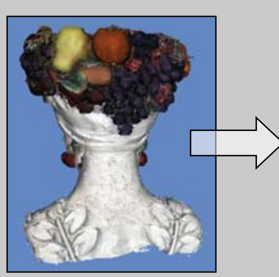
3D Point Cloud



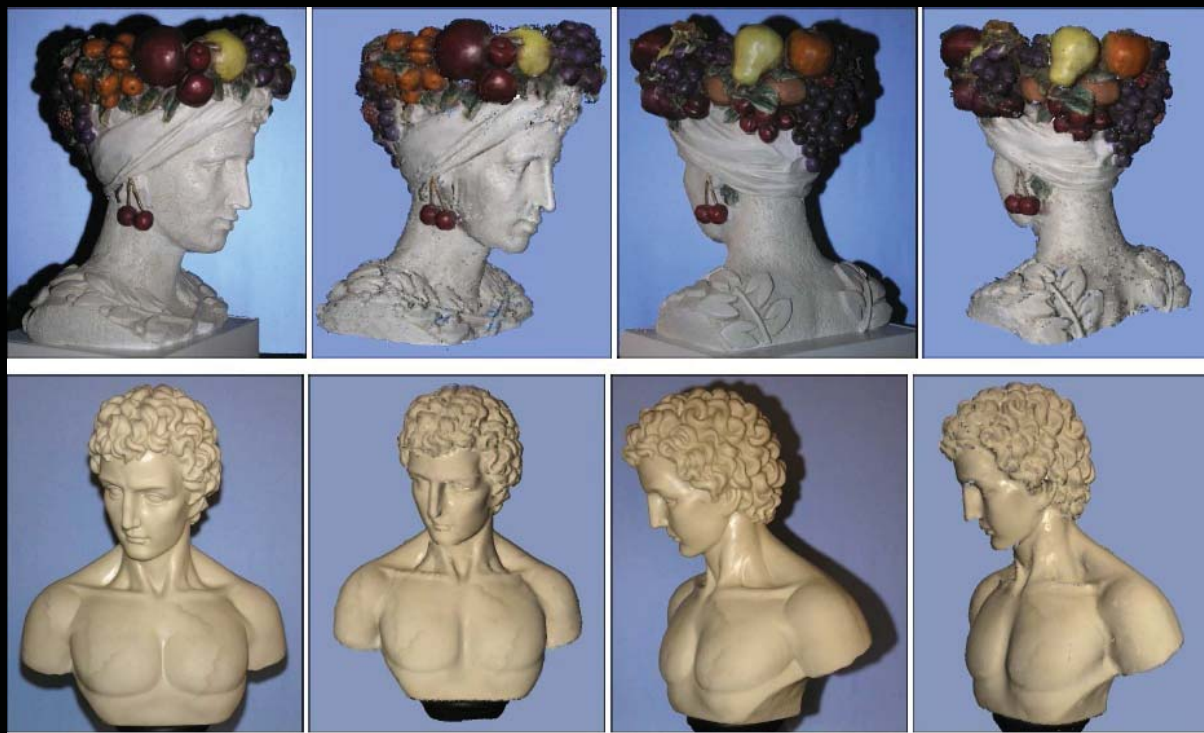
Implicit Surface

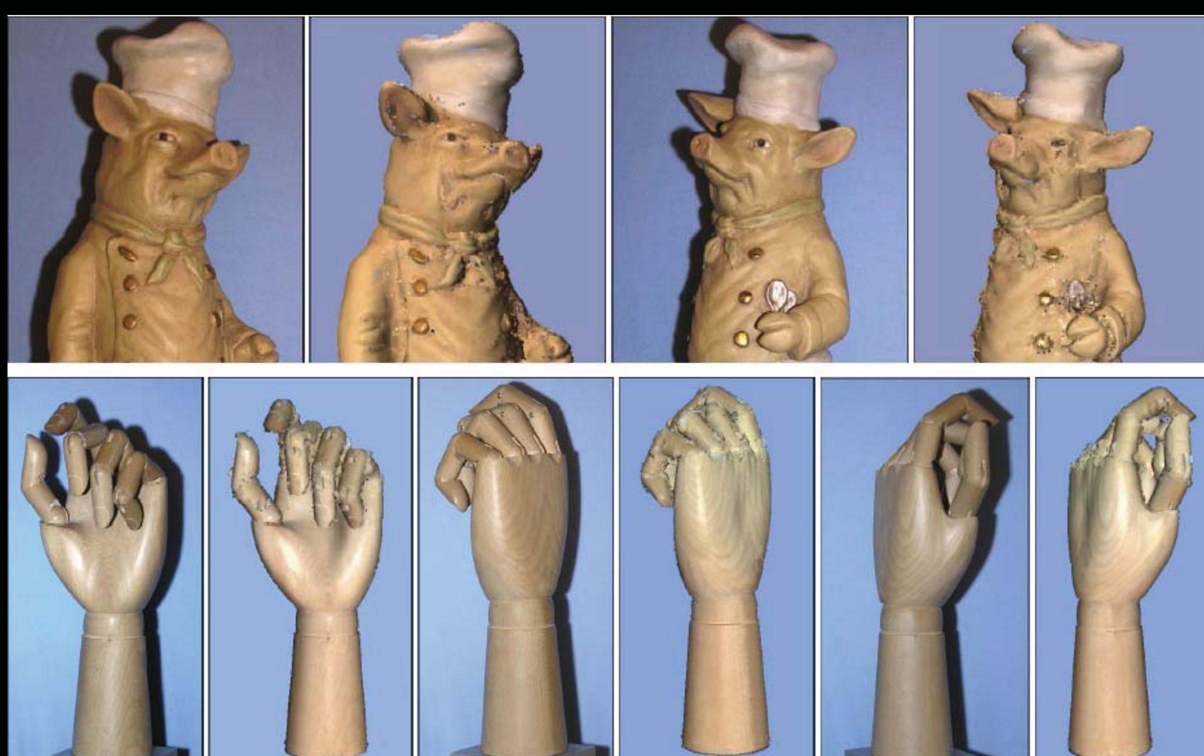


Phong (Diffuse)

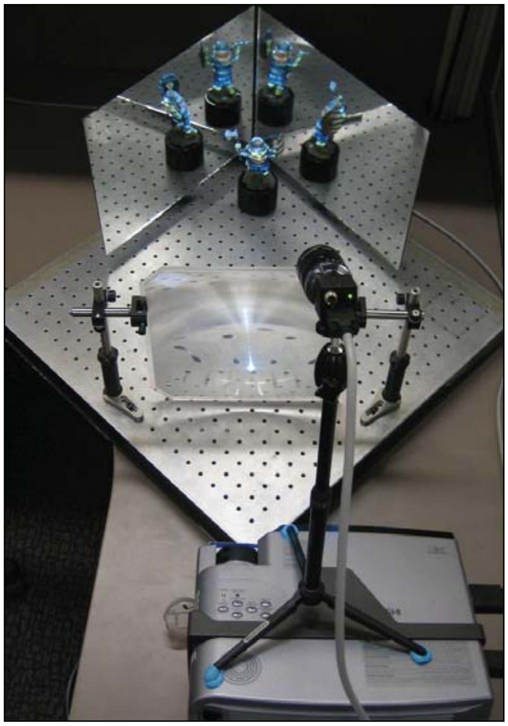


Estimated Phong Appearance Model

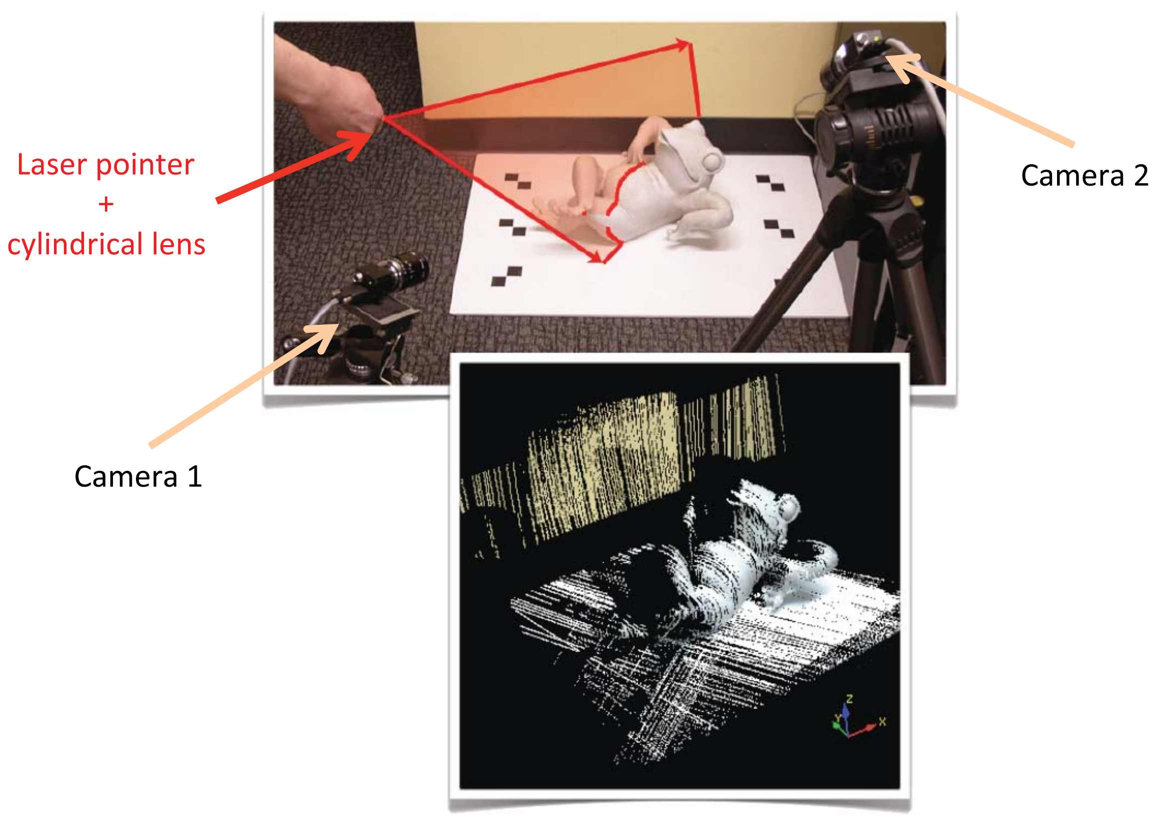




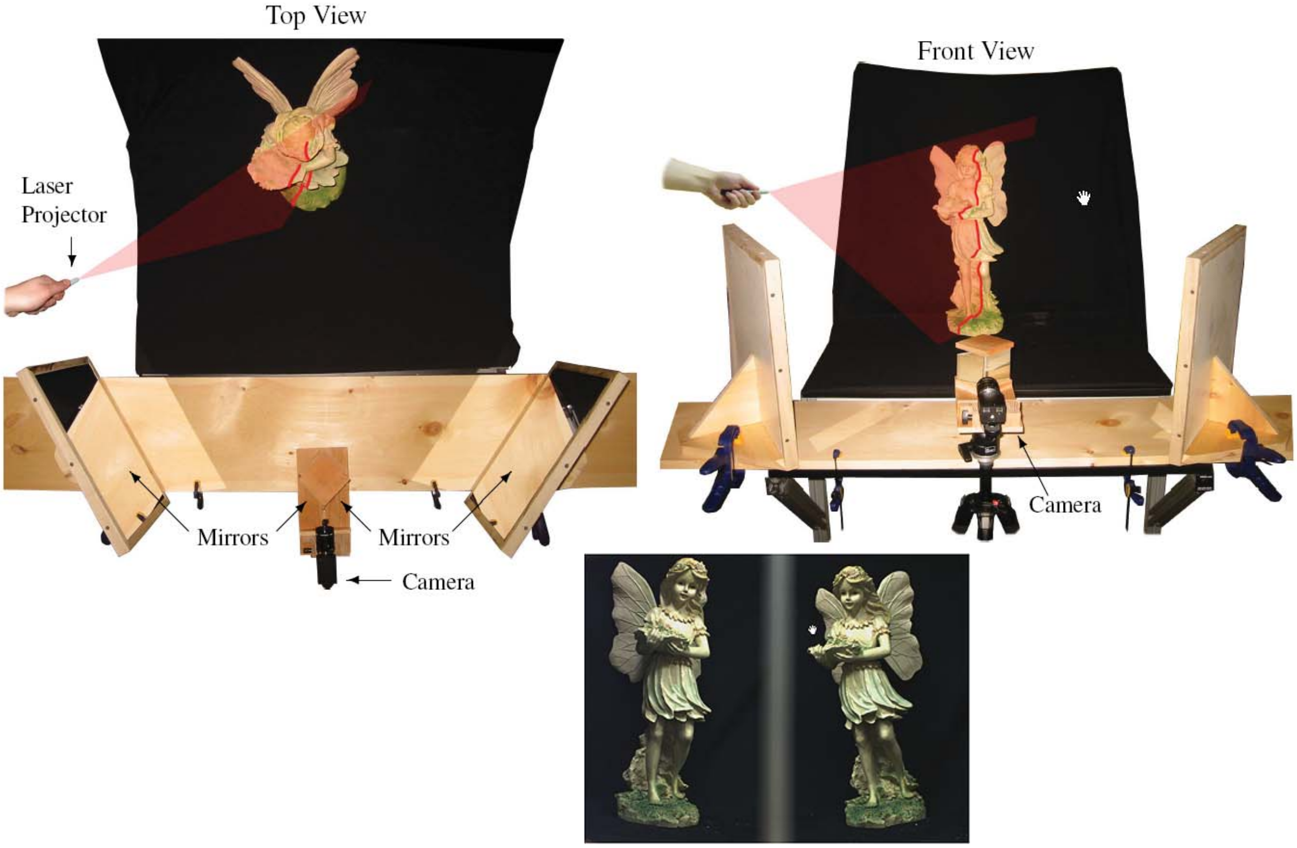
Surround Structured Lighting:
3-D Scanning with Orthographic Illumination
D. Lanman, D. Crispell, G. Taubin [CVIU 2009]



3D Slit Scanning with Planar Constraints
M. Leotta, A. Vandergon, and G. Taubin [CGF 2008]

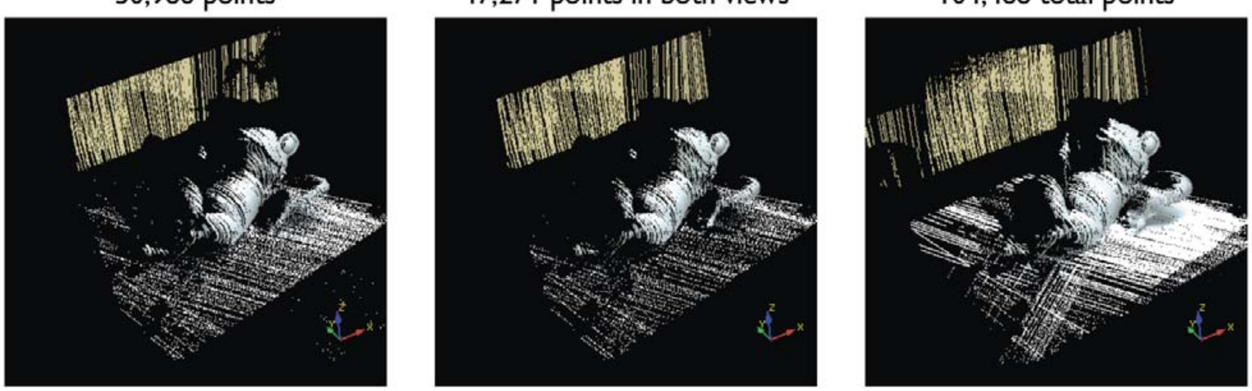


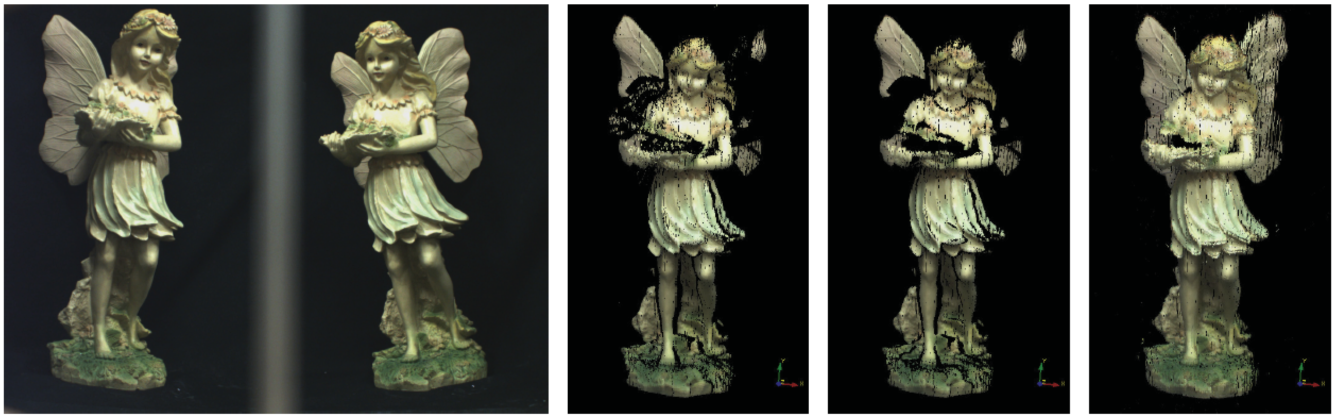
Catadioptric Stereo Implementation



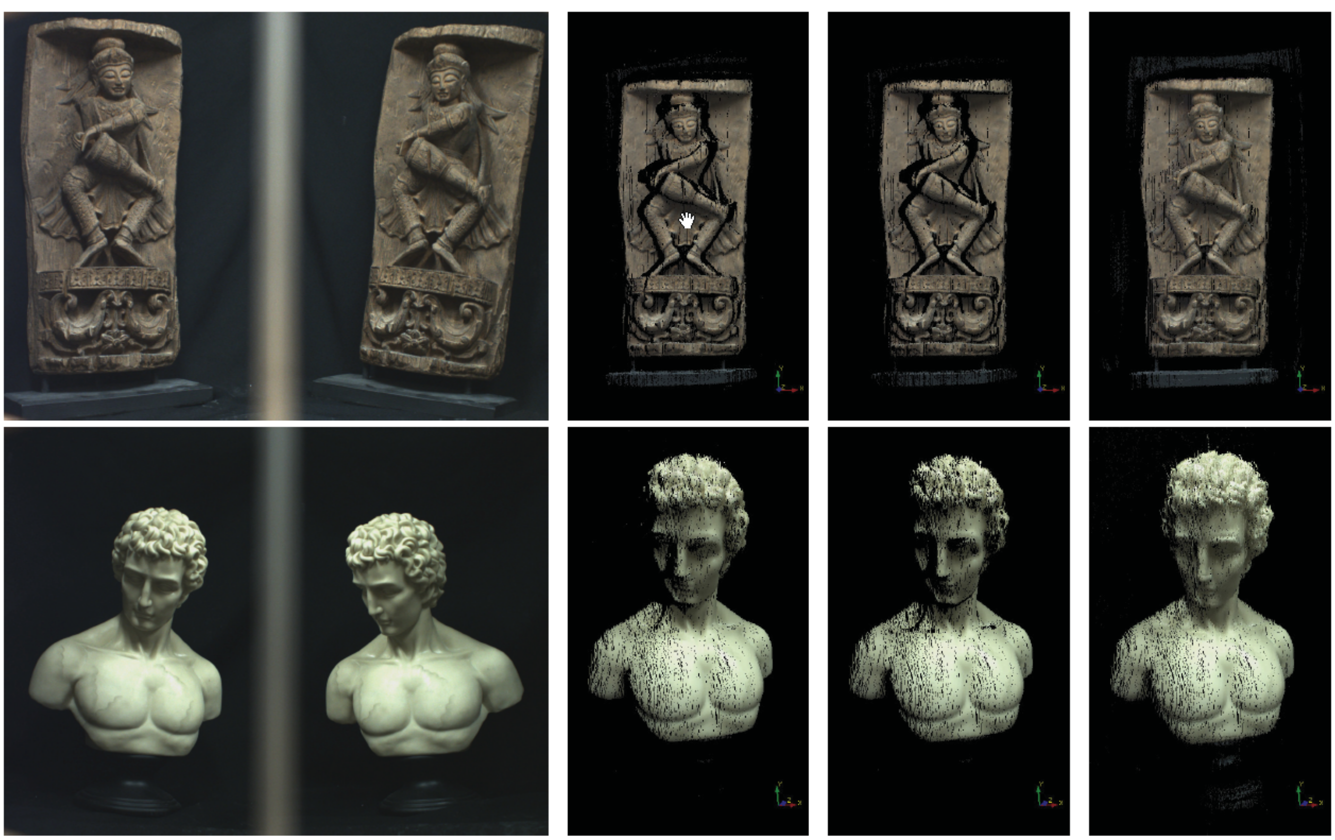
Can Estimate Points Visible From One Camera

50,986 points | 47,271 points in both views | 104,488 total points





(a) Camera View (b) Triangulated (c) Planar Optimal (d) All Points



Schedule for this week and next

- **Introduction to 3D Scanning**
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes | Slit scanner
- Camera and Swept-Plane Light Source Calibration

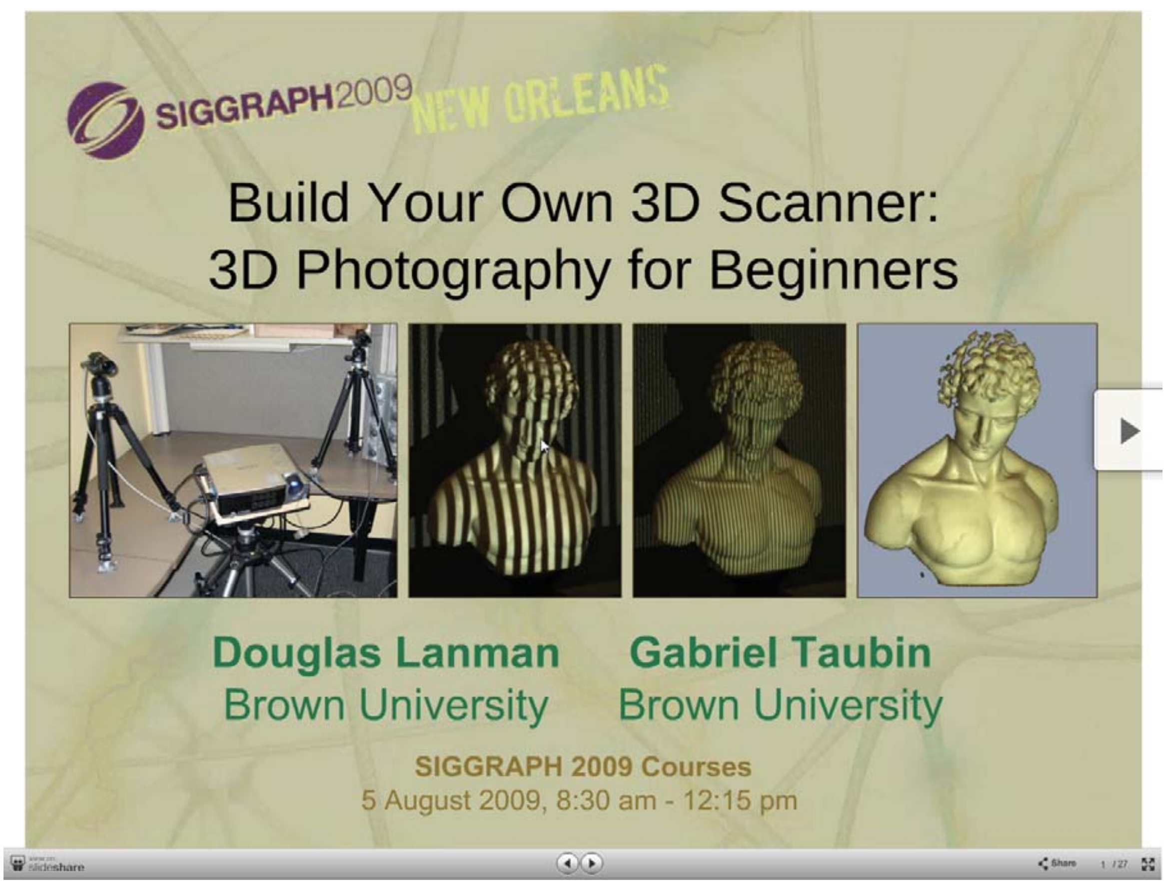
Lecture Notes and Additional Resources

- Download Course Notes from
- **http://mesh.brown.edu/byo3d**
- Or just Google search for BYO3D

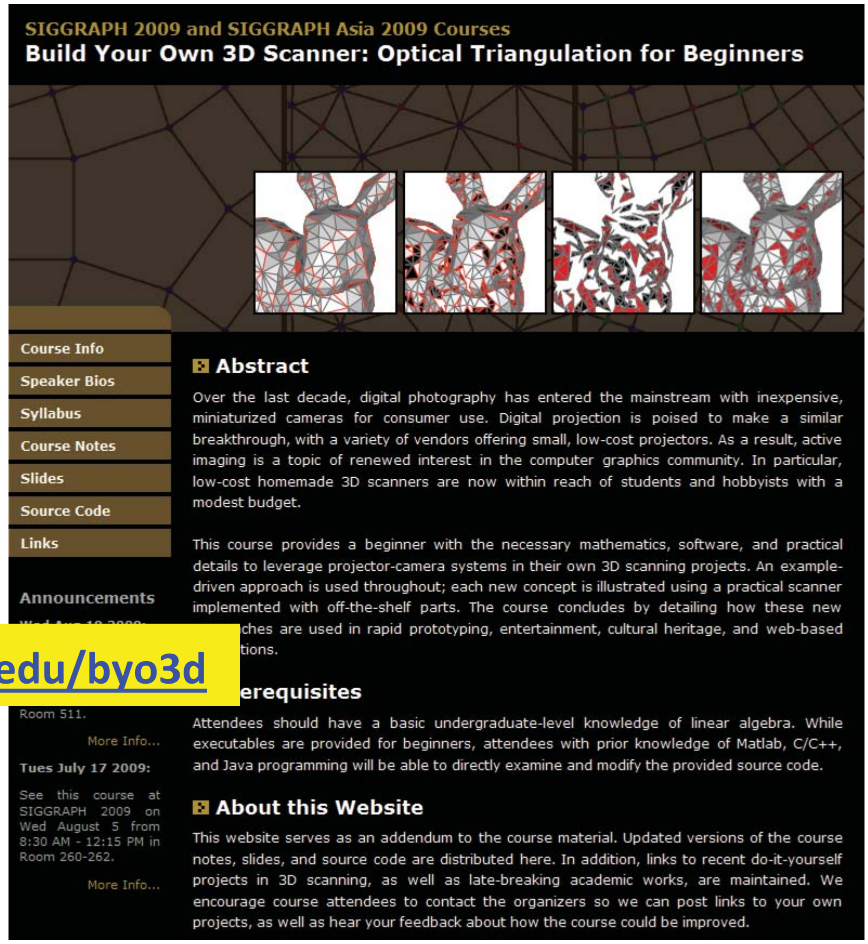
- Courses on 3D Photography taught at Brown University
- **http://mesh.brown.edu/3DP**

- This Course
- **http://mesh.brown.edu/3DP-FCEN-2013**

http://mesh.brown.edu/byo3d/



http://mesh.brown.edu/byo3d


SIGGRAPH 2009 and SIGGRAPH Asia 2009 Courses

Build Your Own 3D Scanner: Optical Triangulation for Beginners

- Course Info
- Speaker Bios
- Syllabus
- Course Notes
- Slides
- Source Code
- Links

Announcements

Room 511.

More Info...

Tues July 17 2009:

See this course at SIGGRAPH 2009 on Wed August 5 from 8:30 AM - 12:15 PM in Room 260-262.

More Info...

Abstract

Over the last decade, digital photography has entered the mainstream with inexpensive, miniaturized cameras for consumer use. Digital projection is poised to make a similar breakthrough, with a variety of vendors offering small, low-cost projectors. As a result, active imaging is a topic of renewed interest in the computer graphics community. In particular, low-cost homemade 3D scanners are now within reach of students and hobbyists with a modest budget.

This course provides a beginner with the necessary mathematics, software, and practical details to leverage projector-camera systems in their own 3D scanning projects. An example-driven approach is used throughout; each new concept is illustrated using a practical scanner implemented with off-the-shelf parts. The course concludes by detailing how these new ...ches are used in rapid prototyping, entertainment, cultural heritage, and web-based ...tions.

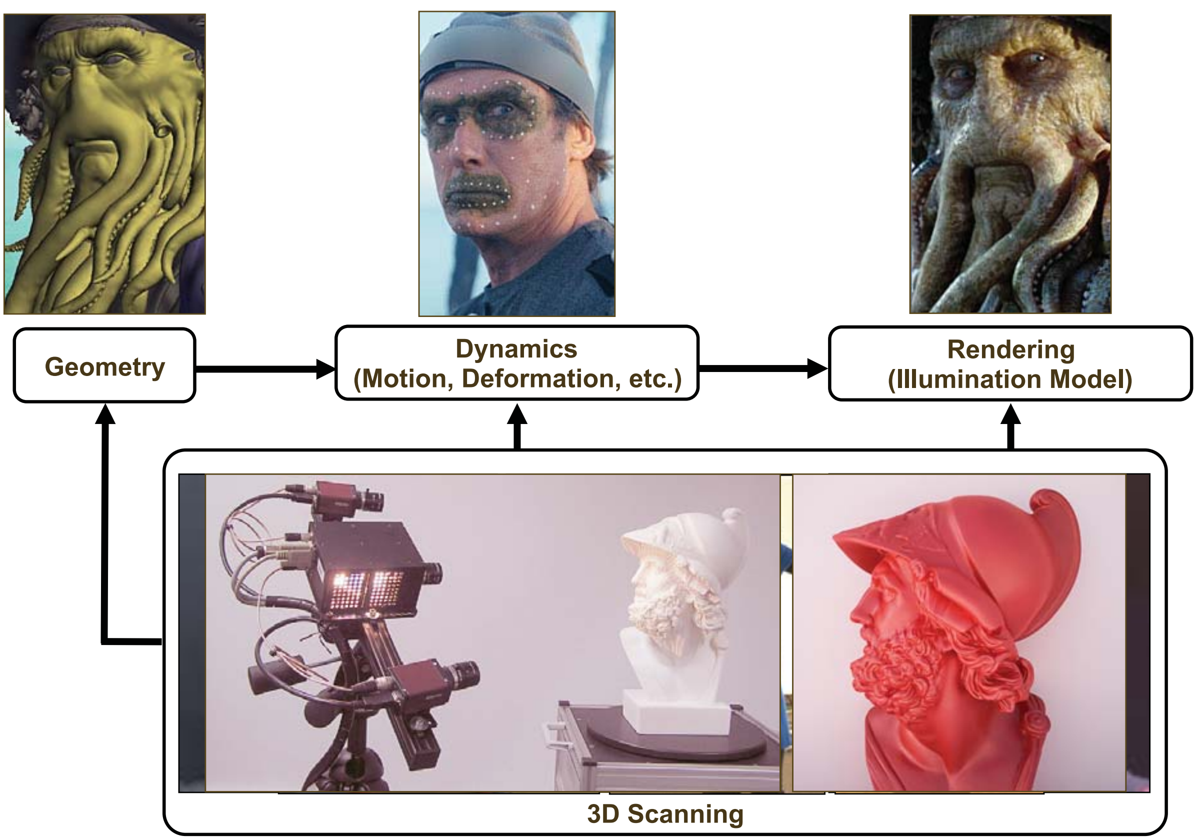
...erequisites

Attendees should have a basic undergraduate-level knowledge of linear algebra. While executables are provided for beginners, attendees with prior knowledge of Matlab, C/C++, and Java programming will be able to directly examine and modify the provided source code.

About this Website

This website serves as an addendum to the course material. Updated versions of the course notes, slides, and source code are distributed here. In addition, links to recent do-it-yourself projects in 3D scanning, as well as late-breaking academic works, are maintained. We encourage course attendees to contact the organizers so we can post links to your own projects, as well as hear your feedback about how the course could be improved.

Introduction to 3D Scanning

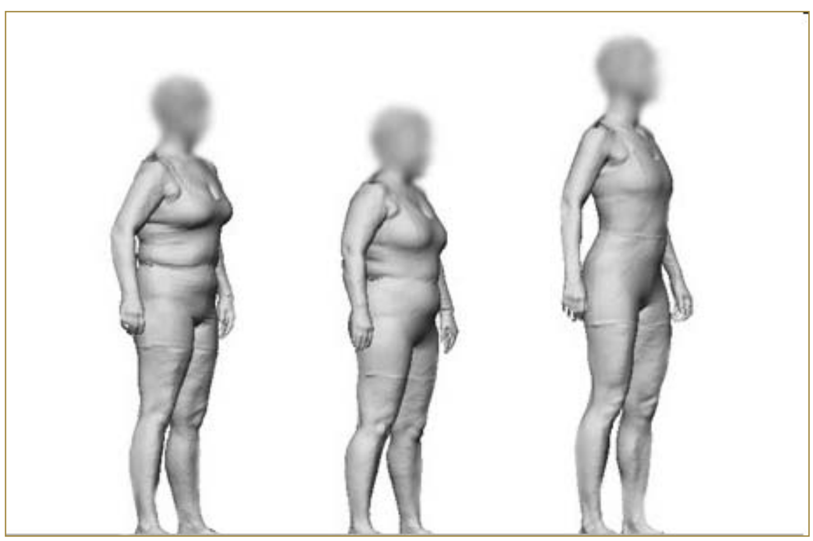


Applications of 3D Scanning: Entertainment and Consumer Applications

- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

Andreas Wenger et al. Performance Relighting and Reflectance Transformation with Time-Multiplexed Illumination. ACM SIGGRAPH, 2005

Applications of 3D Scanning: Entertainment and Consumer Applications

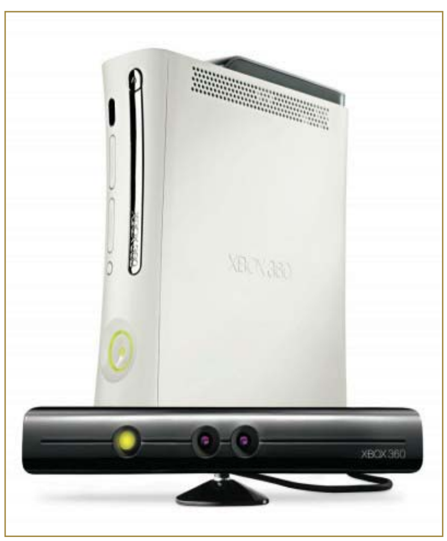


- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

Applications of 3D Scanning:
Entertainment and Consumer Applications

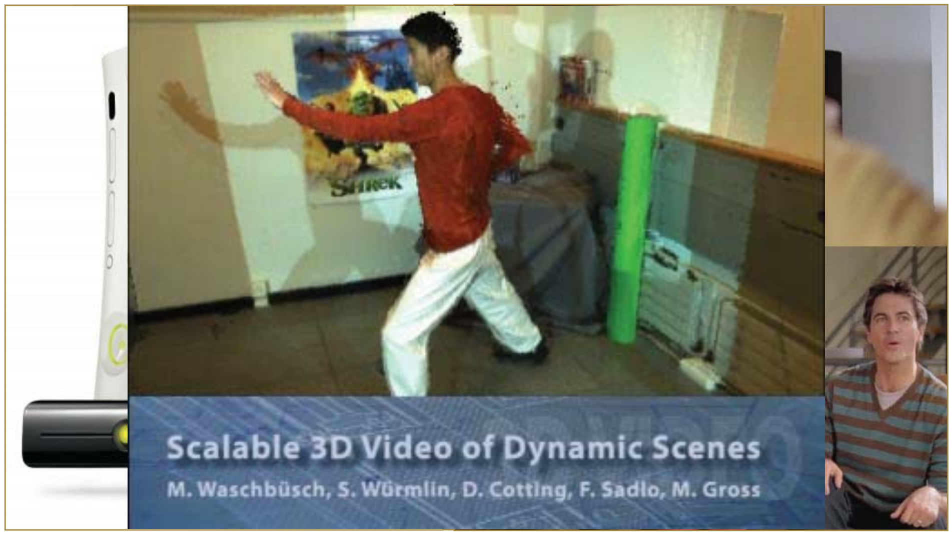
- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

Applications of 3D Scanning:
Entertainment and Consumer Applications



- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

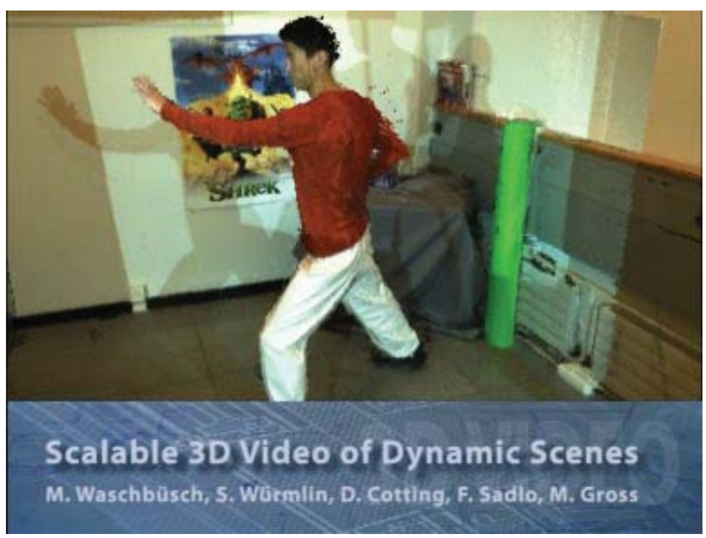
Applications of 3D Scanning:
Entertainment and Consumer Applications



- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

Andreas Wenger et al. Performance Relighting and Reflectance Transformation with Time-Multiplexed Illumination. ACM SIGGRAPH, 2005
M. Waschbüsch et al. Scalable 3D Video of Dynamic Scenes. *The Visual Computer*, 2005.

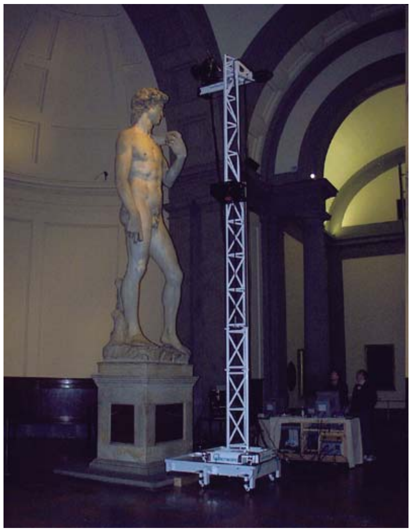
Applications of 3D Scanning:
Entertainment and Consumer Applications

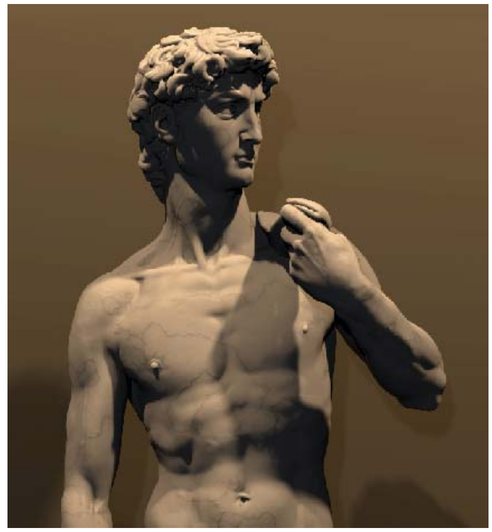


- Import sculptures into a 3D modeling/rendering pipeline
- Capture geometric (and photometric) properties for relighting
- Fit clothes, track 3D interaction, free-viewpoint video (3D TV), etc.

M. Waschbüsch et al. Scalable 3D Video of Dynamic Scenes. *The Visual Computer*, 2005.

Applications of 3D Scanning:
Historical Preservation

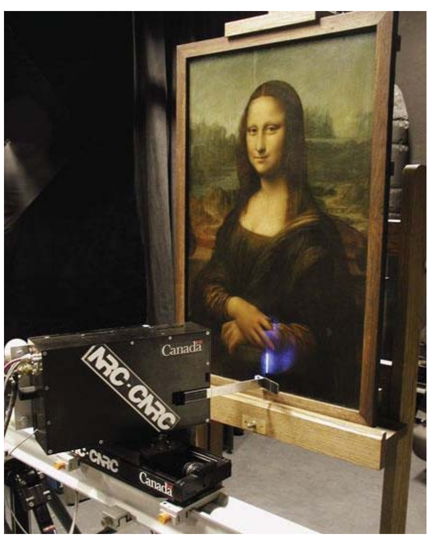




- Preserve/restore deteriorating works and unite dispersed collections
- Facilitate academic study (tooling, lighting, pentimenti, revision history)
- Replicate collections (souvenirs, retain repatriated works, etc.)

M. Levoy et al. The Digital Michelangelo Project: 3D Scanning of Large Statues. ACM SIGGRAPH, 2000

Applications of 3D Scanning:
Historical Preservation



- Preserve/restore deteriorating works and unite dispersed collections
- Facilitate academic study (tooling, lighting, pentimenti, revision history)
- Replicate collections (souvenirs, retain repatriated works, etc.)

L. Borgeat et al. Visualizing and Analyzing the Mona Lisa. *IEEE Computer Graphics and Applications,* 2007

Applications of 3D Scanning: Historical Preservation

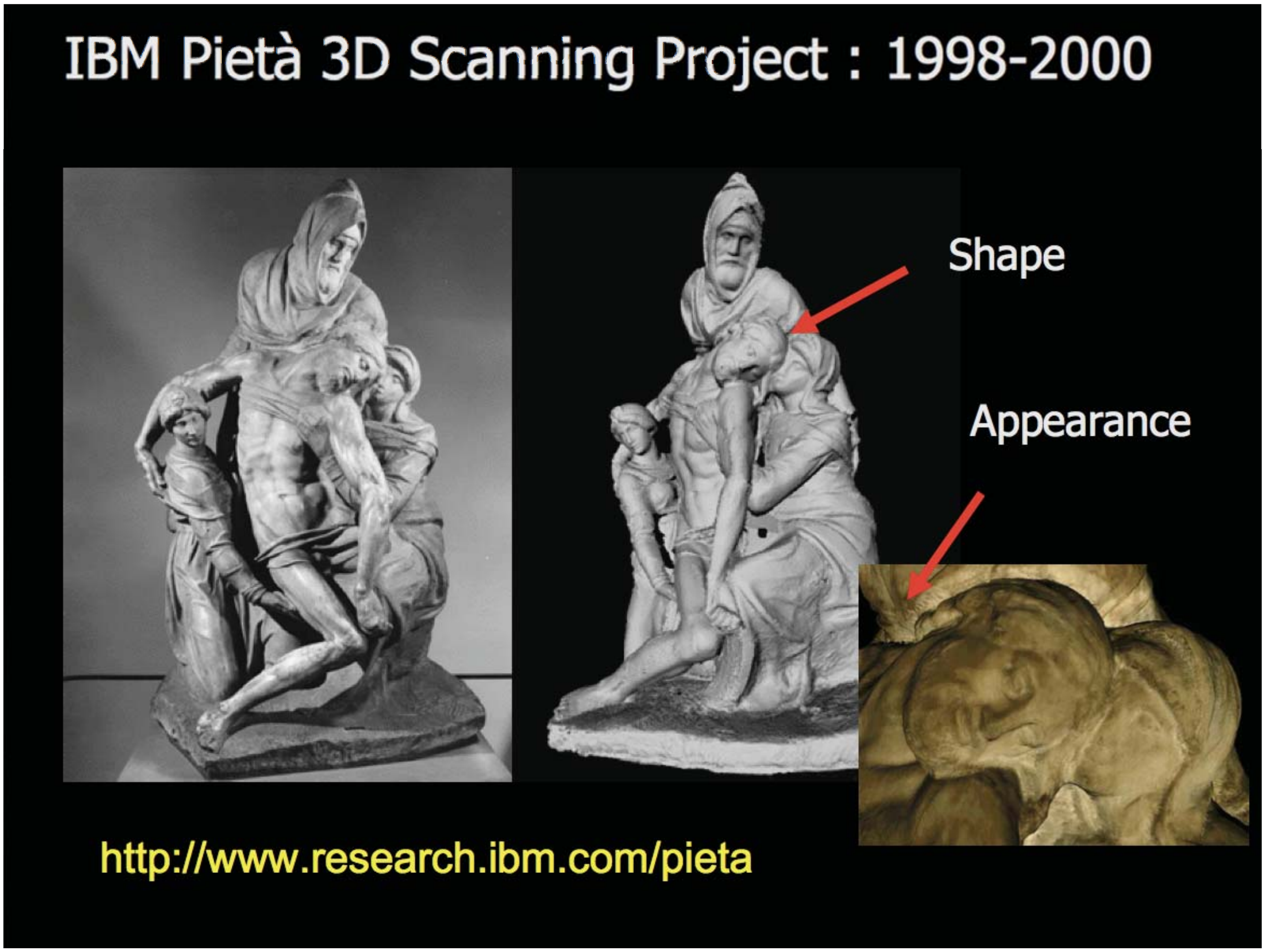
- Preserve/restore deteriorating works and unite dispersed collections
- Facilitate academic study (tooling, lighting, pentimenti, revision history)
- Replicate collections (souvenirs, retain repatriated works, etc.)

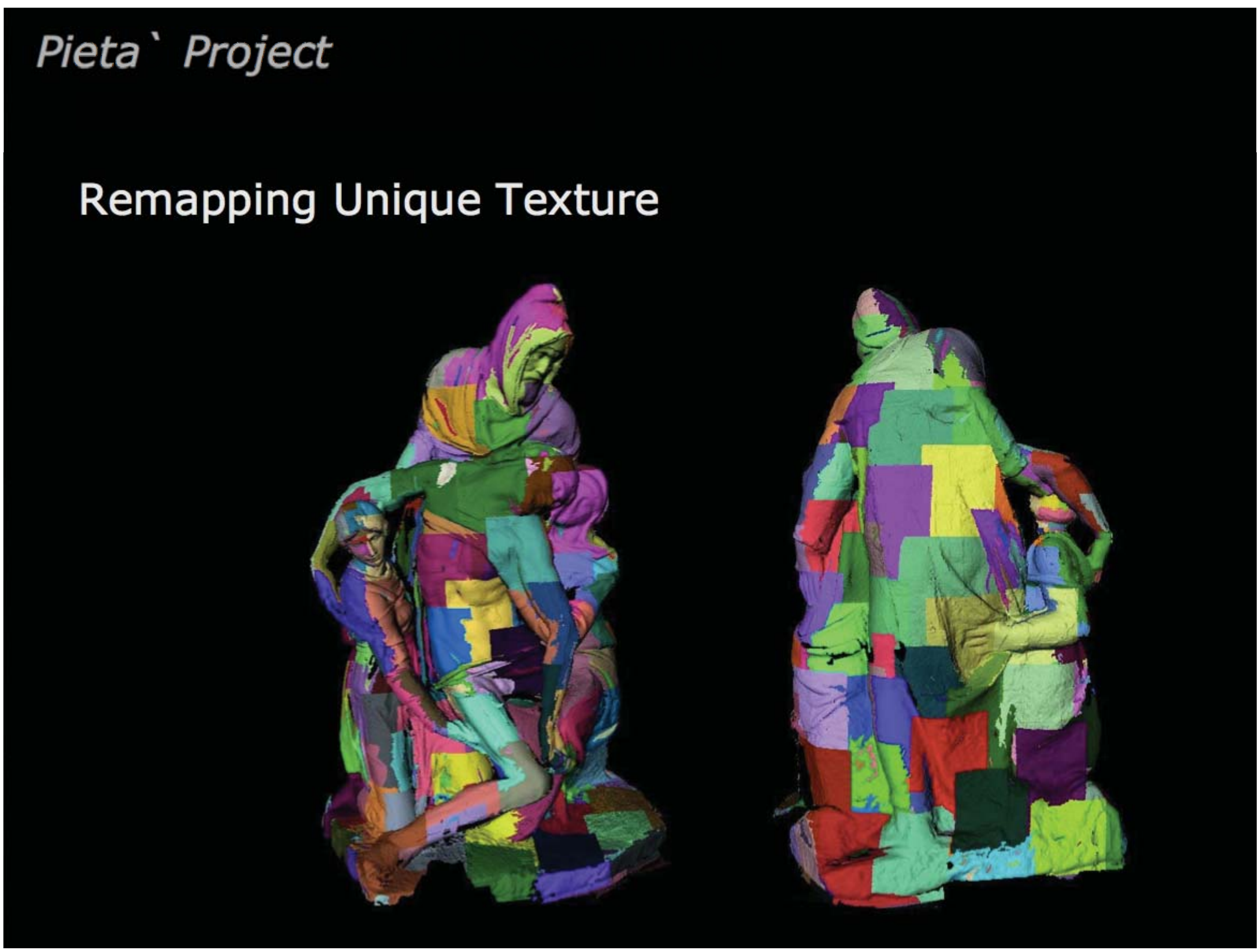
P. Debevec. Making "The Parthenon". Intl. Sym. on Virtual Reality, Archaeology, and Cultural Heritage, 2005

Applications of 3D Scanning: Historical Preservation

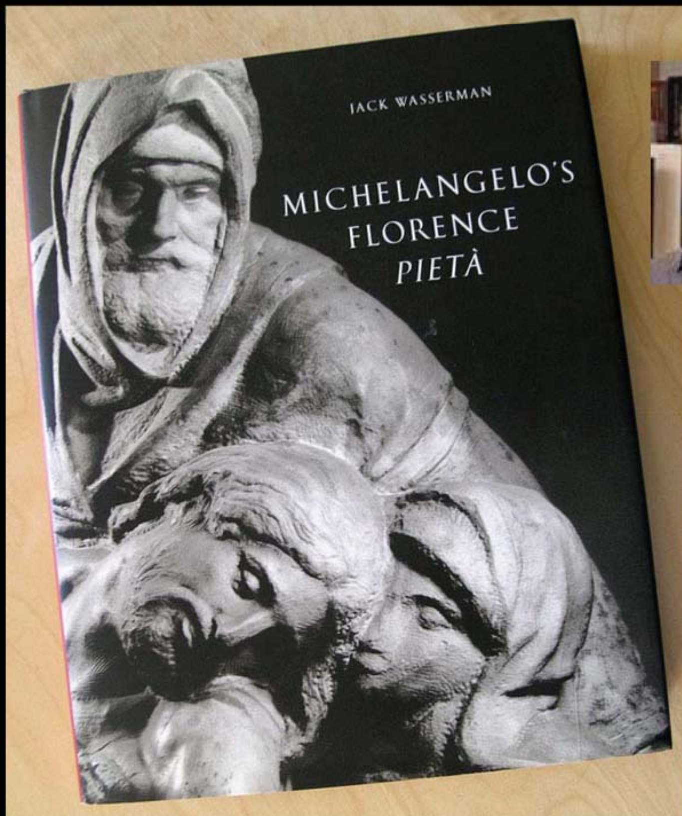
- Preserve/restore deteriorating works and unite dispersed collections
- Facilitate academic study (tooling, lighting, pentimenti, revision history)
- Replicate collections (souvenirs, retain repatriated works, etc.)

P. Debevec. Making "The Parthenon". Intl. Sym. on Virtual Reality, Archaeology, and Cultural Heritage, 2005


IBM Pietà 3D Scanning Project : 1998-2000
Shape
Appearance
http://www.research.ibm.com/pieta


Pieta` Project
Remapping Unique Texture

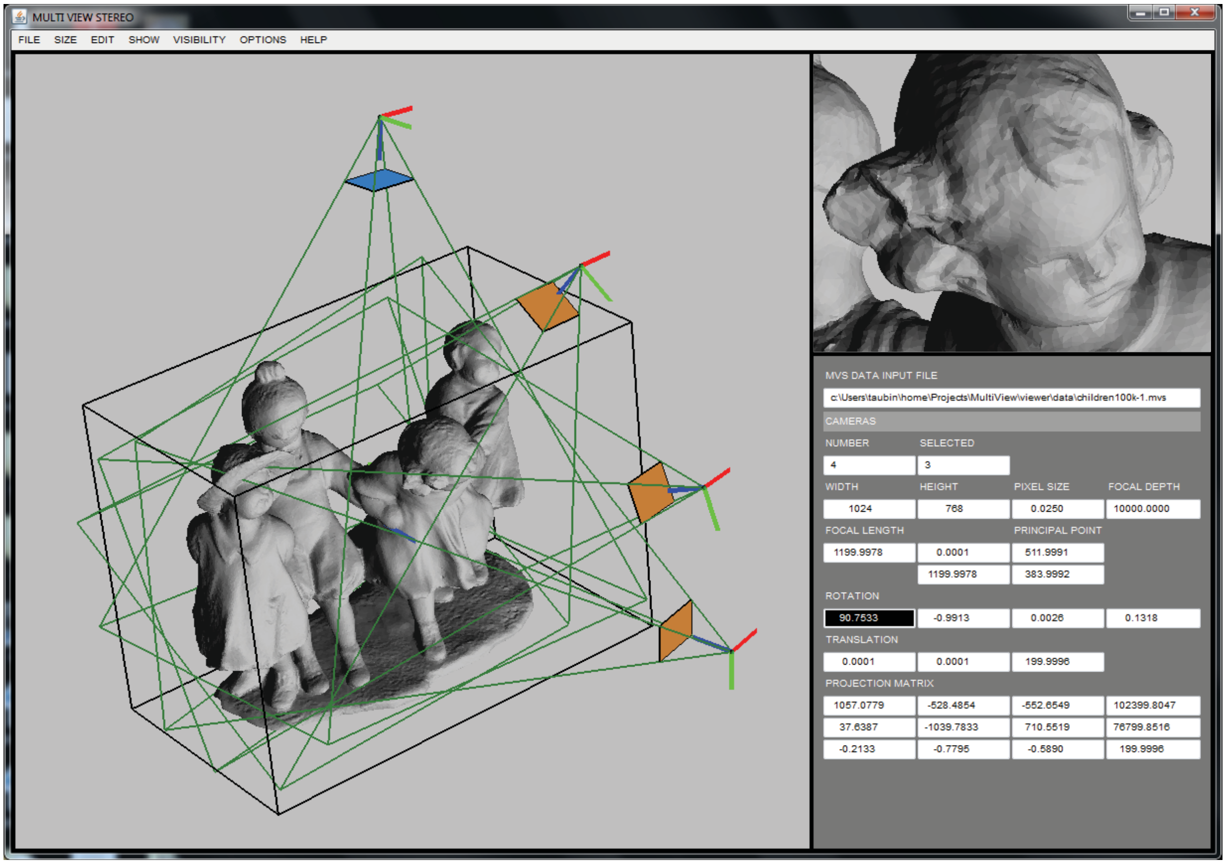
Wasserman's Pietà Book

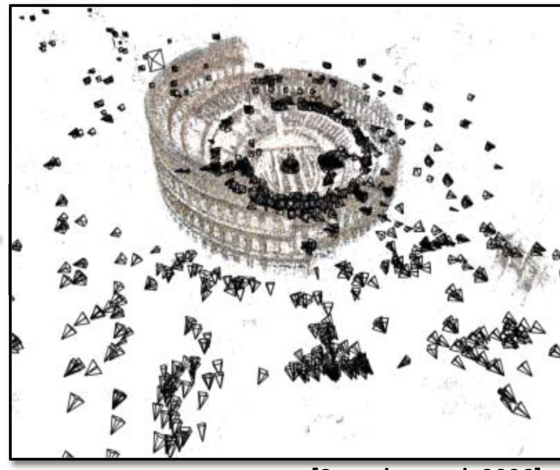


Princeton University Press 2003

Capturing ~800 scans (1998)

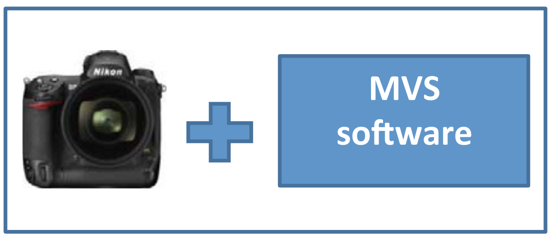
Surface Reconstruction from Multi-View Data





[Snavely et. al. 2006]

http://phototour.cs.washington.edu/bundler/



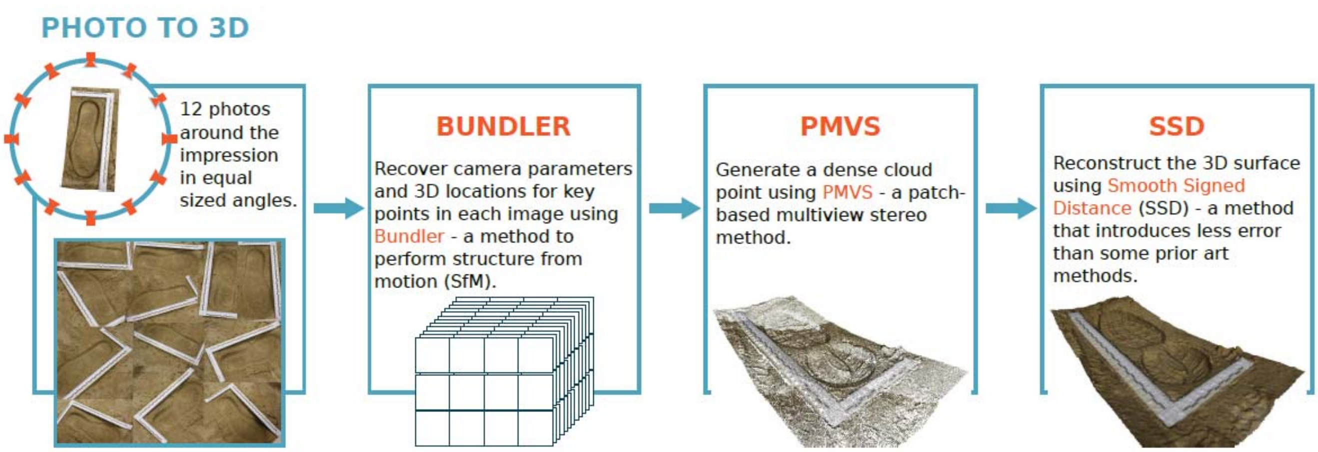
Patch-based **M**ulti-**V**iew **S**tereo (PMVS)
http://grail.cs.washington.edu/software/pmvs/

[Furukawa and Ponce 2008]

Fig. 1 Reconstruction of the side of a castle model: The input point cloud (top-left), Surface reconstructed by the proposed algorithm(top-right), Two views from the surface and color map reconstructed by the proposed algorithm (bottom).

Fig. 6 Reconstruction of a brick with cuneiform, Mesopotamian, 859-840 BCE, clay, Overall 35 x 35 x 11 cm, Williams College Museum of Art, Williamstown, MA, Gift of Professor Edgar J. Banks and Dr. John Henry Haynes, Class of 1876,(20.1.33.A). Top row: the input point cloud (left), and surface geometry (right) reconstructed by the proposed algorithm. Middle row: Two views from surface and color map reconstructed by the proposed algorithm. Bottom row: 6 examples from the set of 21 images that are used for shape acquisition.

Accurate 3D Footwear Impression Recovery From Photographs,
F. A. Andalo, F. Calakli, G. Taubin, and S. Goldenstein,
International Conference on Imaging for Crime Detection and Prevention (ICDP-2011).



Comparable to 3D Laser Scanner

EXPERIMENTAL RESULTS

3D SCANNER

SHOE SOLES

d_g : Haursdoff distance map between the scanned shoe print and the scanned shoe sole.

d_m : Haursdoff distance map between our 3D model and the scanned shoe sole.

Shoeprint #	$\overline{d_g}$	$\overline{d_m}$	$\overline{d_m} - \overline{d_g}$
1	9.996	10.002	0.006
2	8.157	8.660	0.503
3	8.715	9.480	0.765
4	8.816	9.114	0.298

(mm)

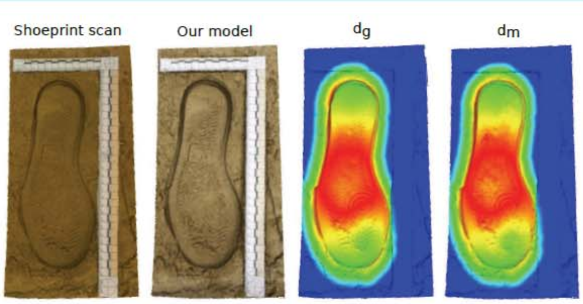
CONCLUSIONS

We presented a pipeline to recover footwear impressions from crime scenes using multiview stereo, which has not been considered for this kind of application until now.

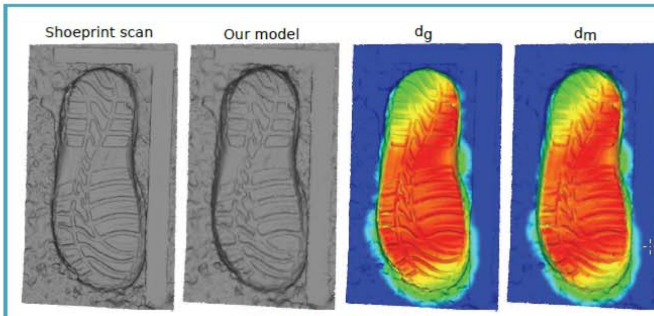
Despite the simplicity, the obtained surfaces are comparable with 3D scanning.

Future work: more experiments - number of images, angle between images, comparison with casting.

EXAMPLES

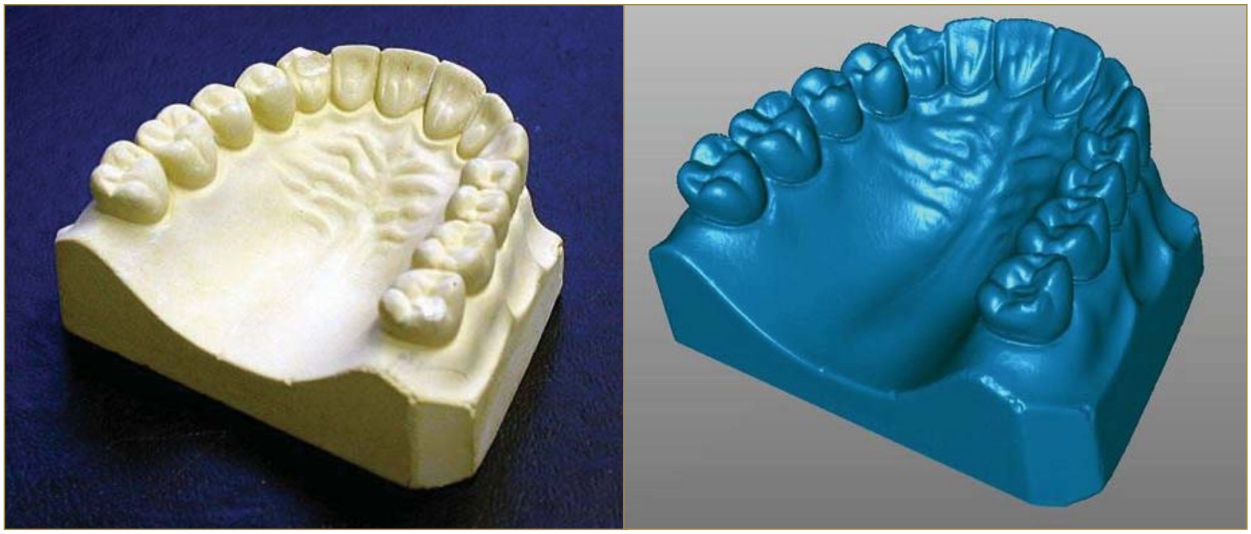


1st Shoeprint



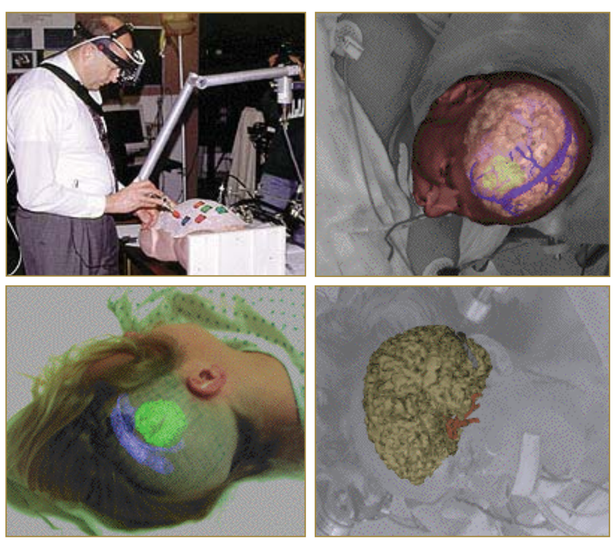
3rd Shoeprint

Applications of 3D Scanning: Medical Imaging and Surgical Planning



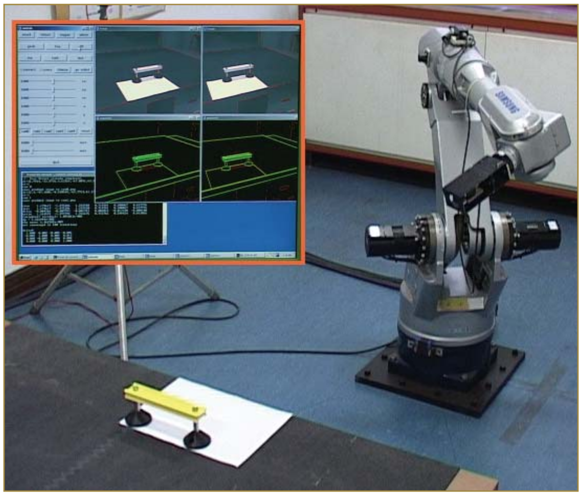
- Medical imaging (X-ray, CT, MRI, etc.) and surgical planning
- Measuring dimensions (dental impressions and hip replacement surgery)
- Tele-surgery (augmented virtual reality, video see-through, etc.)

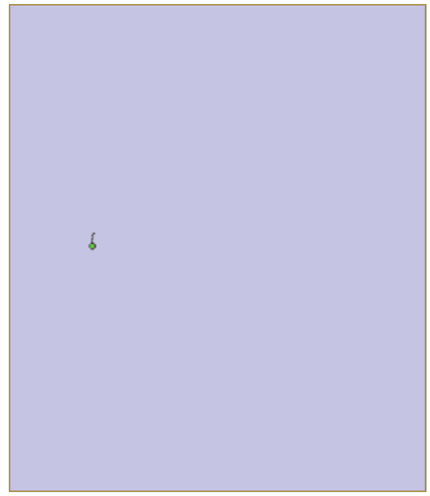
Applications of 3D Scanning: Medical Imaging and Surgical Planning



- Medical imaging (X-ray, CT, MRI, etc.) and surgical planning
- Measuring dimensions (dental impressions and hip replacement surgery)
- Tele-surgery (augmented virtual reality, video see-through, etc.)

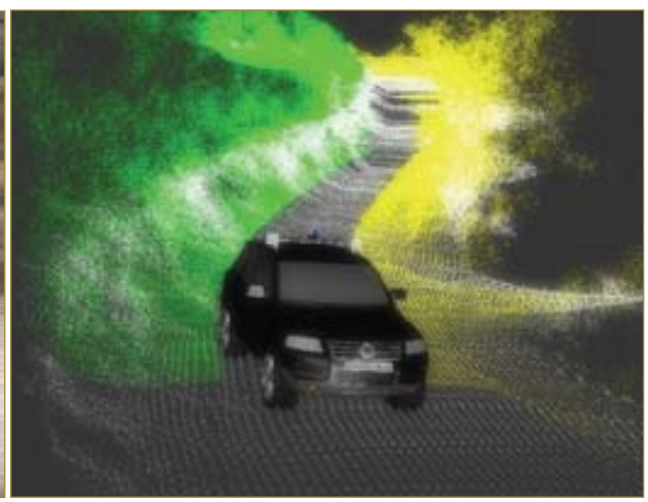
Applications of 3D Scanning:
Robotics (Interaction and Navigation)





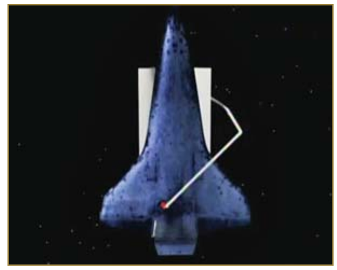
- Motion planning (manipulation, gripping, pushing/pulling, etc.)
- Simultaneous localization and mapping (SLAM)
- Autonomous navigation (DARPA Grand/Urban Challenge)

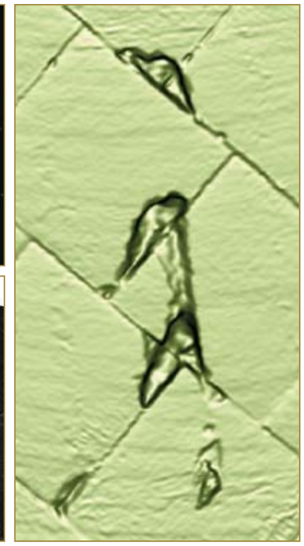
Applications of 3D Scanning:
Robotics (Interaction and Navigation)



- Motion planning (manipulation, gripping, pushing/pulling, etc.)
- Simultaneous localization and mapping (SLAM)
- Autonomous navigation (DARPA Grand/Urban Challenge)

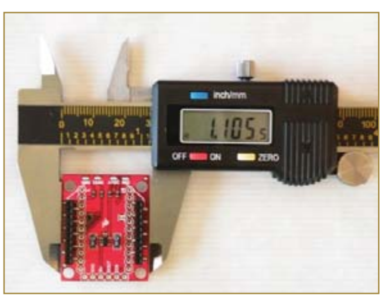
Applications of 3D Scanning:
Inspection and Reverse Engineering

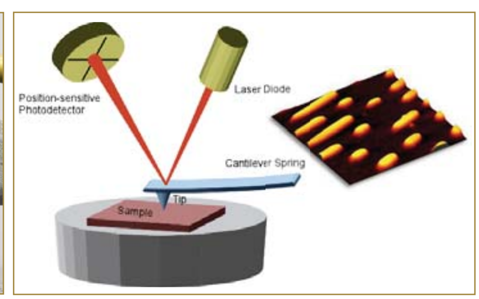




- Manufacturing and process control (tolerances and alignment)
- Reverse engineering (repairing antiques and replicating designs)
- Remote inspection (inaccessible or dangerous environments)

Taxonomy of 3D Scanning:
Direct Contact





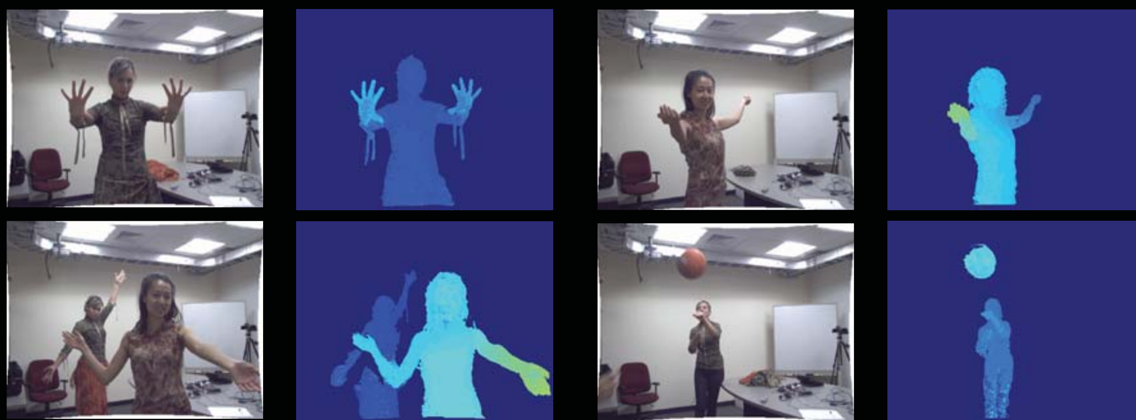
Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact

Taxonomy adapted from Szymon Rusinkiewicz

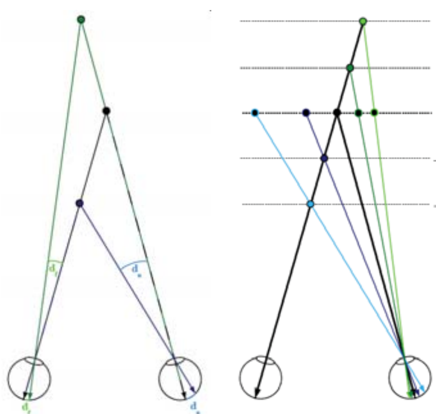
Real-Time High-Definition Stereo on GPGPU using Progressive Multi-Resolution Adaptive Windows

Y. Zhao, and G. Taubin, Image and Vision Computing 2011.



Screen shots of our real-time stereo system working on the field

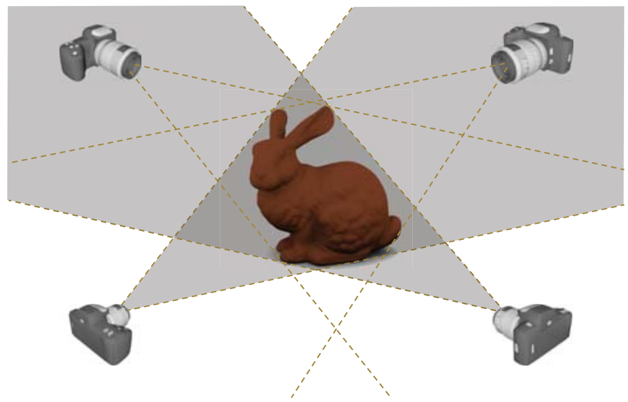
Taxonomy of 3D Scanning:
Stereo/Multi-view Photography



Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
(stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

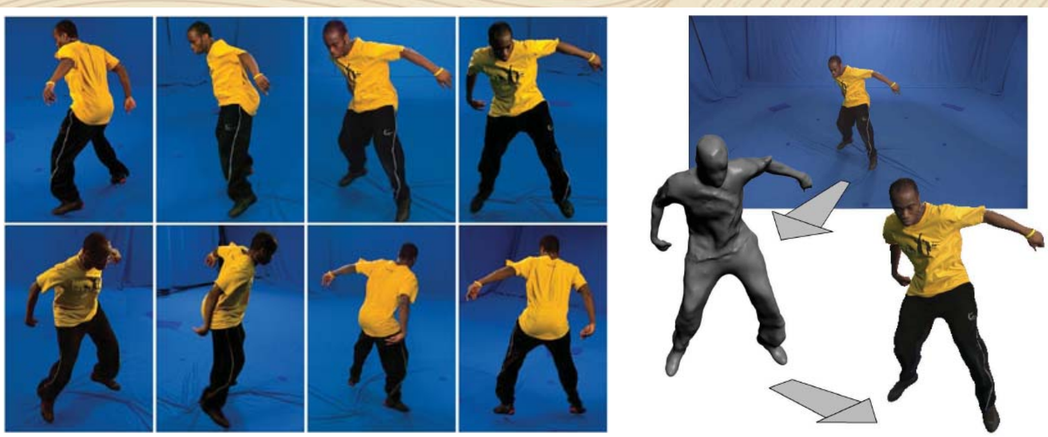
Taxonomy of 3D Scanning:
Shape-from-Silhouettes



Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-frm-X
  (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

Taxonomy of 3D Scanning:
Shape-from-Silhouettes

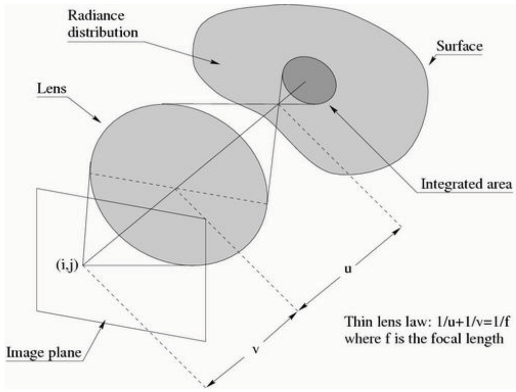


Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
  (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

J. Starck and A. Hilton. Surface Capture for Performance-Based Animation. *IEEE Computer Graphics and Applications*, 2007

Taxonomy of 3D Scanning:
Shape-from-Focus/Defocus

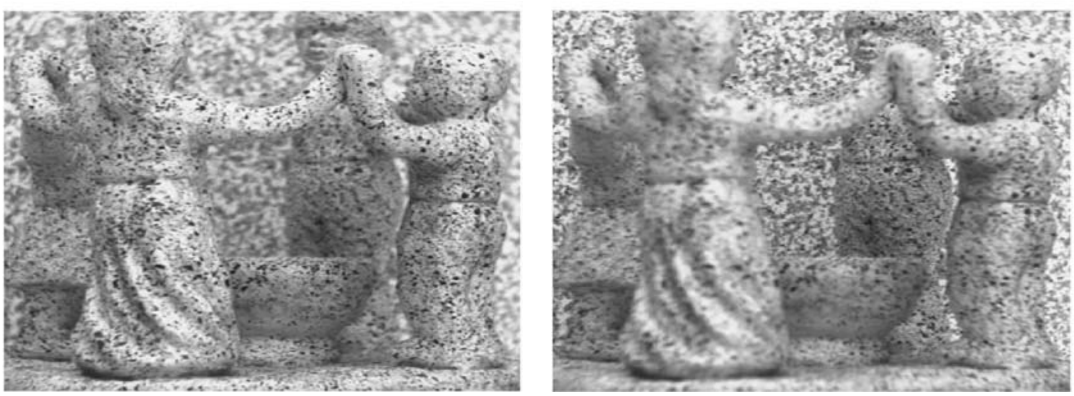


Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
(stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

M. Watanabe and S. Nayar. Rational filters for passive depth from defocus. *Intl. J. of Comp. Vision*, 27(3):203-225, 1998

Taxonomy of 3D Scanning:
Shape-from-Focus/Defocus

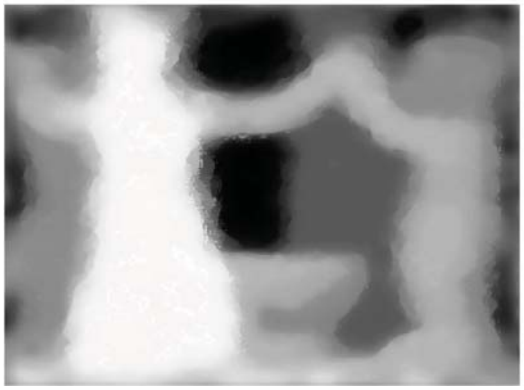


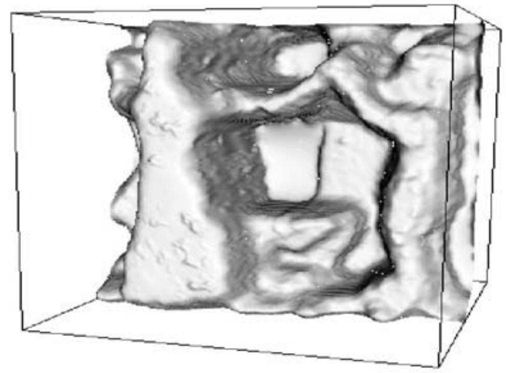
Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
(stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

M. Watanabe and S. Nayar. Rational filters for passive depth from defocus. *Intl. J. of Comp. Vision*, 27(3):203-225, 1998

Taxonomy of 3D Scanning:
Shape-from-Focus/Defocus



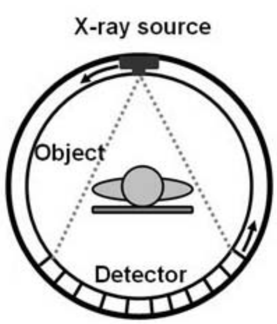


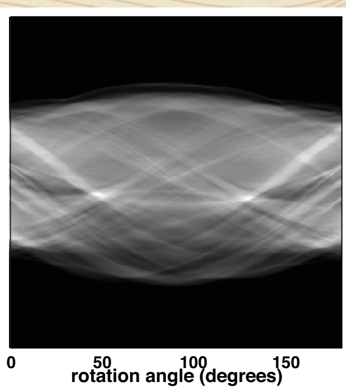
Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
  (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**

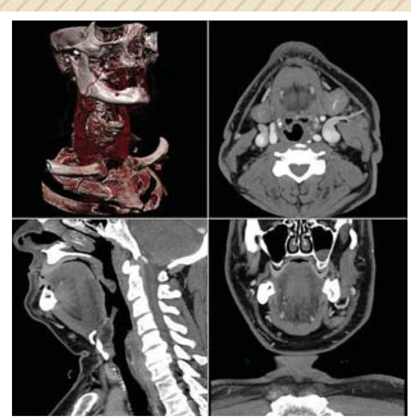
M. Watanabe and S. Nayar. Rational filters for passive depth from defocus. *Intl. J. of Comp. Vision*, 27(3):203-225, 1998

Taxonomy of 3D Scanning:
Computed Tomography (CT)





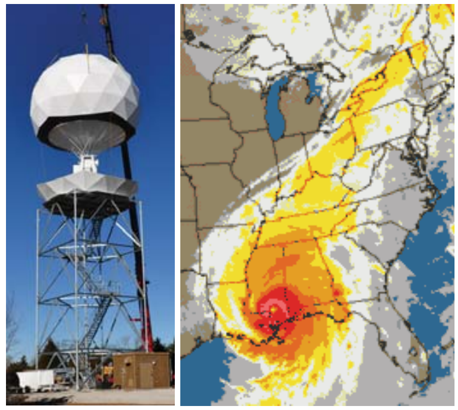
Parallel/Fan-beam Projections

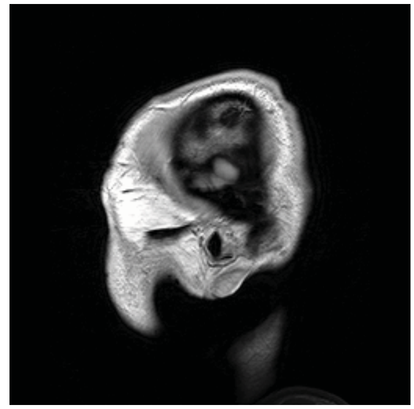


Contact — Direct Measurements
(rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)

Non-Contact
- **Passive** — Shape-from-X
  (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
- **Active**
  - **Transmissive**
    - Computed Tomography (CT)
    - Transmissive Ultrasound
  - **Reflective**

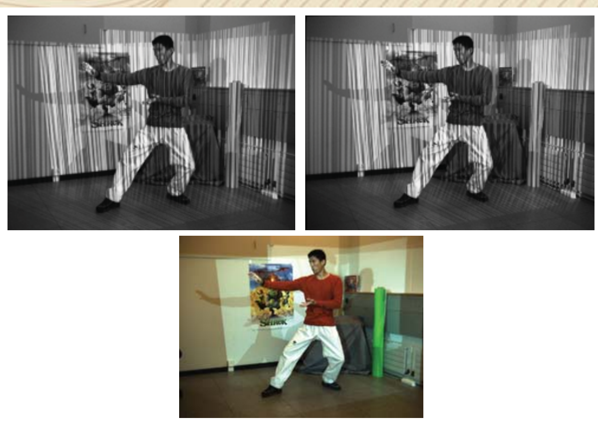
Taxonomy of 3D Scanning: Non-optical Active Methods





- **Contact** — Direct Measurements (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods (reflective ultrasound, radar, sonar, MRI)

Taxonomy of 3D Scanning: Active Variants of Passive Methods



- **Contact** — Direct Measurements (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods (reflective ultrasound, radar, sonar, MRI)

      - Active Variants of Passive Methods (stereo/focus/defocus using projected patterns)

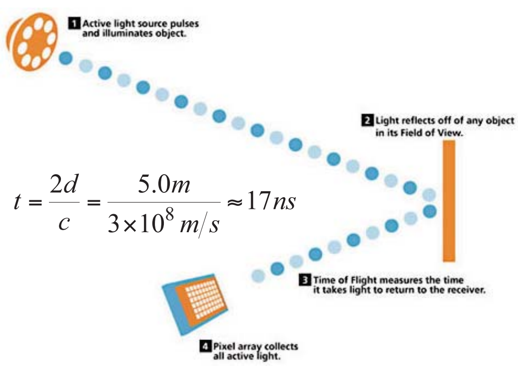
M. Waschbüsch et al. Scalable 3D Video of Dynamic Scenes.
The Visual Computer, pp. 629-638, 2005

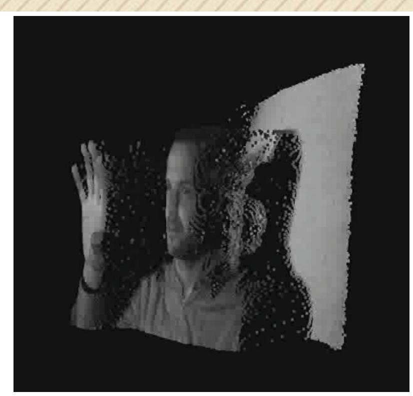
Taxonomy of 3D Scanning:
Active Variants of Passive Methods

- **Contact** — Direct Measurements
  (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X
    (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods
        (reflective ultrasound, radar, sonar, MRI)
      - Active Variants of Passive Methods
        (stereo/focus/defocus using projected patterns)

M. Watanabe and S. Nayar. Rational Filters for Passive Depth from Defocus. *Intl. J. of Comp. Vision*, 27(3):203-225, 1998

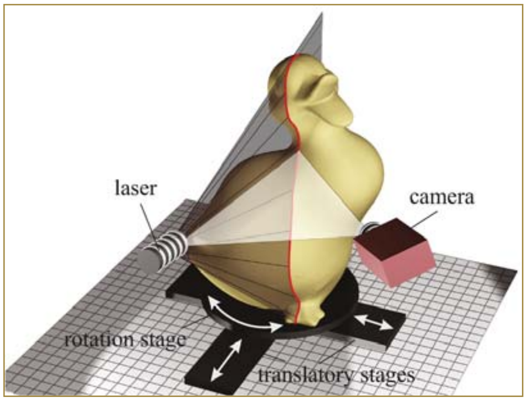
Taxonomy of 3D Scanning:
Time-of-Flight

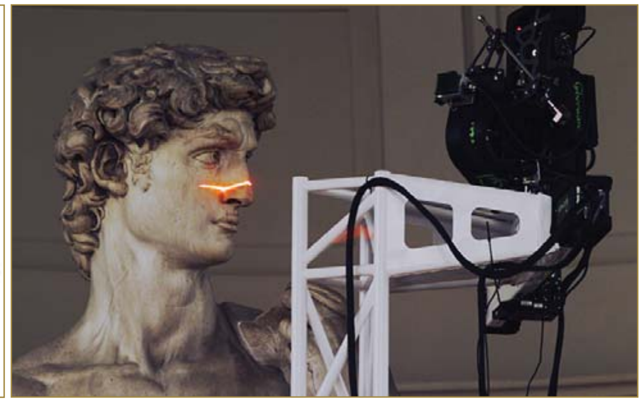




- **Contact** — Direct Measurements
  (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X
    (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods
        (reflective ultrasound, radar, sonar, MRI)
      - Active Variants of Passive Methods
        (stereo/focus/defocus using projected patterns)
      - Time-of-Flight

Taxonomy of 3D Scanning:
Triangulation with Laser Striping

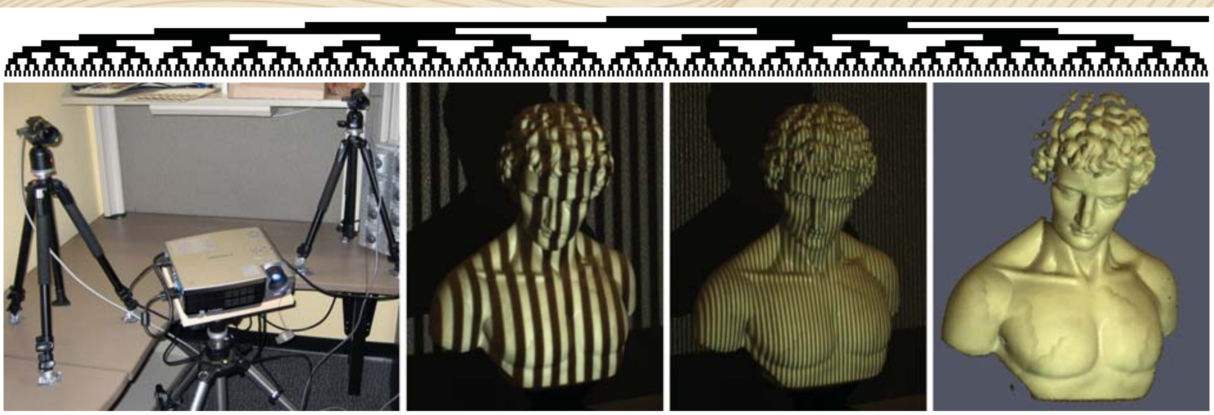




- **Contact** — Direct Measurements
  (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X
    (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods
        (reflective ultrasound, radar, sonar, MRI)
      - Active Variants of Passive Methods
        (stereo/focus/defocus using projected patterns)
      - Time-of-Flight
      - Triangulation
        (laser striping and structured lighting)

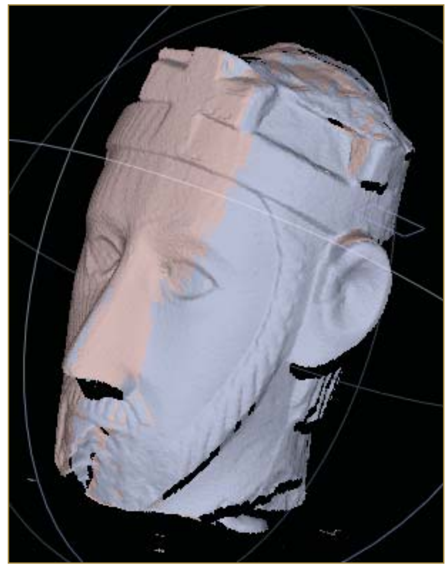
M. Levoy et al. The Digital Michelangelo Project: 3D Scanning of Large Statues. Proc. ACM SIGGRAPH, 2000

Taxonomy of 3D Scanning:
Triangulation with Structured Lighting



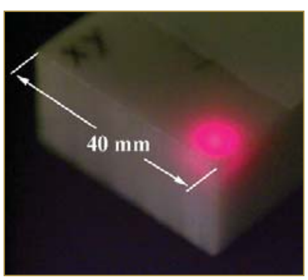
- **Contact** — Direct Measurements
  (rulers, calipers, pantographs, coordinate measuring machines (CMM), AFM)
- **Non-Contact**
  - **Passive** — Shape-from-X
    (stereo/multi-view, silhouettes, focus/defocus, motion, texture, etc.)
  - **Active**
    - **Transmissive**
      - Computed Tomography (CT)
      - Transmissive Ultrasound
    - **Reflective**
      - Non-optical Methods
        (reflective ultrasound, radar, sonar, MRI)
      - Active Variants of Passive Methods
        (stereo/focus/defocus using projected patterns)
      - Time-of-Flight
      - Triangulation
        (laser striping and structured lighting)

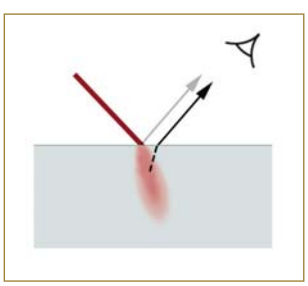
Challenges of Optical 3D Scanning

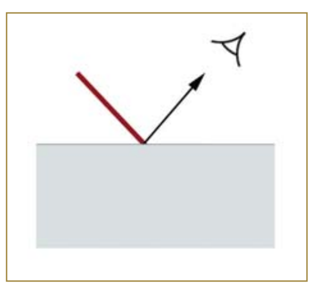


- Must be simultaneously *illuminated* and *imaged* (occlusion problems)
- Non-Lambertian BRDFs (transparency, reflections, subsurface scattering)
- Acquisition time (dynamic scenes), large (or small) features, etc.

Challenges of Optical 3D Scanning





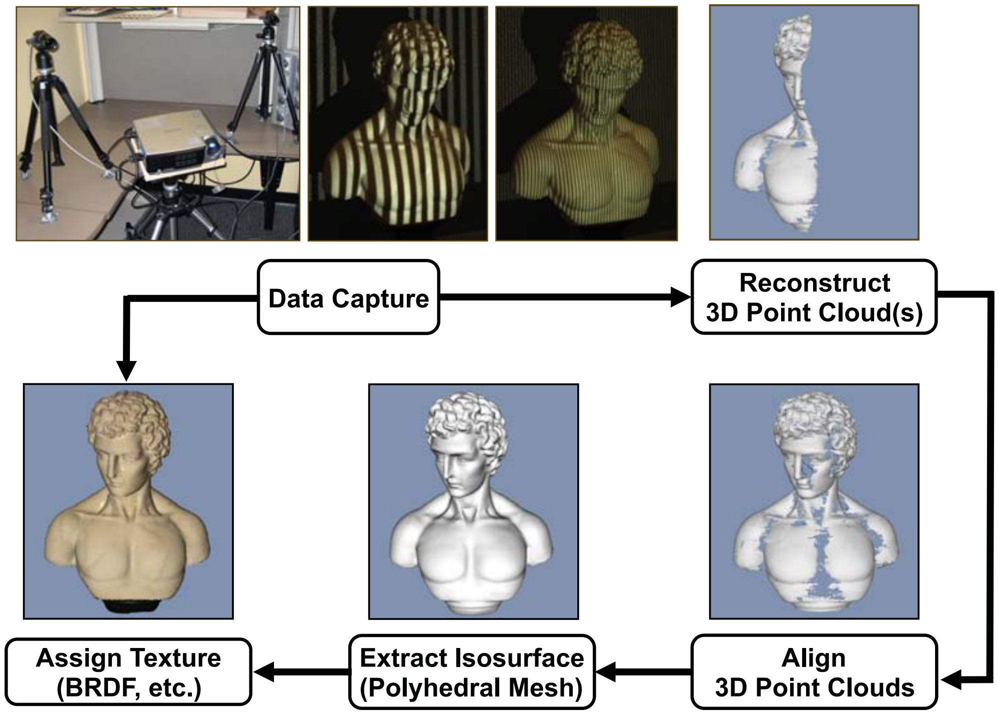


- Must be simultaneously *illuminated* and *imaged* (occlusion problems)
- Non-Lambertian BRDFs (transparency, reflections, subsurface scattering)
- Acquisition time (dynamic scenes), large (or small) features, etc.

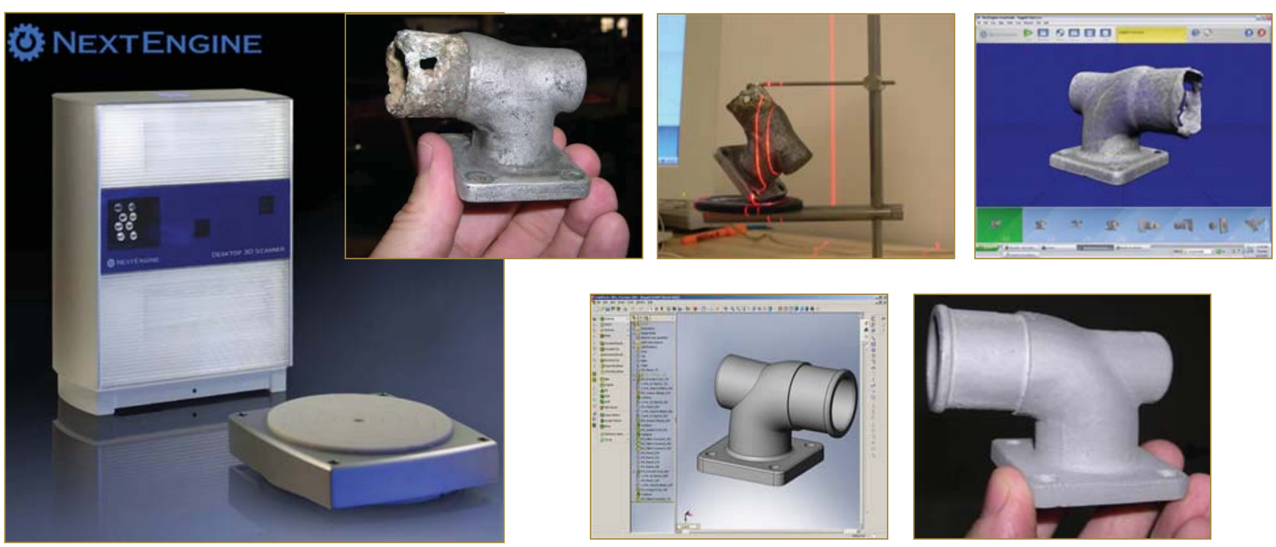
M. Levoy. Why is 3D scanning hard? 3DPVT, 2002

Godin et al. An Assessment of Laser Range Measurement on Marble Surfaces. Intl. Conf. Optical 3D Measurement Techniques, 2001

The 3D Scanning Pipeline



Commercial 3D Scanners



Features, Limitations, and Benefits

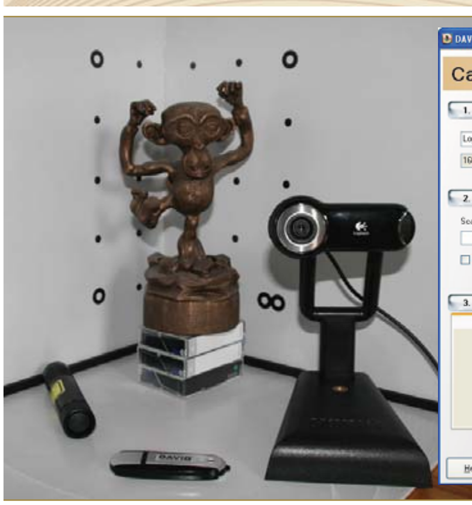
- Most commercial scanners use laser striping + turntables/fiducials
- Cost varies (NextEngine ~$3,000 USD, others more expensive)
- Complete pipeline (including registration and isosurface extraction)

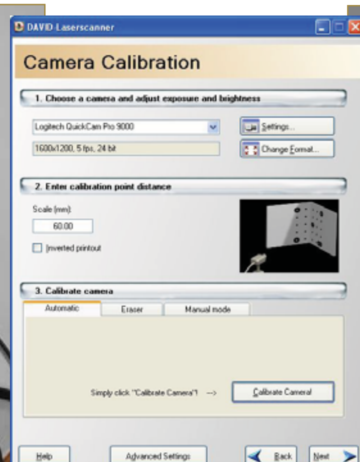
Commercial 3D Scanners

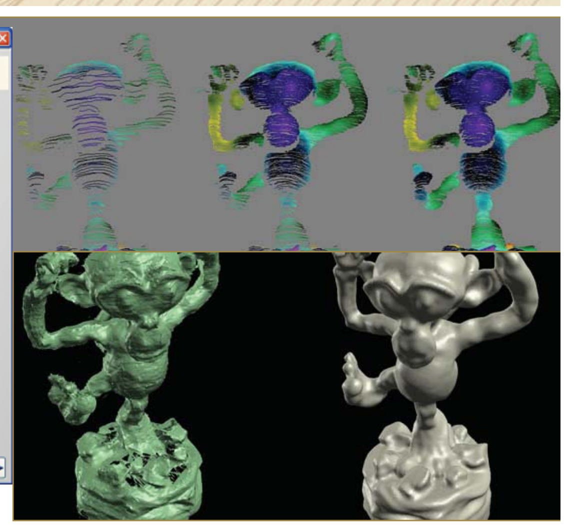
Features, Limitations, and Benefits

- Most commercial scanners use laser striping + turntables/fiducials
- Cost varies (NextEngine ~$3,000 USD, others more expensive)
- Complete pipeline (including registration and isosurface extraction)

Do-It-Yourself (DIY) 3D Scanners



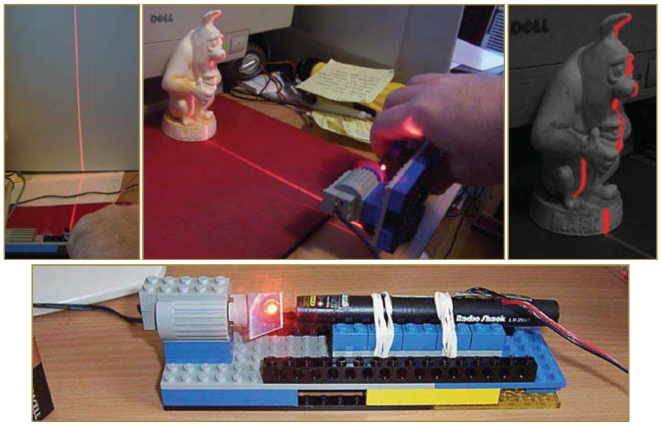


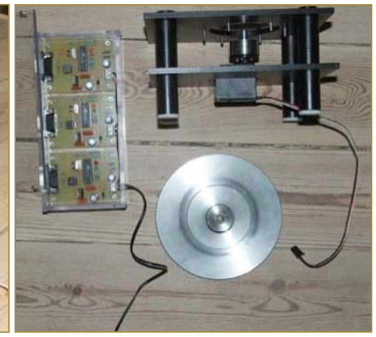


Features, Limitations, and Benefits

- Most DIY scanners also use laser striping + turntables
- Relatively inexpensive (DAVID laser scanner ~$550 USD for starter kit)
- Incomplete pipeline (lacking registration and isosurface extraction)
- Most (but not all) lack proper camera and light source calibration

Do-It-Yourself (DIY) 3D Scanners

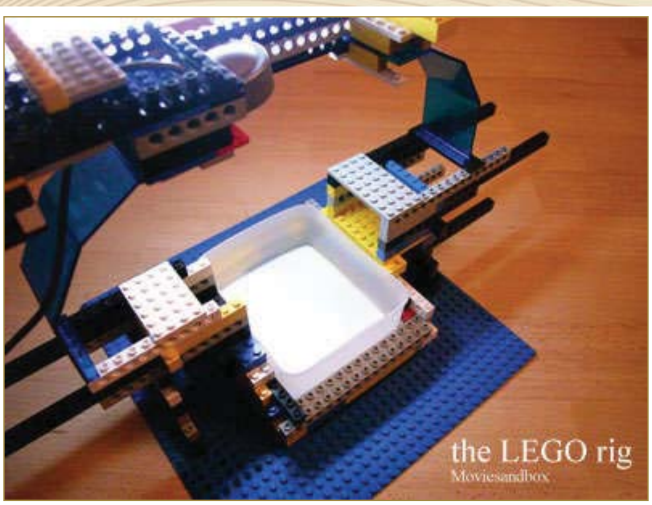




Features, Limitations, and Benefits

- Most DIY scanners also use laser striping + turntables
- Relatively inexpensive (DAVID laser scanner ~$550 USD for starter kit)
- Incomplete pipeline (lacking registration and isosurface extraction)
- Most (but not all) lack proper camera and light source calibration

Do-It-Yourself (DIY) 3D Scanners



Features, Limitations, and Benefits

- Most DIY scanners also use laser striping + turntables
- Relatively inexpensive (DAVID laser scanner ~$550 USD for starter kit)
- Incomplete pipeline (lacking registration and isosurface extraction)
- Most (but not all) lack proper camera and light source calibration

Do-It-Yourself (DIY) 3D Scanners

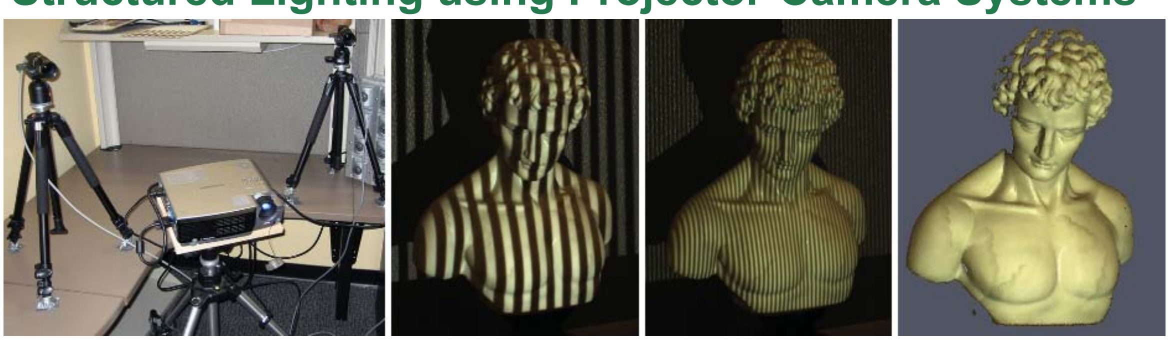
Features, Limitations, and Benefits

- Most DIY scanners also use laser striping + turntables
- Relatively inexpensive (DAVID laser scanner ~$550 USD for starter kit)
- Incomplete pipeline (lacking registration and isosurface extraction)
- Most (but not all) lack proper camera and light source calibration

Topics/Scanners in this Course

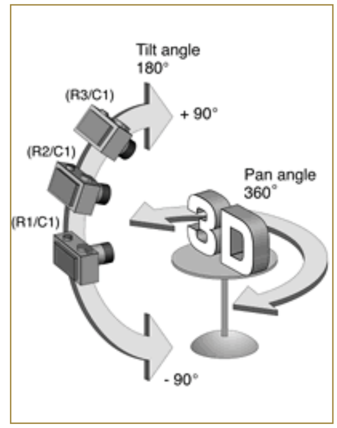
1) Scanning with Swept-Planes

2) Structured Lighting using Projector-Camera Systems



3) Post-processing Pipeline: Registration and Isosurfaces

When not to Scan?



- Scanning is (usually) unnecessary when output is another image!
- Better to use image-based rendering (light fields, QTVR, etc.)

Marc Levoy. Stanford Spherical Gantry. On-line, 2005

Next Class

- Introduction to 3D Scanning
- **The Mathematics of 3D Triangulation**
- 3D Scanning with Swept-Planes | Slit scanner
- Camera and Swept-Plane Light Source Calibration

Introducción a la Fotografia 3D
UBA/FCEN Marzo 27 – Abril 12 2013
Clase 2 : Miercoles Abril 3

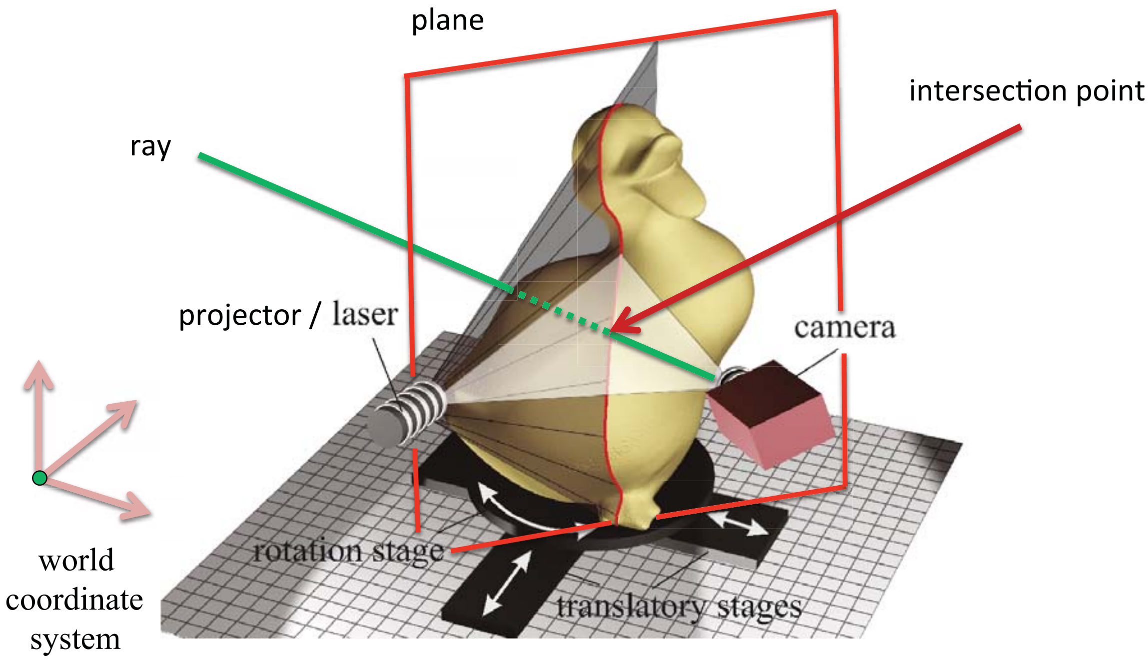
Gabriel Taubin
Brown University

BROWN

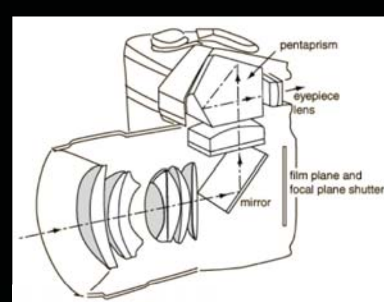
Las matemáticas de la triangulacion 3D

- Triangulation por interseccion de linea y plano
- Triangulation por interseccion de linea y linea
- Puntos, vectores, lineas, rayos, and planos
- Que es una camara?
- Que es una image?
- …

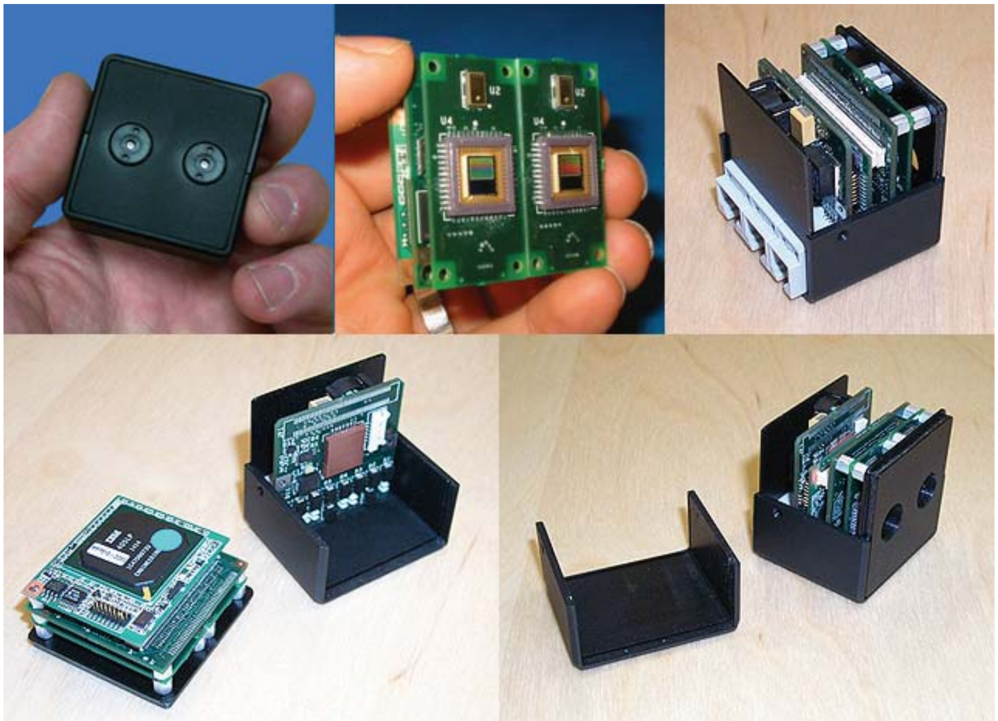
3D triangulation: ray-plane Intersection



Que es una Camara?



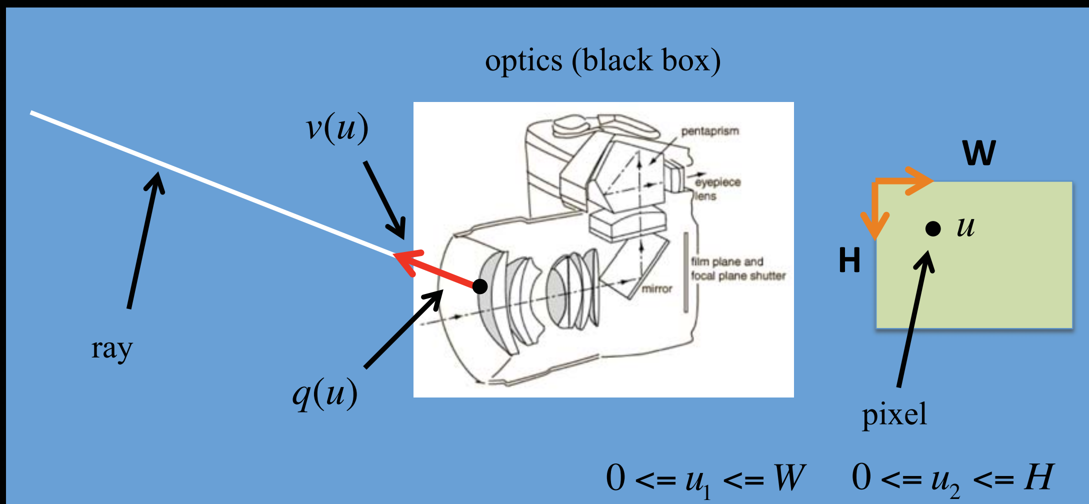
What is a Camera ?



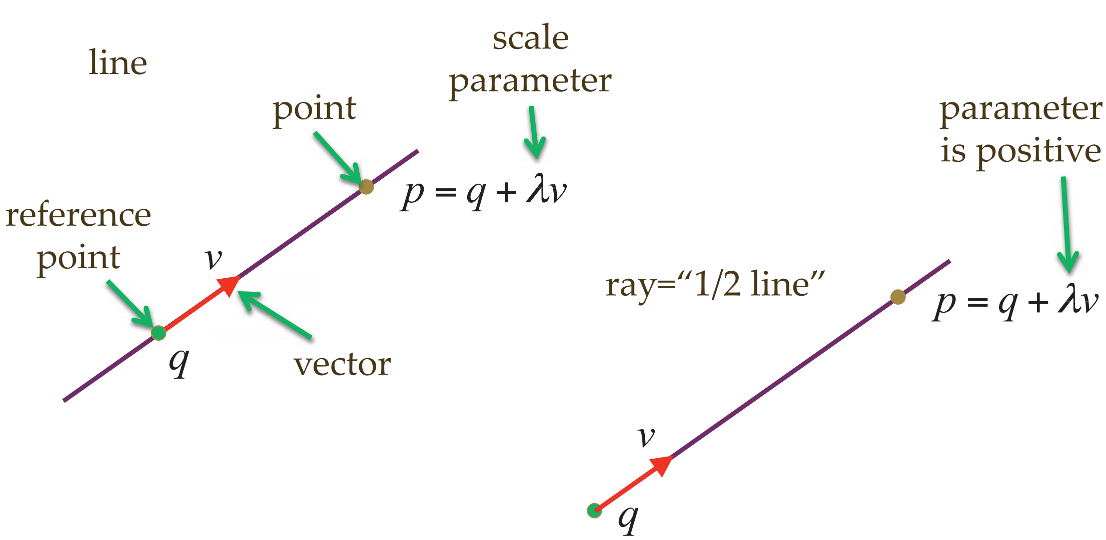
What is a Camera ?

Modelo Geometrico General de una Camara

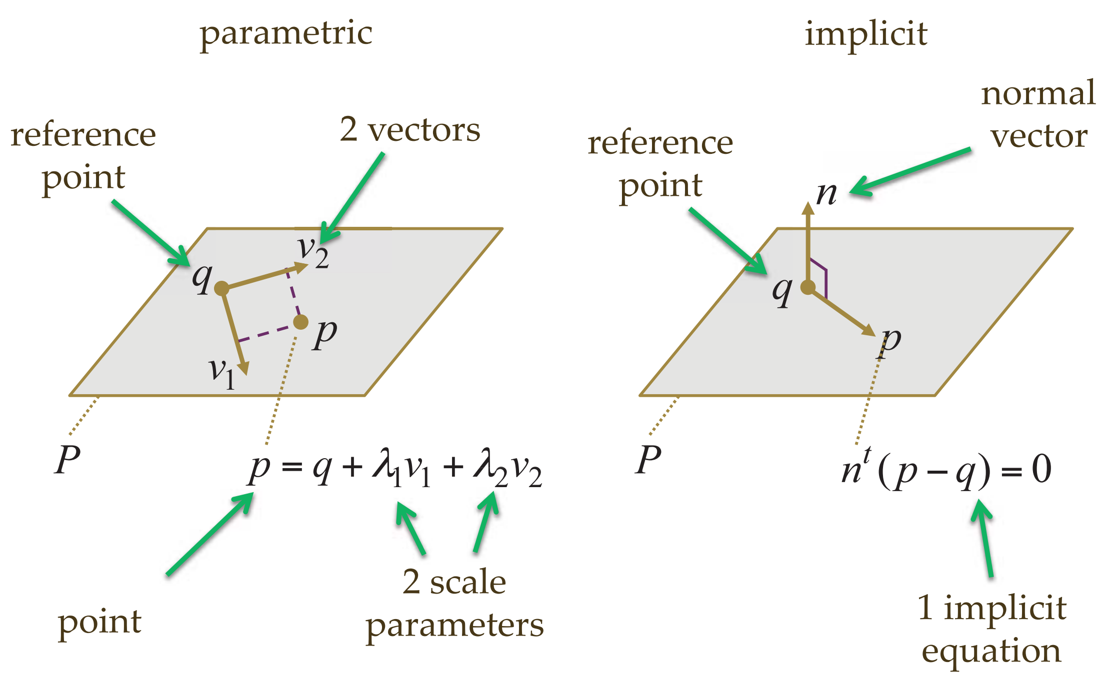
- Funcion que a cada punto de la imagen le hace corresponder un rayo
- El dominio esta contenido en un rectangulo y la funcion es continua
- En muchos casos el análisis es mas simple en el espacio de los rayos



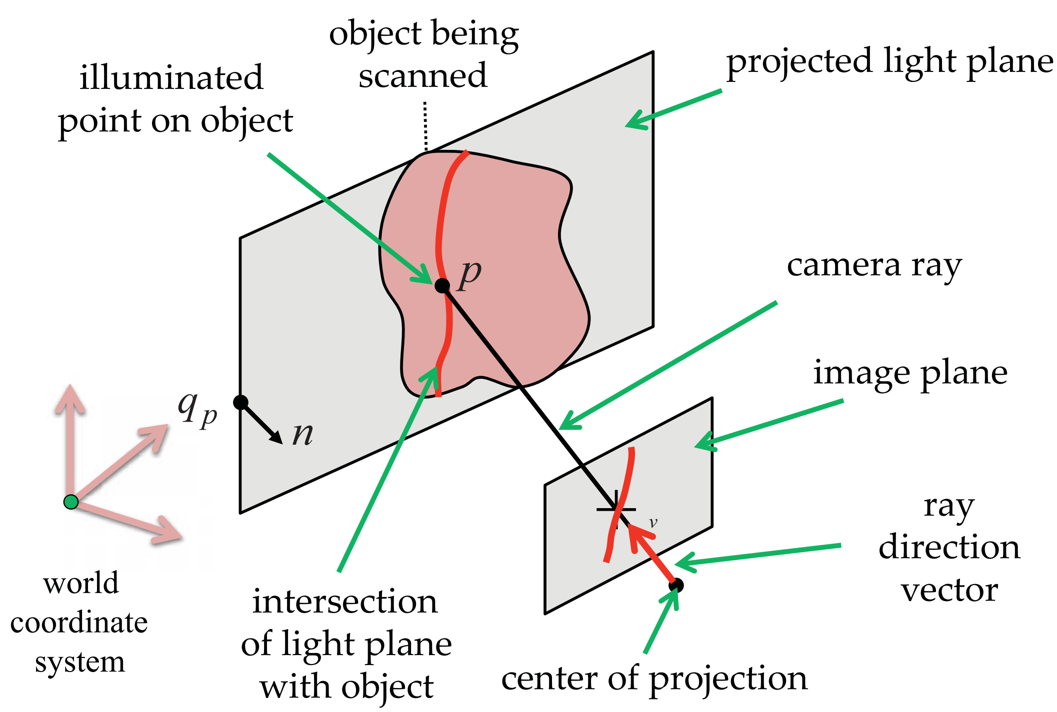
Representation of Lines and Rays



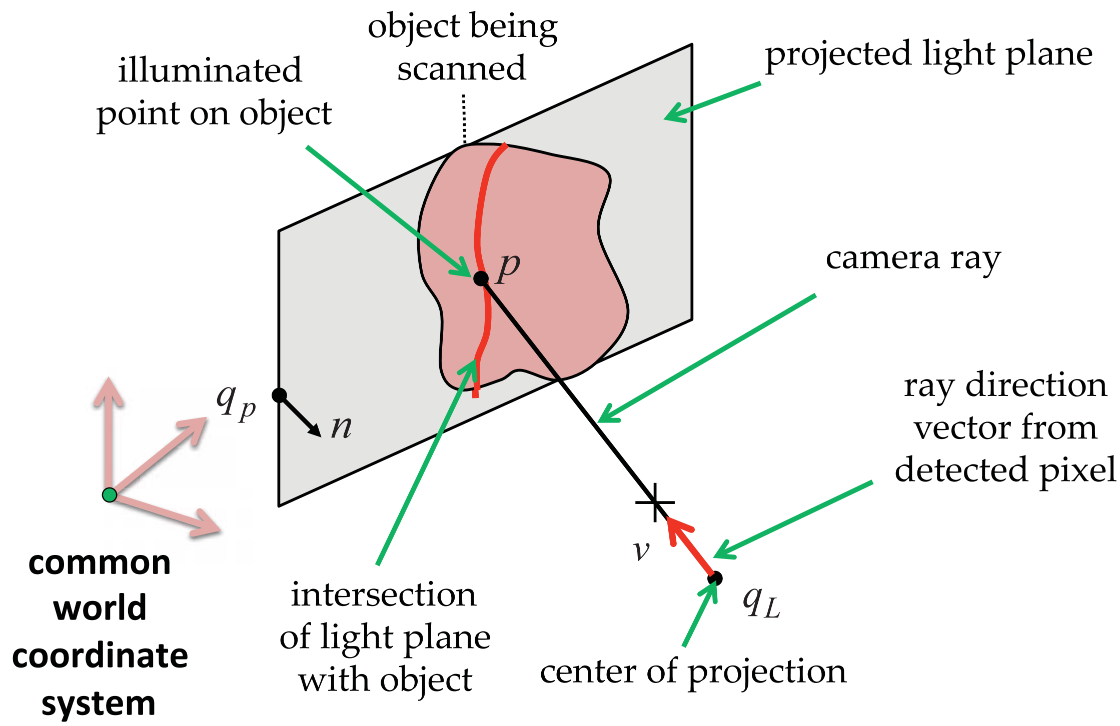
Representation of Planes



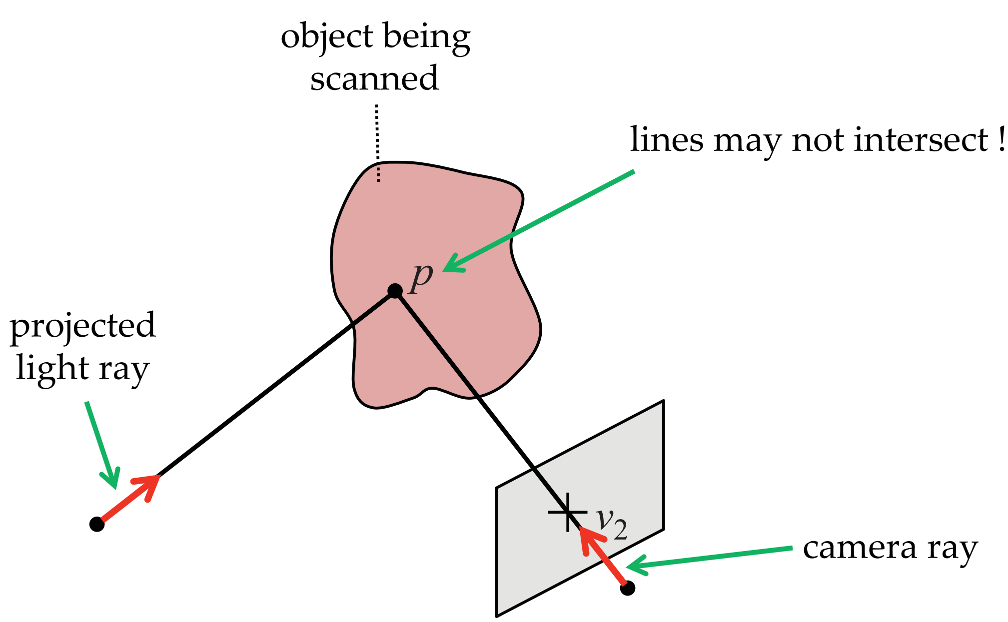
3D Triangulation by Line-Plane intersection



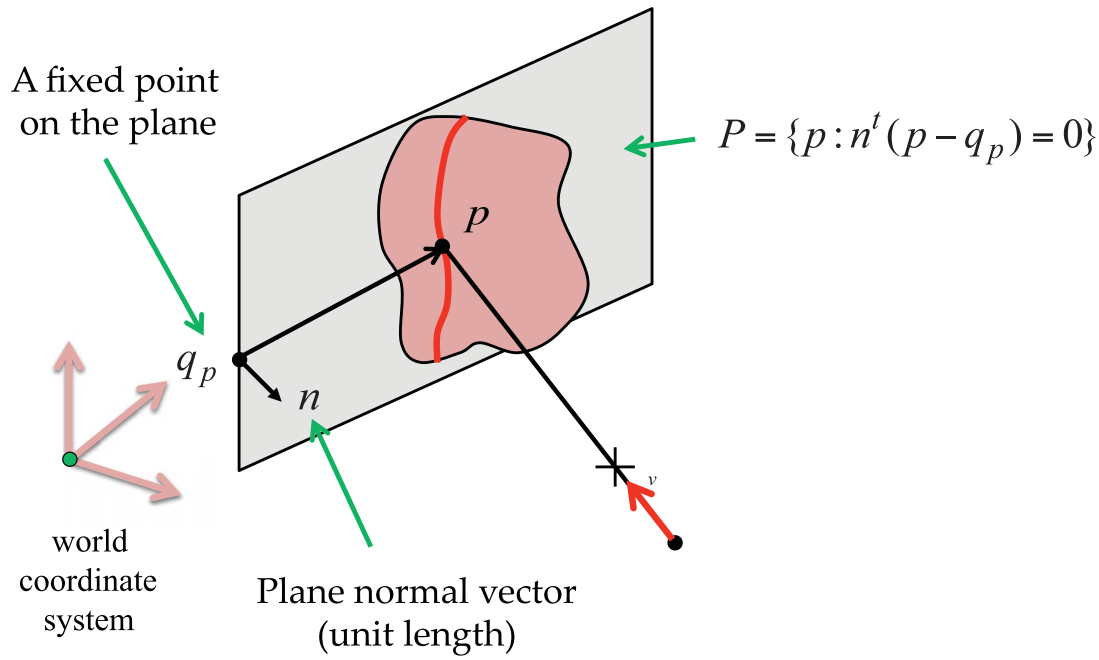
If camera and projector are calibrated



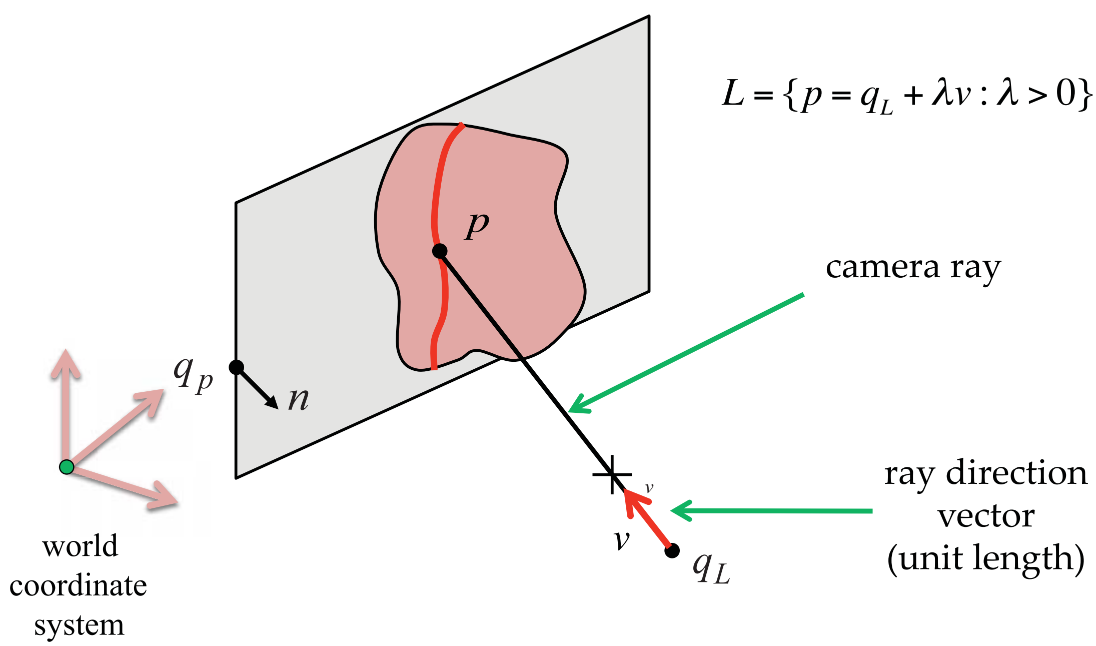
3D Triangulation by Line-Line Intersection



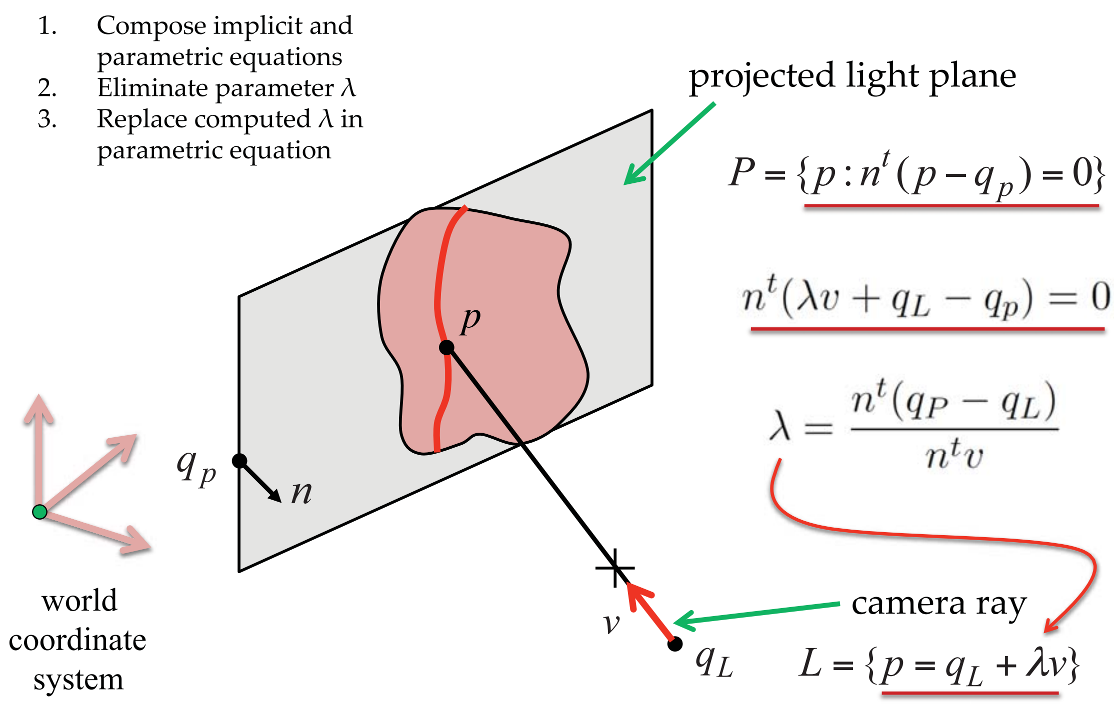
Implicit equation of the plane



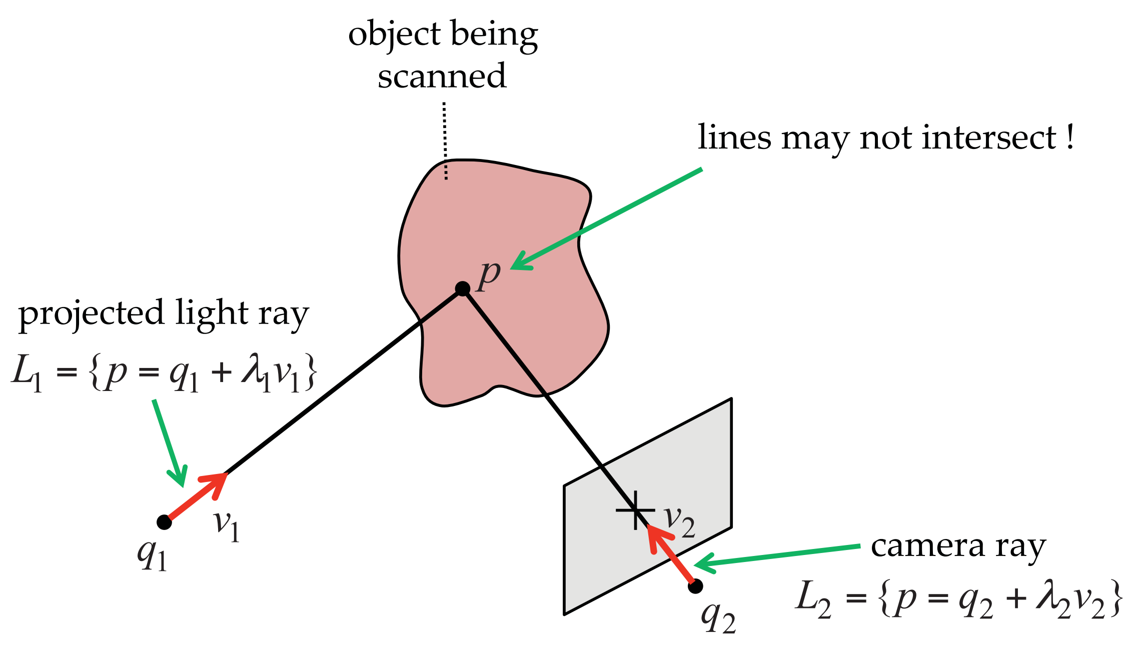
Parametric equation of the ray



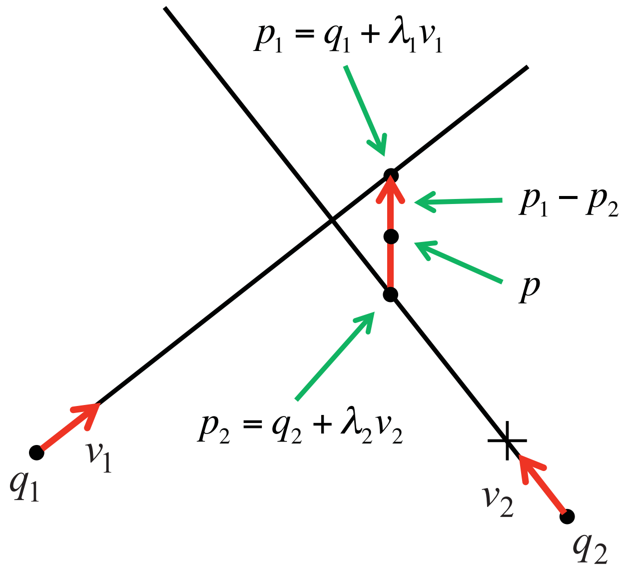
Triangulation by Line-Plane Intersection



Triangulation by Line-Line Intersection



Triangulation by Line-Line Intersection



$$L_1 = \{p_1 = q_1 + \lambda_1 v_1\}$$
$$L_2 = \{p_2 = q_2 + \lambda_2 v_2\}$$

Minimize

$$E(\lambda_1, \lambda_2) = dist(p_2 - p_1)^2$$

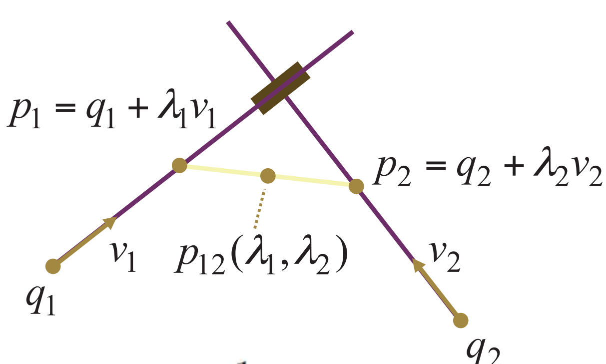
Necessary conditions

$$v_1^t(p_1 - p_2) = 0$$
$$v_2^t(p_2 - p_1) = 0$$

$$p = (p_1 + p_2)/2$$

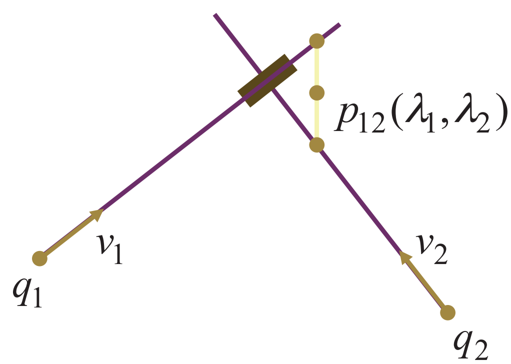
Approximate Line-Line Intersection

Midpoint of segment joining arbitrary points in the two lines



$$p_{12} = p_1 + \frac{1}{2}(p_2 - p_1)$$

Least-squares approach



Find parameters which minimize

$$\|(q_2 + \lambda_2 v_2) - (q_1 + \lambda_1 v_1)\|^2$$

Approximate Line-Line Intersection

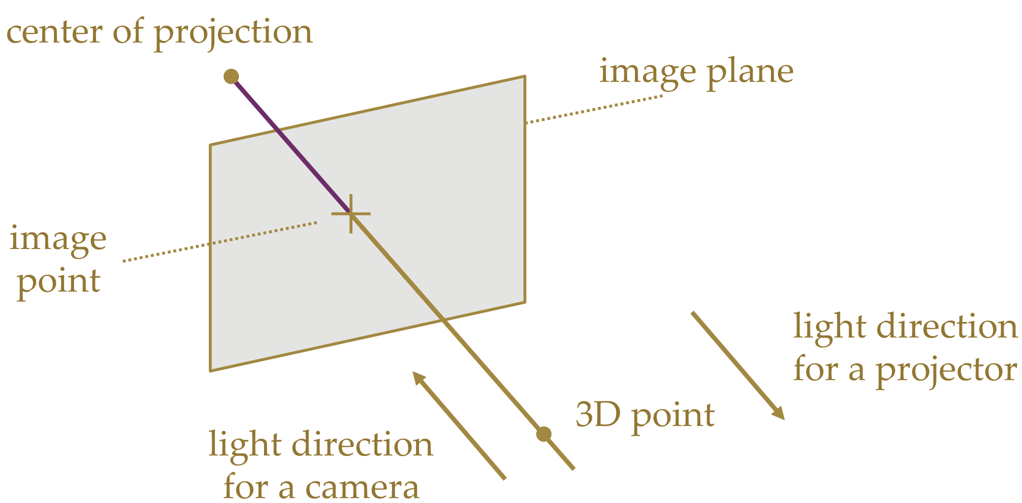
$$p_{12} = p_1 + \frac{1}{2}(p_2 - p_1)$$

$$p_1 = q_1 + \lambda_1 v_1$$

$$p_2 = q_2 + \lambda_2 v_2$$

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} \|v_1\|^2 & -v_1^t v_2 \\ -v_2^t v_1 & \|v_2\|^2 \end{pmatrix}^{-1} \begin{pmatrix} v_1^t(q_2 - q_1) \\ v_2^t(q_1 - q_2) \end{pmatrix}$$

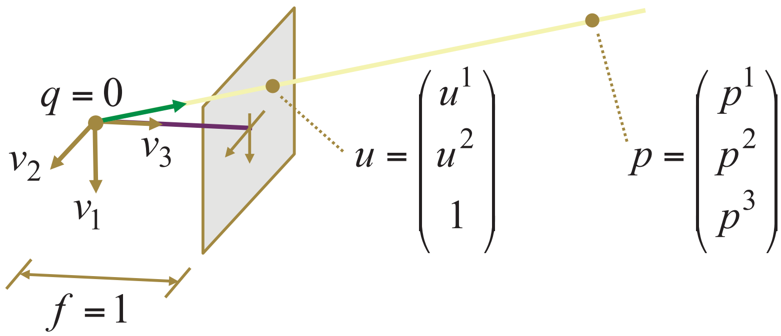
Perspective Projection (Pinhole Model)



Calibration: mapping from image points to rays

The Ideal Pinhole Camera

camera coordinate system = world coordinate system

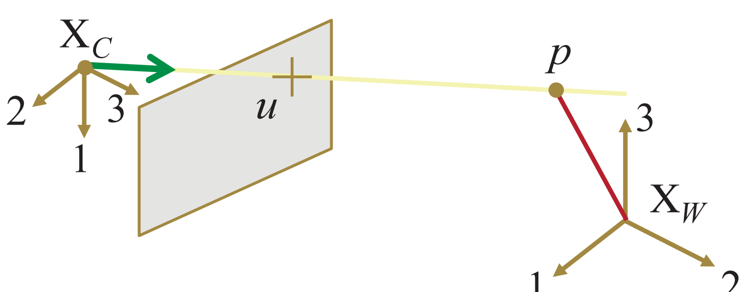


$$\begin{pmatrix} p^1 \\ p^2 \\ p^3 \end{pmatrix} = \lambda \; [v_1|v_2|q] \begin{pmatrix} u^1 \\ u^2 \\ 1 \end{pmatrix} \qquad [v_1|v_2|q] = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The General Pinhole Model

camera coordinate system

world coordinate system



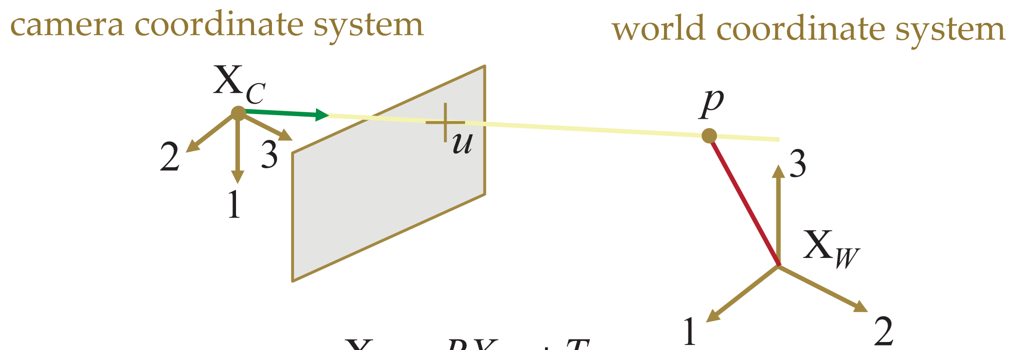
$$X_C = RX_W + T$$

extrinsic parameters

$$\lambda u = R p_W + T$$

Ideal assumptions
- Image lengths = world lengths
- Focal length = 1
- Image origin = optical center
- Image plane spanned by two basis vectors

The General Pinhole Model



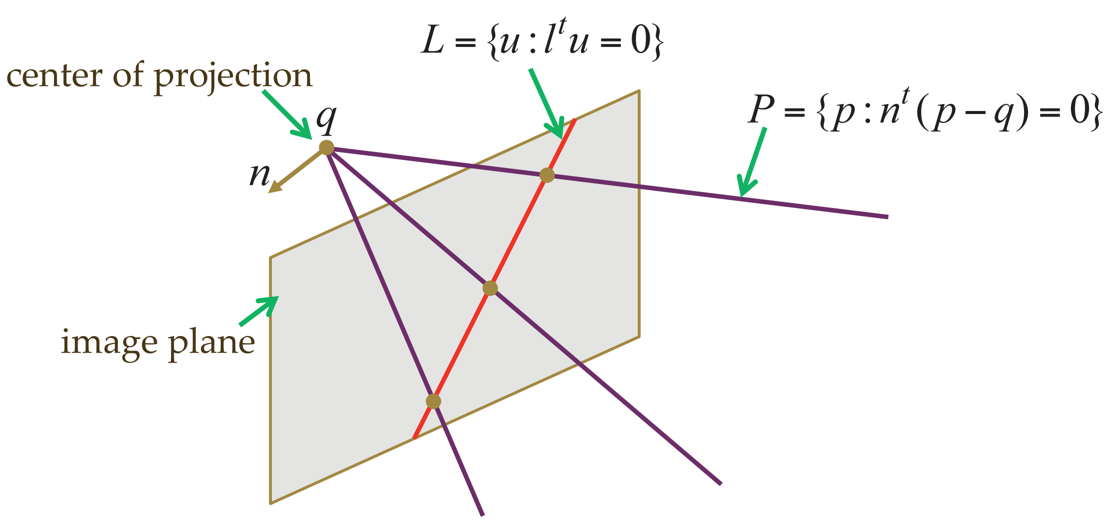
$$X_C = R X_W + T$$

intrinsic parameters

$$\lambda u = K(R p_W + T) \qquad K = \begin{pmatrix} f s_1 & f s_\theta & o^1 \\ 0 & f s_2 & o^2 \\ 0 & 0 & 1 \end{pmatrix}$$

Plane Defined by Image Line and Projection Center

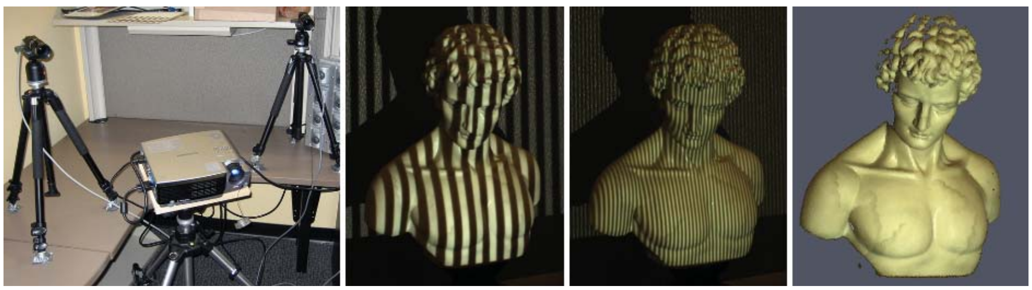
Implicit equation of line in image coordinates



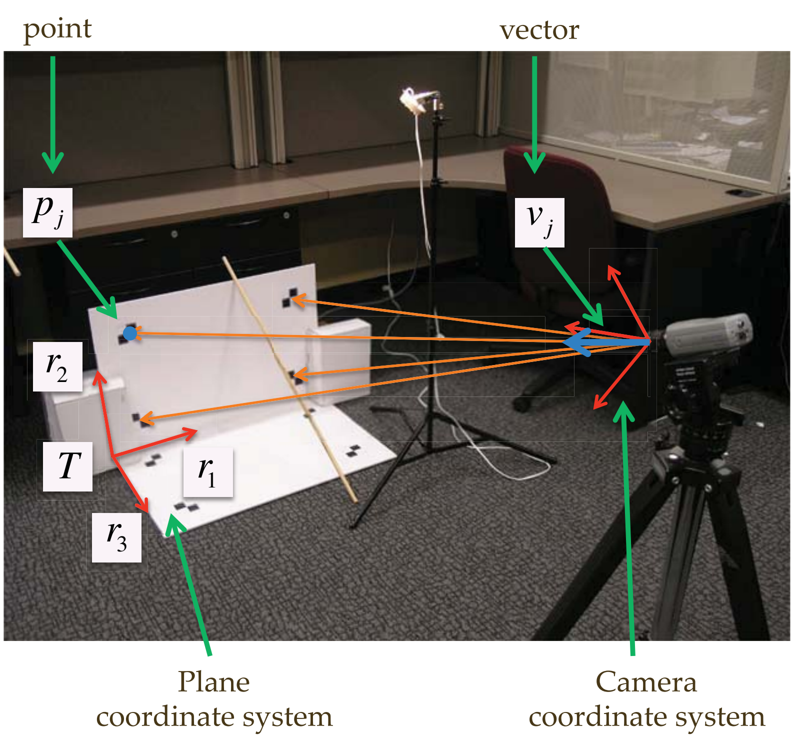
$$0 = \lambda l^t u = l^t (R p_W + T) = (R^t l)^t (p_W - (-R^t T)) .$$

Desktop 3D Photography

Structured Lighting



Estimating the equation of a plane



Given

$$\lambda_j v_j = R p_j + T$$

$$j = 1, \ldots, N$$

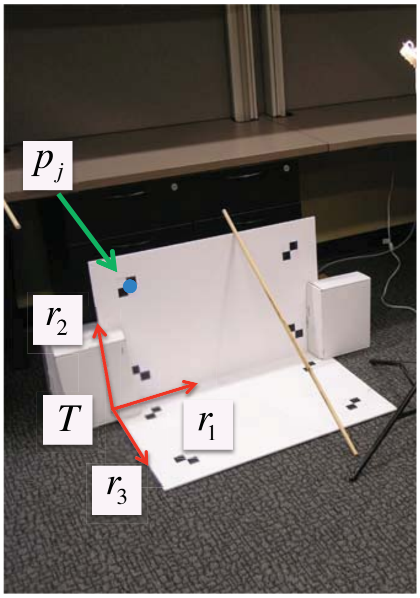
We want to estimate the rotation R and the translation T

$$R = \left[r_1 r_2 r_3 \right]$$

How to solve this problem?

What is the minimum number of points necessary to solve the problem?

Estimating the equation of a plane



$$\lambda_j v_j = R\, p_j + T$$

Camera coordinate system

Plane coordinate system

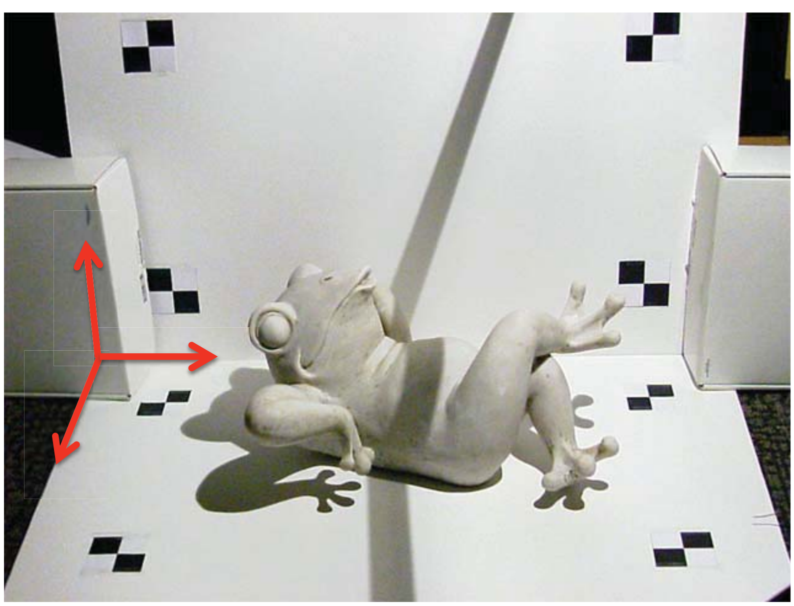
Implicit equation of plane in the camera coordinate system

$$\{q : r_3^t (q - T) = 0\}$$

Parametric equation of plane in the camera coordinate system

$$\{q = T + x\, r_1 + y\, r_2 + 0\ \ r_3\} \qquad p = [x, y, 0]^t$$

Estimating the equation of two planes



Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

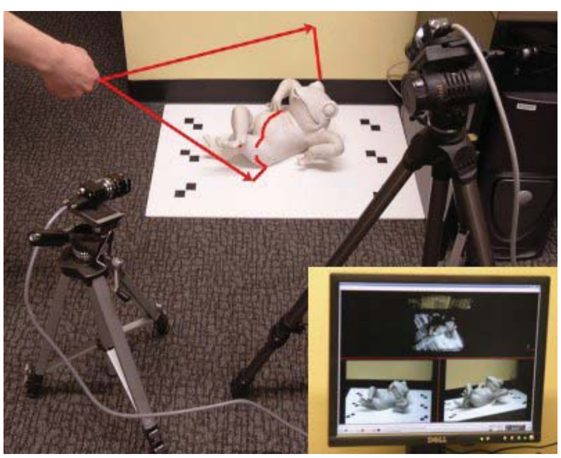
Clase 3 : Jueves Abril 4

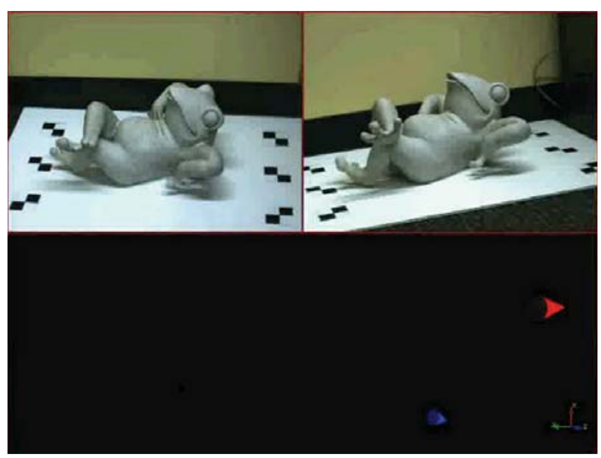
Gabriel Taubin

Brown University

BROWN

Triangulation by Laser Striping

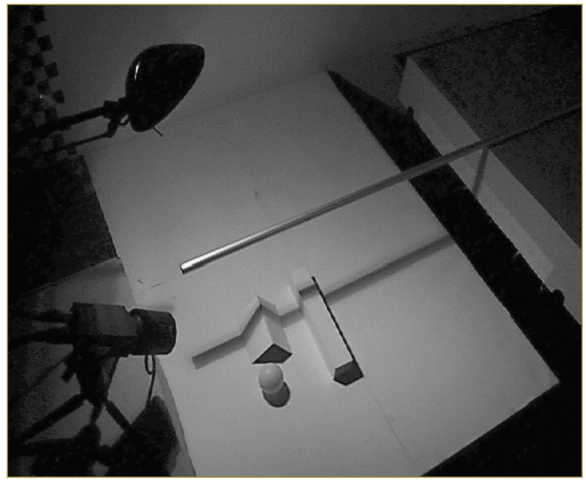




- Manually or mechanically translated laser stripe
- Per-pixel depth by ray-plane triangulation
- Requires accurate camera and laser plane calibration
- Popular solution for commercial and DIY 3D scanners

M. J. Leotta, A. Vandergon, and G. Taubin. 3D Slit Scanning With Planar Constraints. *Computer Graphics Forum*, 2008

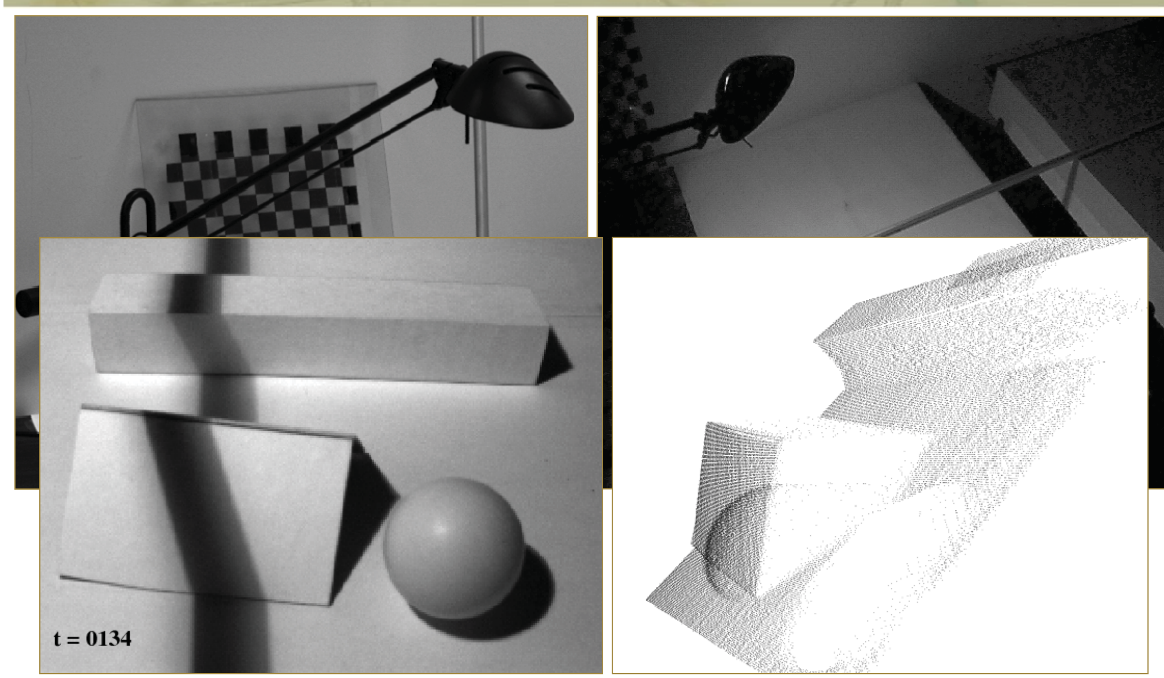
3D Photography on Your Desk: Bouguet and Perona [ICCV 1998]



- DIY scanner using only a camera, a halogen lamp, and a stick
- Per-pixel depth by ray-plane triangulation
- Requires accurate camera and shadow plane calibration

J.-Y. Bouguet and P. Perona. 3D photography on your desk.
Intl. Conf. Comp. Vision, 1998

3D Photography on Your Desk: Bouguet and Perona [ICCV 1998]



J.-Y. Bouguet and P. Perona. 3D photography on your desk.
Intl. Conf. Comp. Vision, 1998

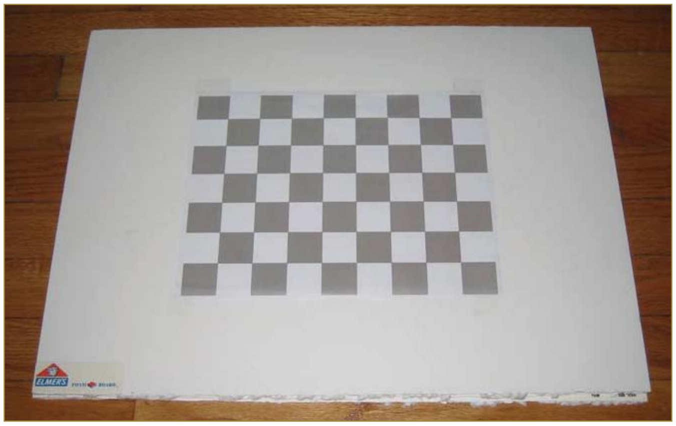
Assembling Your Own Scanner

- Parts: camera (QuickCam 9000), lamp, stick, two planar objects [~$100]
- Step 1: Build the calibration boards (include fiducials and chessboard)
- Step 2: Build the point light source (remove reflector and place in scene)
- Step 3: Arrange the camera, light source, and calibration boards

Assembling Your Own Scanner

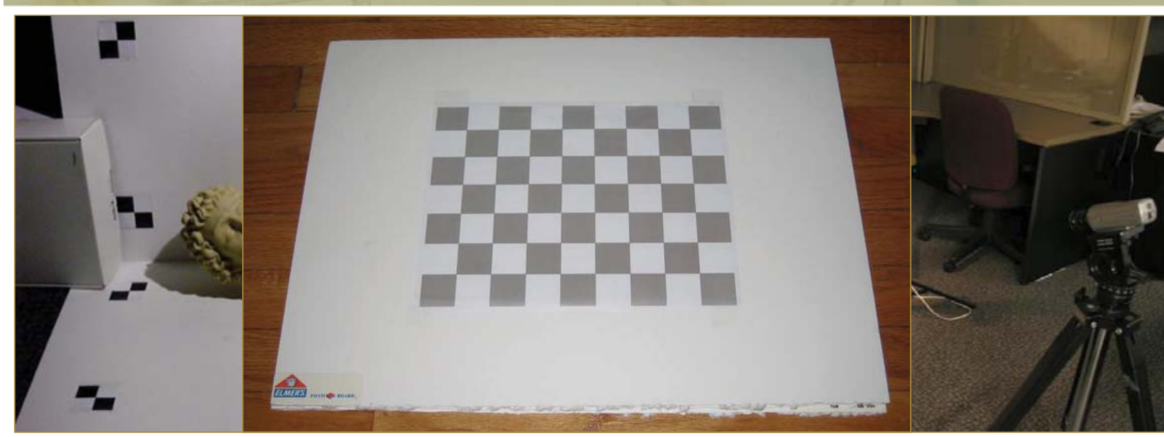
- Parts: camera (QuickCam 9000), lamp, stick, two planar objects [~$100]
- Step 1: Build the calibration boards (include fiducials and chessboard)
- Step 2: Build the point light source (remove reflector and place in scene)
- Step 3: Arrange the camera, light source, and calibration boards

Assembling Your Own Scanner



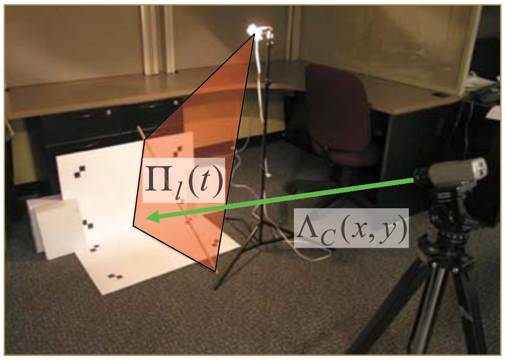
- Parts: camera (QuickCam 9000), lamp, stick, two planar objects [~$100]
- Step 1: Build the calibration boards (include fiducials and chessboard)
- Step 2: Build the point light source (remove reflector and place in scene)
- Step 3: Arrange the camera, light source, and calibration boards

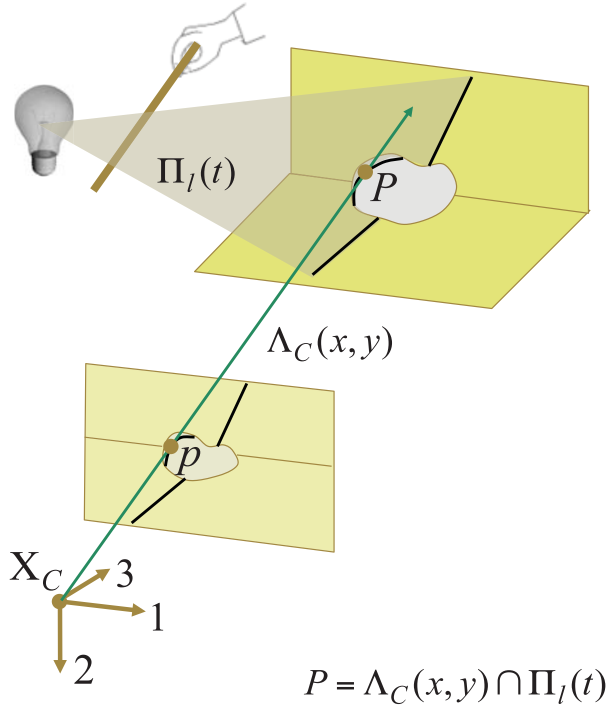
Assembling Your Own Scanner



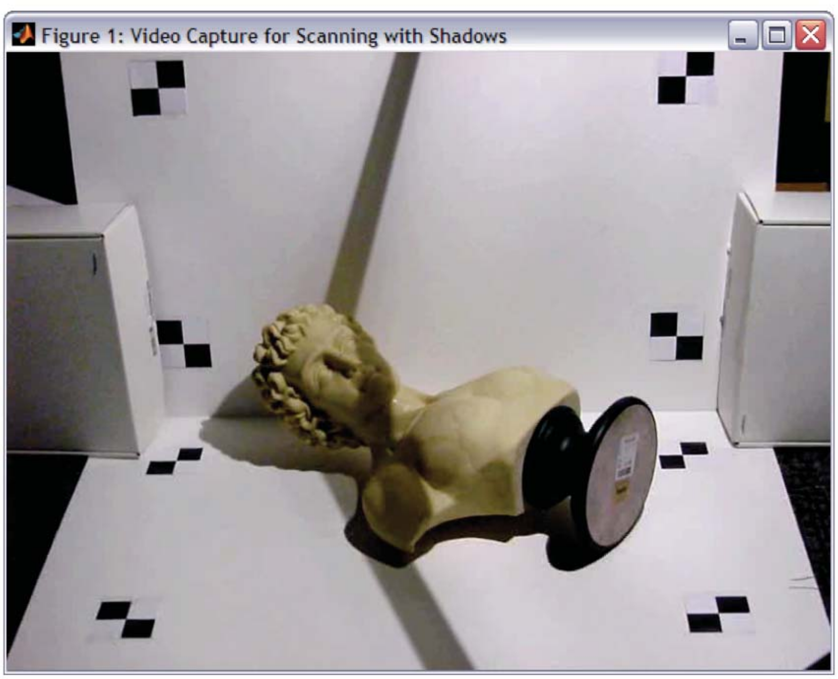
- Parts: camera (QuickCam 9000), lamp, stick, two planar objects [~$100]
- Step 1: Build the calibration boards (include fiducials and chessboard)
- Step 2: Build the point light source (remove reflector and place in scene)
- Step 3: Arrange the camera, light source, and calibration boards

Swept-Plane Reconstruction Geometry





Demo: Data Capture

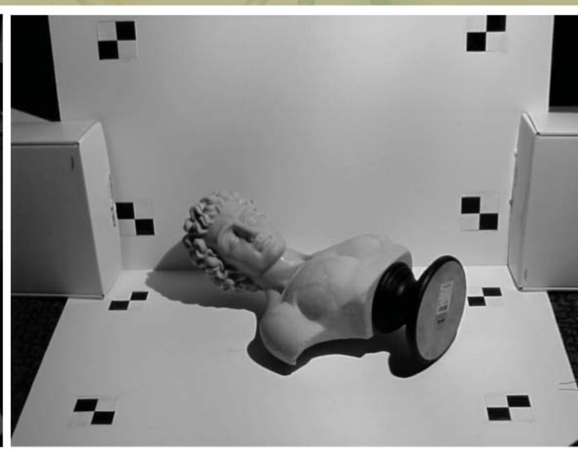


Video Processing: Assigning Per-Pixel Shadow Thresholds

$$I_{min}(x;y) = \min_t I(x;y;t)$$

- Convert from RGB to grayscale (for luminance-domain processing)
- Determine per-pixel minimum and maximum value over sequence

Video Processing: Assigning Per-Pixel Shadow Thresholds

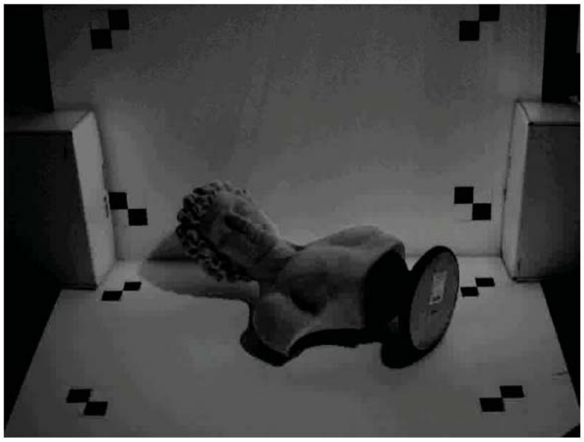


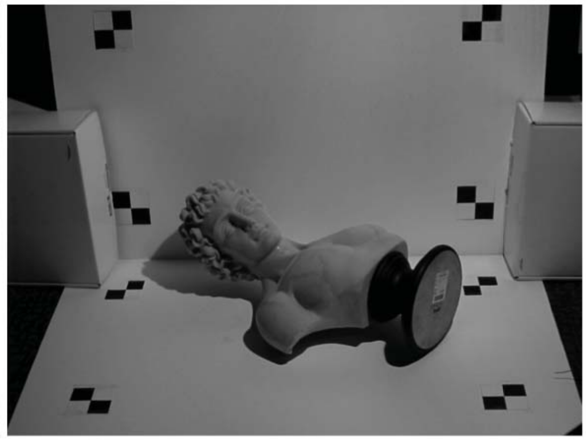
$$I_{min}(x;y) = \min_t I(x;y;t)$$

$$I_{max}(x;y) = \max_t I(x;y;t)$$

- Convert from RGB to grayscale (for luminance-domain processing)
- Determine per-pixel minimum and maximum value over sequence

Video Processing:
Assigning Per-Pixel Shadow Thresholds

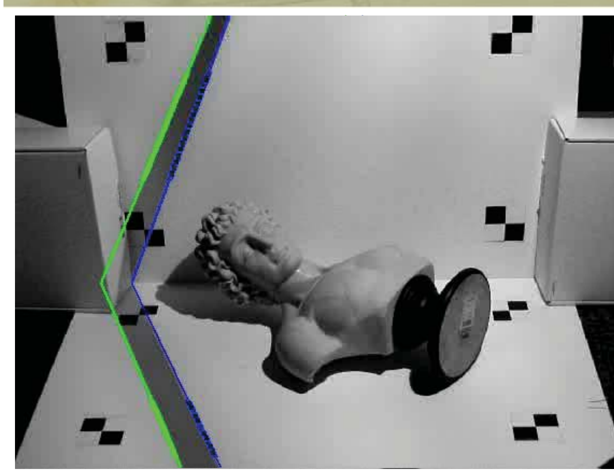


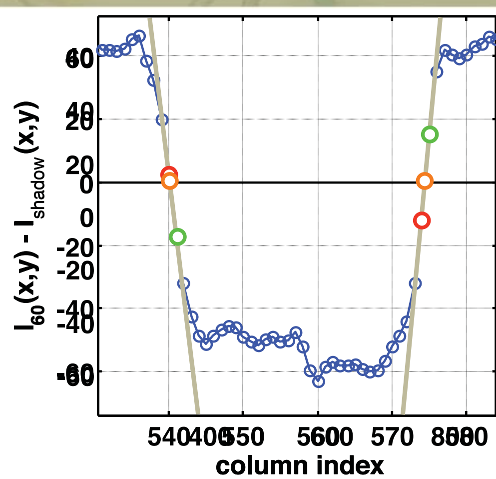


$$I_{min}(x;y) = \min_t I(x;y;t) \qquad I_{shadow}(x;y) = \frac{I_{max}(x;y)+I_{min}(x;y)}{2}$$

- Convert from RGB to grayscale (for luminance-domain processing)
- Determine per-pixel minimum and maximum value over sequence
- Evaluate per-pixel “shadow threshold” as average of min. and max.

Video Processing:
Spatial Shadow Edge Localization

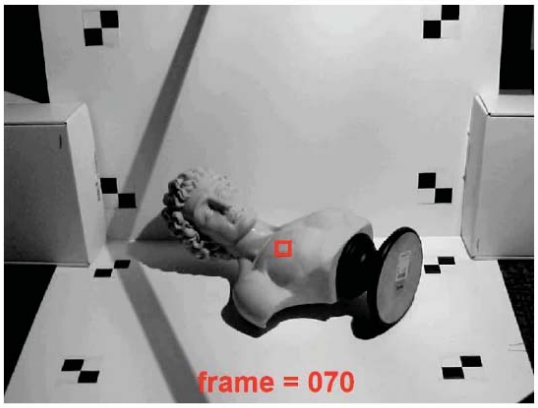


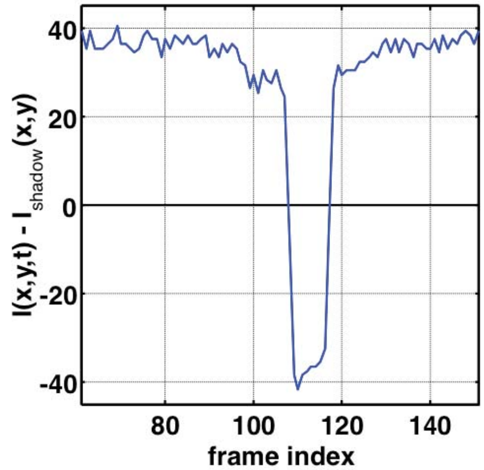


- Select region of interest on each calibration plane (occlusion-free)
- Estimate zero-crossings to find leading and trailing shadow boundaries
- Fit a line to the set of points along each shadow boundary

➔ **Result: Best-fit 2D lines for each shadow edge (in image coordinates)**

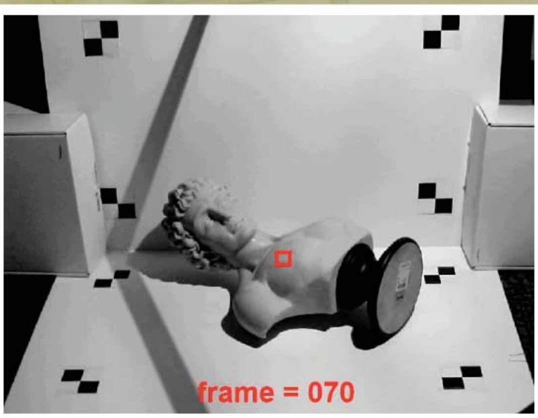
Video Processing: Temporal Shadow Edge Localization

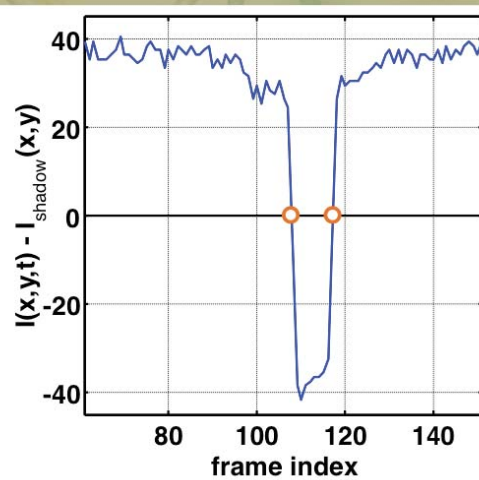




- Tabulate per-pixel temporal sequence (minus shadow threshold)

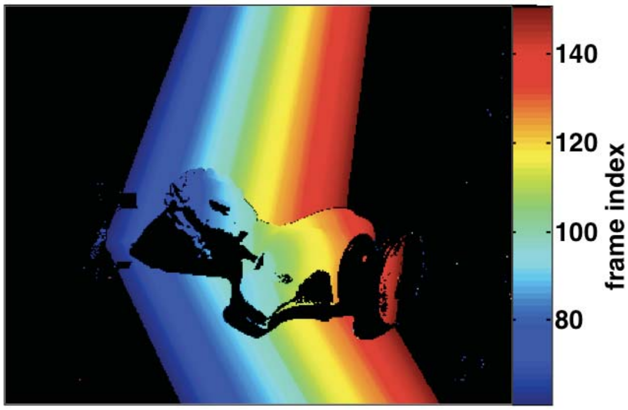
Video Processing: Temporal Shadow Edge Localization

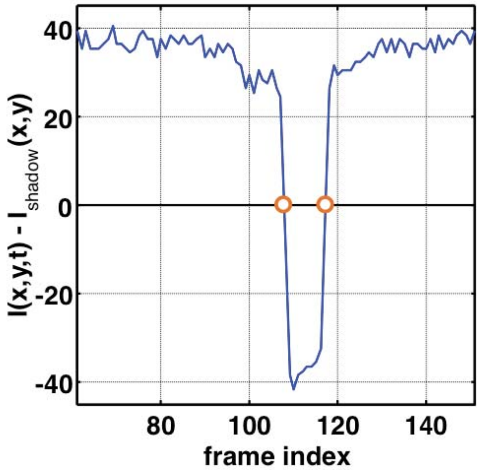




- Tabulate per-pixel temporal sequence (minus shadow threshold)
- Estimate zero-crossings to find shadow-crossing times

Video Processing:
Temporal Shadow Edge Localization

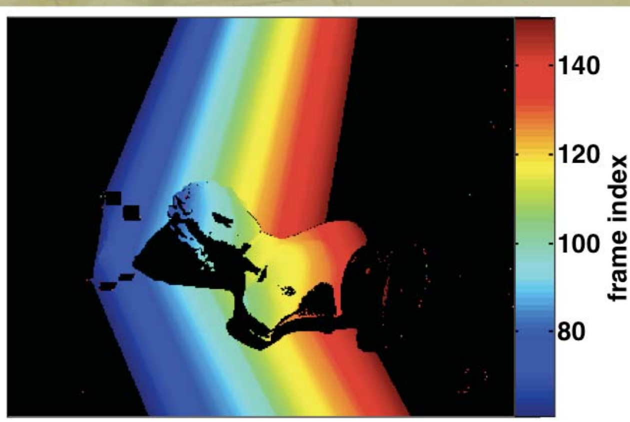


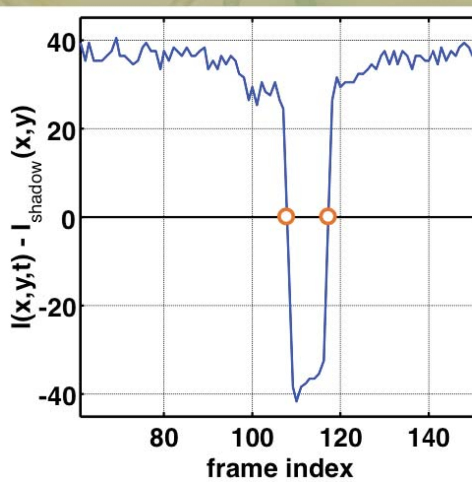


crossing frame index for leading shadow

- Tabulate per-pixel temporal sequence (minus shadow threshold)
- Estimate zero-crossings to find shadow-crossing times

Video Processing:
Temporal Shadow Edge Localization

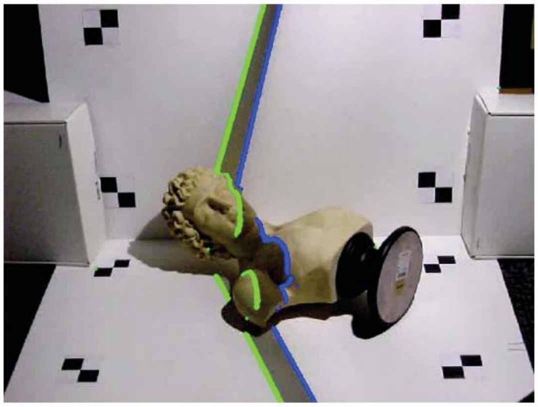


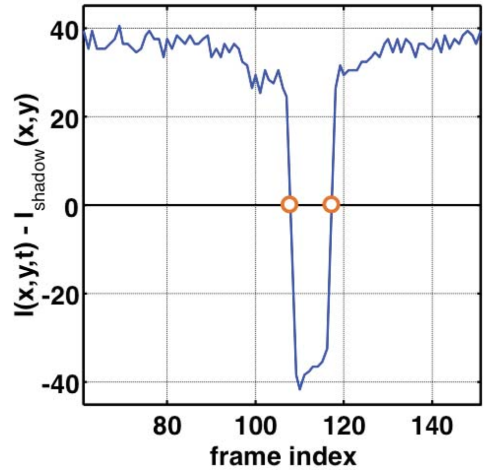


crossing frame index for trailing shadow

- Tabulate per-pixel temporal sequence (minus shadow threshold)
- Estimate zero-crossings to find shadow-crossing times

Video Processing:
Temporal Shadow Edge Localization





- Tabulate per-pixel temporal sequence (minus shadow threshold)
- Estimate zero-crossings to find shadow-crossing times

➔ **Result: Use shadow-crossing time to lookup corresponding 3D plane**

Course Schedule

- Introduction
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes
- ➢ ***Camera and Swept-Plane Light Source Calibration***
- Reconstruction and Visualization using Point Clouds

Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

Clase 4 : Viernes Abril 5

Gabriel Taubin

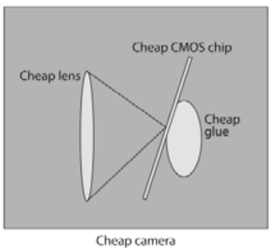
Brown University

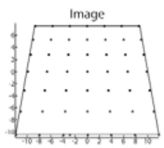
BROWN

Course Schedule

- Introduction
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes
- ***Camera and Swept-Plane Light Source Calibration***
- Reconstruction and Visualization using Point Clouds

Modeling Lens Distortion





Without lens distortion distortion

$$\begin{bmatrix} x_p \\ y_p \end{bmatrix} = \begin{bmatrix} f_x X^W / Z^W + c_x \\ f_y X^W / Z^W + c_y \end{bmatrix}$$

G. Bradski and A. Kaehler. *Learning OpenCV*. O'Reilly Media, 2008

Modeling Lens Distortion

Radial distortion

$$x_{\text{corrected}} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$y_{\text{corrected}} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

G. Bradski and A. Kaehler. *Learning OpenCV*. O'Reilly Media, 2008

Modeling Lens Distortion

Tangencial distortion

$$x_{\text{corrected}} = x + [2p_1 y + p_2(r^2 + 2x^2)]$$

$$y_{\text{corrected}} = y + [p_1(r^2 + 2y^2) + 2p_2 x]$$

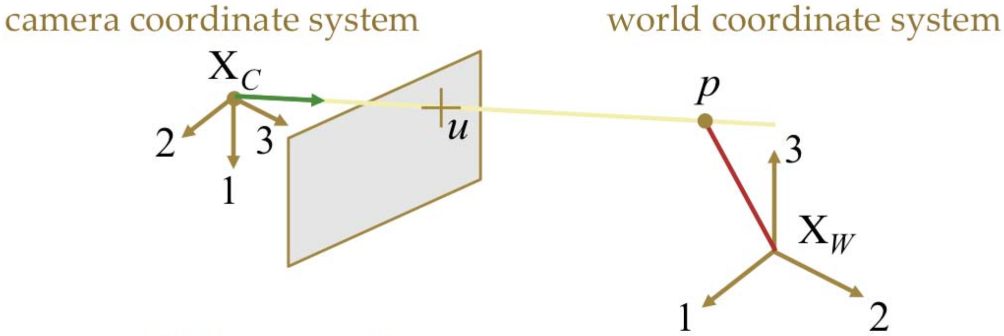
G. Bradski and A. Kaehler. *Learning OpenCV*. O' Reilly Media, 2008

Modeling Lens Distortion

$$\begin{bmatrix} x_p \\ y_p \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \begin{bmatrix} x_d \\ y_d \end{bmatrix} + \begin{bmatrix} 2p_1 x_d y_d + p_2(r^2 + 2x_d^2) \\ p_1(r^2 + 2y_d^2) + 2p_2 x_d y_d \end{bmatrix}$$

G. Bradski and A. Kaehler. *Learning OpenCV*. O' Reilly Media, 2008

Intrinsic Camera Calibration



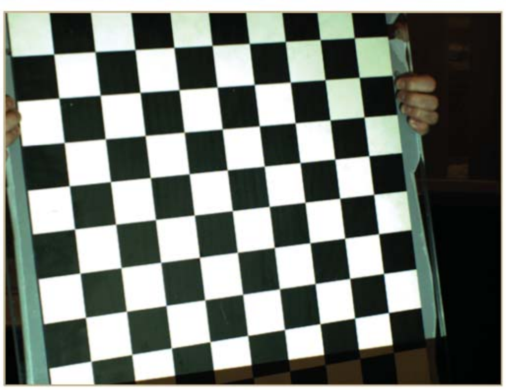
$$\lambda u = K(R p_W + T)$$

$$X_C = R X_W + T$$

intrinsic parameters **extrinsic parameters**

- How to estimate intrinsic parameters and distortion model?
(unknowns: focal length, skew, scale, principal point, and distortion coeffs.)

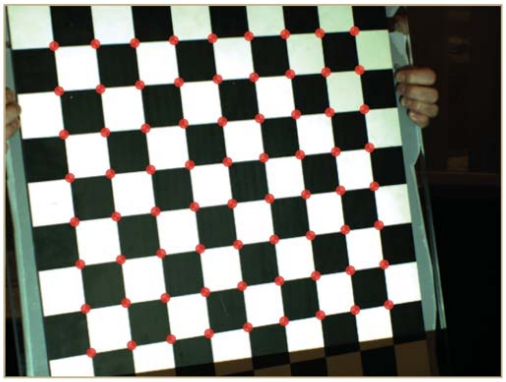
Intrinsic Camera Calibration



Camera Calibration Input

- How to estimate intrinsic parameters and distortion model?
(unknowns: focal length, skew, scale, principal point, and distortion coeffs.)
- Popular solution: Observe a known calibration object (Zhang [2000])

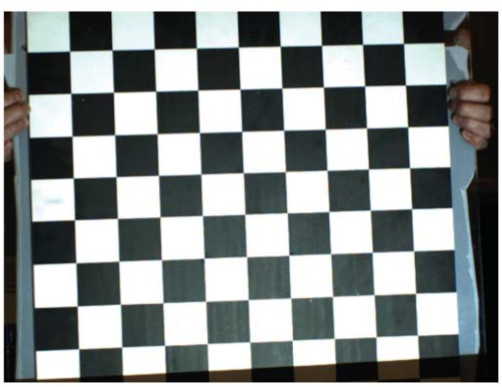
Intrinsic Camera Calibration



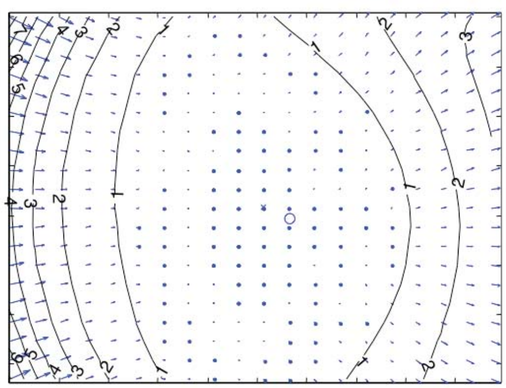
Camera Calibration Input

- How to estimate intrinsic parameters and distortion model?
  (unknowns: focal length, skew, scale, principal point, and distortion coeffs.)
- Popular solution: Observe a known calibration object (Zhang [2000])
- Each 2D chessboard corner yields two constraints on the 6-11 unknowns

Intrinsic Camera Calibration



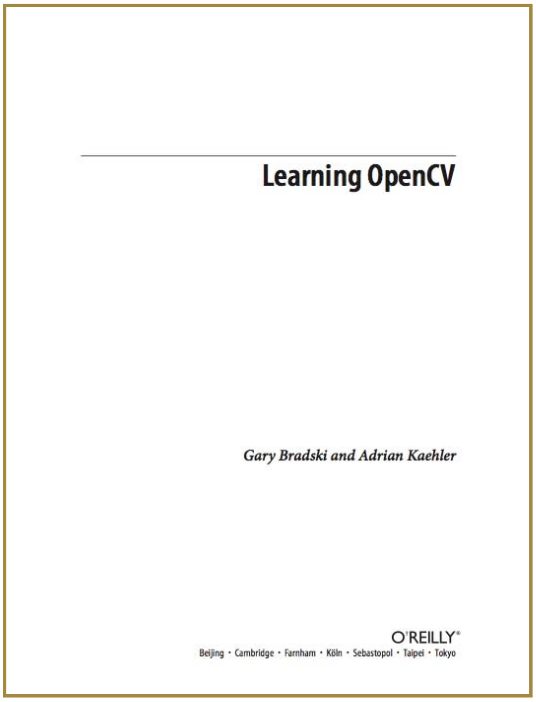
Camera Calibration Input



Estimated Camera Lens Distortion Map

- How to estimate intrinsic parameters and distortion model?
  (unknowns: focal length, skew, scale, principal point, and distortion coeffs.)
- Popular solution: Observe a known calibration object (Zhang [2000])
- Each 2D chessboard corner yields two constraints on the 6-11 unknowns
- But, must also find 6 extrinsic parameters per image (rotation/translation)
- ➔ **Result: Two or more images of a chessboard are sufficient**

OpenCV

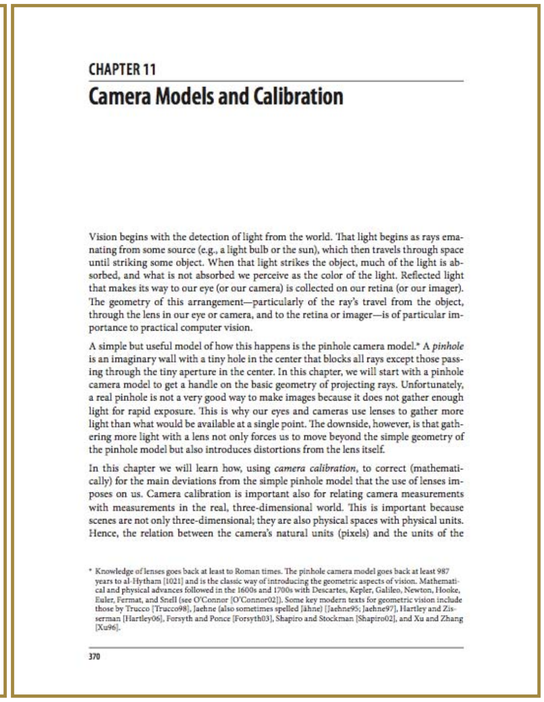


Learning OpenCV

Gary Bradski and Adrian Kaehler

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo



CHAPTER 11

Camera Models and Calibration

Vision begins with the detection of light from the world. That light begins as rays emanating from some source (e.g., a light bulb or the sun), which then travels through space until striking some object. When that light strikes the object, much of the light is absorbed, and what is not absorbed we perceive as the color of the light. Reflected light that makes its way to our eye (or our camera) is collected on our retina (or our imager). The geometry of this arrangement—particularly of the ray's travel from the object, through the lens in our eye or camera, and to the retina or imager—is of particular importance to practical computer vision.

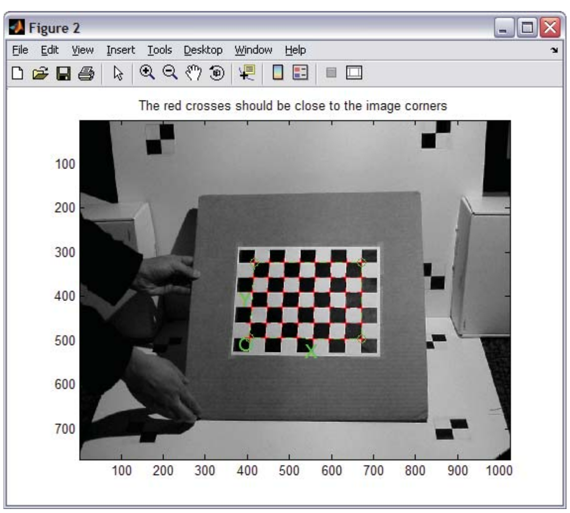
A simple but useful model of how this happens is the pinhole camera model.* A *pinhole* is an imaginary wall with a tiny hole in the center that blocks all rays except those passing through the tiny aperture in the center. In this chapter, we will start with a pinhole camera model to get a handle on the basic geometry of projecting rays. Unfortunately, a real pinhole is not a very good way to make images because it does not gather enough light for rapid exposure. This is why our eyes and cameras use lenses to gather more light than what would be available at a single point. The downside, however, is that gathering more light with a lens not only forces us to move beyond the simple geometry of the pinhole model but also introduces distortions from the lens itself.

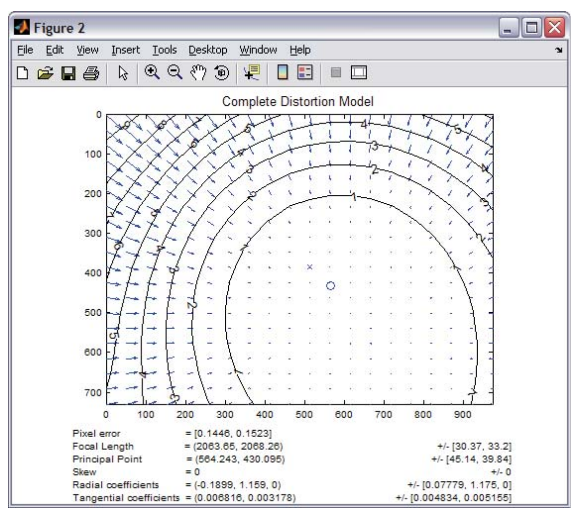
In this chapter we will learn how, using *camera calibration*, to correct (mathematically) for the main deviations from the simple pinhole model that the use of lenses imposes on us. Camera calibration is important also for relating camera measurements with measurements in the real, three-dimensional world. This is important because scenes are not only three-dimensional; they are also physical spaces with physical units. Hence, the relation between the camera's natural units (pixels) and the units of the

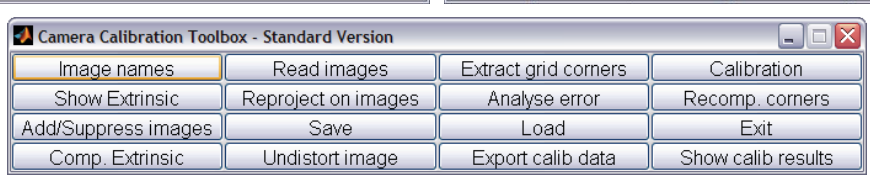
* Knowledge of lenses goes back at least to Roman times. The pinhole camera model goes back at least 987 years to al-Hytham [1021] and is the classic way of introducing the geometric aspects of vision. Mathematical and physical advances followed in the 1600s and 1700s with Descartes, Kepler, Galileo, Newton, Hooke, Euler, Fermat, and Snell (see O'Connor [O'Connor02]). Some key modern texts for geometric vision include those by Trucco [Trucco98], Jaehne (also sometimes spelled Jähne) [Jaehne95; Jaehne97], Hartley and Zisserman [Hartley06], Forsyth and Ponce [Forsyth03], Shapiro and Stockman [Shapiro02], and Xu and Zhang [Xu96].

370

Demo: Camera Calibration in Matlab

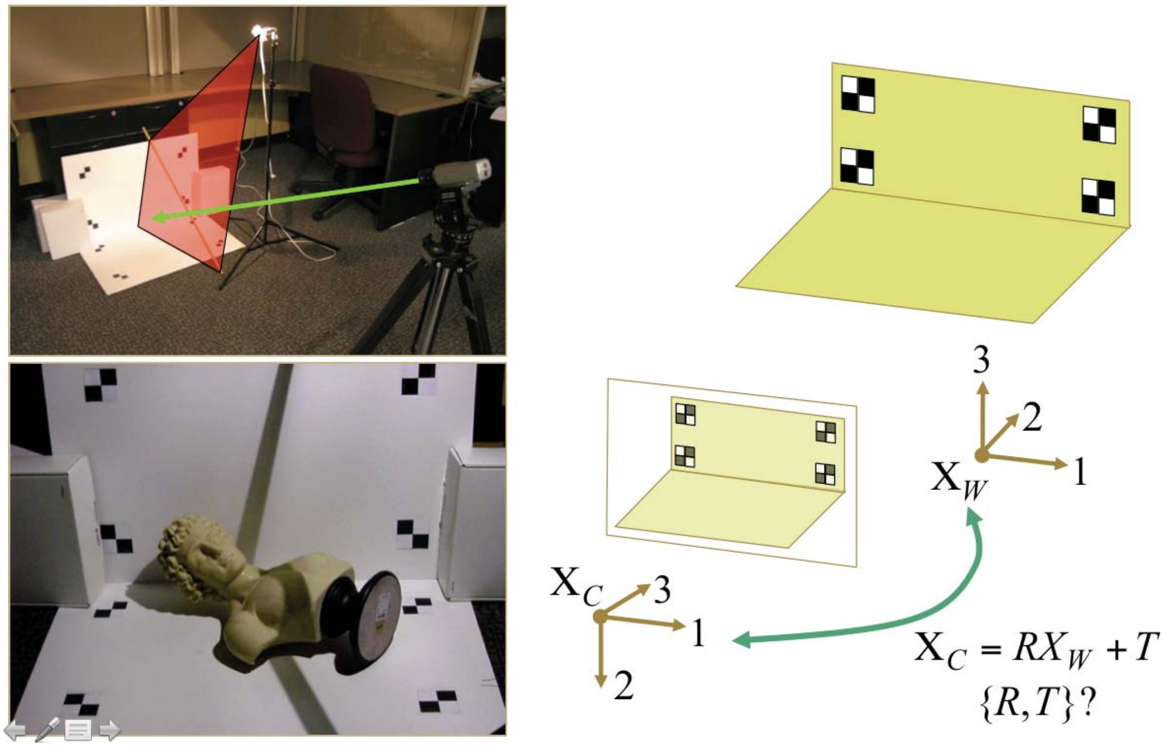




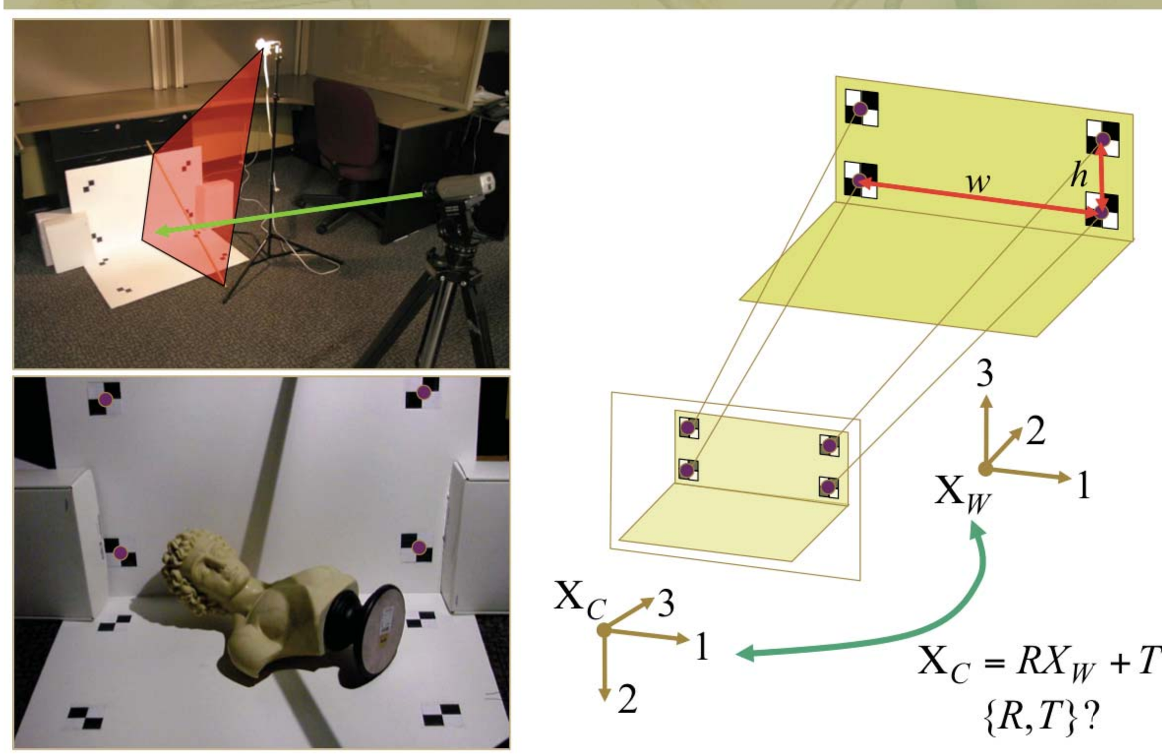


J.-Y. Bouguet. Camera Calibration Toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/

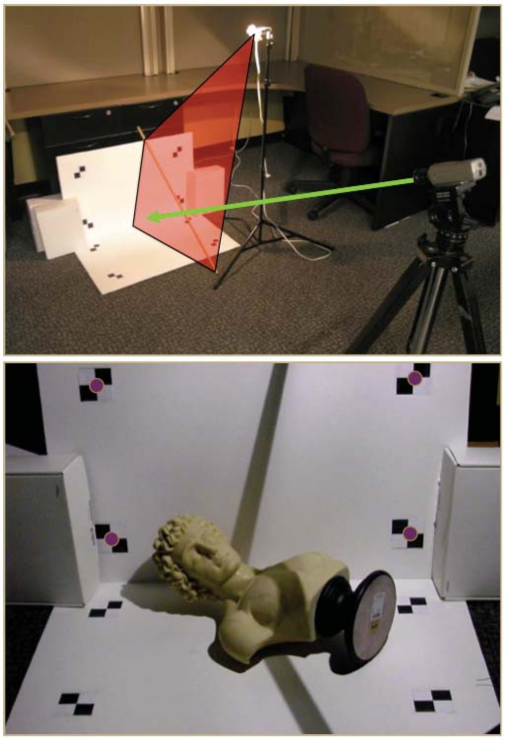
Extrinsic Camera Calibration

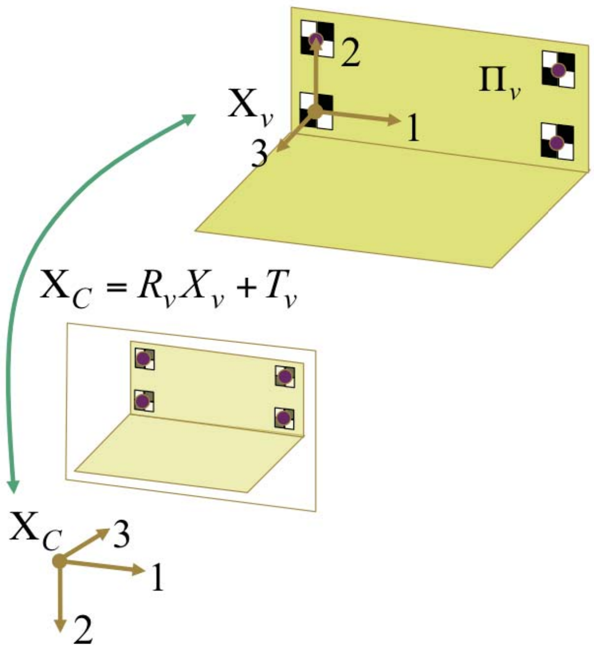


Extrinsic Camera Calibration

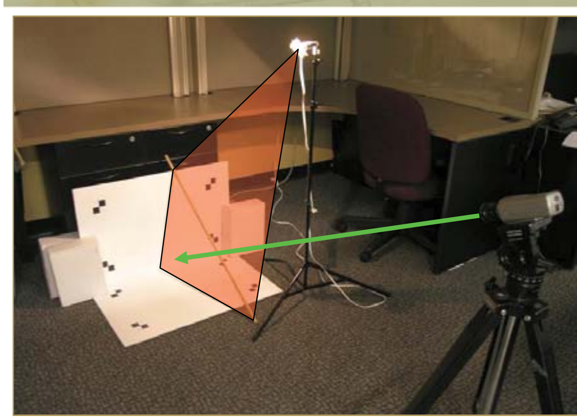


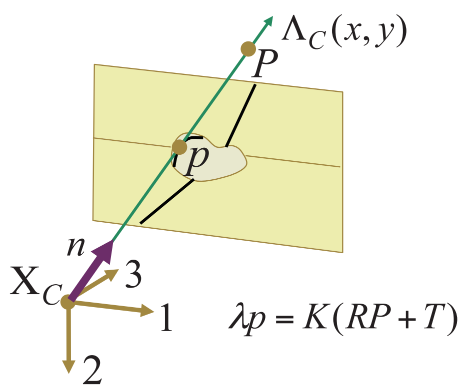
Extrinsic Camera Calibration





Demo: Mapping Pixels to Optical Rays

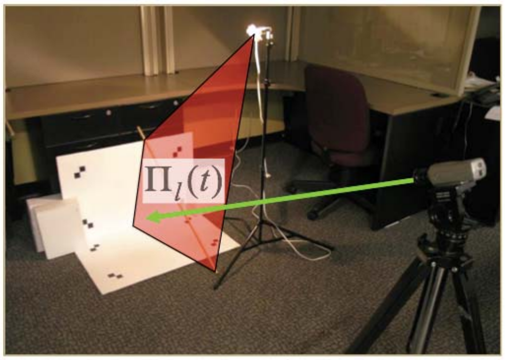


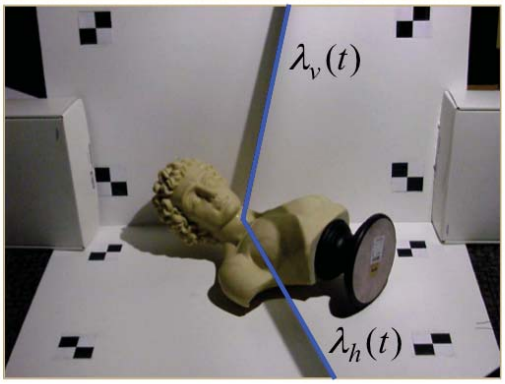


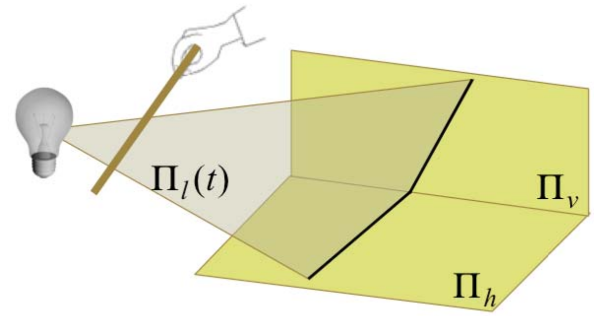
- How to map an image pixel to an optical ray?
- Solution: Invert the *calibrated* camera projection model
- But, also requires inversion of distortion model (which is non-linear)
- Mapping implemented in Camera Calibration Toolbox with **normalize.m**

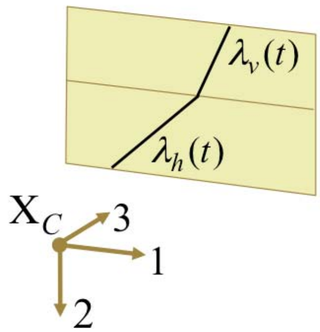
➔ **Result: After calibration, pixels can be converted to optical rays**

Shadow Plane Calibration

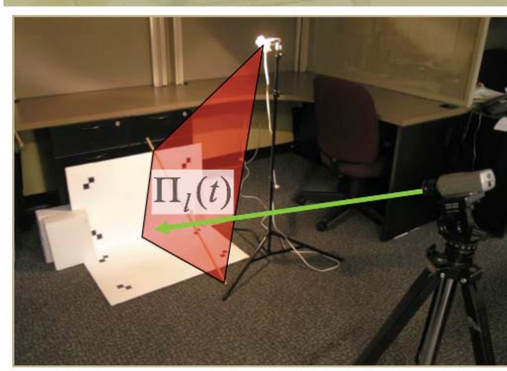


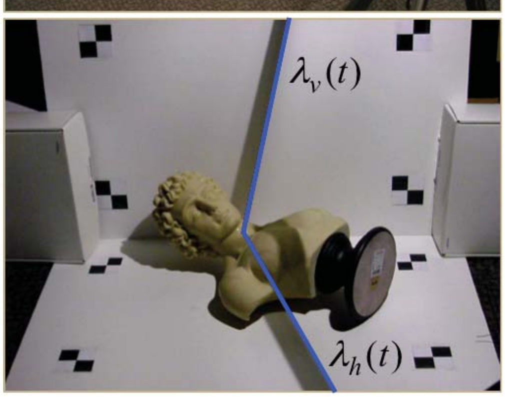


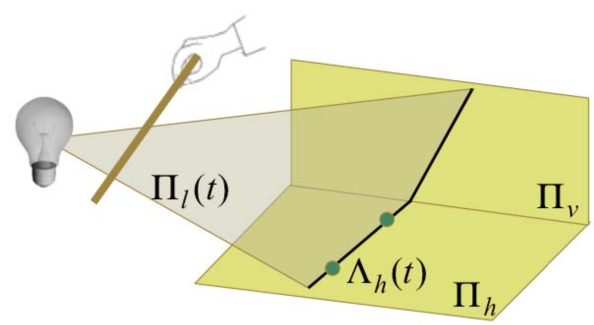


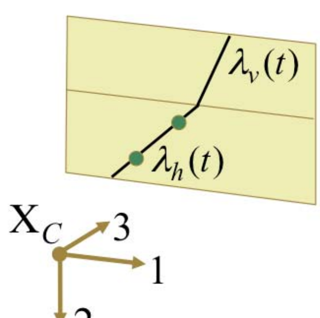


Shadow Plane Calibration

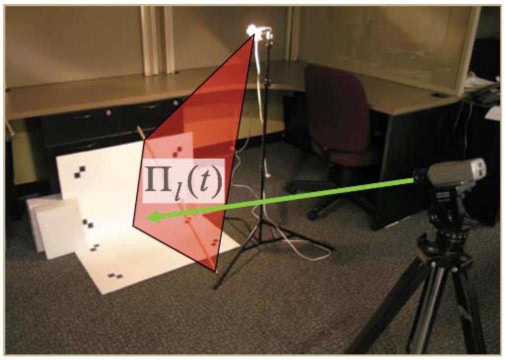


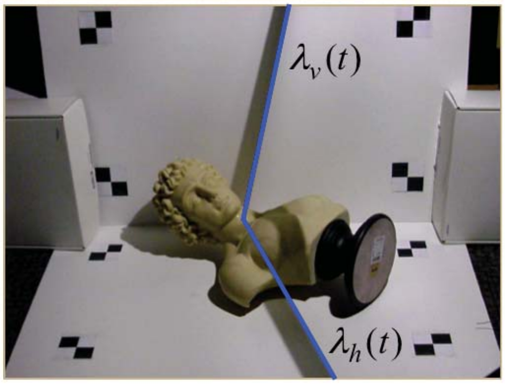


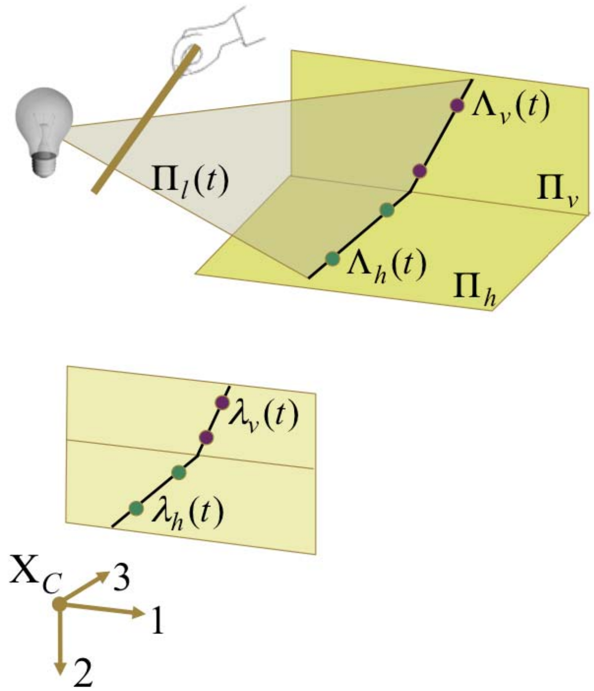




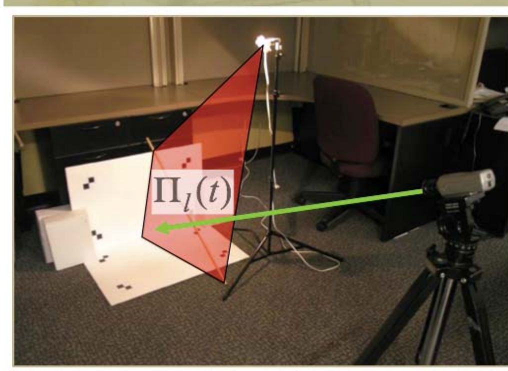
Shadow Plane Calibration

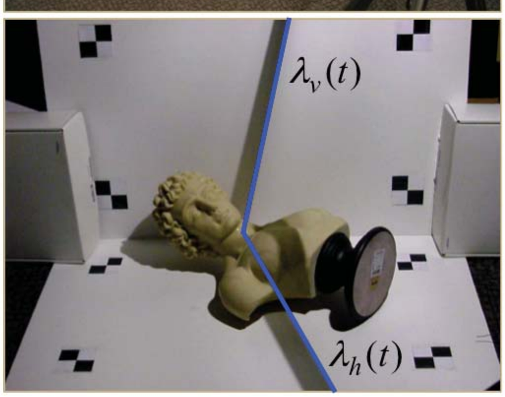


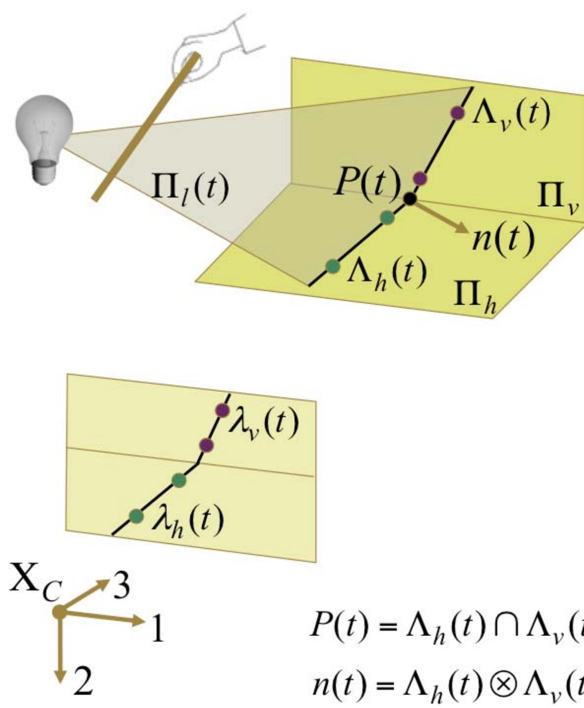




Shadow Plane Calibration



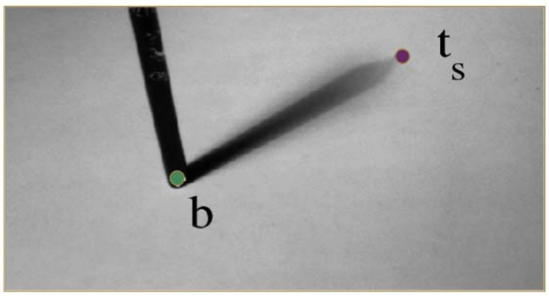


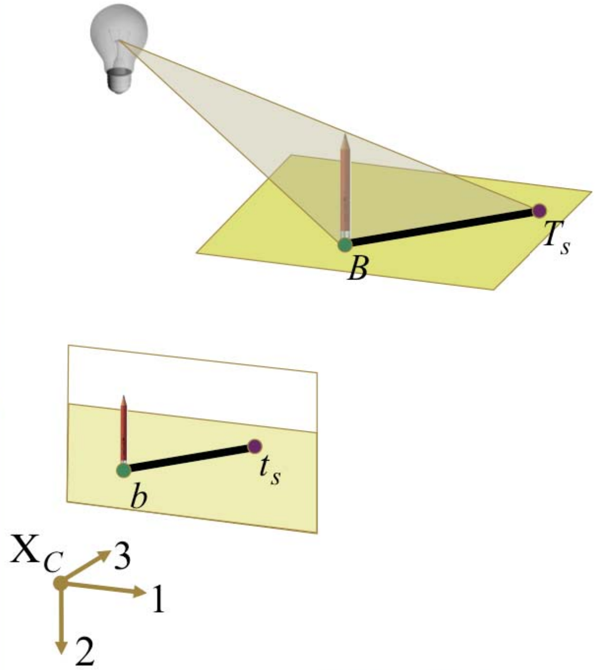


$$P(t) = \Lambda_h(t) \cap \Lambda_v(t)$$

$$n(t) = \Lambda_h(t) \otimes \Lambda_v(t)$$

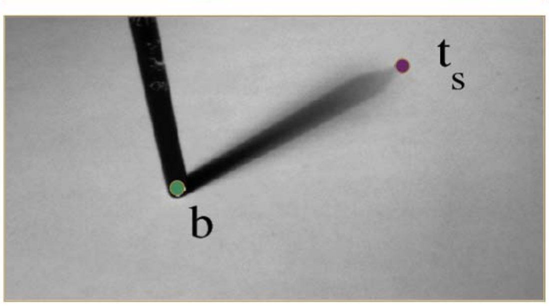
Alternatives for Shadow Plane Calibration

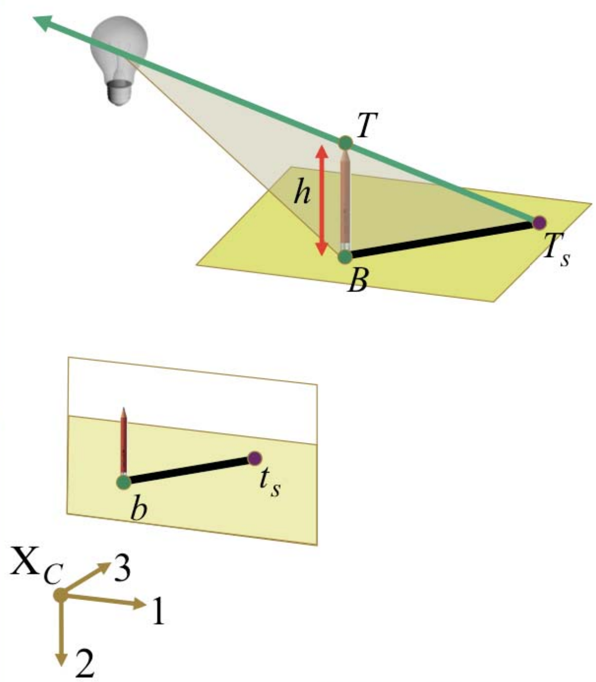




J.-Y. Bouguet and P. Perona. 3D photography on your desk. *Intl. Conf. Comp. Vision*, 1998

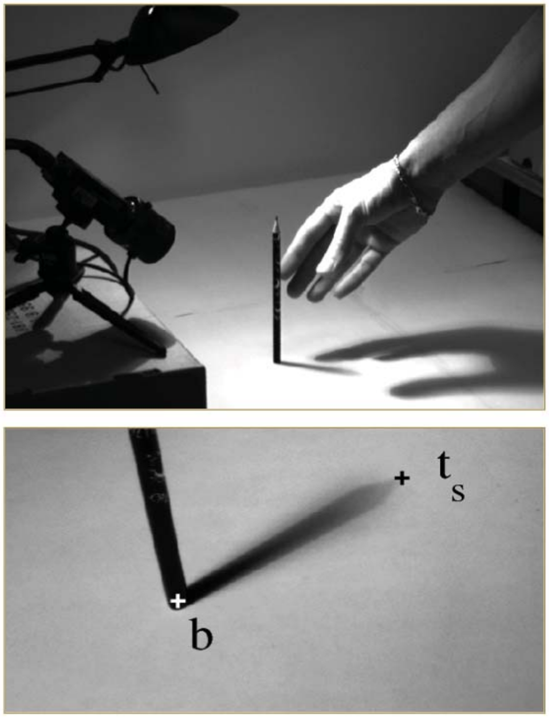
Alternatives for Shadow Plane Calibration

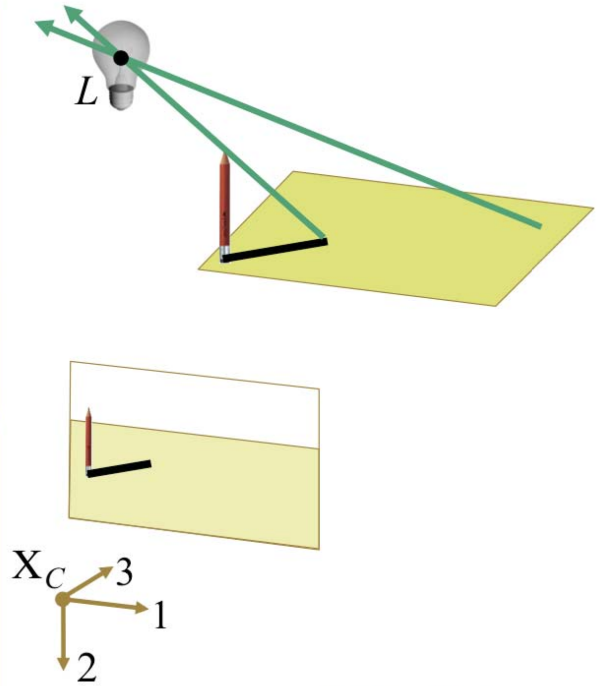




J.-Y. Bouguet and P. Perona. 3D photography on your desk. *Intl. Conf. Comp. Vision*, 1998

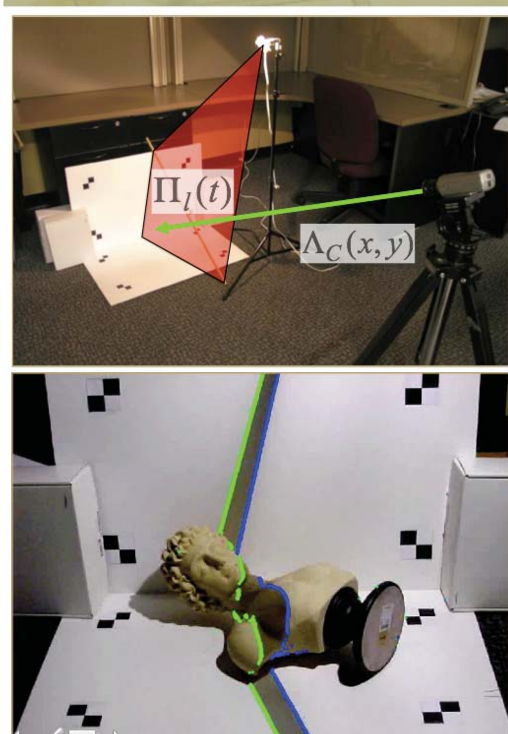
Alternatives for Shadow Plane Calibration

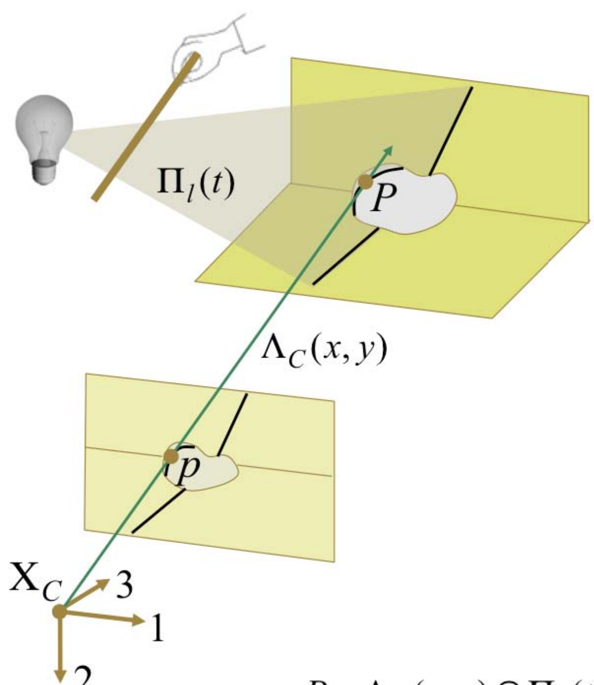




J.-Y. Bouguet and P. Perona. 3D photography on your desk.
Intl. Conf. Comp. Vision, 1998

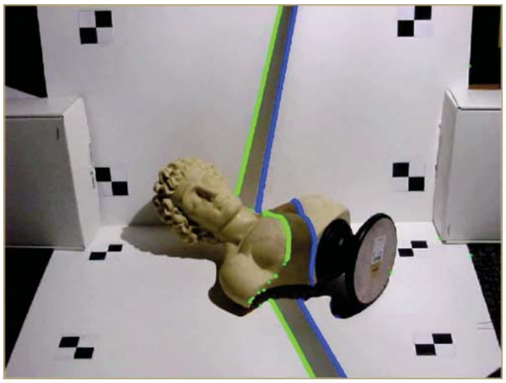
Point Cloud Reconstruction

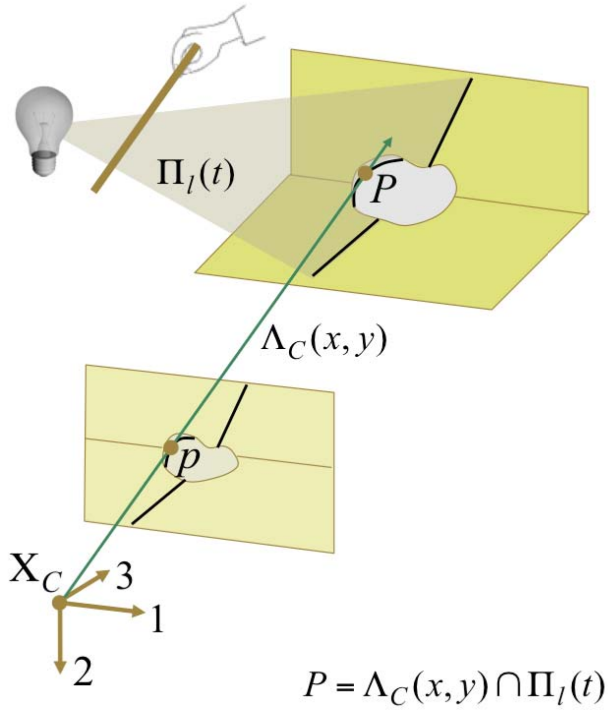




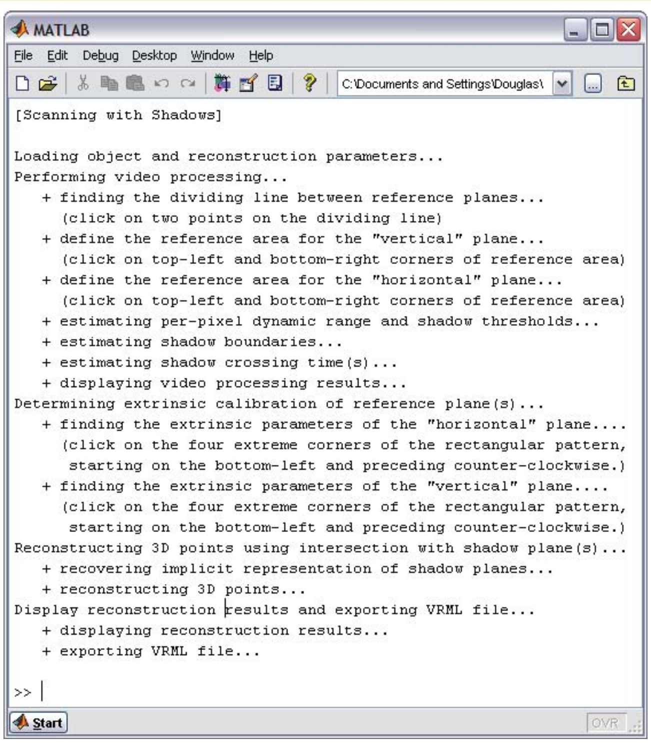
$$P = \Lambda_C(x,y) \cap \Pi_l(t)$$

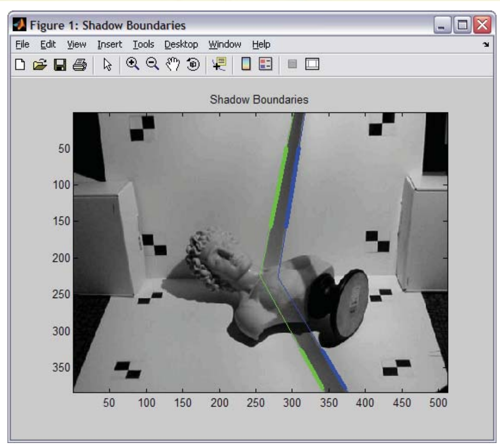
Point Cloud Reconstruction





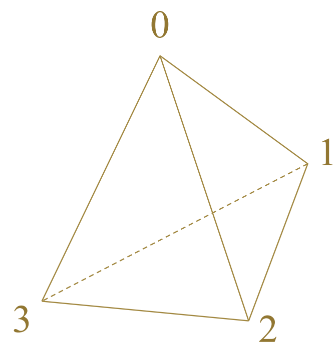
Demo: Putting it All Together



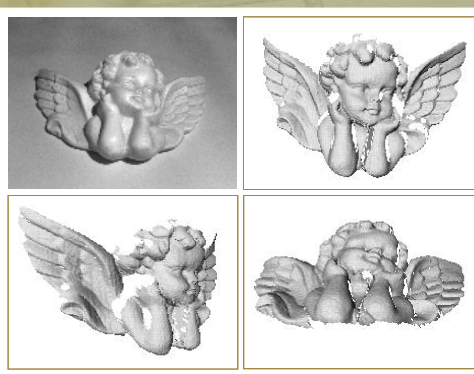


VRML File Format

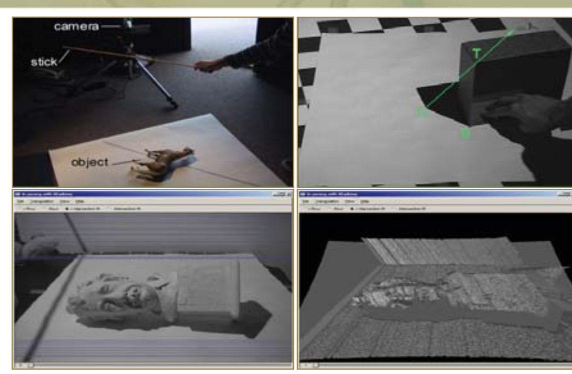
```
#VRML V2.0 utf8
Shape {
  geometry IndexedFaceSet {
    coord Coordinate {
      point [
         1.633 -0.943 -0.667
         0.000  0.000  2.000
        -1.633 -0.943 -0.667
         0.000  1.886 -0.667
      ]
    }
    coordIndex [
      0 1 2 -1 3 1 0 -1 2 1 3 -1 2 3 0 -1
    ]
  }
}
```



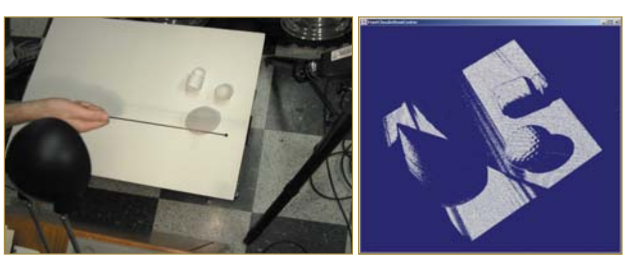
Additional Reconstruction Examples



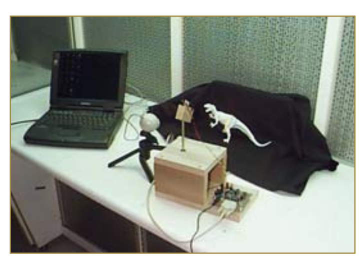
J.-Y. Bouguet and P. Perona. 3D photography on your desk. *Intl. Conf. Comp. Vision*, 1998



J. Kim and J. Wu. Scanning with Shadows. CSE 558 Project Report (U. Washington), 2001



P. Blaer, N. Hasan, C. Tripp, and L. Volchok. 3D Desktop Photography by Eclipse. Project Report (Columbia), 2001



J. Kubicky. Home-Brew 3-D Photography. EE 149 Project Report (Caltech), 1998

Course Schedule

- Introduction
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes
- Camera and Swept-Plane Light Source Calibration
- ***Reconstruction and Visualization using Point Clouds***

Introducción a la Fotografia 3D
UBA/FCEN Marzo 27 – Abril 12 2013
Clase 5 : Lunes Abril 8

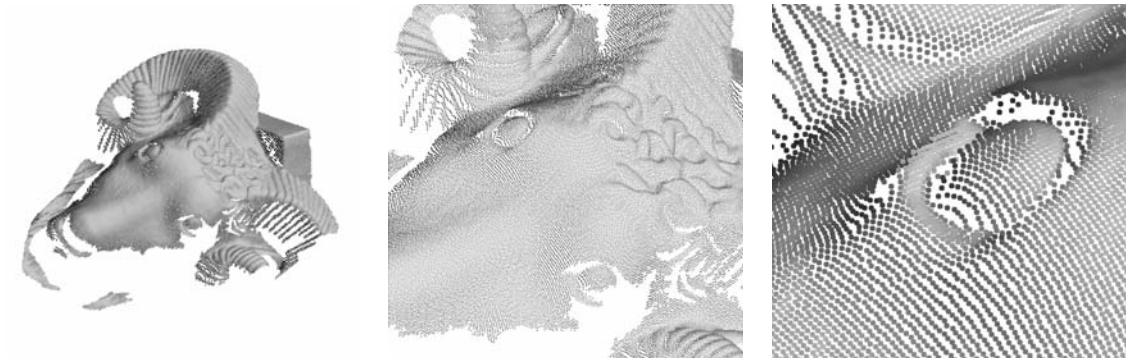
Gabriel Taubin
Brown University

BROWN

Course Schedule

- Introduction
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes
- Camera and Swept-Plane Light Source Calibration
- ***Reconstruction and Visualization using Point Clouds***
- Combining Point Clouds Recovered from Multiple Views

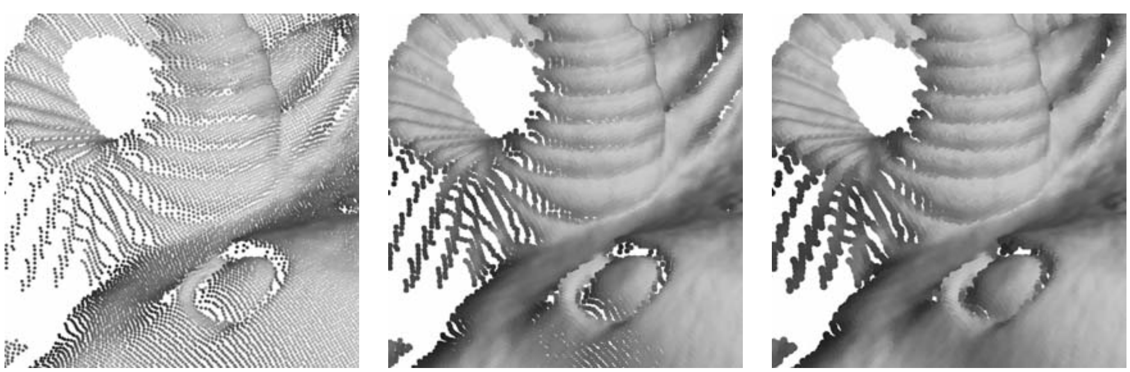
Visualizing Point Clouds:
Point-based Rendering via Splatting



- Swept-plane scanner produces a *colored point cloud*: a set of 3D points
- Problem: how to render a point cloud to make it look like as a continuous surface?
- Splatting: render points as overlapping colored disks
- If normal vectors are measured as well, render points as shaded ellipses

***See the SIGGRAPH 2009 course: Point Based Graphics – State of the Art and Recent Advances by Markus Gross.**

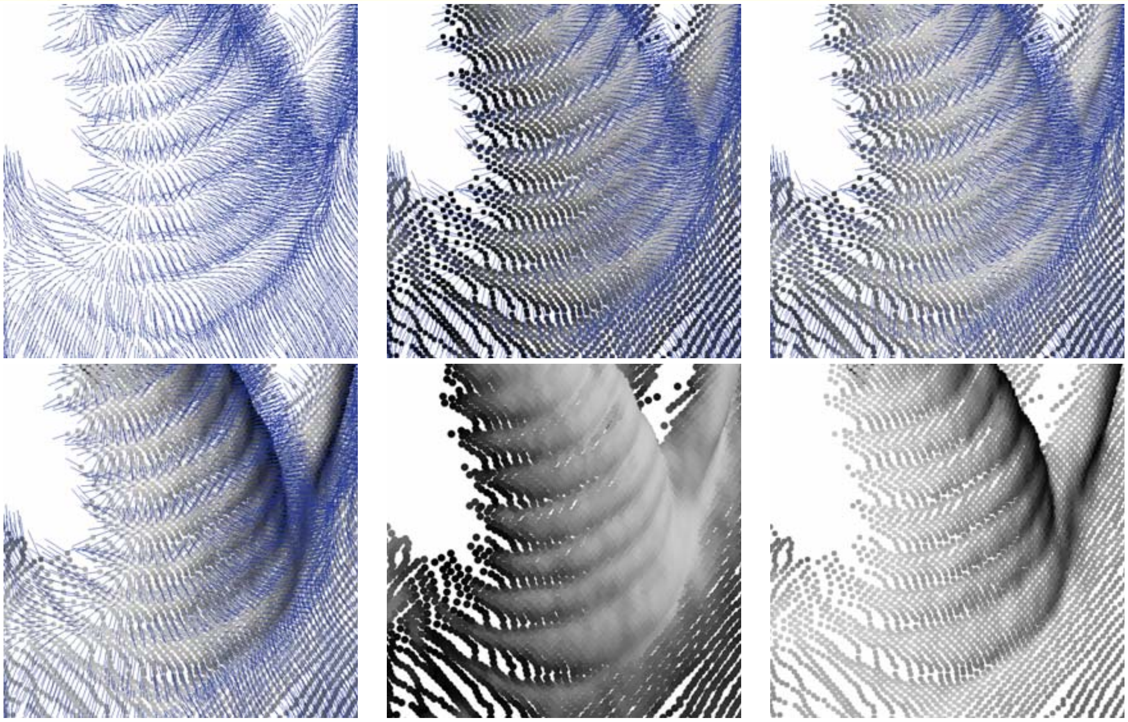
Visualizing Point Clouds:
Point-based Rendering via Splatting



- Swept-plane scanner produces a *colored point cloud*: a set of 3D points
- Problem: how to render a point cloud to make it look like as a continuous surface?
- Splatting: render points as overlapping colored disks
- If normal vectors are measured as well, render points as shaded ellipses

***See the SIGGRAPH 2009 course: Point Based Graphics – State of the Art and Recent Advances by Markus Gross.**

Visualizing Point Clouds:
Splatting with normal vectors and colors



Visualizing Point Clouds:
File Formats

- No standard file format to store point clouds
- Point = (x,y,z) plus (R,G,B) and/or (Nx,Ny,Nz)
- It is easy to create an ad-hoc file format
- Scene graph based file format: VRML
- International standard: ISO/IEC 14772-1:97 VRML'97
- PointSet node includes coordinates (x,y,z) and optional colors (R,G,B), but no normals

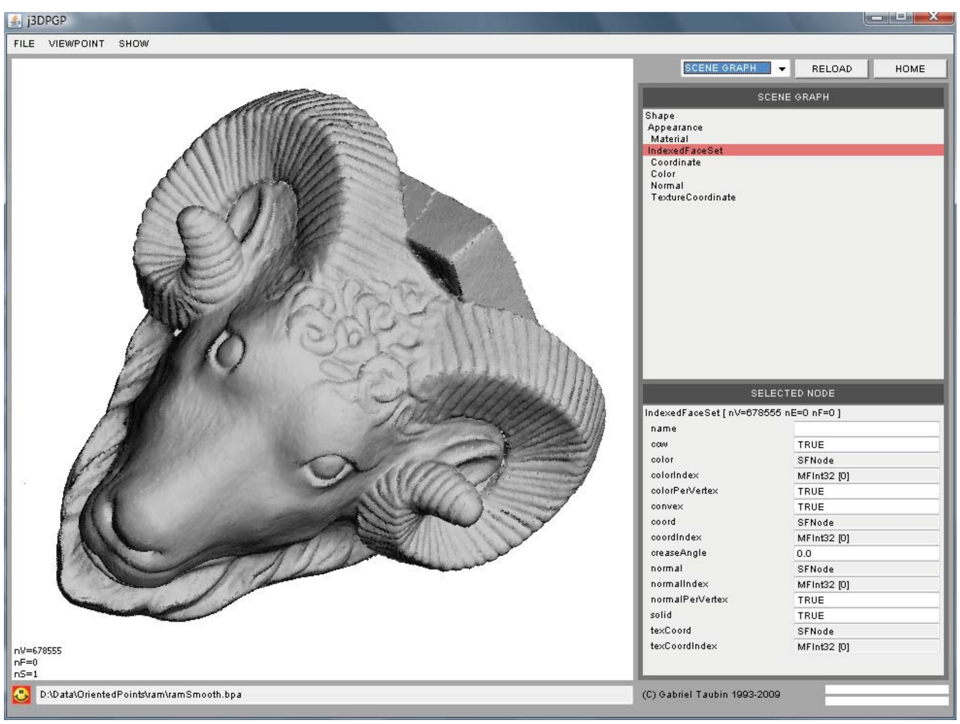
```
PointSet {
 coord Coordinate {
  point [
    0 -1  2,     1  0  0,
-2  3 -1
   ]
  }
 color Color {
  color [
    1 0 0,    0 1 0,     1 1 0
   ]
  }
}
```

Visualizing Point Clouds:
File Formats

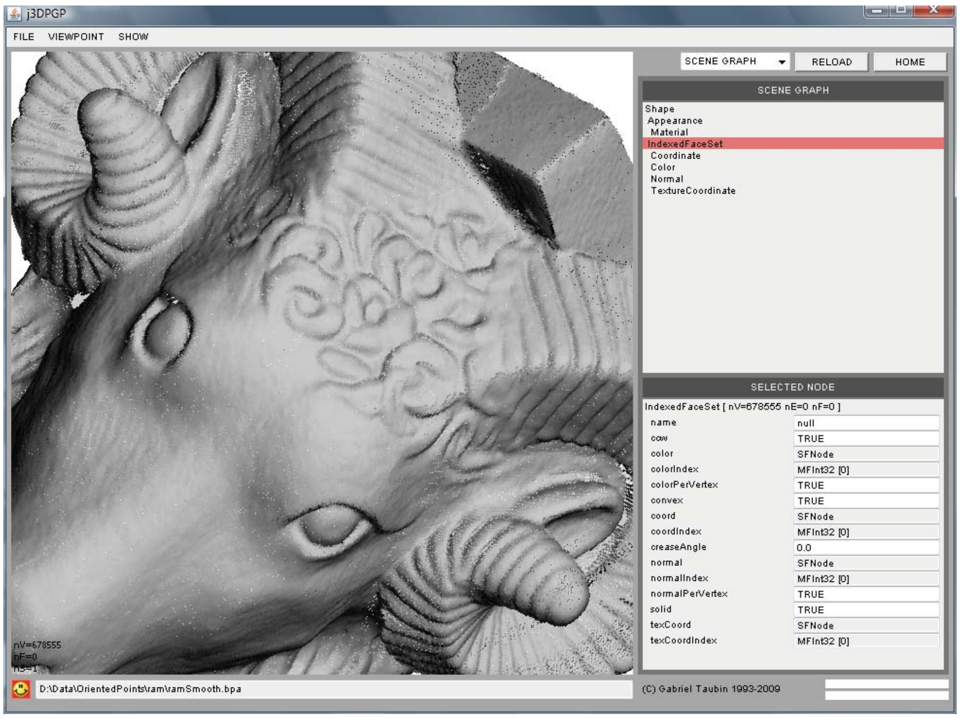
- IndexedFaceSet node designed to store a polygon mesh can be used to store point clouds with optional colors and/or normal vectors
- Store point coordinates as vertices
- Store point colors as colors per vertex
- Store point normal vectors as normals per vertex
- Degenerate polygon mesh with no faces is valid VRML syntax

```
IndexedFaceSet {
 coord Coordinate {
  point [
    0 -1  2,
    1  0  0,
   -2  3 -1
  ]
 }
 colorPerVertex TRUE
 color Color {
  color [
   1 0 0,
   0 1 0,
   1 1 0
  ]
 }
 normalPerVertex TRUE
 normal Normal {
  vector [
   1 0 0,
   0 1 0,
   0 0 1
  ]
 }
}
```

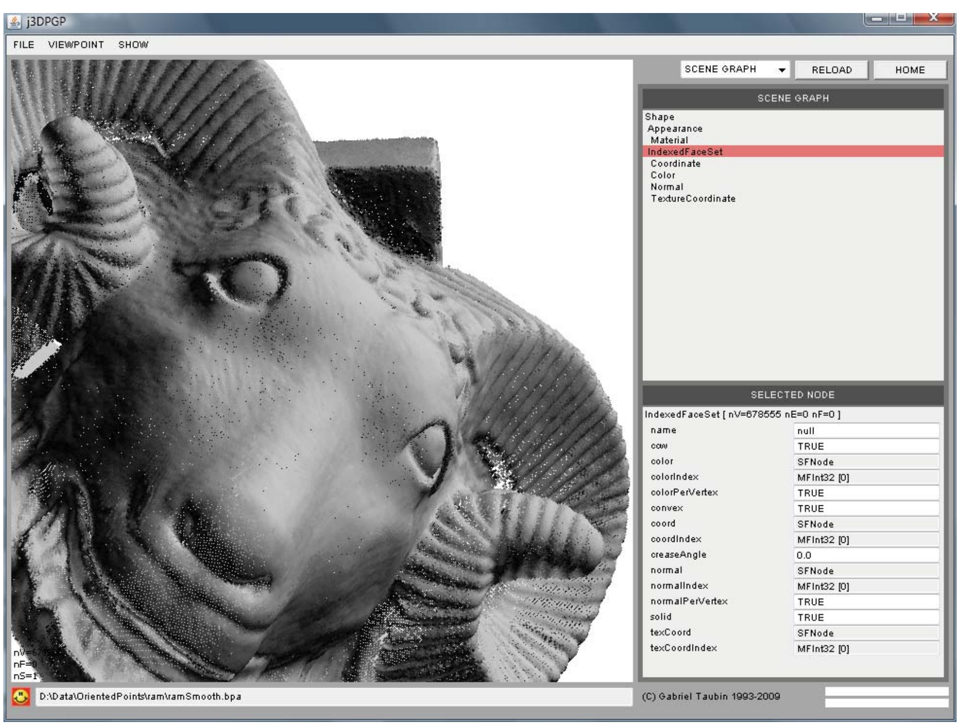
Visualizing Point Clouds:
BYO3D Java Viewer



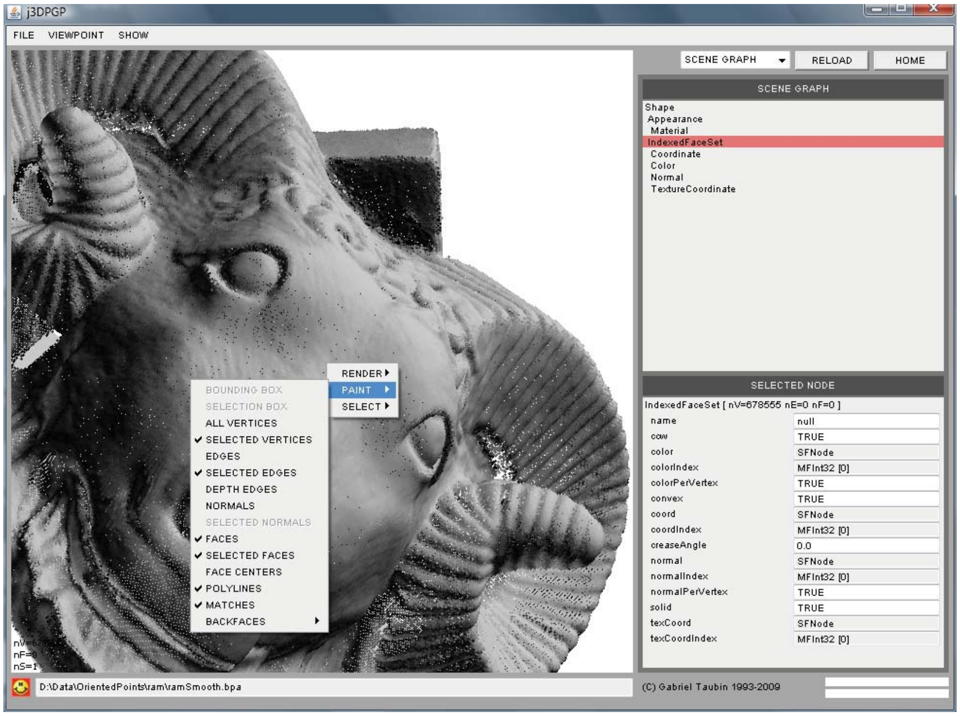
Visualizing Point Clouds: BYO3D Java Viewer



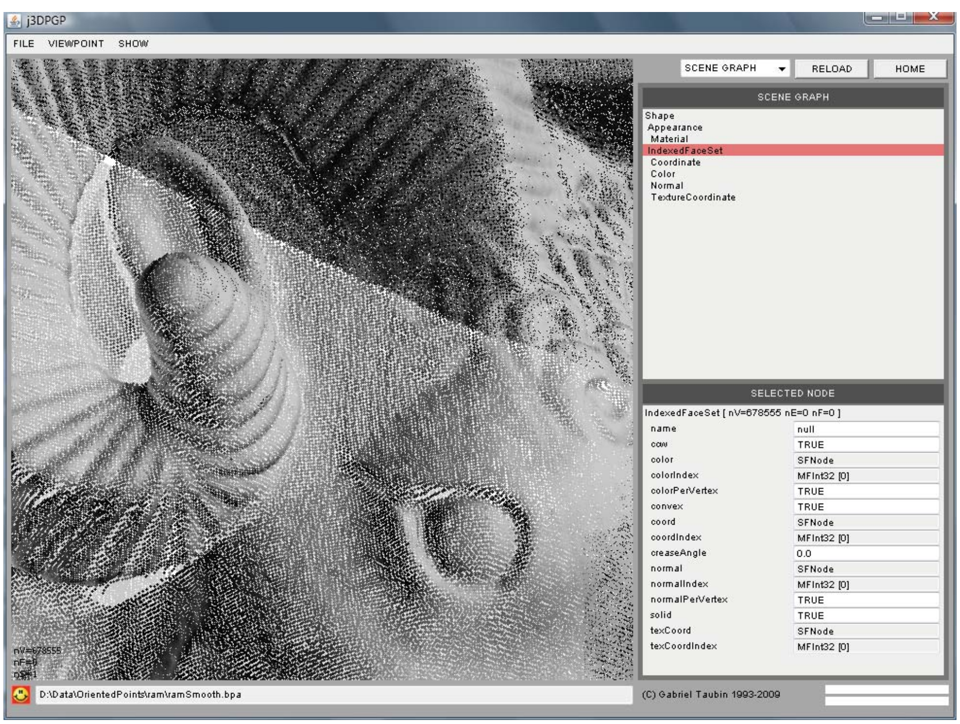
Visualizing Point Clouds: BYO3D Java Viewer



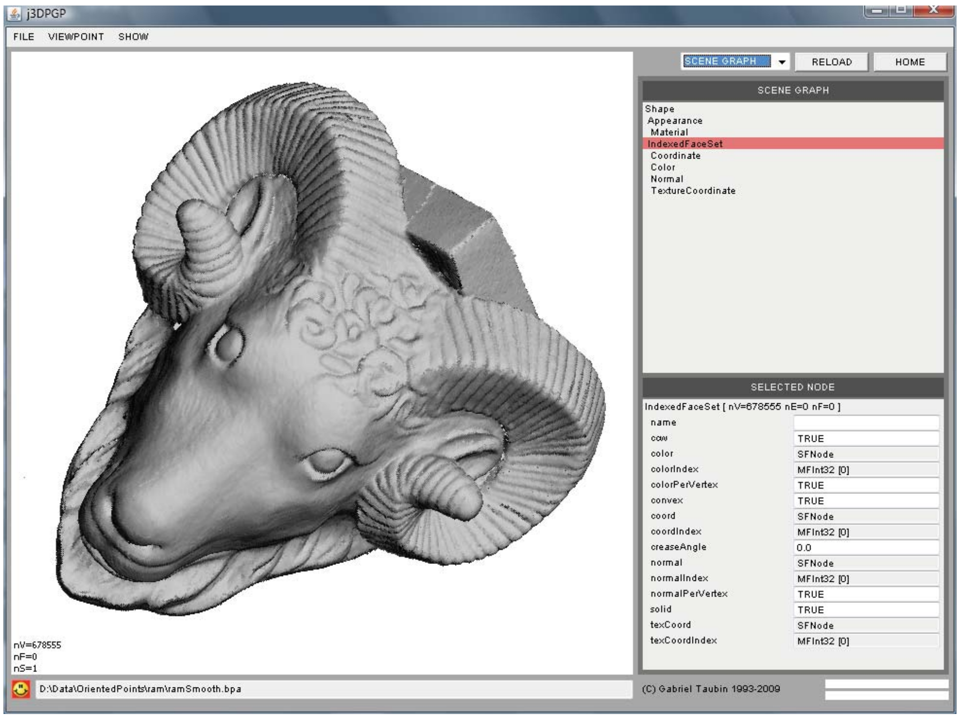
Visualizing Point Clouds: BYO3D Java Viewer



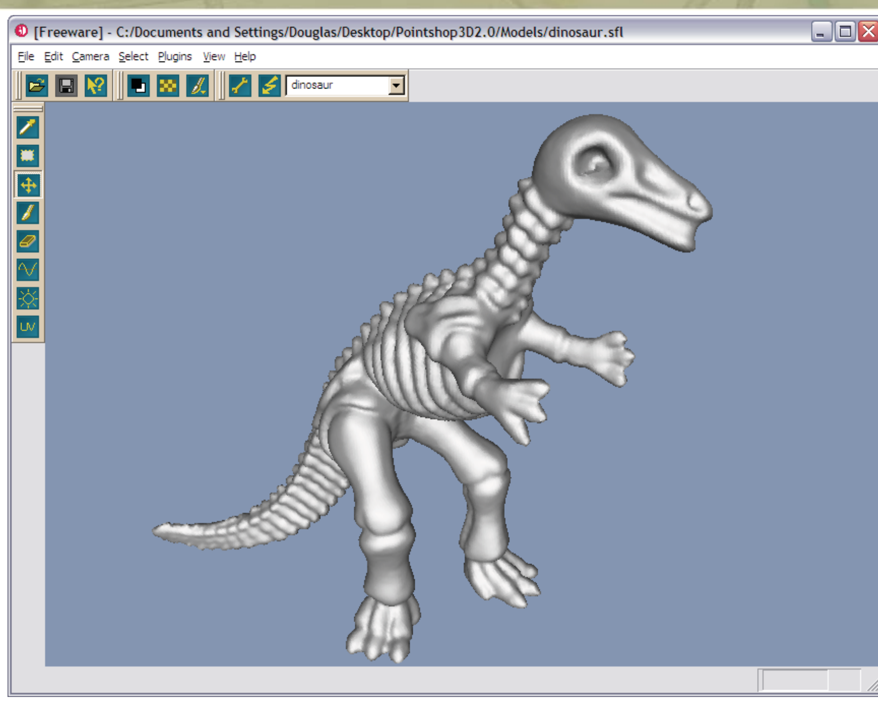
Visualizing Point Clouds: BYO3D Java Viewer



Visualizing Point Clouds:
BYO3D Java Viewer

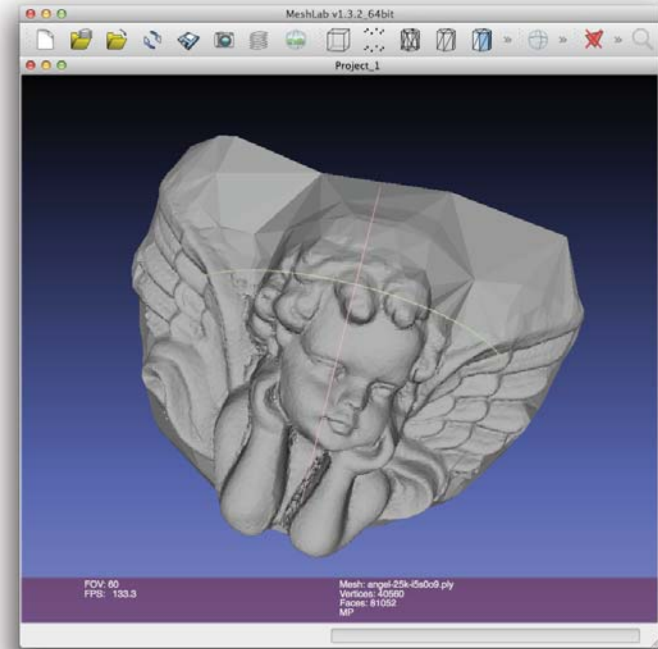


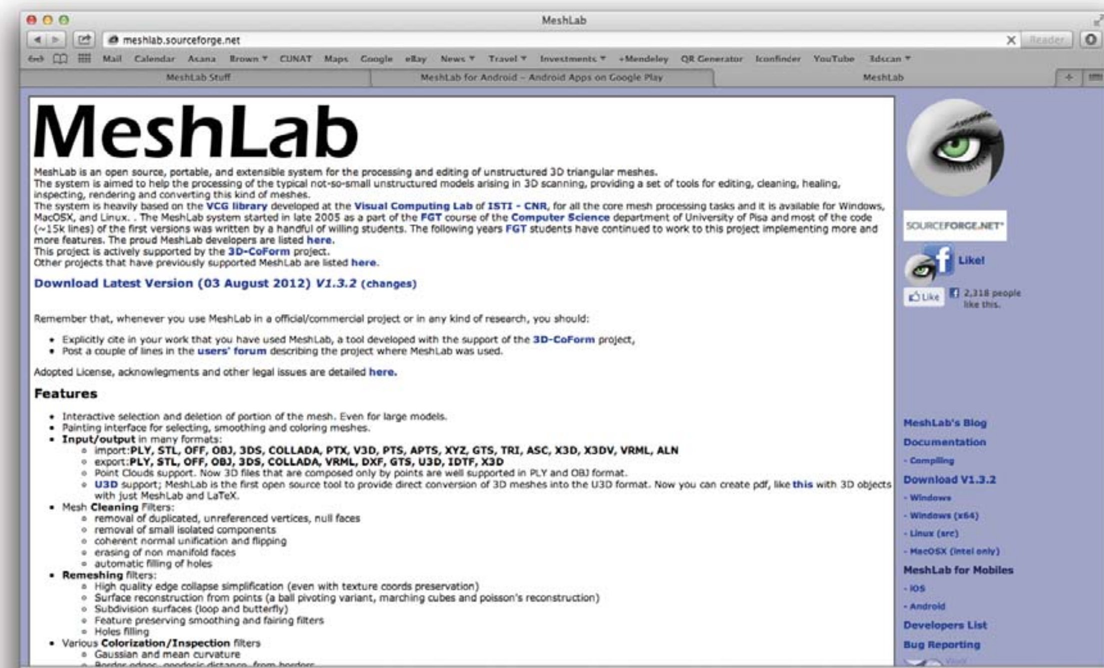
Visualizing Point Clouds:
Pointshop 3D [Zwicker et al. 2002]



M. Zwicker, M. Pauly, O. Knoll, M. Gross. Pointshop 3D: An Interactive System for Point-Based Surface Editing. ACM SIGGRAPH, 2002

Visualizing Point Clouds: MeshLab

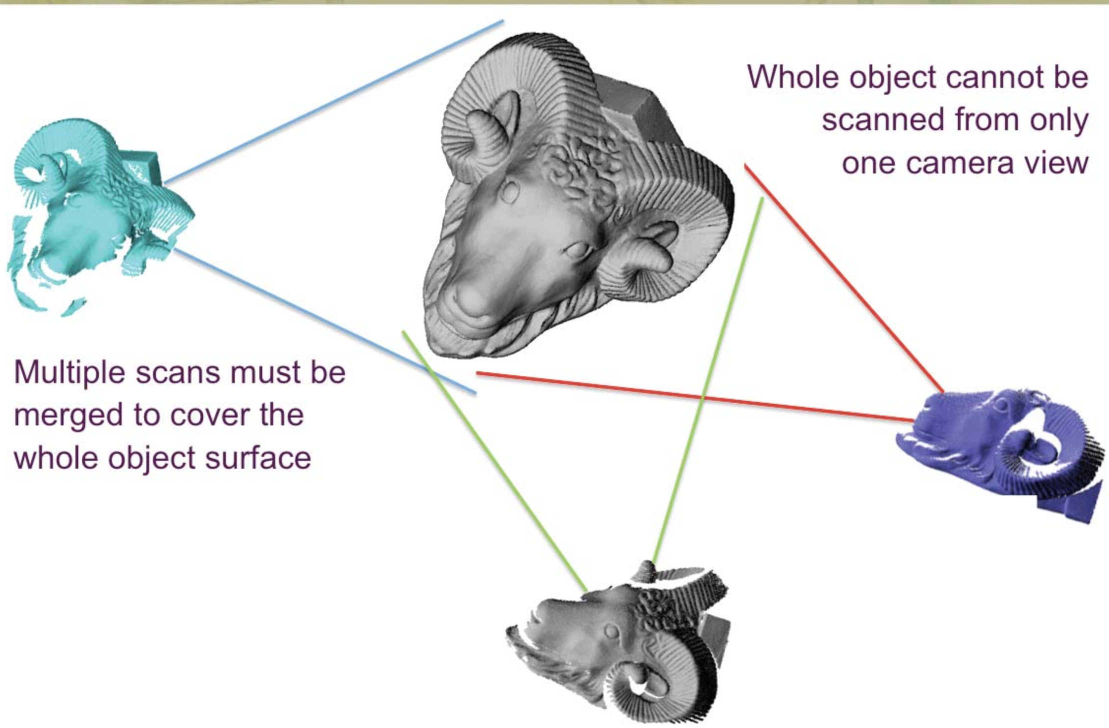




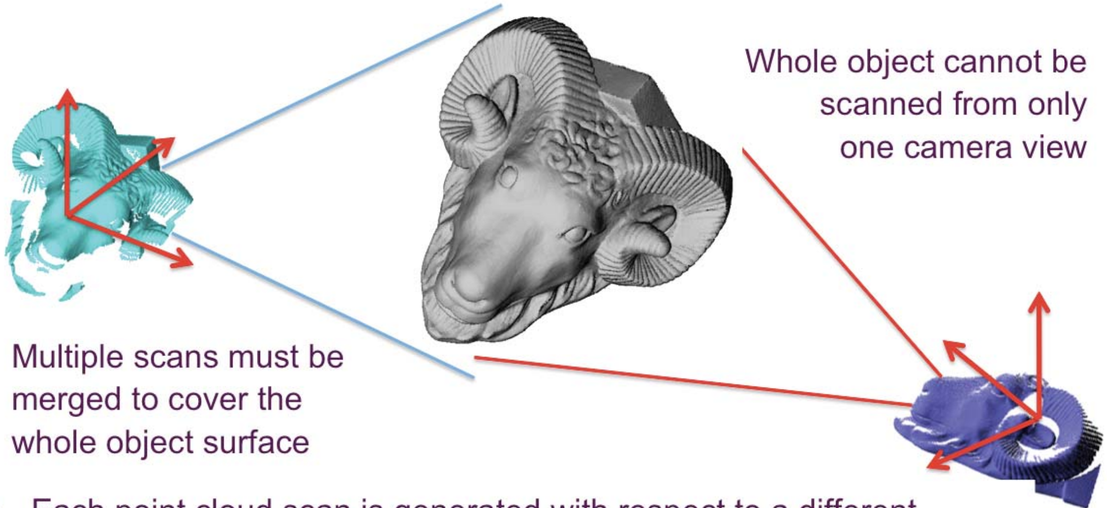
Course Schedule

- Introduction
- The Mathematics of 3D Triangulation
- 3D Scanning with Swept-Planes
- Camera and Swept-Plane Light Source Calibration
- Reconstruction and Visualization using Point Clouds
- ***Combining Point Clouds Recovered from Multiple Views***

Merging Multiple Point Cloud Scans

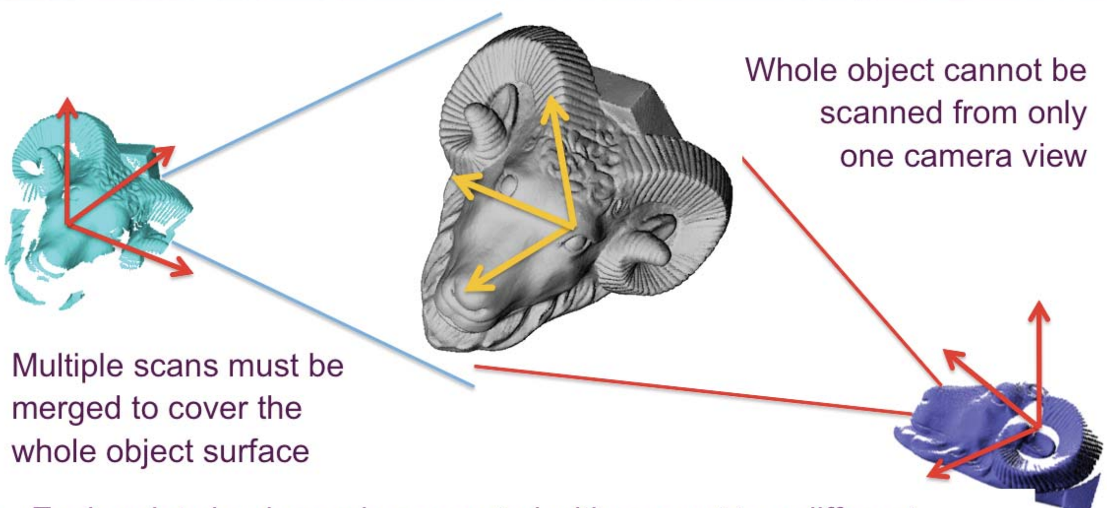


Merging Multiple Point Cloud Scans



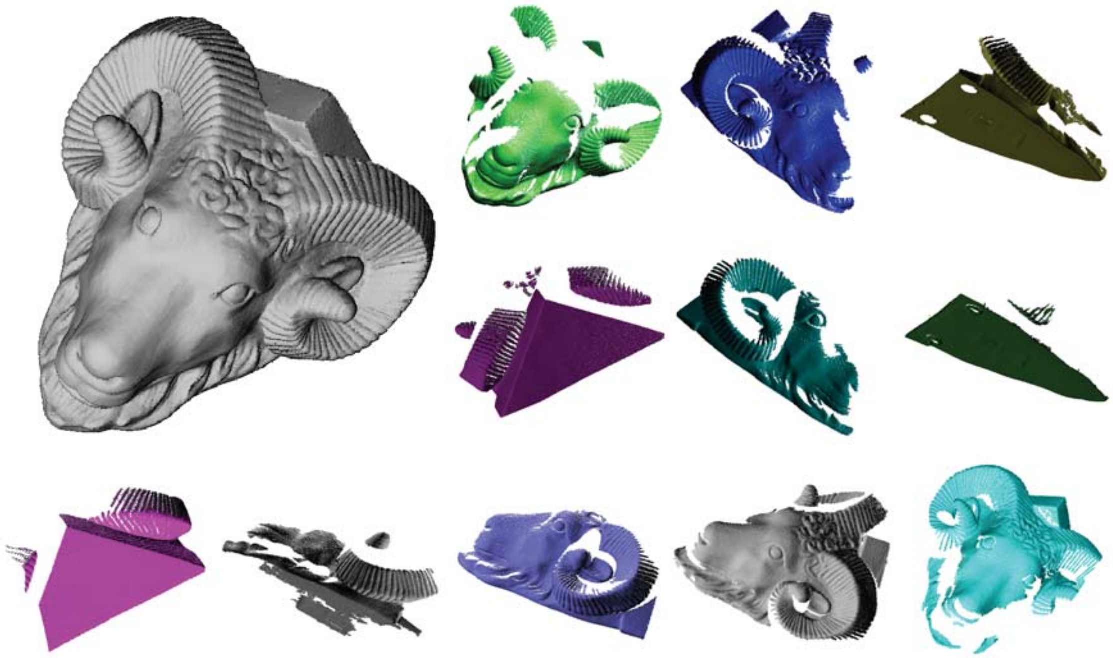
- Each point cloud scan is generated with respect to a different camera coordinate system

Merging Multiple Point Cloud Scans



- Each point cloud scan is generated with respect to a different camera coordinate system
- Relative position and orientation of each scan with respect to a global coordinate system must be determined to produce a single merged point cloud

Merging Point Cloud Scans

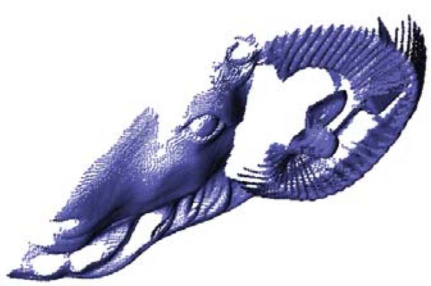


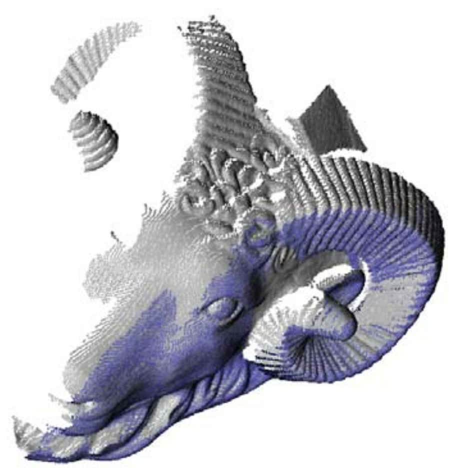
Complex Models May Require 100s of Scans

http://www.research.ibm.com/pieta

Merging Point Cloud Scans

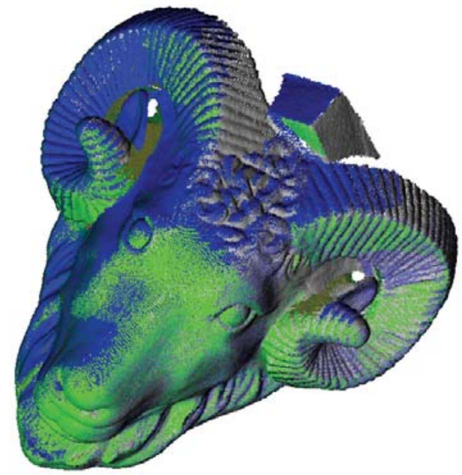
- Incremental registration and merging
- Followed by global relaxation to remove accumulated errors





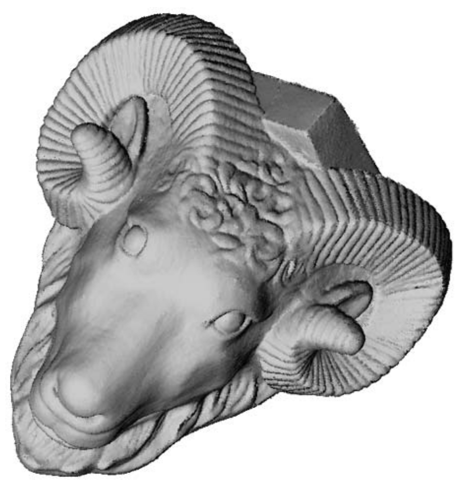
Merging Point Cloud Scans

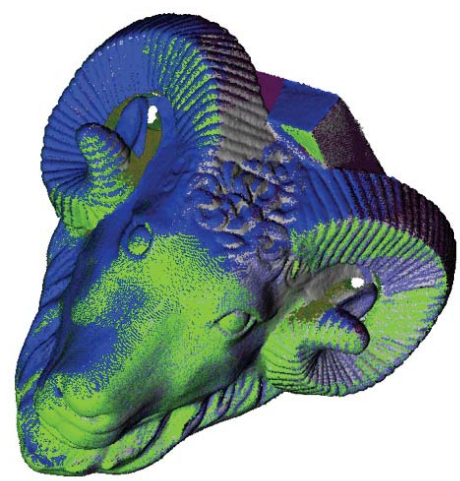
- Incremental registration and merging
- Followed by global relaxation to remove accumulated errors



Merging Point Cloud Scans

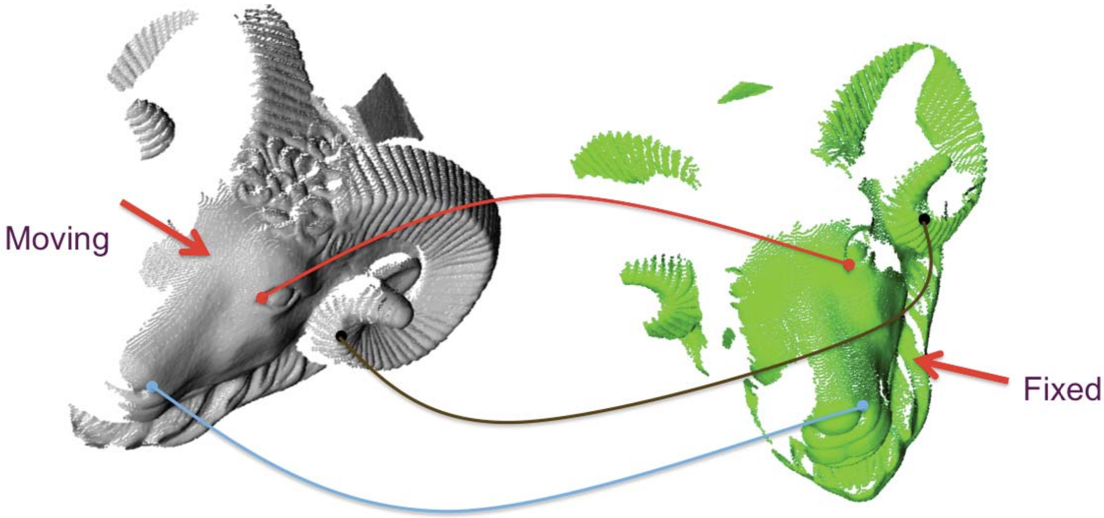
- Incremental registration and merging
- Followed by global relaxation to remove accumulated errors





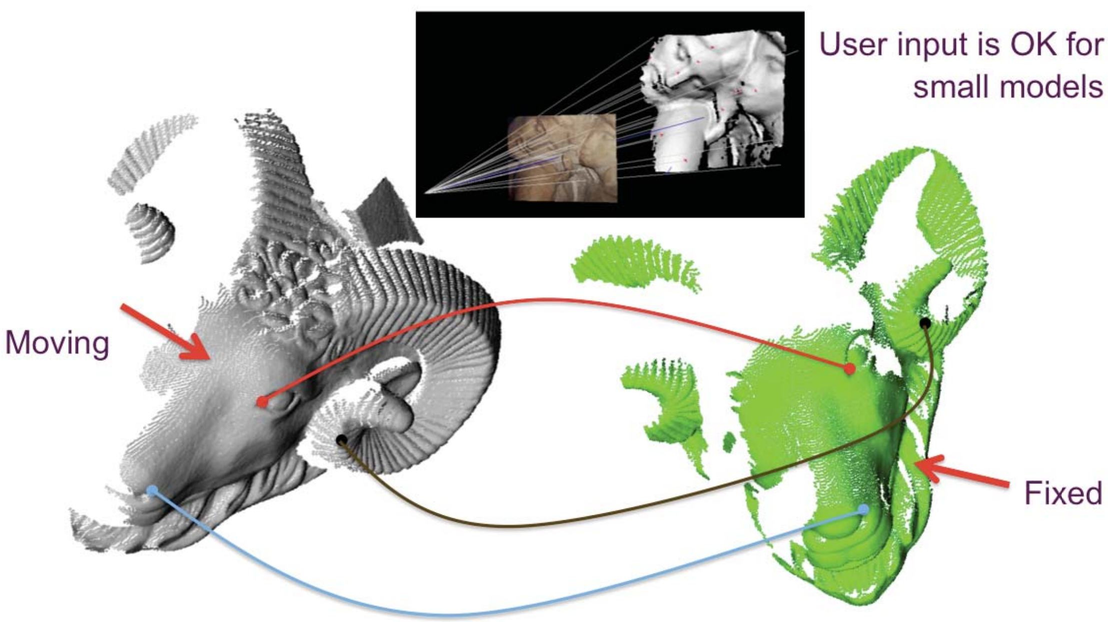
Merging Two Point Cloud Scans

- Select at least 3 pairs of corresponding points, but ideally N>>3



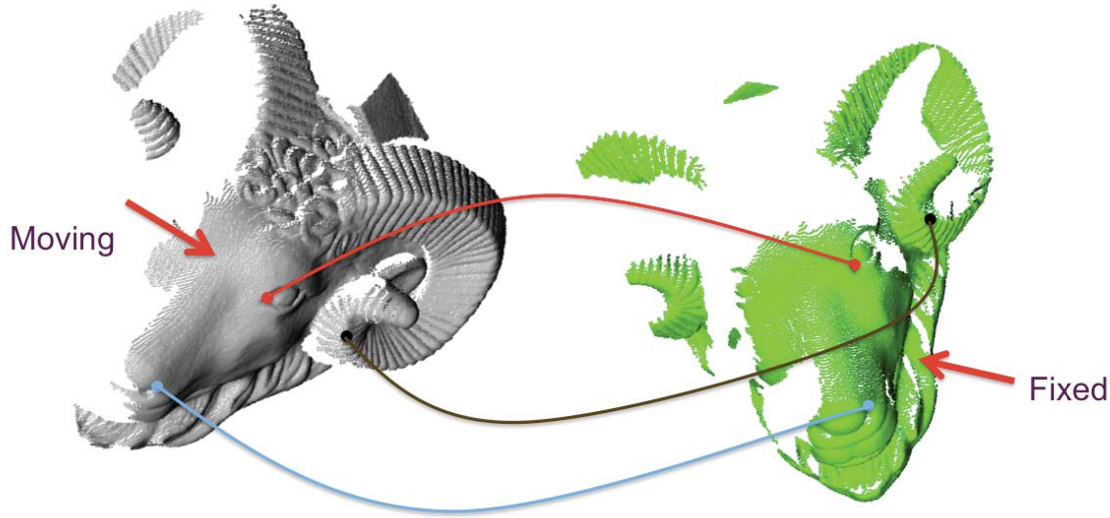
Merging Two Point Cloud Scans

- Select at least 3 pairs of corresponding points, but ideally N>>3



Merging Two Point Cloud Scans

- Select at least 3 pairs of corresponding points, but ideally N>>3
- Solve in close form for the matching rigid body transformation
- Refine solution using the Iterative Closest Point Algorithm (ICP)



P. Besl, N.D. McKey, A method for Registration of 3D Shapes.
IEEE Transactions on PAMI, 1992

Computing the Matching Transformation

- Given N pairs of corresponding 3D points $(p_1,q_1),\dots,(p_n,q_n)$ we are looking for a rotation matrix R and a translation vector T so that

$$Rp_j + T = q_j \quad j = 1,\dots,n$$

- In general, solution does not exists: solve in the Least-Squares sense
- Now we are looking for the minimizer of the quadratic energy function

$$E(R,T) = \frac{1}{n}\sum_{j=1}^{n}\left\| Rp_j + T - q_j \right\|^2$$

- This problem has a closed form solution

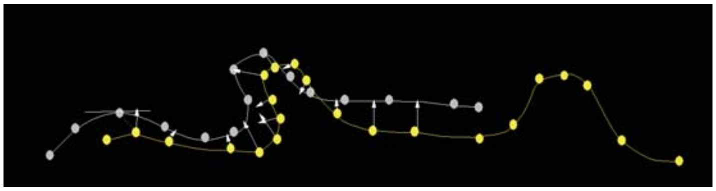
$$R = V^t U, \quad T = \overline{q} - R\,\overline{p}$$

- Where

$$\overline{p} = \frac{1}{n}\sum_{j=1}^{n} p_j \quad \overline{q} = \frac{1}{n}\sum_{j=1}^{n} q_j \quad M = \frac{1}{n}\sum_{j=1}^{n}(p_j - \overline{p})(q_j - \overline{q})^t$$

- And $M = U\Delta V^t$ is the Singular Value Decomposition (SVD) of M

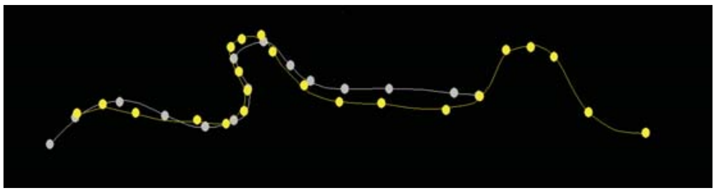
Iterative Closest Point Algorithm (ICP)



1. Automatically select N points $p_1,\ldots,p_n$
2. Find closest corresponding points $q_1,\ldots,q_n$
3. Solve in close form for the matching rigid body transformation which minimizes the energy function

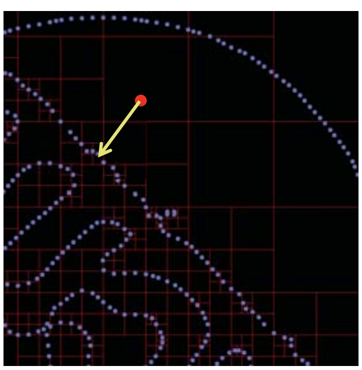
$$E(R,T) = \frac{1}{n}\sum_{j=1}^{n}\left\| Rp_j + T - q_j \right\|^2$$

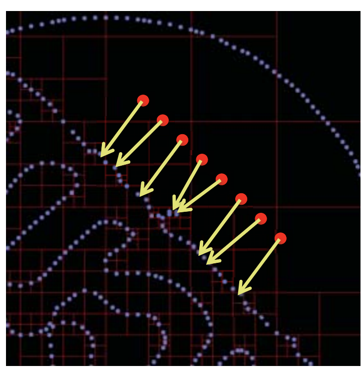
4. Repeat 1-3 while until convergence



Finding Closest Points

- Problem: find the point of the set $D = \{p_1,\ldots,p_n\}$ closest to the point q
- Naïve algorithm: sequential search O(N)
- Too expensive if the same computation must be performed for many points $q_1,\ldots,q_n$
- Efficient algorithm requires space partition data structure Quadtree/Octree, BSP tree





Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

Clase 6 : Martes Abril 9

Gabriel Taubin

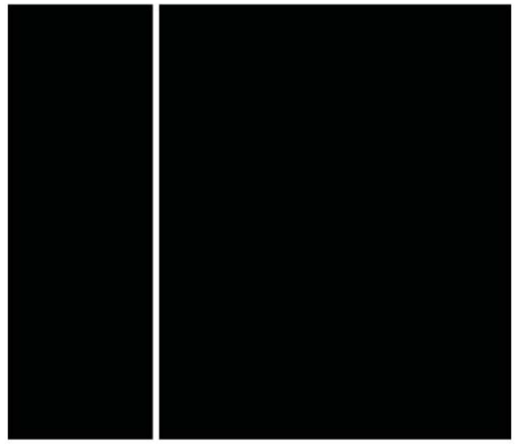
Brown University

BROWN

Course Schedule

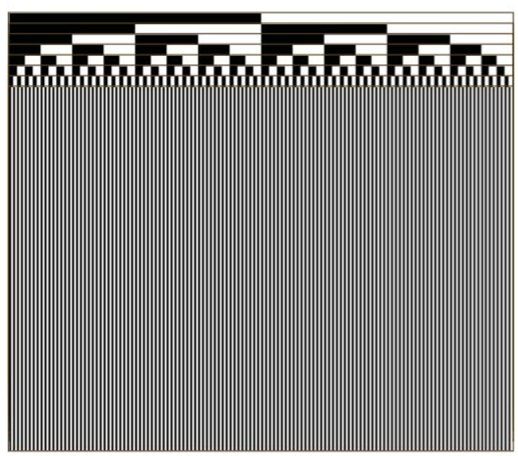
- ***Structured Lighting***
  - Projector Calibration and Structured Light Reconstruction

Structured Lighting: Swept-Planes Revisited



- Swept-plane scanning recovers 3D depth using ray-plane intersection
- Use a data projector to replace manually-swept laser/shadow planes
- How to assign correspondence from projector planes to camera pixels?
- Solution: Project a spatially- and temporally-encoded image sequence
- **What is the optimal image sequence to project?**

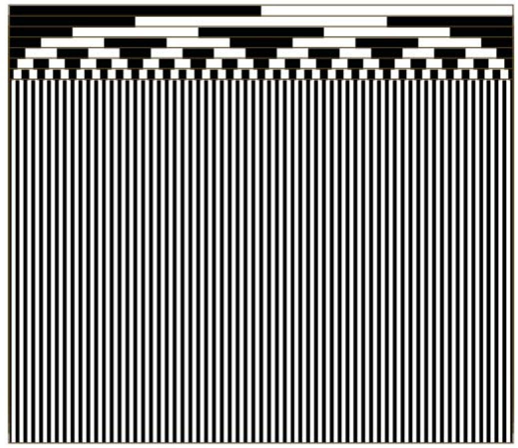
Structured Lighting: Binary Codes



Binary Image Sequence [Posdamer and Altschuler 1982]

- Each image is a bit-plane of the binary code for projector row/column
- Minimum of 10 images to encode 1024 columns or 768 rows
- In practice, 20 images are used to encode 1024 columns or 768 rows
- Projector and camera(s) must be synchronized

Structured Lighting: Gray Codes

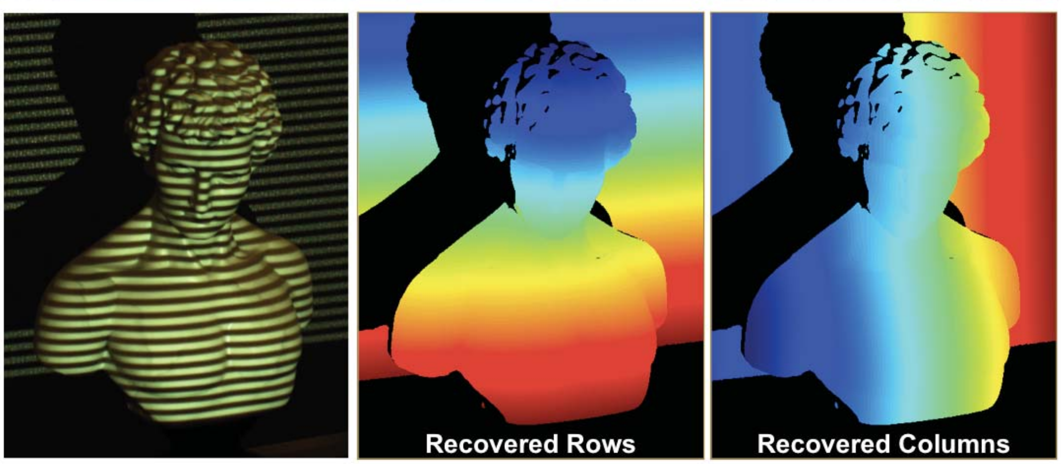


Gray Code Image Sequence [Inokuchi 1984]

- Each image is a bit-plane of the Gray code for each projector row/column
- Requires same number of images as a binary image sequence, but has better performance in practice

```
Bin2Gray(B,G)
G ← B
for i ← n-1 downto 0
    G[i] ← B[i+1] xor B[i]
```

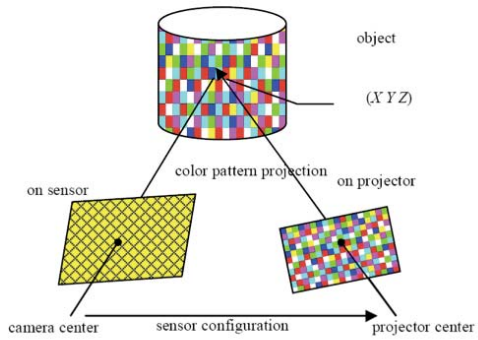
Gray Codes: Decoding Performance



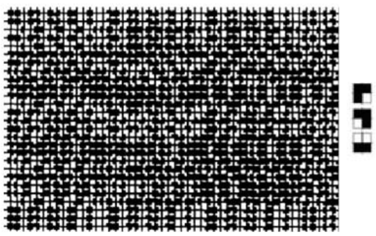
3D Reconstruction using Structured Light [Inokuchi 1984]

- Implemented using a total of 42 images
(2 to measure dynamic range, 20 to encode rows, 20 to encode columns)
- Individual bits assigned by detecting if bit-plane (or its inverse) is brighter
- Decoding algorithm: Gray code → binary code → integer row/column index

Additional Structured Lighting Patterns



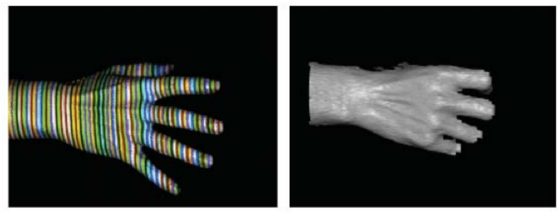
Spatial encoding strategies [Chen et al. 2007]



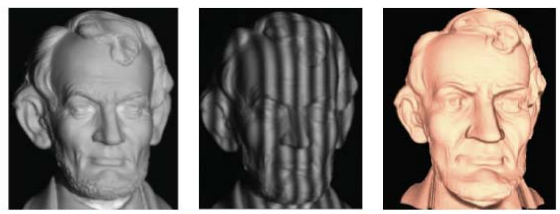
Pseudorandom and M-arrays [Griffin 1992]

J. Salvi, J. Pagès, and J. Batlle. Pattern Codification Strategies in Structured Light Systems. *Pattern Recognition*, 2004

"Single-shot" patterns (N-arrays, grids, random, etc.)



De Bruijn sequences [Zhang et al. 2002]



Phase-shifting [Zhang et al. 2004]

Assembling Your Own Scanner

Parts List [$600-$800, but only $300 without mounting hardware]

- [1] **Camera (Logitech QuickCam Pro 9000) [$70]**
- [1] **Projector (InFocus LP 70+) [$200-$400 from eBay]**
- [1] **Manfrotto 190XB 3 Section Aluminum Tripod [$130]**
- [1] **Manfrotto 486RC2 Compact Ball Head [$75]**
- [1] **Custom Aluminum Rail (with 3/8 inch holes for mounting ball heads) [NA]**
- [1] **Custom Aluminum Tripod Adaptor Plate (for mounting projector to ball head) [NA]**
- [2] **Manfrotto 484 Mini Ball Heads [$45 each]**
- [1] **Sunpak 620-500C Versipod (with adjustable, locking ball head) [$15]**
- [1] **~11x17 inch foamcore board (with an affixed calibration chessboard) [$10]**

Assembling Your Own Scanner

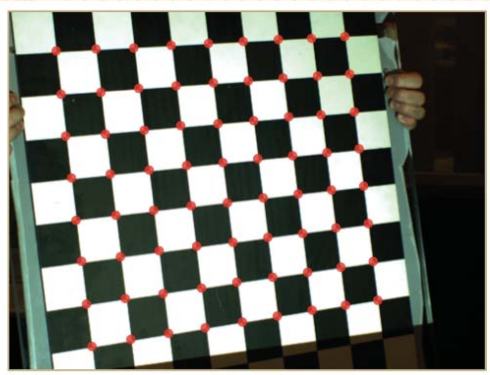
Parts List [$600-$800, but only $300 without mounting hardware]

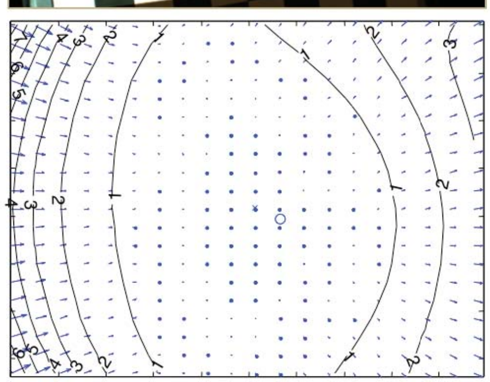
- [1] **Camera (Logitech QuickCam Pro 9000) [$70]**
- [1] **Projector (InFocus LP 70+) [$200-$400 from eBay]**
- [1] **Manfrotto 190XB 3 Section Aluminum Tripod [$130]**
- [1] **Manfrotto 486RC2 Compact Ball Head [$75]**
- [1] **Custom Aluminum Rail (with 3/8 inch holes for mounting ball heads) [NA]**
- [1] **Custom Aluminum Tripod Adaptor Plate (for mounting projector to ball head) [NA]**
- [2] **Manfrotto 484 Mini Ball Heads [$45 each]**
- [1] **Sunpak 620-500C Versipod (with adjustable, locking ball head) [$15]**
- [1] **~11x17 inch foamcore board (with an affixed calibration chessboard) [$10]**

Developing Your Own Software

Key Functions

- Camera and projector calibration





Developing Your Own Software

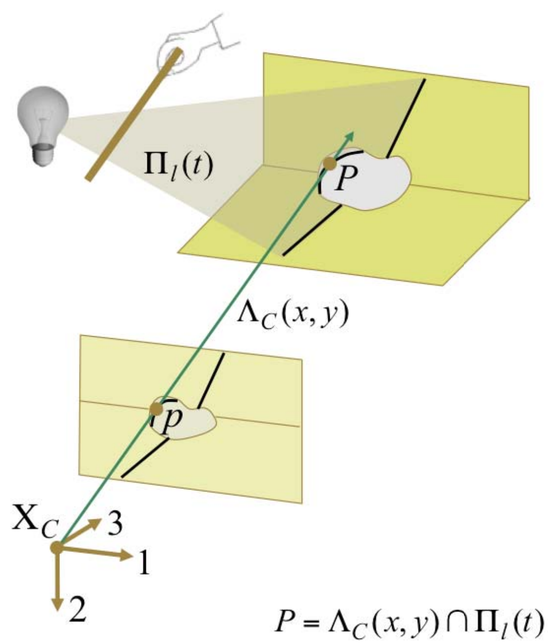
Key Functions

- Camera and projector calibration
- Camera and projector interfaces

Developing Your Own Software

Key Functions

- Camera and projector calibration
- Camera and projector interfaces
- Geometric operations



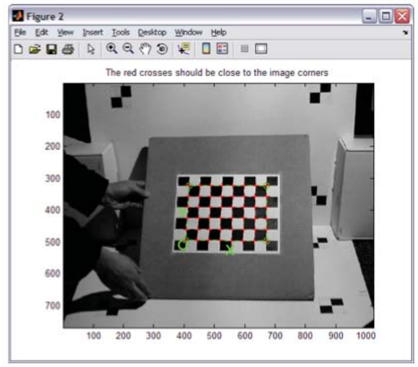
Developing Your Own Software

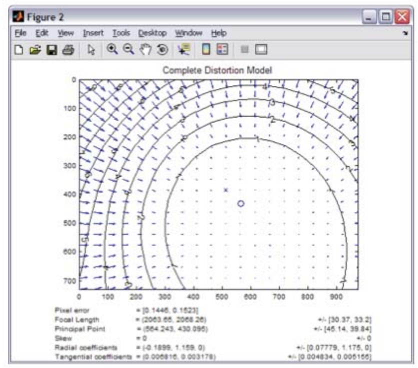
Key Functions

- Camera and projector calibration
- Camera and projector interfaces
- Geometric operations

Matlab

- Camera Calibration Toolbox
  (updated to support projector calibration)





Developing Your Own Software

Key Functions

- Camera and projector calibration
- Camera and projector interfaces
- Geometric operations

Matlab

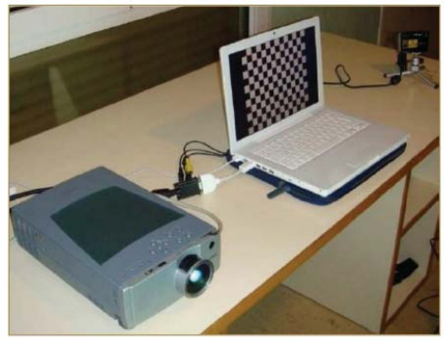
- Camera Calibration Toolbox
  (updated to support projector calibration)
- Projector-Camera Calibration Toolbox
  (just released, supports full calibration)

http://code.google.com/p/procamcalib/

Developing Your Own Software

Key Functions
- Camera and projector calibration
- Camera and projector interfaces
- Geometric operations

Matlab
- Camera Calibration Toolbox (updated to support projector calibration)
- Projector-Camera Calibration Toolbox (just released, supports full calibration)
- Image Acquisition Toolbox, etc.
- Geometric operations (course source)

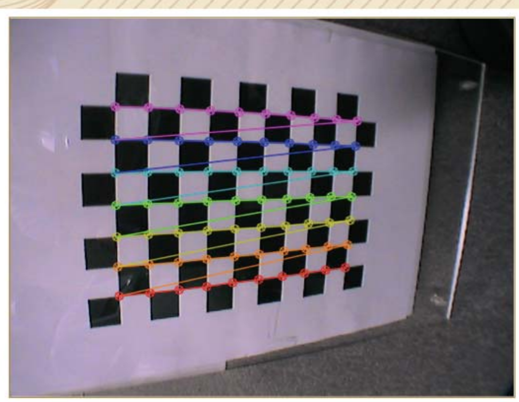


http://code.google.com/p/procamcalib/

Developing Your Own Software

Key Functions
- Camera and projector calibration
- Camera and projector interfaces
- Geometric operations

Matlab
- Camera Calibration Toolbox (updated to support projector calibration)
- Projector-Camera Calibration Toolbox (just released, supports full calibration)
- Image Acquisition Toolbox, etc.
- Geometric operations (course source)



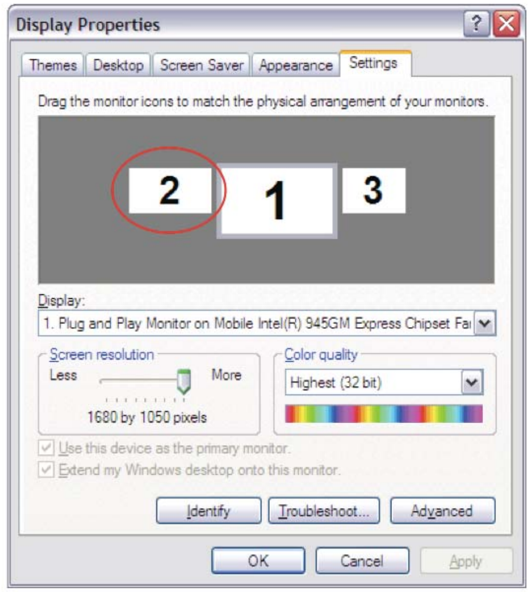
OpenCV
- Supports camera calibration
- Updated to support projector calibration (course source)

http://opencv.willowgarage.com/wiki/

Controlling a Projector for Structured Lighting

Operating System

- A projector is just another display…



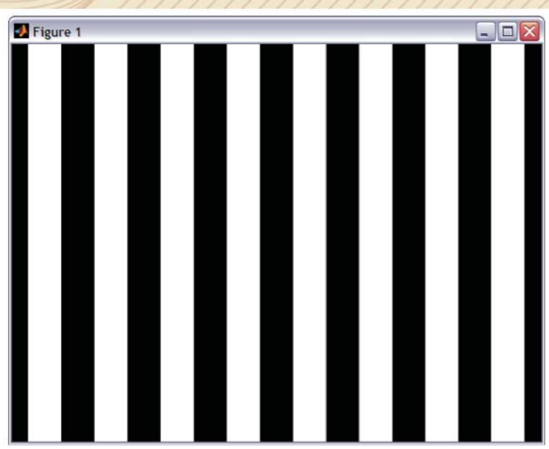
Controlling a Projector for Structured Lighting

Operating System

- A projector is just another display…

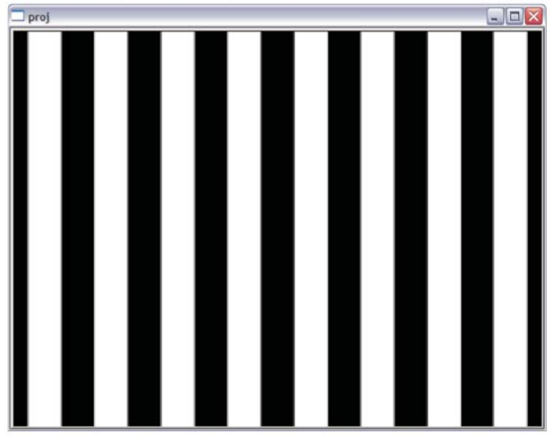
Matlab

- Fullscreen display not fully supported (cannot remove window decorations)




```
imagesc(P{1}(:,:,5));
axis image off;
set(gca,'Pos',[0 0 1 1]);
set(gcf,'MenuBar','none');
set(gcf,'Pos',[-1023 283 1024 768]);
```

Controlling a Projector for Structured Lighting



Operating System
- A projector is just another display…

Matlab
- Fullscreen display not fully supported (cannot remove window decorations)
- Use Java to control displays (slow, multiple display problems)
- Use the *Psychophysics Toolbox* (Screen.m wraps OpenGL functions)

OpenCV
- Cannot remove window decorations
- cvNamedWindow/cvMoveWindow

General
- Use OpenGL wrappers

```
cvNamedWindow("proj",CV_WINDOW_AUTOSIZE);
IplImage* proj_frame =
   cvCreateImage(cvSize(w,h),8,3);
cvSet(proj_frame,cvScalar(0,0,0));
cvShowImage("proj",proj_frame);
cvMoveWindow("proj",-w-7,-33);
cvWaitKey(1);
```

http://opencv.willowgarage.com/wiki

Course Schedule

- Structured Lighting
- ➢ ***Robust Pixel Classification***
- Projector Calibration / Structured Light Reconstruction
- Surface Reconstruction from Point Clouds
- Elementary Mesh Processing

Robust Pixel Classification

- **Robust pixel classification for 3d modeling with structured light**, by Yi Xu and Daniel G. Aliaga, in Proceedings of Graphics Interface 2007, GI ’07, pages 233–240, New York, NY, USA, 2007. ACM.
- **Fast separation of direct and global components of a scene using high frequency illumination**, by Shree K. Nayar, Gurunandan Krishnan, Michael D. Grossberg, and Ramesh Raskar, in ACM Trans. Graph., 25(3):935–944, July 2006.

Robust Pixel Classification

- The intensity of a pixel can be decomposed into the direct component and indirect (or global) component.
- The direct component is due to light bouncing off the surface in a single reflection.
- The indirect component is due to multiple reflections (e.g. inter-reflections, subsurface, scattering etc.).

$$L(p) = \begin{cases} L_d(p) + \alpha L_g(p) & \text{if p is on} \\ (1-\alpha) L_g(p) & \text{if p is off,} \end{cases}$$

- where $0 \leq \alpha \leq 1$ is a fraction of activated source pixels. For example, for a random binary pattern $\alpha = 1$, and for a white pattern $\alpha = 1$.

Robust Pixel Classification

- Estimate direct and global components for each pixel using the method by Nayar
- Use two images of the scene, one with the scene lit with high-frequency illumination, and the other lit with the complementary illumination.

$$L_d(p) = L_{max}(p) - \frac{\alpha}{1-\alpha} L_{min}(p)$$
$$L_g(p) = \frac{1}{1-\alpha} L_{min}(p),$$

Robust Pixel Classification

- Projecting the code pattern i and its inverse yields two values for each pixel.
- Classify p according to the rules

$$p \text{ is } \begin{cases} \text{uncertain} & \text{if } L_d(p) < m \\ \text{on} & \text{if } L_d(p) > L_g(p) \text{ and } L_i^+(p) > L_i^-(p) \\ \text{off} & \text{if } L_d(p) > L_g(p) \text{ and } L_i^+(p) < L_i^-(p) \\ \text{off} & \text{if } L_i^+(p) < L_d(p) \text{ and } L_i^-(p) > L_g(p) \\ \text{on} & \text{if } L_i^+(p) > L_g(p) \text{ and } L_i^-(p) < L_d(p) \\ \text{uncertain} & \text{otherwise,} \end{cases}$$

Introducción a la Fotografia 3D
UBA/FCEN Marzo 27 – Abril 12 2013
Clase 7 : Miercoles Abril 10

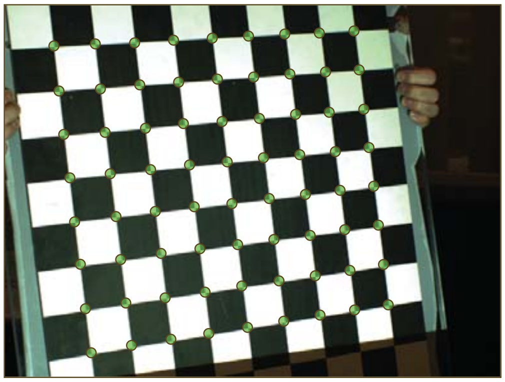
Gabriel Taubin
Brown University

BROWN

Course Schedule

- Structured Lighting
- Robust Pixel Classification
- ***Projector Calibration / Structured Light Reconstruction***
- Projector Calibration / Structured Light Reconstruction
- Surface Reconstruction from Point Clouds
- Elementary Mesh Processing

Summary of Camera Calibration



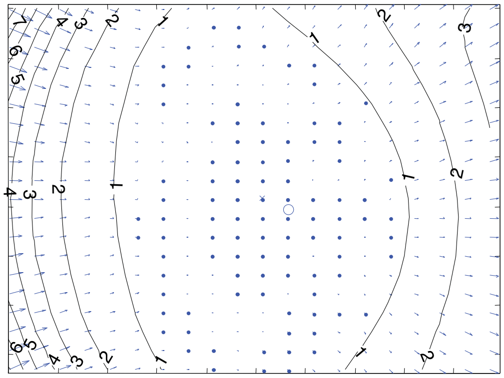


Image Sequence for Camera Calibration

Estimated Camera Lens Distortion

Camera Calibration Procedure

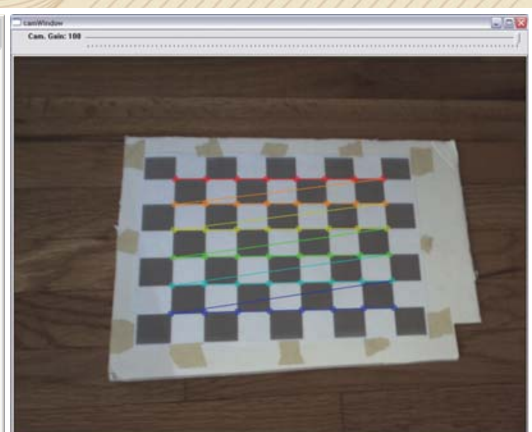
- Use the *Camera Calibration Toolbox for Matlab* or *OpenCV*

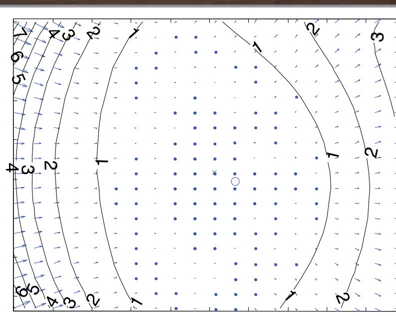
Normalized Ray	Distorted Ray (4th-order radial + tangential)	Predicted Image-plane Projection
$x_n = \begin{bmatrix} X_c/Z_c \\ Y_c/Z_c \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}$	$x_d = \begin{bmatrix} x_d(1) \\ x_d(2) \end{bmatrix} = \left(1 + kc(1)\, r^2 + kc(2)\, r^4 + kc(5)\, r^6\right) x_n + dx$ $dx = \begin{bmatrix} 2\, kc(3)\, x\, y + kc(4)\left(r^2 + 2x^2\right) \\ kc(3)\left(r^2 + 2y^2\right) + 2\, kc(4)\, x\, y \end{bmatrix}$	$x_p = fc(1)\left(x_d(1) + alpha\_c * x_d(2)\right) + cc(1)$ $y_p = fc(2)\, x_d(2) + cc(2)$

Demo: Camera Calibration in OpenCV

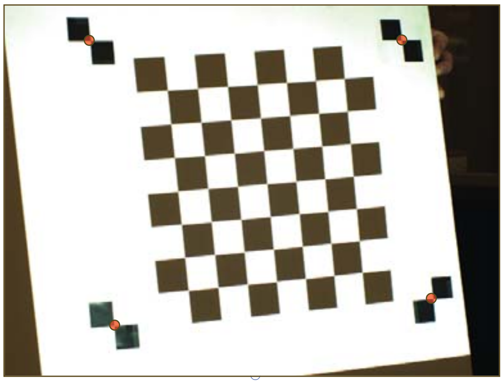
c:\Documents and Settings\Douglas\Desktop\Active Projects\3D Scanning ...

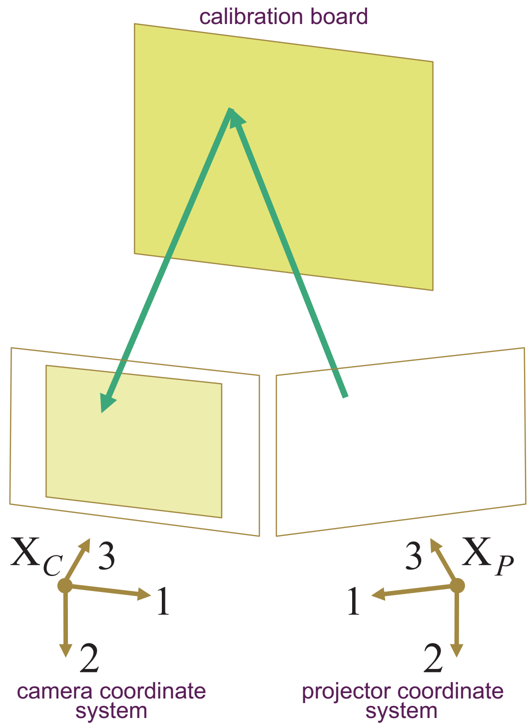
```
[Structured Lighting for 3D Scanning]
Reading configuration file "./config.xml"...
Initializing camera and projector...
Enabling Bayer mode for Logitech QuickCam 9000...
Creating output directory (overwrites existing object data)...
Camera has not been intrinsically calibrated!
Projector has not been intrinsically calibrated!
Projector-camera system has not been extrinsically calibrated!

Press the following keys for the corresponding functions.
'S': Run scanner
'B': Estimate background
'R': Reset background
'C': Calibrate camera
'P': Calibrate projector
'A': Calibrate camera and projector simultaneously
'E': Calibrate projector-camera alignment
'ESC': Exit application

> Calibrating camera...
Creating camera calibration directory (overwrites existing data)...
Enter the maximum number of calibration images, then press return.
+ Maximum number of images = 15
Press 'n' (in 'camWindow') to capture next image, or 'ESC' to quit.
+ Captured frame 1 of 15.
+ Captured frame 2 of 15.
+ Captured frame 3 of 15.
+ Captured frame 4 of 15.
+ Captured frame 5 of 15.
+ Captured frame 6 of 15.
+ Captured frame 7 of 15.
+ Captured frame 8 of 15.
+ Captured frame 9 of 15.
+ Captured frame 10 of 15.
+ Captured frame 11 of 15.
+ Captured frame 12 of 15.
+ Captured frame 13 of 15.
+ Captured frame 14 of 15.
+ Captured frame 15 of 15.
Calibrating camera...
Saving calibration images and parameters...
Camera calibration was successful.
Camera calibration:
+ Intrinsic parameters =
   1311.368   0.000 770.787
      0.000 1315.449 609.661
      0.000   0.000   1.000
+ Distortion coefficients =
      0.051  -0.147   0.003  -0.005   0.000
```





Projector Calibration

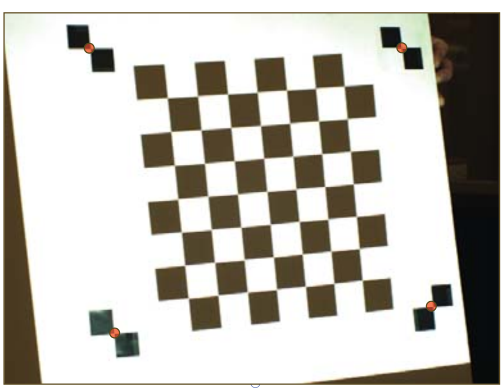


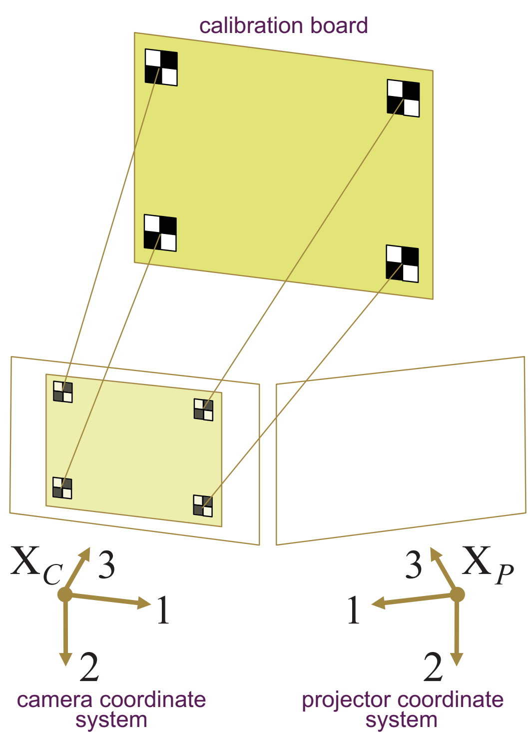


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)

Projector Calibration

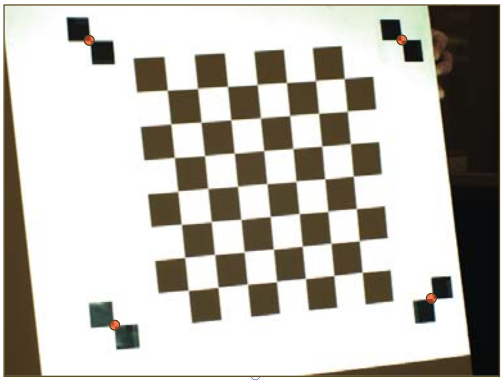


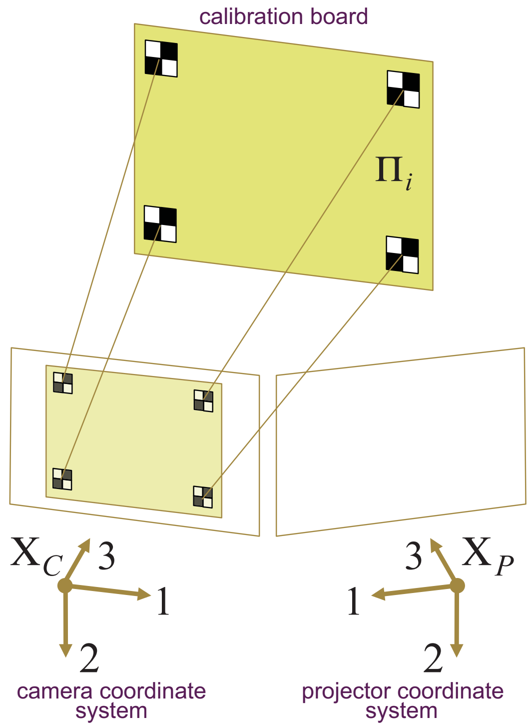


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image

Projector Calibration

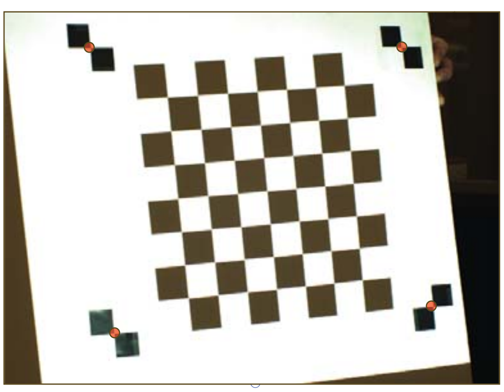


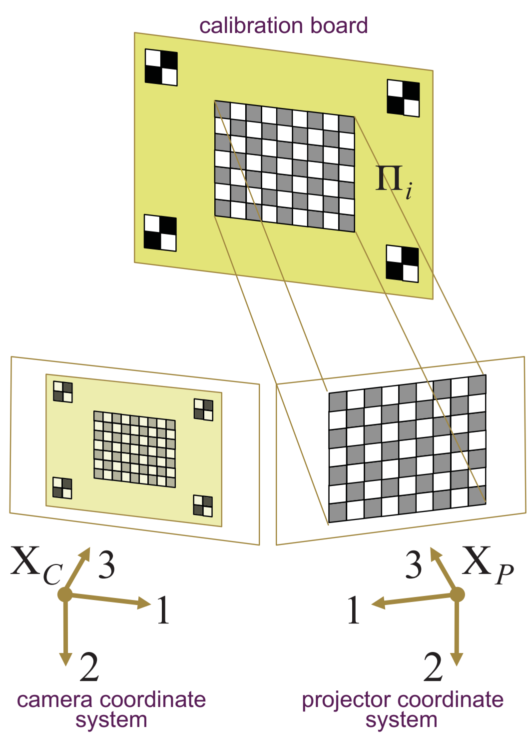


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image
- Use fidicials to find 3D calibration plane

Projector Calibration

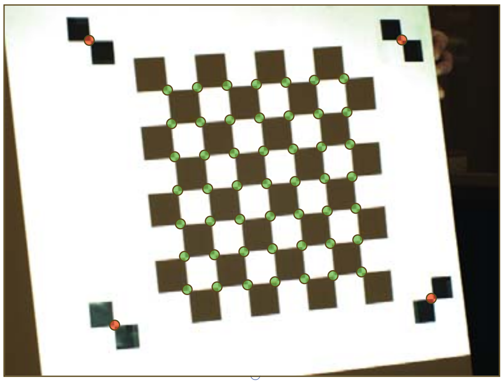


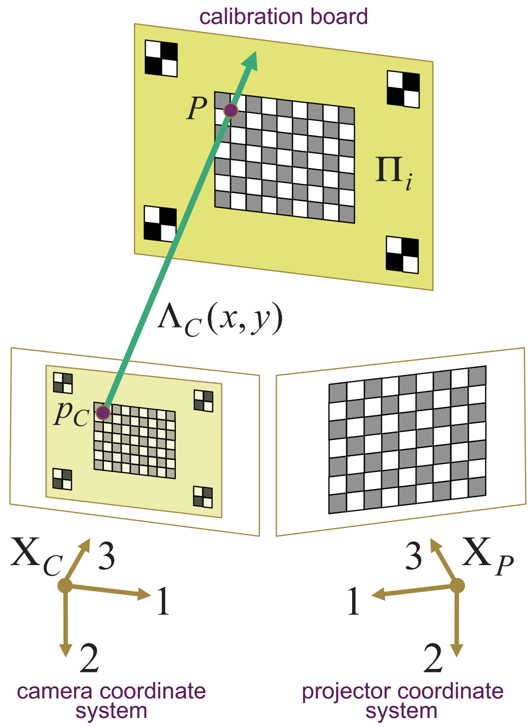


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image
- Use fidicials to find 3D calibration plane
- Project checkerboard on calibration board

Projector Calibration

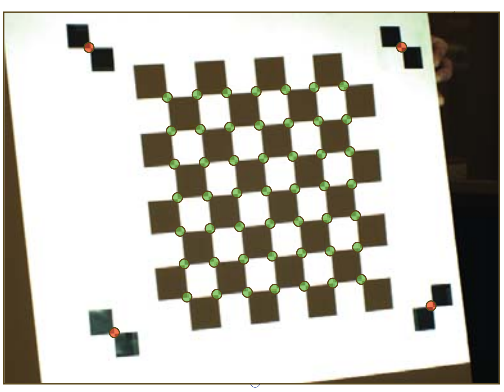


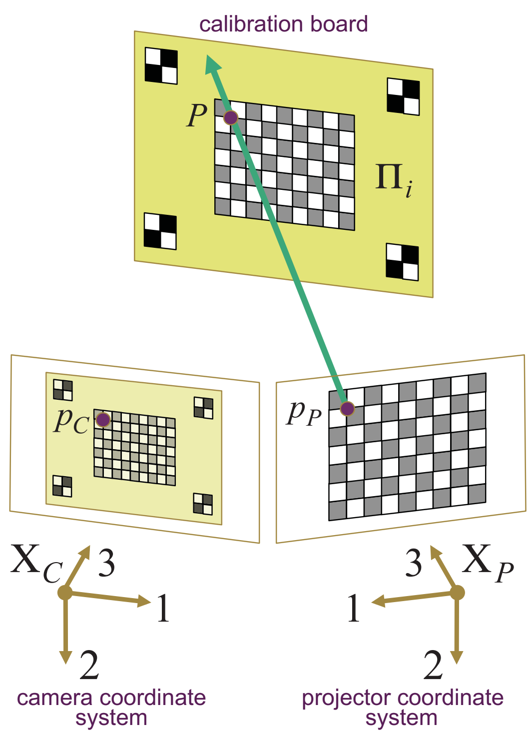


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image
- Use fidicials to find 3D calibration plane
- Project checkerboard on calibration board
- Find ray-plane intersection for each corner

Projector Calibration

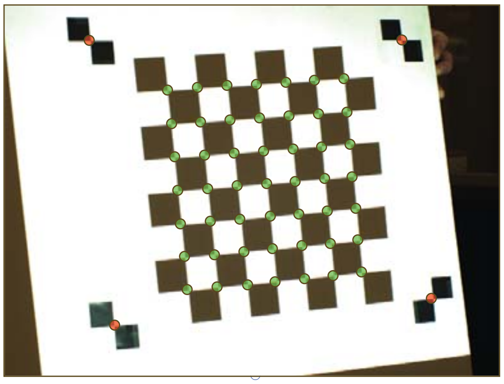


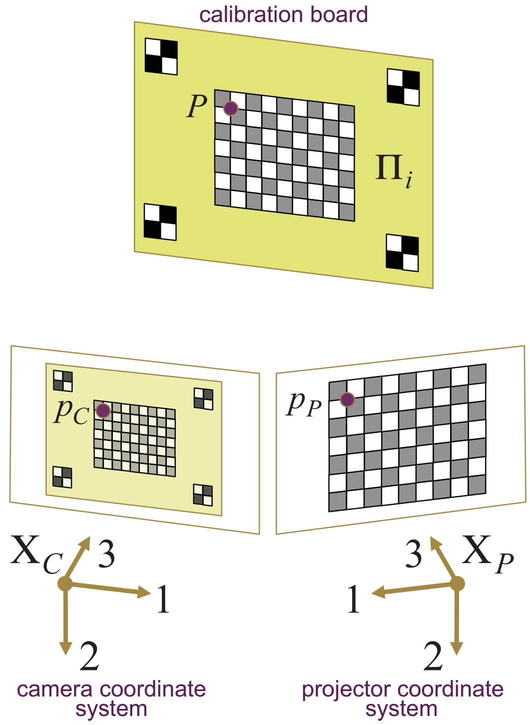


Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image
- Use fidicials to find 3D calibration plane
- Project checkerboard on calibration board
- Find ray-plane intersection for each corner

Projector Calibration

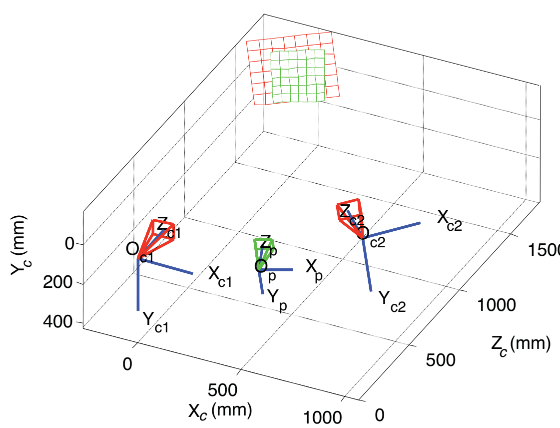




Calibration Procedure

- Consider projector an inverse camera (maps intensities to 3D rays)
- Identify printed fiducials in each image
- Use fidicials to find 3D calibration plane
- Project checkerboard on calibration board
- Find ray-plane intersection for each corner
- Use 2D→3D correspondences to estimate intrinsic/extrinsic projector calibration (and radial distortion model)

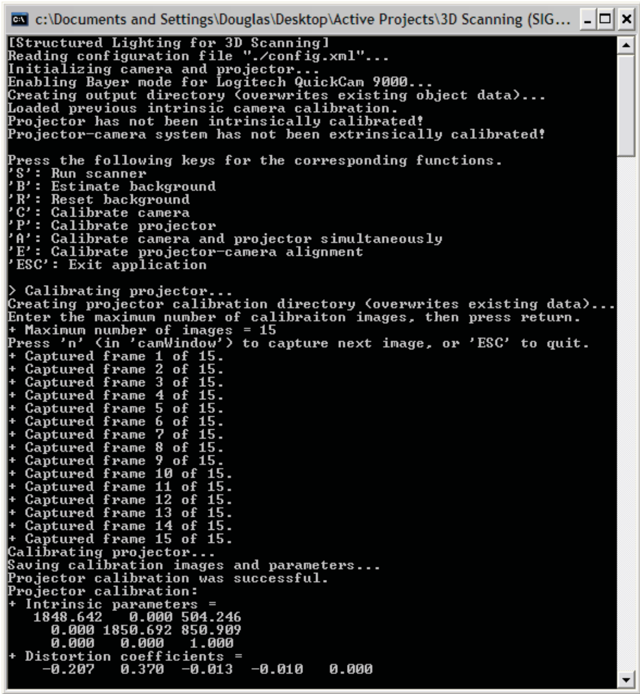
Projector-Camera Calibration Results

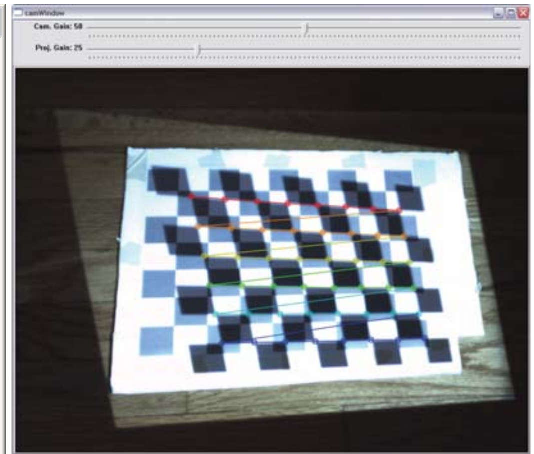


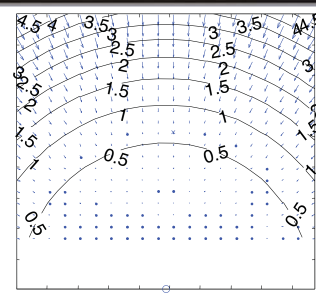
Projector-Camera Calibration Results

- Implemented complete toolbox for projector-camera calibration (available for *Matlab* and *OpenCV*)
- Sufficient accuracy for structured lighting applications
- Software and documentation available on the course website

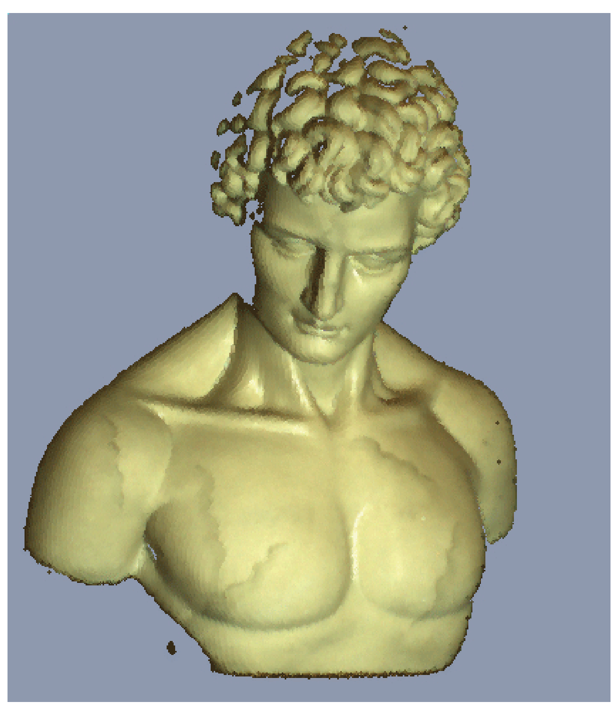
Demo: Projector Calibration in OpenCV



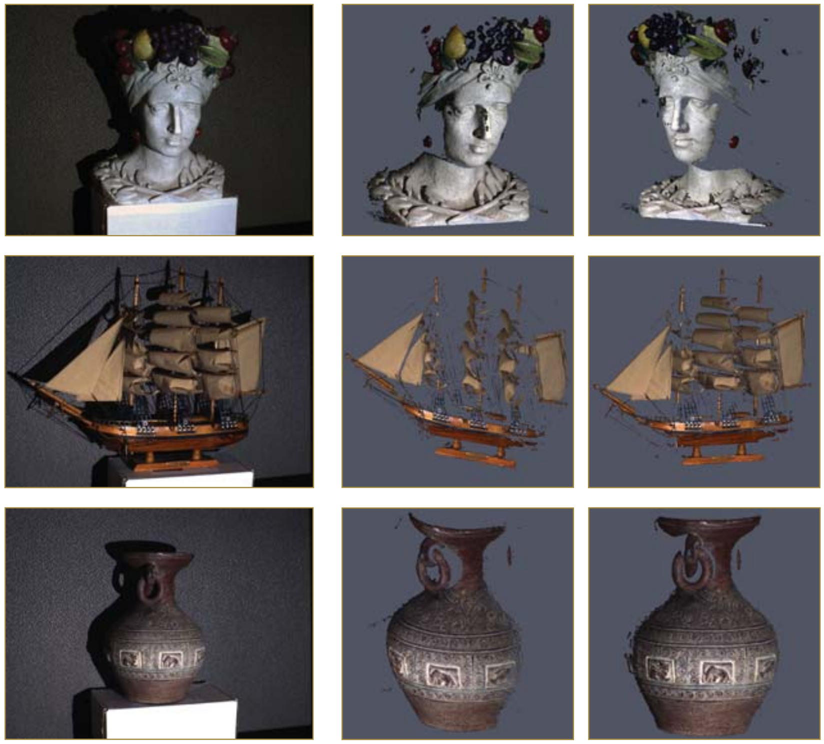




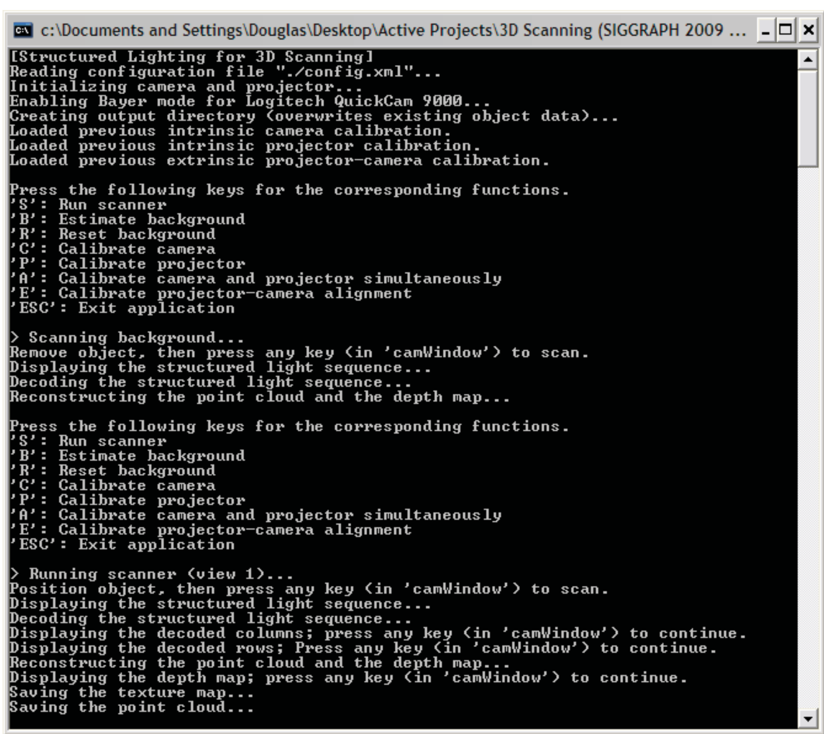
Structured Lighting Reconstruction Results



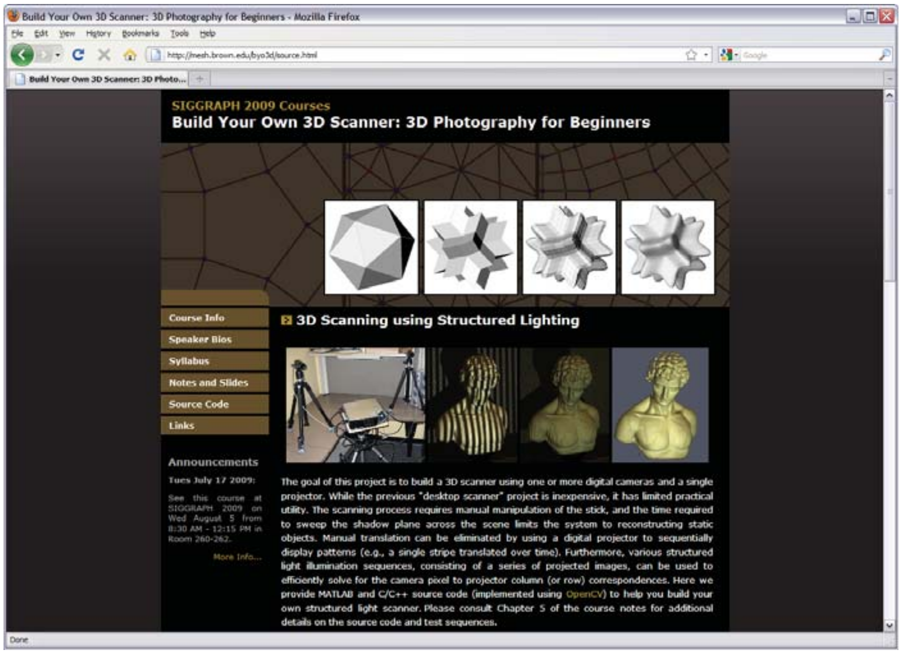
Additional Reconstruction Examples



Demo: Putting it All Together

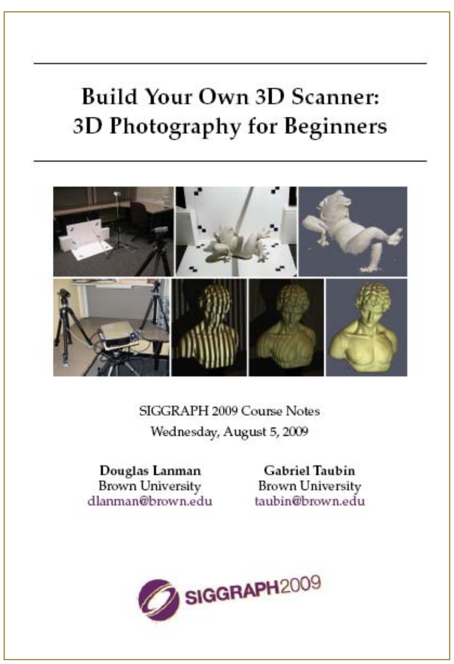


How to Get the Source Code



http://mesh.brown.edu/byo3d

For More Details

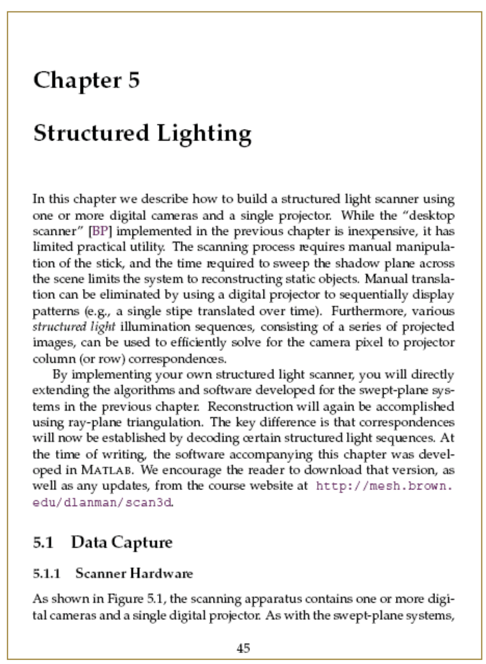

Build Your Own 3D Scanner:
3D Photography for Beginners

SIGGRAPH 2009 Course Notes
Wednesday, August 5, 2009

Douglas Lanman
Brown University
dlanman@brown.edu

Gabriel Taubin
Brown University
taubin@brown.edu

SIGGRAPH2009


Chapter 5

Structured Lighting

In this chapter we describe how to build a structured light scanner using one or more digital cameras and a single projector. While the "desktop scanner" [BP] implemented in the previous chapter is inexpensive, it has limited practical utility. The scanning process requires manual manipulation of the stick, and the time required to sweep the shadow plane across the scene limits the system to reconstructing static objects. Manual translation can be eliminated by using a digital projector to sequentially display patterns (e.g., a single stipe translated over time). Furthermore, various *structured light* illumination sequences, consisting of a series of projected images, can be used to efficiently solve for the camera pixel to projector column (or row) correspondences.

By implementing your own structured light scanner, you will directly extending the algorithms and software developed for the swept-plane systems in the previous chapter. Reconstruction will again be accomplished using ray-plane triangulation. The key difference is that correspondences will now be established by decoding certain structured light sequences. At the time of writing, the software accompanying this chapter was developed in MATLAB. We encourage the reader to download that version, as well as any updates, from the course website at http://mesh.brown.edu/dlanman/scan3d.

5.1 Data Capture

5.1.1 Scanner Hardware

As shown in Figure 5.1, the scanning apparatus contains one or more digital cameras and a single digital projector. As with the swept-plane systems,

45

http://mesh.brown.edu/byo3d

Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

Clase 8 : Jueves Abril 11

Gabriel Taubin

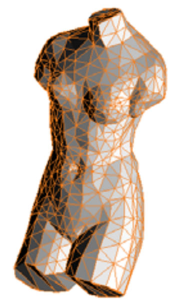
Brown University

Course Schedule

- Structured Lighting
- Projector Camera Calibration
- ***Surface Reconstruction from Point Clouds***
- Elementary Mesh Processing
- Related Projects
- Conclusion / Q & A

Surface Representations

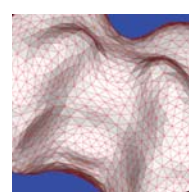
- Surfaces in Mathematics
  - Parametric $S = \{p = x(u) : u = (u_1, u_2) \in R^2\}$
  - Implicit $S = \{p : f(p) = 0\} \quad f : V \to R \quad V \subset R^3$ (level set)
- We can only operate on a surface representation
  - A data structure defined by a finite number of parameters
  - Efficient to perform certain geometric operations
- Point clouds (surfaces represented as sets of samples)
  - Positions
  - Optional properties: normals, colors, etc



- Polygon meshes (piecewise planar surfaces)
  - vertices, edges, and faces
  - Optional properties: normals, color, texture coordinates, etc.

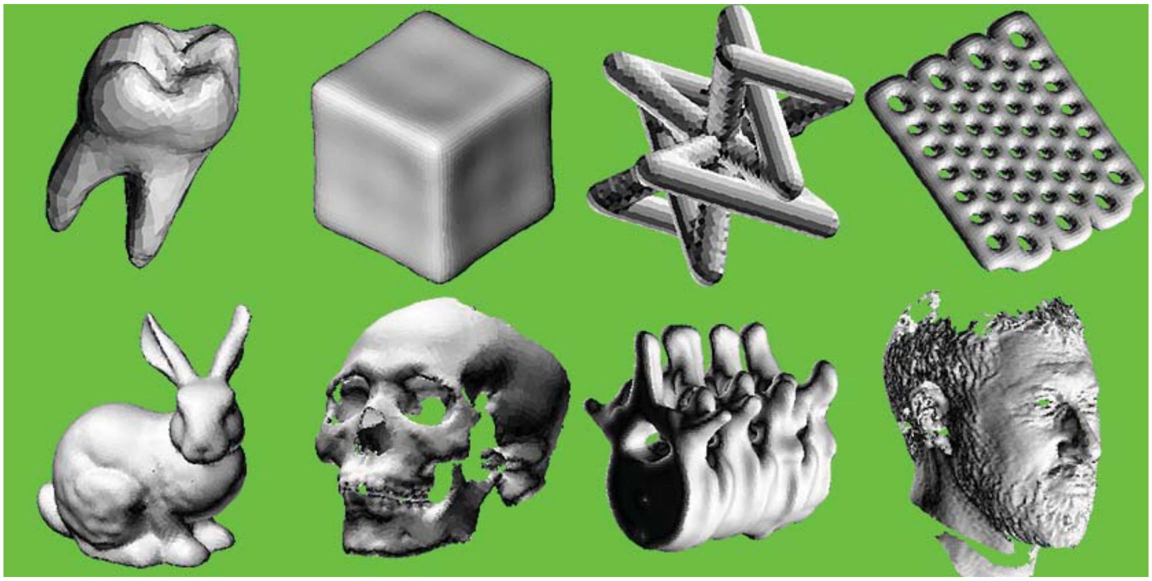
Connectivity / Watertight Surfaces

- Most applications require **connectivity information**
  - Efficient ways to find points in close proximity to each other
- Point clouds do not provide connectivity information
  - Additional data structures are needed to efficiently find neighboring points



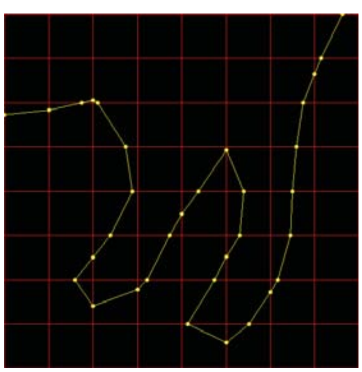
- Connectivity is explicit in polygon meshes: **edges**
- **Triangulate** the point cloud to get connectivity information
  - Find an interpolating or approximating triangle mesh
- Many applications require **watertight** surfaces: continuous closed surfaces which partition 3D space into an **inside** and an **outside**
  - Point clouds are not watertight
  - Polygon meshes may be watertight
- Will the triangulation constructed from the point cloud be watertight ?

Polygon Meshes



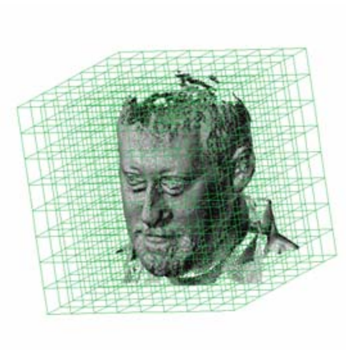
Surface Reconstruction from Point Clouds

- Every regular **Implicit surface** is watertight $S = \{p : f(p) = 0\}$
- An **Isosurface** is a polygonal approximation of an implicit function associated with a volumetric grid



Surface Reconstruction from Point Clouds

- Every regular **Implicit surface** is watertight $S = \{p : f(p) = 0\}$
- An **Isosurface** is a polygonal approximation of an implicit function associated with a volumetric grid
- **Marching Cubes** is an algorithm to compute an isosurface from an implicit surface evaluated on the vertices of a regular hexahedral grid



W.E. Lorensen, H.E. Cline. Marching Cubes:A high resolution 3D surface reconstruction algorithm. Siggraph,1987

Surface Reconstruction from Point Clouds

- Every regular **Implicit surface** is watertight $S = \{p : f(p) = 0\}$
- An **Isosurface** is a polygonal approximation of an implicit function associated with a volumetric grid
- **Marching Cubes** is an algorithm to compute an isosurface from an implicit surface evaluated on the vertices of a regular hexahedral grid
- Similar simple algorithms exists to generate isosurfaces from an implicit function evaluated on the vertices of a tetrahedral grid
- We will only discuss here approximation algorithms to fit implicit surfaces to point clouds
- Algorithms related to the Poisson Equation

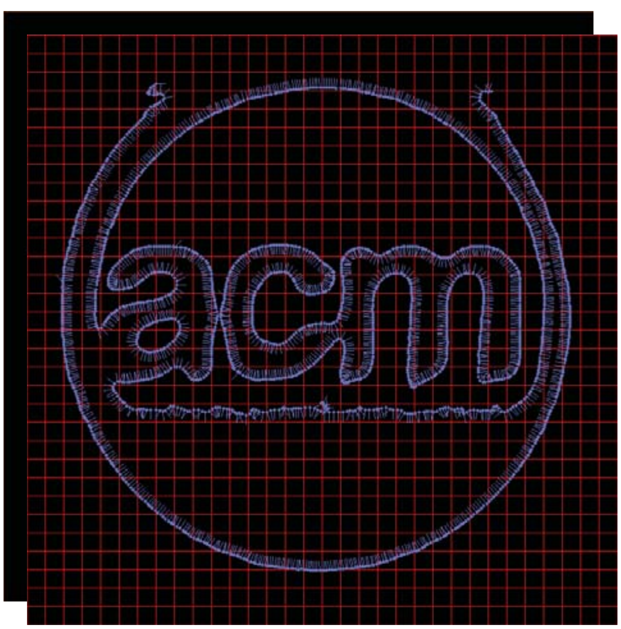
W.E. Lorensen, H.E. Cline. Marching Cubes:A high resolution 3D surface reconstruction algorithm. Siggraph,1987

M. Kazhdan, M. Bolitho, H. Hoppe. Poisson Surface Reconstruction. European Symposium on Geometry Processing, 2006

Curve Reconstruction from Point Clouds

- Oriented points

Curve Reconstruction from Point Clouds

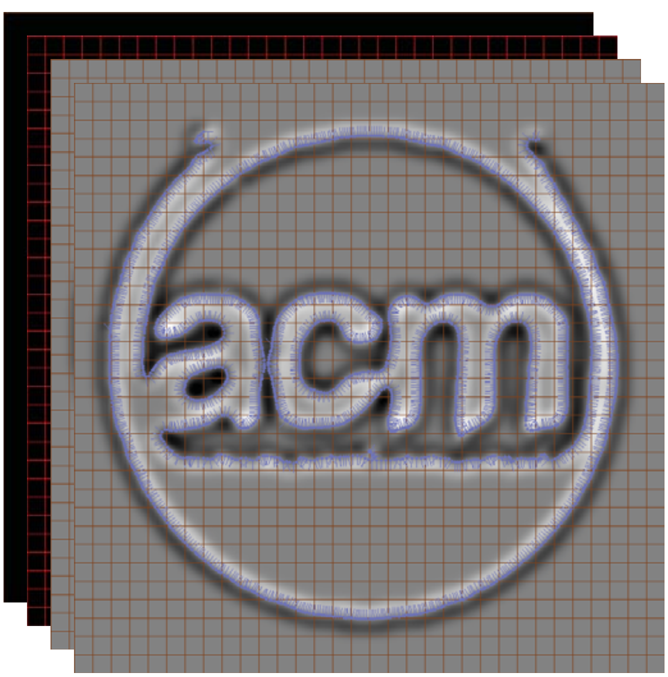


- Oriented points
- Regular grid

Curve Reconstruction from Point Clouds

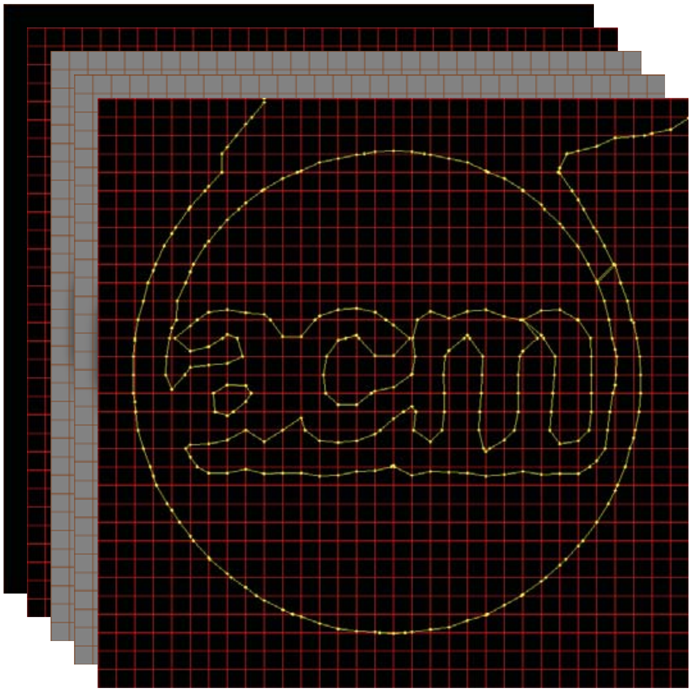
- Oriented points
- Regular grid
- Implicit function

Curve Reconstruction from Point Clouds



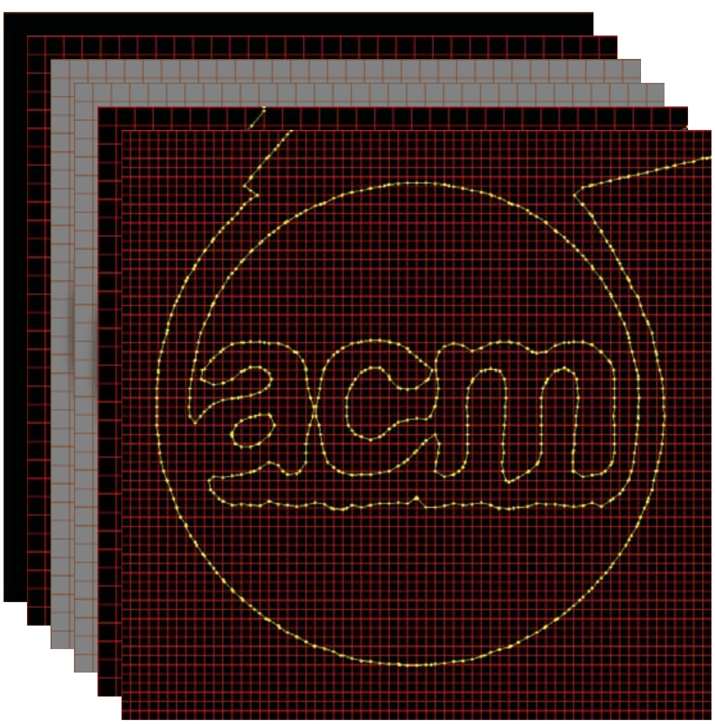
- Oriented points
- Regular grid
- Implicit function

Curve Reconstruction from Point Clouds



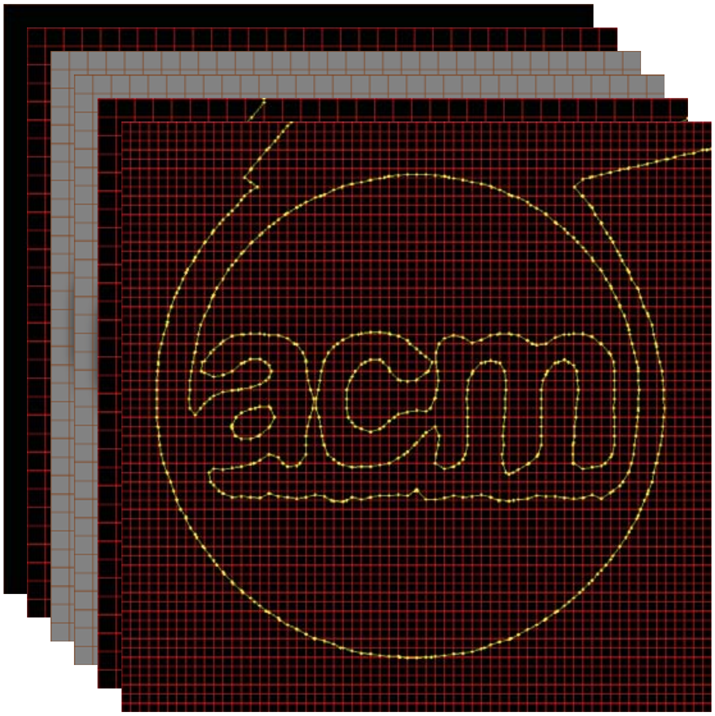
- Oriented points
- Regular grid
- Implicit function
- Isocurve

Curve Reconstruction from Point Clouds



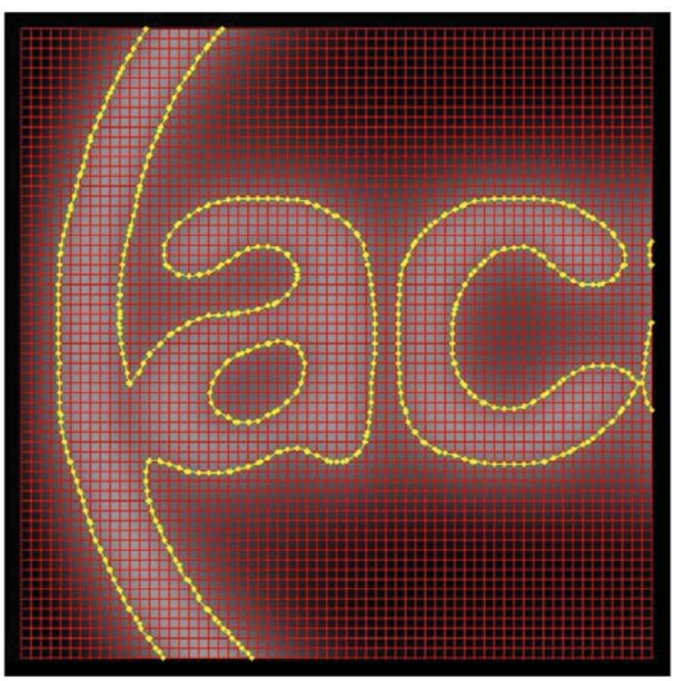
- Oriented points
- Regular grid
- Implicit function
- Isocurve
- Grid too coarse: Aliasing
- Finer grid resolves topology

Curve Reconstruction from Point Clouds



- Oriented points
- Regular grid
- Implicit function
- Isocurve
- Grid too coarse: Aliasing
- Finer grid resolves topology

IsoCurves



- Given a continuous function

$$f(x_1, x_2)$$

- Sampled on a regular grid

$$G = (V, E, C)$$

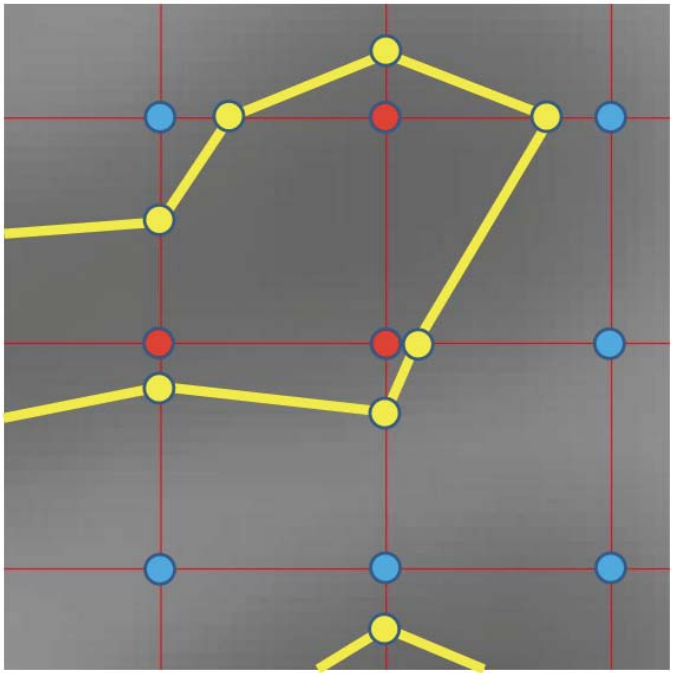
$$F = \{f_v : v \in V\}$$

- Compute a polygonal approximation of a level set

$$\mathcal{C}_\lambda = \{x : f(x) = \lambda\}$$

- Increase grid resolution if necessary

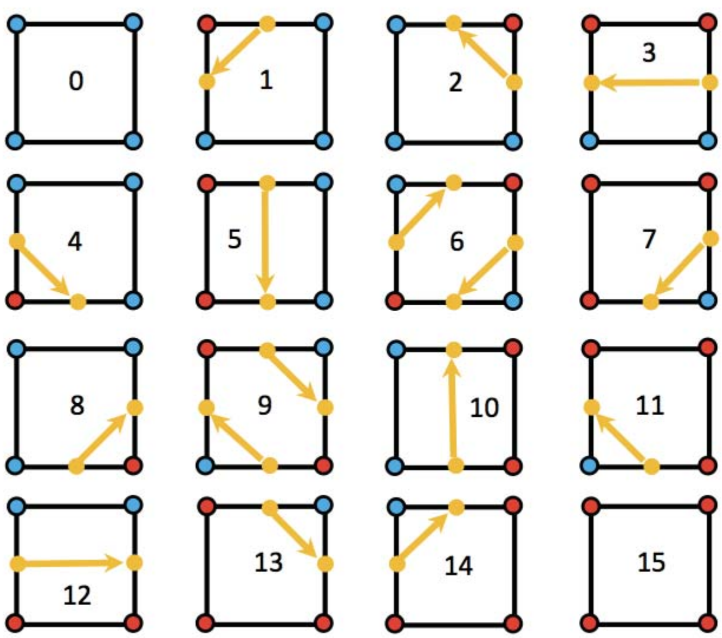
The Marching Lines Algorithm (ML)



4 STEPS

1. Determine grid vertex sign bits
2. Determine supporting grid edges
3. Compute location of Isovertices along supporting grid edges
4. Interconnect isovertices by table look-up within each cell

The Marching Lines Table



4 STEPS

1. Determine grid vertex sign bits
2. Determine supporting grid edges
3. Compute location of Isovertices along supporting grid edges
4. Interconnect isovertices by table look-up within each cell

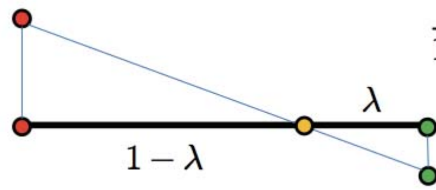
Choices for 6 & 9

The 4 Steps

1. Determine grid vertex sign bits $b_v = \begin{cases} 1 & f_v \geq 0 \\ 0 & f_v < 0 \end{cases}$

2. Determine supporting grid edges  $b_v \neq b_w$

3. Compute location of IsoVertices along supporting grid edges



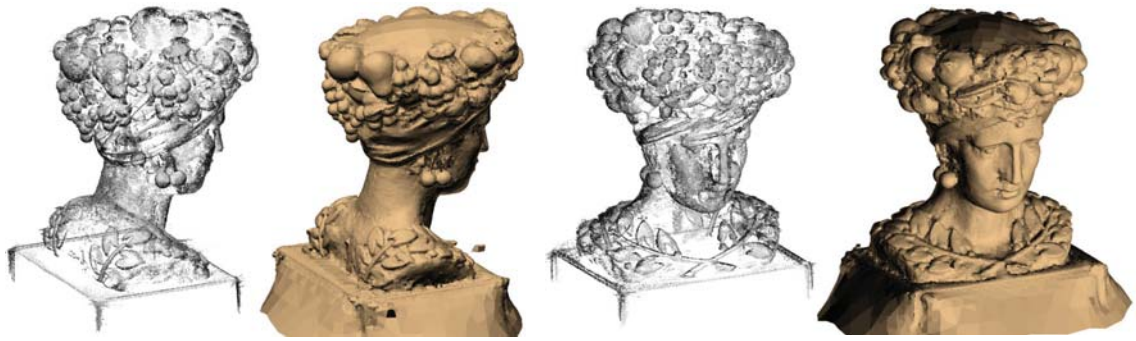
$$\frac{f_v}{1-\lambda} + \frac{f_w}{\lambda} = 0 \quad \Leftrightarrow \quad \lambda = \frac{f_w}{f_w - f_v}$$

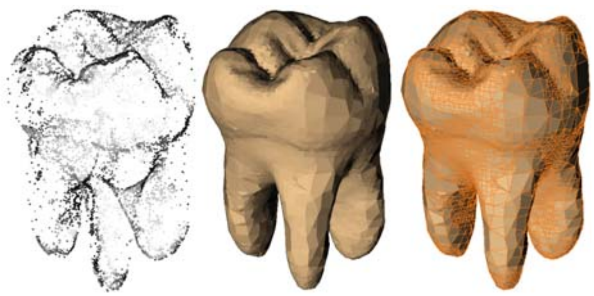
4. Interconnect IsoVertices by table look-up within each cell

Related Papers & Projects

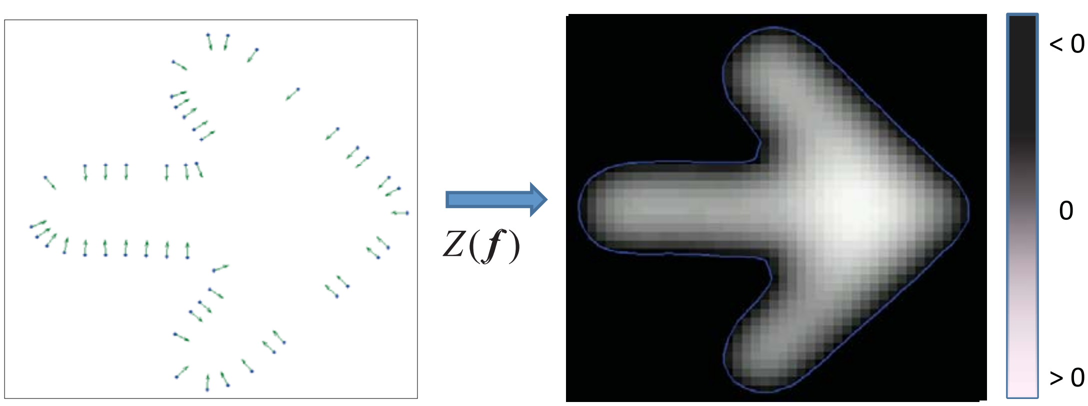
- Vector Field Isosurface-Based Reconstruction From Oriented Points, by Sibley & Taubin, Siggraph 2005 (Sketch).
- **Smooth Signed Distance Surface Reconstruction, by Calakli & Taubin, PG 2011 & Computer Graphics Forum 2011.**
- Smooth Signed Distance Colored Surface Reconstruction, by Calakli & Taubin, chapter in State-of-the-Art Volume on Computer Graphics, Visualization, Visual Analytics, VR and HCI, 2012.
- Accurate 3D Footwear Impression Recovery from Photographs, by Andalo, Calakli, Taubin, and Goldenstein, Proceedings of the 4th. International Conference on Imaging for Crime Detection and Prevention (ICDP-2011).
- High Resolution Surface Reconstruction from Multi-view Aerial Imagery, Calakli, Ulusoy, Restrepo, Mundy & Taubin, 3DIMPVT 2012
- REVEAL Digital Archaeology Project
- Cuneiform Automatic Translation Project

Particularly Good at Extrapolating Missing Data





Implicit function framework



Oriented Points, ***D***
(samples from unknown surface ***S***)

Computed Implicit Surface, ***S'***

Find a scalar valued function $f : D \rightarrow \Re$, whose zero level set ***Z(f) = S'*** is the estimate for true surface ***S***

Implicit Curve and Surface Reconstruction

- Input: oriented point set:

  $D = \{ (\mathbf{p}_i, \mathbf{n}_i) \; i=1,\ldots,N\}$

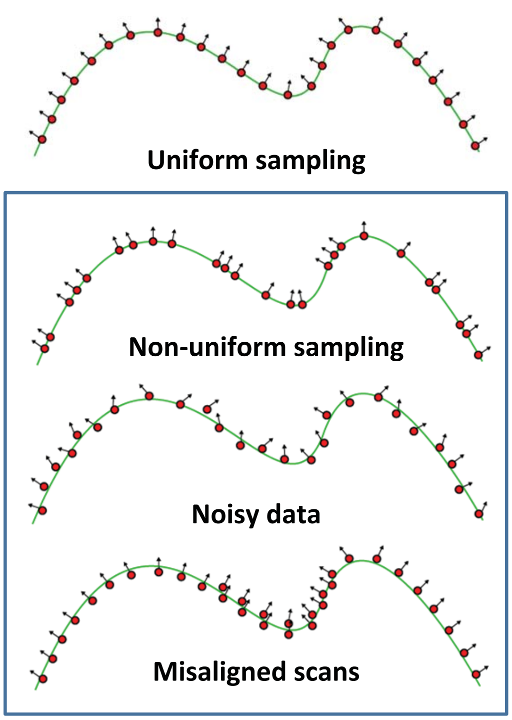
  contained in a bounding volume V
- Output: implicit surface

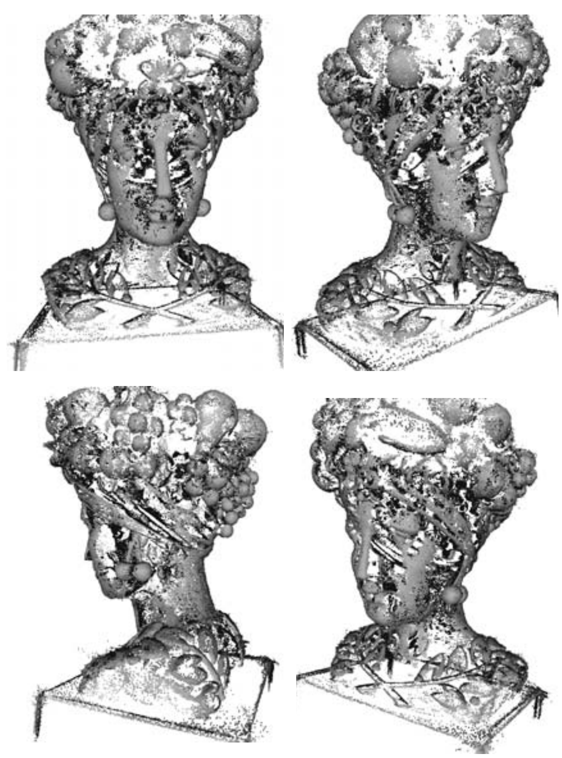
  $S = \{ \mathbf{x} \mid f(\mathbf{x}) = 0 \}$

  with the function defined on V, such that

  $f(\mathbf{p}_i) = 0 \quad \text{and} \quad \nabla f(\mathbf{p}_i) = \mathbf{n}_i \quad \forall (\mathbf{p}_i, \mathbf{n}_i) \in D$
- A family of implicit functions with a finite number of parameters has to be chosen
- Parameters must be estimated so that the conditions stated above are satisfied, if not exactly, then in the least-squares sense

Challenges





General Approaches

- Interpolating polygon meshes
  Boissonnat [1984], Edelsbrunner [1984]
  Amenta et al. [1998], Bernardini et al. [1999]
  Dey et al. [2003][2007], …
- Implicit function fitting
  Taubin [1991], Hoppe et al. [1992], Curless et al. [1996]
  Whitaker [1998], Carr et al.[2001], Davis et al. [2002],
  Ohtake et al. [2004], Turk et al. [2004], Shen et al. [2004]
  Sibley-Taubin [2005]

Poisson Surface Reconstruction

Kazhdan et al. [2006]

Manson et al. [2008]

Poisson Surface Reconstruction

1. Extend oriented points to continuous vector field $v(p)$ defined on the whole volume, so that

$$v(\mathbf{p_i}) \approx n_{\mathrm{i}}$$

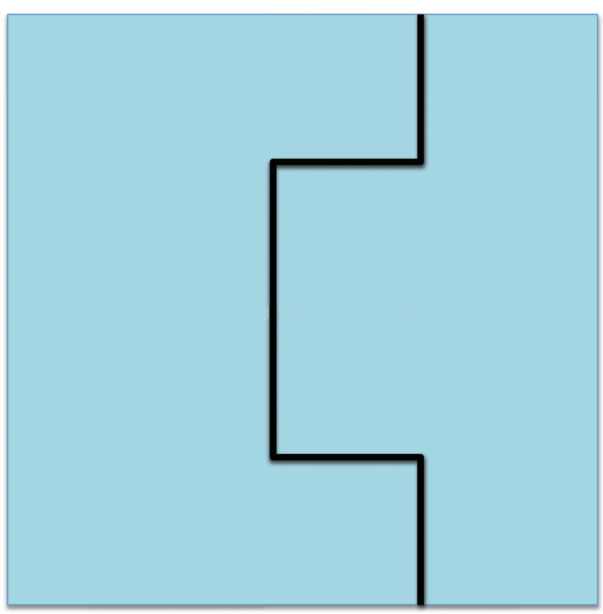
2. Integrate vector field, by minimizing

$$\int_V \|\nabla f(p) - v(p)\|^2 \, dp$$

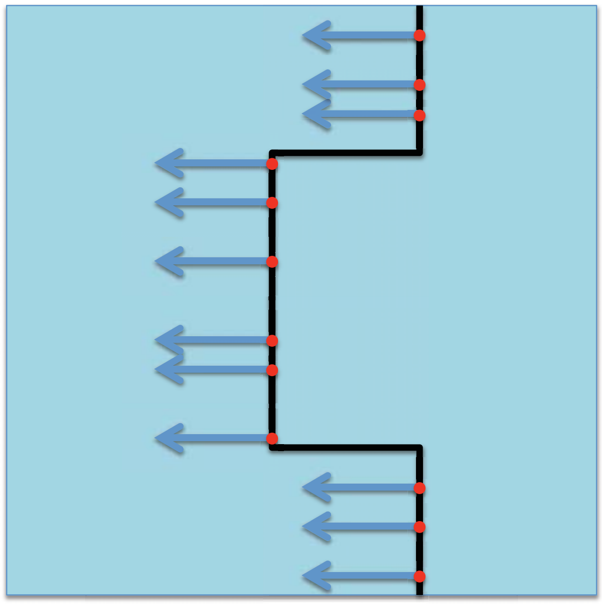
3. Determine isolevel, by minimizing

$$\sum_{i=1}^{N} (f(\mathbf{p_i}) - f_0)^2$$

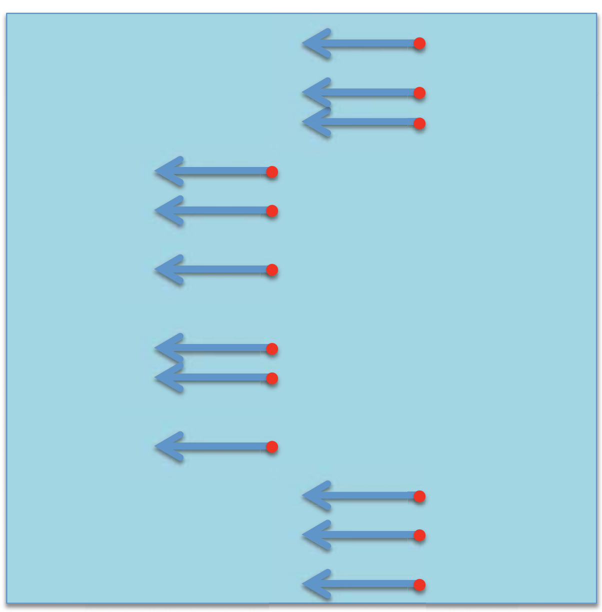
Main problem with this approach



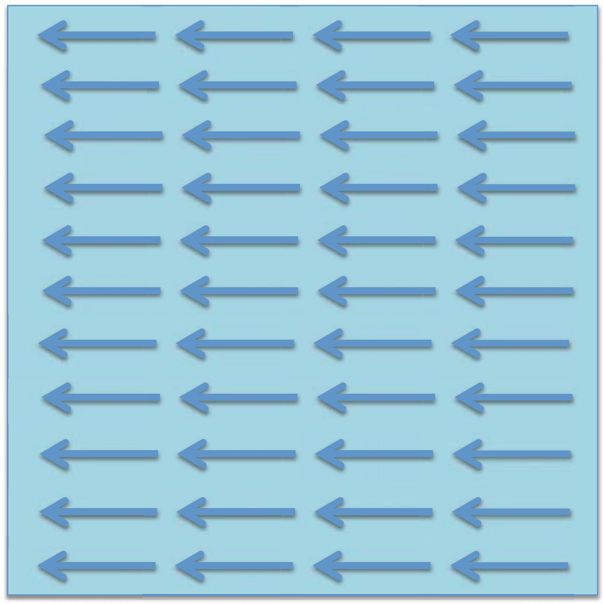
Main problem with this approach



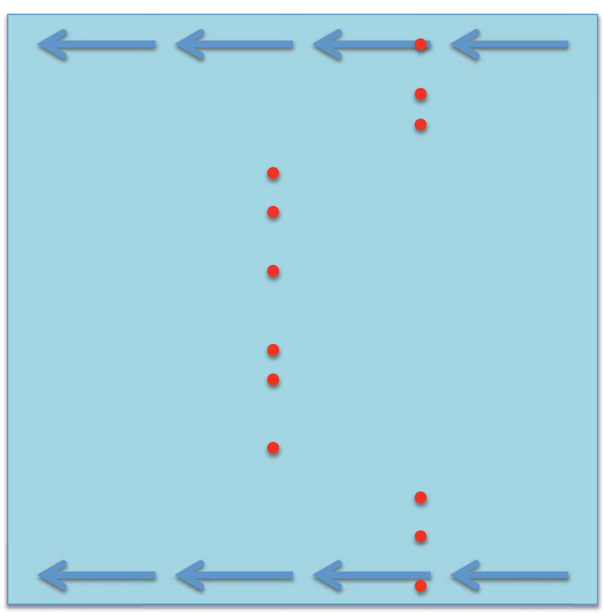
Main problem with this approach



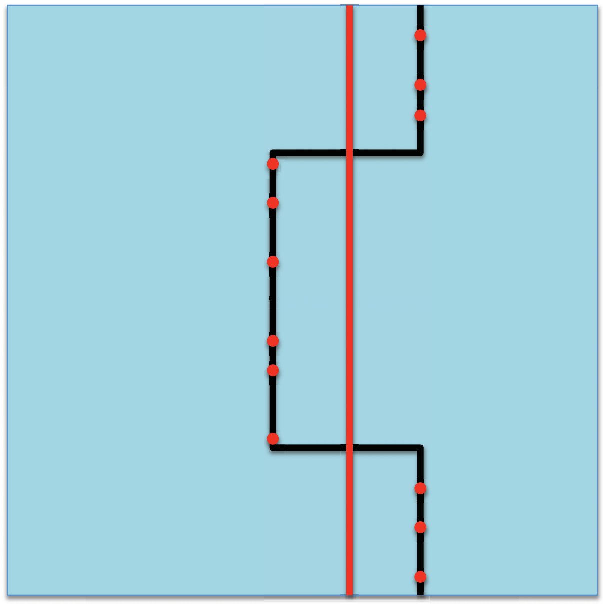
Main problem with this approach



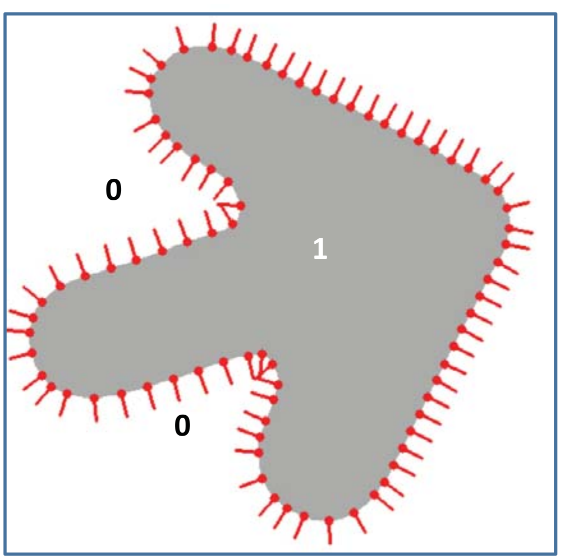
Main problem with this approach



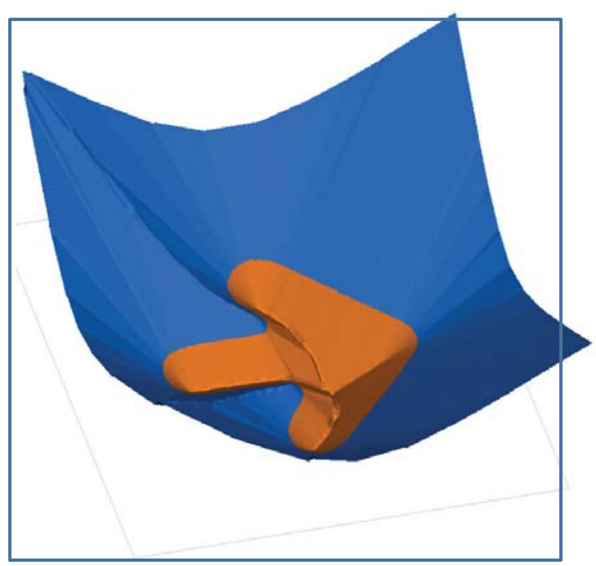
Main problem with this approach



What kind of implicit function?



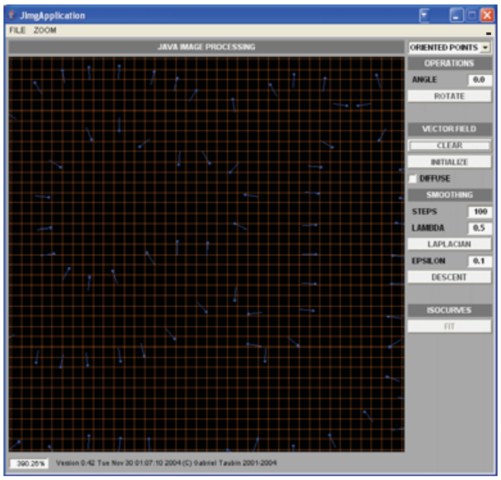
Indicator Function

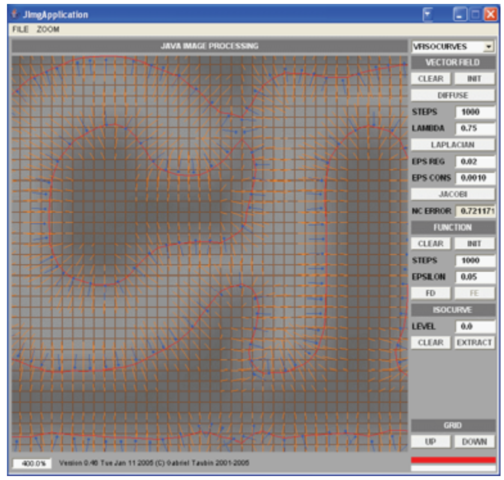


Smooth Signed Distance Function

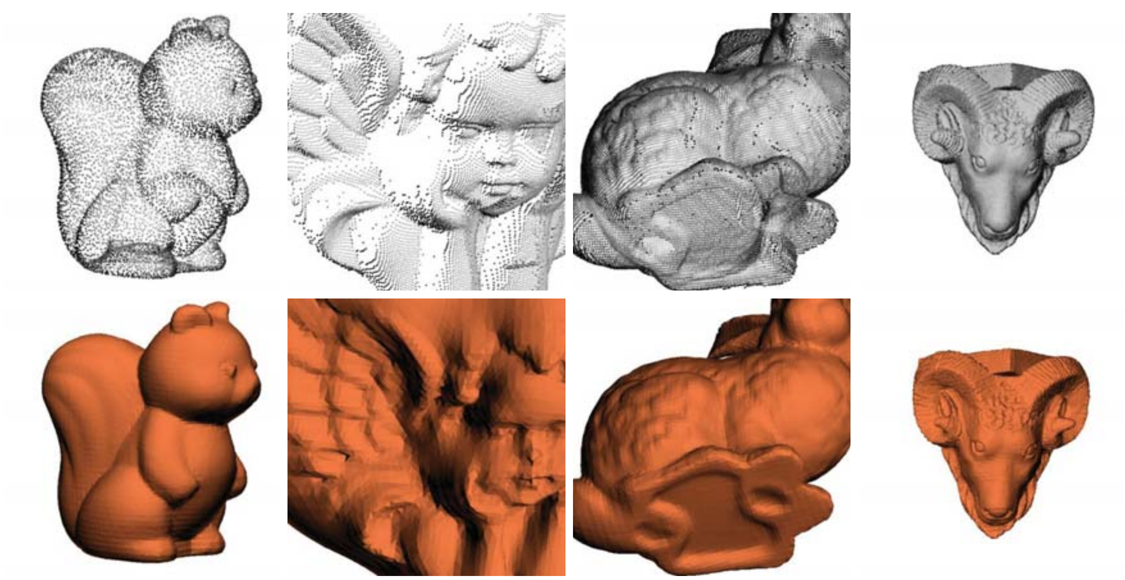
Vectorfield Isosurface-Based Reconstruction From Oriented Points
P. Sibley and G. Taubin [Siggraph 2005 Sketch]

- Surface reconstruction from cloud of oriented points
- Implicit representation can deal with missing data
- Rather than fitting analytic function (RBFs, etc), and then extract isosurface for visualization, fit isosurface directly to data

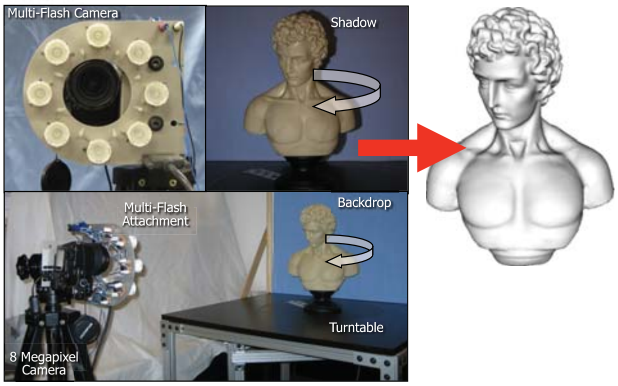


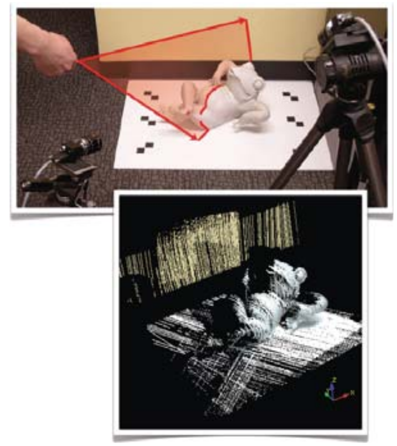


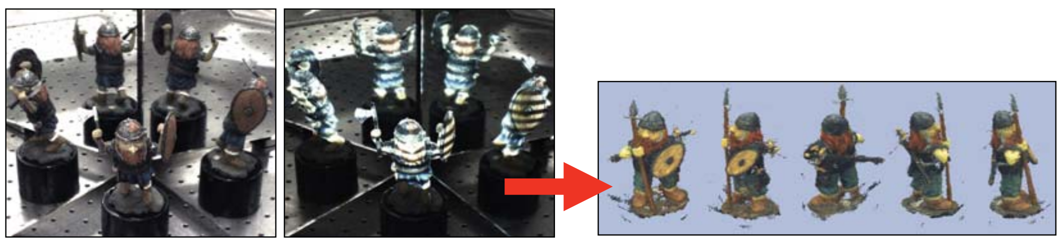
Squirrel(9K) Angel(24K) Bunny(35K) Ram(678K)



Some Methods to Capture 3D Point Clouds







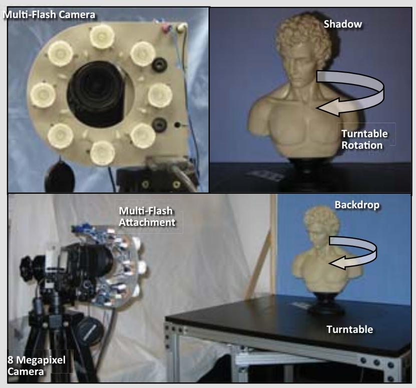
Beyond Silhouettes: Surface Reconstruction using Multi-Flash Photography

D. Crispell, D. Lanman, P. Sibley, Y. Zhao and G. Taubin [3DPVT 2006]

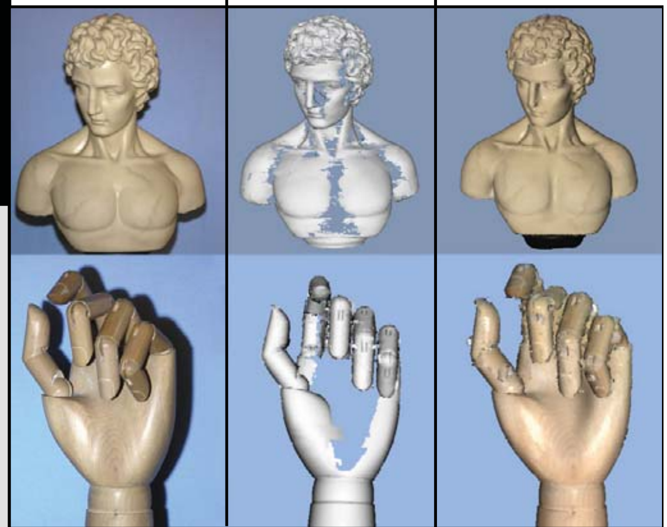
Multi-Flash 3D Photography: Capturing the Shape and Appearance of 3D Objects

A new approach for reconstructing 3D objects using shadows cast by depth discontinuities, as detected by a multi-flash camera. Unlike existing stereo vision algorithms, this method works *even with plain surfaces*, including unpainted ceramics and architecture.

Data Capture: A turntable and a digital camera are used to acquire data from 670 viewpoints. For each viewpoint, we capture a set of images using illumination from four different flashes. Future embodiments will include a small, inexpensive **handheld multi-flash camera**.



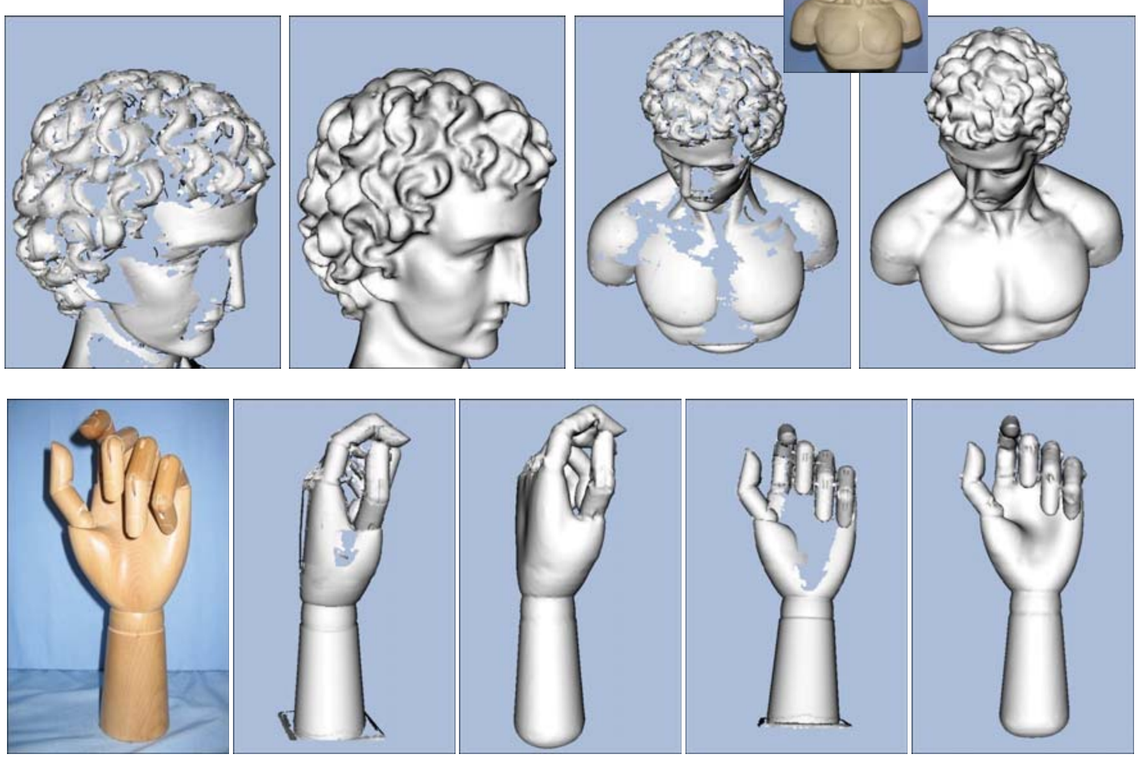
Multi-Flash Turntable Sequence: Input Image	**Estimated Shape: 3D Point Cloud**	**Recovered Appearance: Phong BRDF Model**

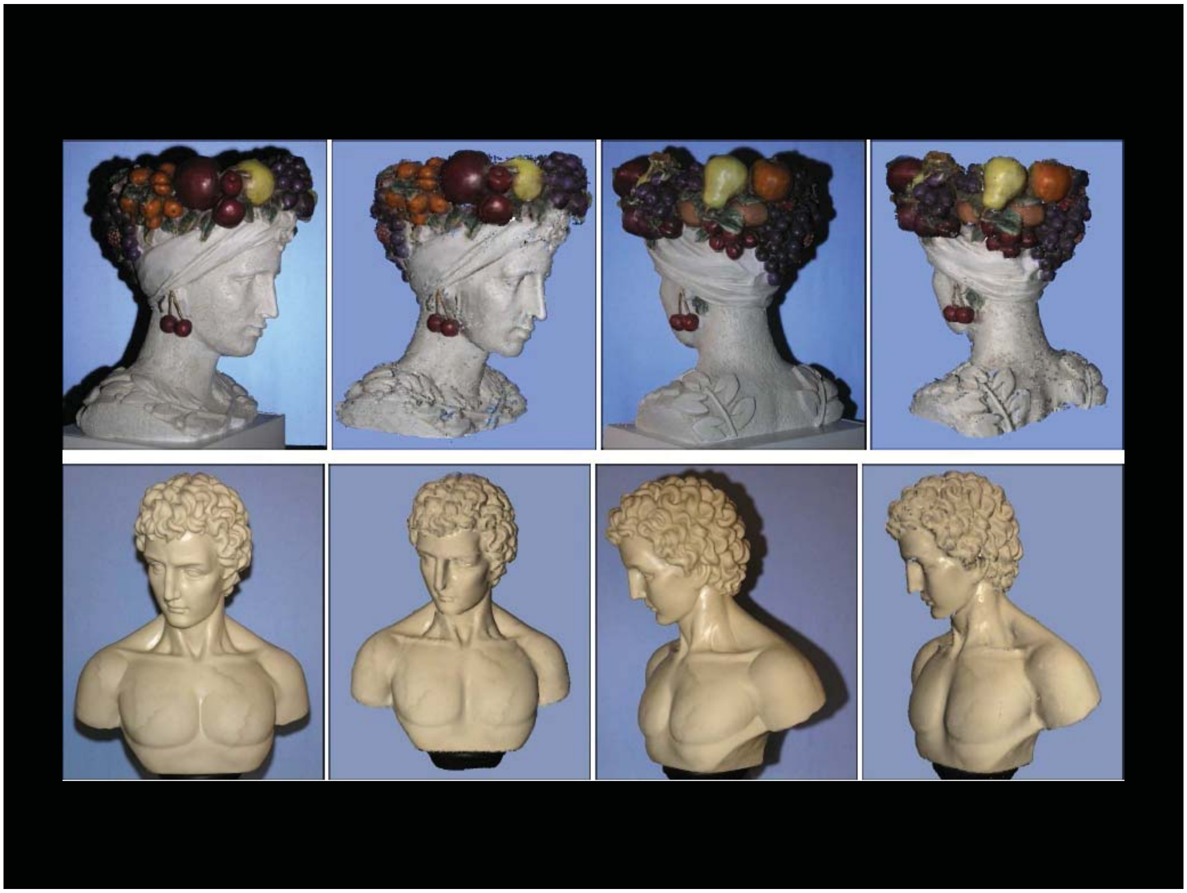


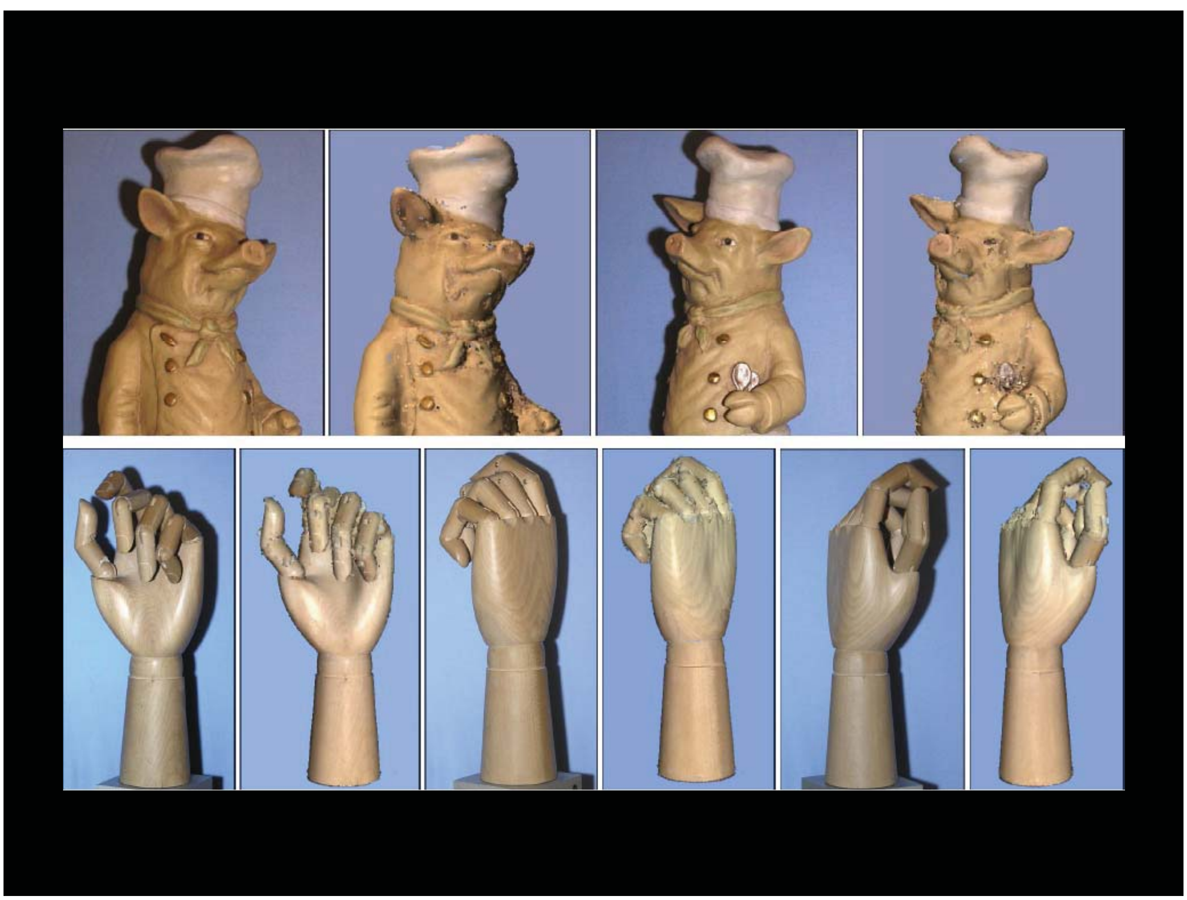
Recovering a Smooth Surface

The reconstructed point cloud can possess errors, including gaps and noise. To minimize these effects, we find an implicit surface which interpolates the 3D points. This method can be applied to any 3D point cloud, including those generated by laser scanners.

VFIso Results [2006 110x110x110 grid]







SSD Continuous Formulation

- Oriented point set:

  $D = \{ (\mathbf{p_i}, \mathbf{n_i}) \}$ sampled from a surface S

- Implicit surface:

  $S = \{ \mathbf{x} \mid f(\mathbf{x}) = 0 \}$ such that

  $f(\mathbf{p_i}) = 0$ and $\nabla f(\mathbf{p_i}) = \mathbf{n_i} \;\; \forall (\mathbf{p_i}, \mathbf{n_i}) \in D$

- Least squares energy:

$$E(f) = \sum_{i=1}^{N} f(\mathbf{p_i})^2 + \lambda_1 \sum_{i=1}^{N} \left\| \nabla f(\mathbf{p_i}) - \mathbf{n_i} \right\|^2 + \lambda_2 \int_V \| \mathrm{H} f(\mathbf{x}) \|^2 \, d\mathbf{x}$$

What does the regularization term do ?

$$\frac{\lambda_0}{N} \sum_{i=1}^{N} f(p_i)^2 + \frac{\lambda_1}{N} \sum_{i=1}^{N} \|\nabla f(p_i) - n_i\|^2 + \frac{\lambda_2}{|V|} \int_V \|Hf(x)\|^2 \, dx$$

$$Hf(x) = \left[\frac{\partial \nabla f(x)}{\partial x_1} \; \frac{\partial \nabla f(x)}{\partial x_2} \; \frac{\partial \nabla f(x)}{\partial x_3} \right]$$

- Near data points: since the data terms dominate, the function approximates the signed distance
- Away from data points: the regularization term dominates and forces the gradient to be smooth and close to constant

Role of each energy term

$$E(f,v,M) = \sum_{i=1}^{N} f(\mathbf{p_i})^2 + \lambda_1 \sum_{i=1}^{N} \left\| v(\mathbf{p_i}) - \mathbf{n_i} \right\|^2 + \lambda_2 \int_V \| M(\mathbf{x}) \|^2 \, d\mathbf{x}$$

Quadratic energy in f, v, and M

If f, v, and M are **linear functions of the same parameters**, then the minimization reduces to a least squares problem

Linear families of functions

$$f(x) = \sum_{\alpha \in \Lambda} f_\alpha \phi_\alpha(x) = \Phi(x)^t F$$

- Popular Smooth Basis Functions
  - Monomials [Taubin'91]
  - Radial basis functions [Carr et al., '01],
  - Compactly supported basis functions [Othake et al. '04],
  - Trigonometric polynomials [Kazhdan et al. '05],
  - B-splines [Kazhdan et al., 06],
  - Wavelets [Manson et al. '08],

Non-homogenous, Quadratic energy

$$E(F) = F^t AF - 2b^t F + c$$

Global minimum

$$AF = b$$

We can use Independent Discretizations

- Hybrid FE/FD discretization
  - Trilinear interpolant for the function $f(\mathbf{x})$
  - Primal finite differences for the gradient $\nabla f(\mathbf{x})$
  - Dual finite differences for the Hessian $Hf(\mathbf{x})$

- As long as $f(\mathbf{x})$, $\nabla f(\mathbf{x})$, and $Hf(\mathbf{x})$ are written as a linear combinations of **the same** parameter vector *F*

Non-homogenous, Quadratic energy

$$E(F) = F^t AF - 2b^t F + c$$

Global minimum

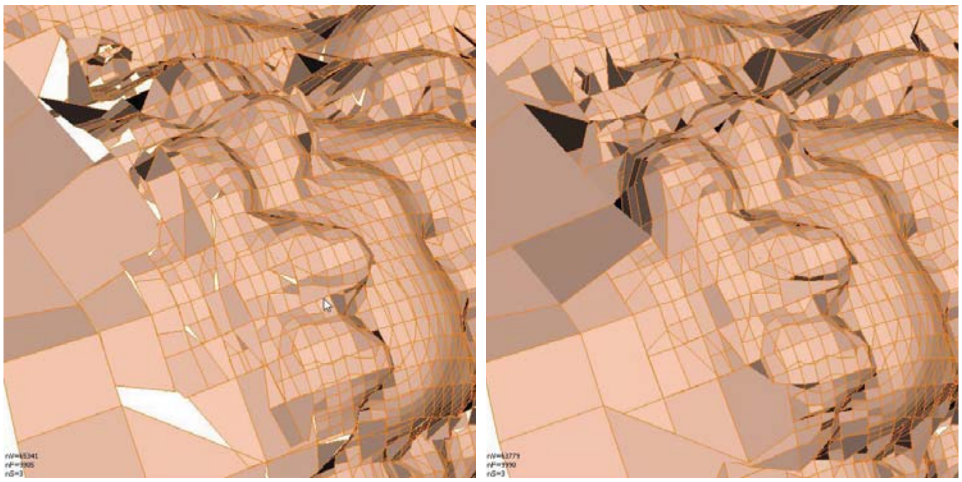
$$AF = b$$

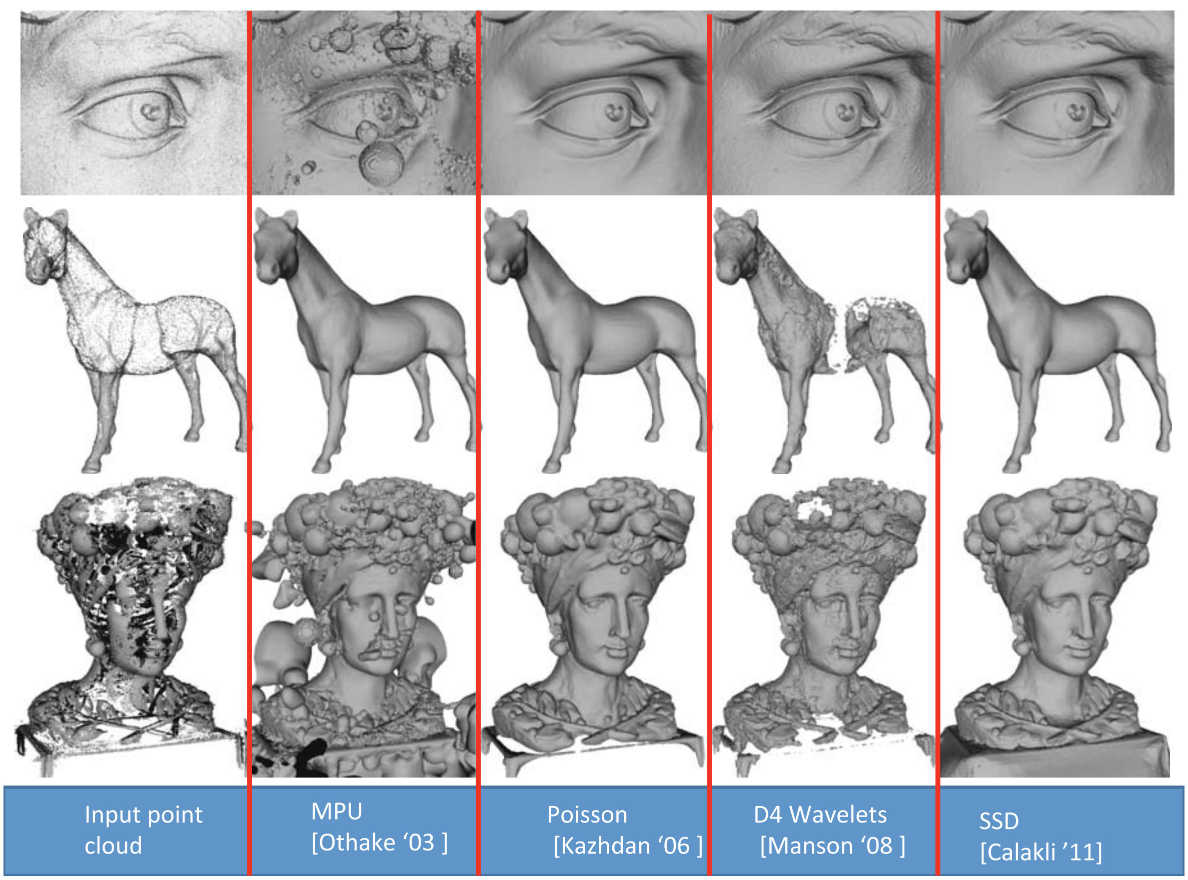
Implementation

- Primal-Dual octree data structure
- Cascading multi-grid iterative solver (conjugate gradient):
  Solve the problem on a much coarser level
  - Use the solution at that level to initialize the solution at the next level
  - Refine with the iterative solver
- Iso-surface extraction (crack-free)
  - Dual marching cubes [Schaefer 2005]

Marching Cubes on Octrees

- Non-conforming hexahedral mesh
- Results in crack problem.
- Problem solved by Dual Marching Cubes





SSD Surface Reconstruction

- Theoretical contributions:
  - Oriented point samples regarded as samples of Euclidean signed distance function
  - Reconstruction as global minimization problem
  - Yet sparse system of linear equations
- Empirical advantages:
  - Robust to noise and uneven sampling density
- Future work:
  - Streaming out-of-core implementations
  - Parallel/Multi-core/GPU implementations
  - Dynamic shapes

Course Schedule

- Structured Lighting
- Projector-Camera Calibration
- Surface Reconstruction from Point Clouds
- ➢ **Elementary Mesh Processing**
- Related Projects
- Conclusion / Q & A

Introducción a la Fotografia 3D

UBA/FCEN Marzo 27 – Abril 12 2013

Clase 9 : Viernes Abril 12

Gabriel Taubin

Brown University

BROWN

Course Schedule

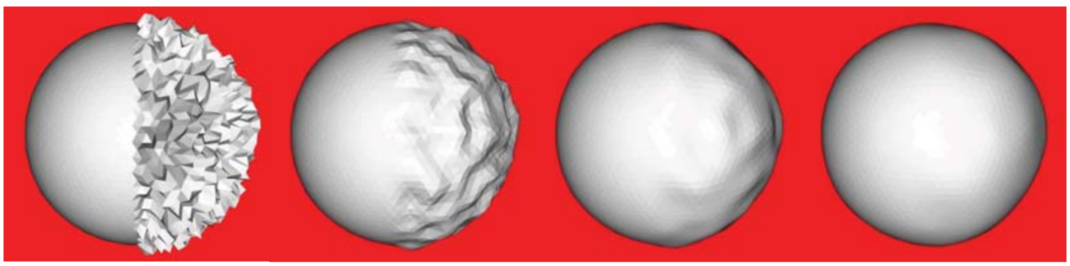
- Structured Lighting
- Projector Calibration / Structured Light Reconstruction
- Combining Point Clouds Recovered from Multiple Views
- Surface Reconstruction from Point Clouds
- ➢ ***Elementary Mesh Processing***
- Related Projects
- Conclusion / Q & A

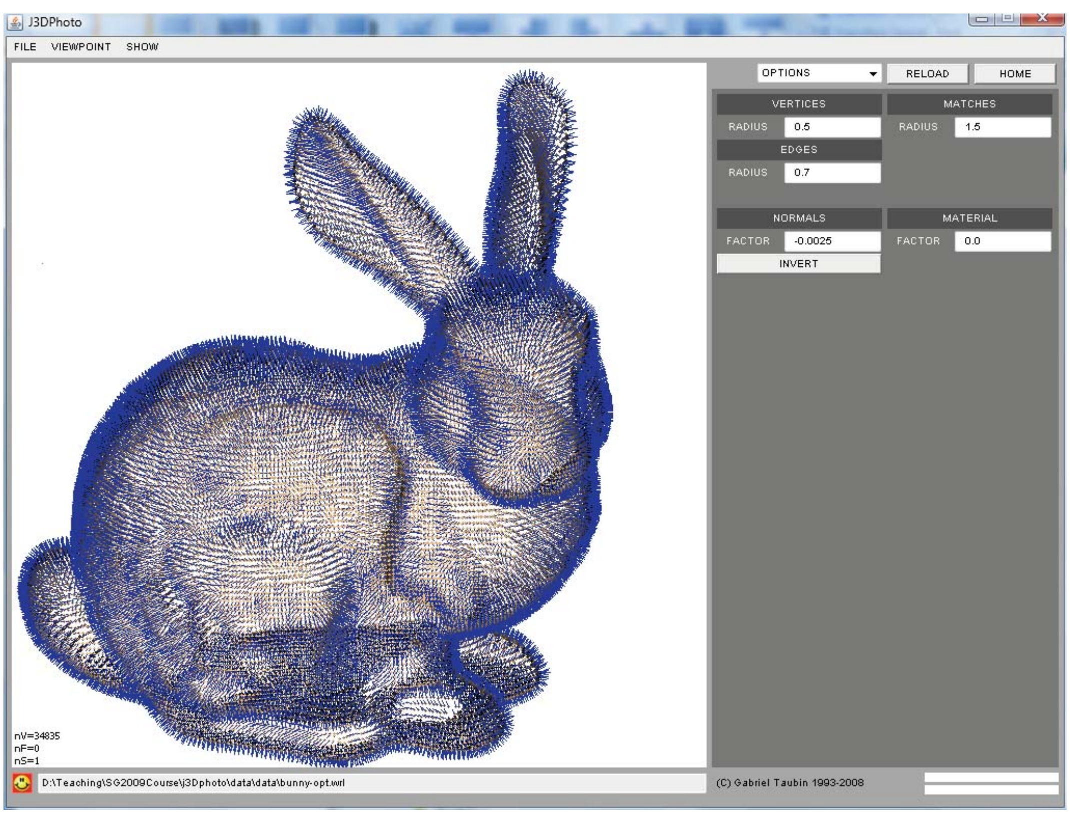
Elementary Mesh Processing

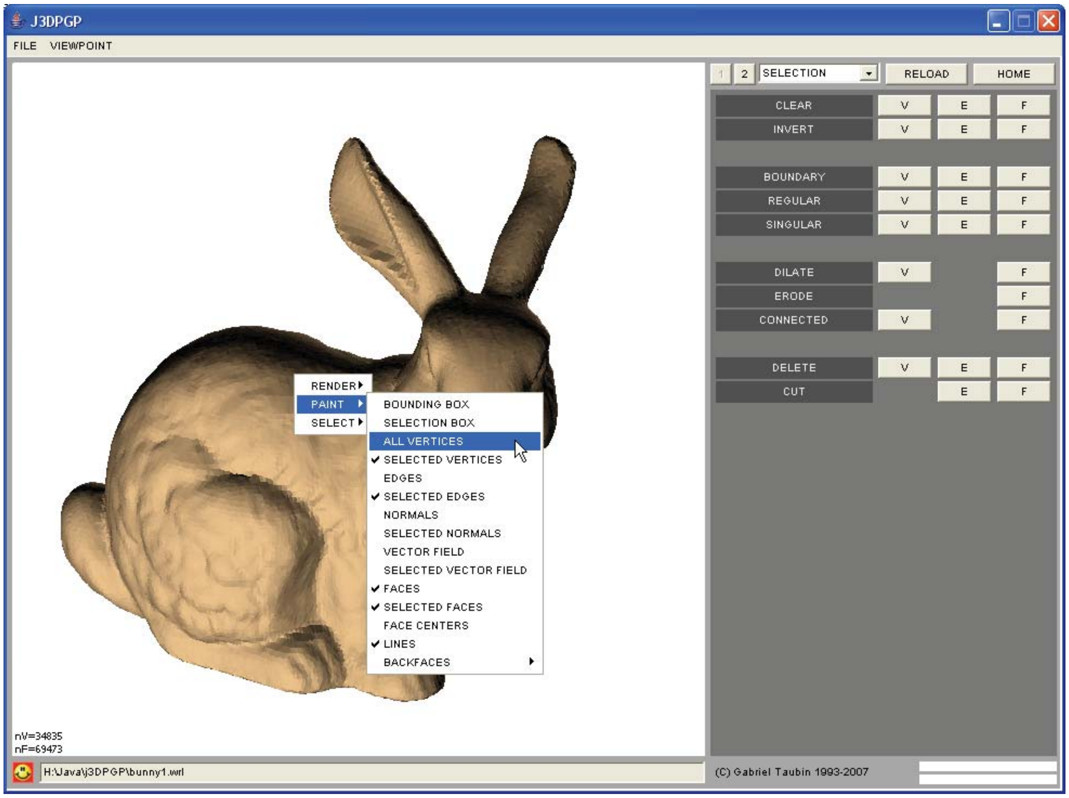
- We will talk about a few simple algorithms which can be applied interactively

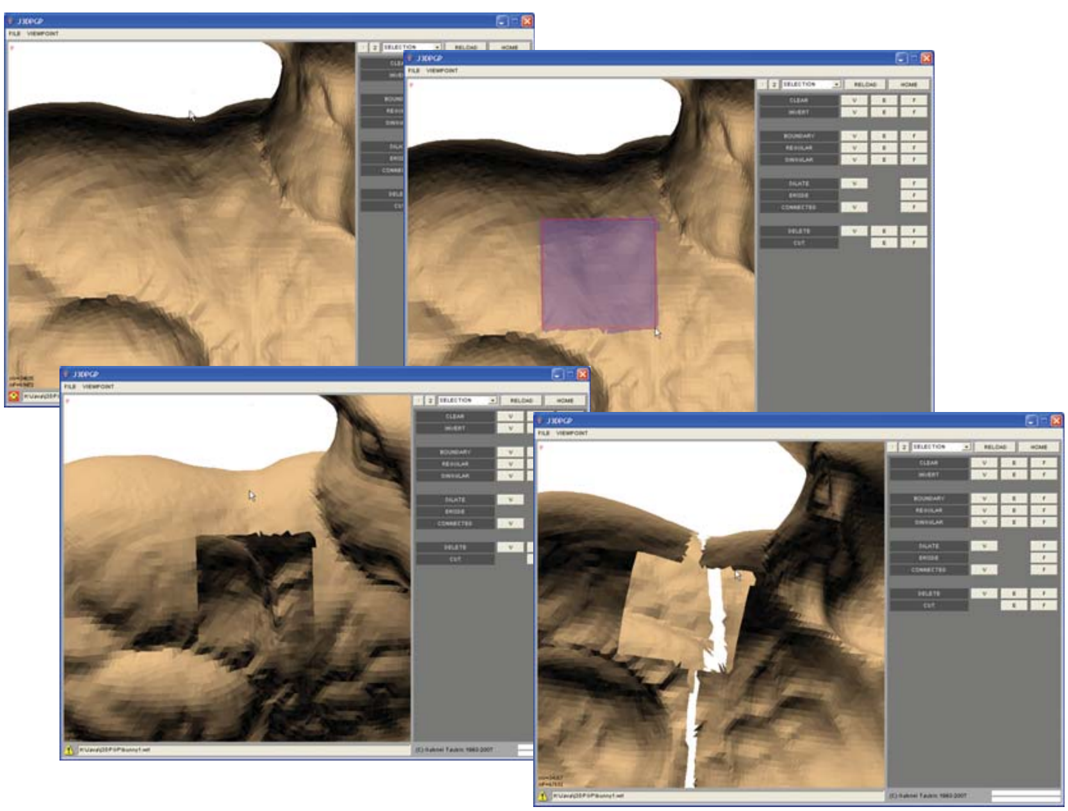
- Polygon mesh smoothing / denoising
- Polygon mesh simplification

- j3DPGP (Java 3D Photography and Geometry Processing)

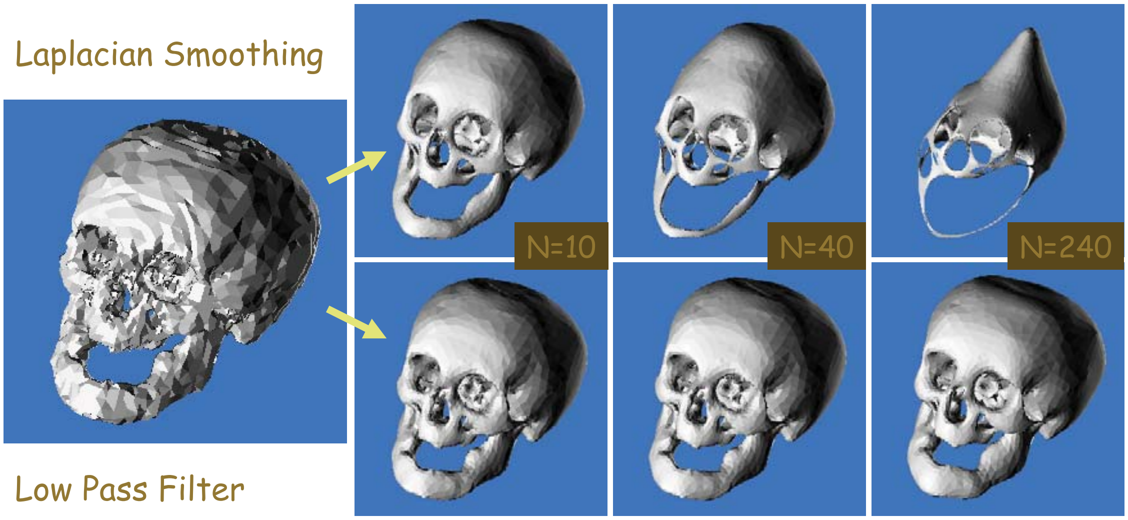





J3DPGP
FILE VIEWPOINT
2
SELECTION
RELOAD
HOME
CLEAR
INVERT
BOUNDARY
REGULAR
SINGULAR
DILATE
ERODE
CONNECTED
DELETE
CUT
V
E
F
RENDER
PAINT
SELECT
BOUNDING BOX
SELECTION BOX
ALL VERTICES
SELECTED VERTICES
EDGES
SELECTED EDGES
NORMALS
SELECTED NORMALS
VECTOR FIELD
SELECTED VECTOR FIELD
FACES
SELECTED FACES
FACE CENTERS
LINES
BACKFACES
nV=34835
nF=69473
H:\Java\j3DPGP\bunny1.wrl
(C) Gabriel Taubin 1993-2007



Polygon Mesh Smoothing / Denoising

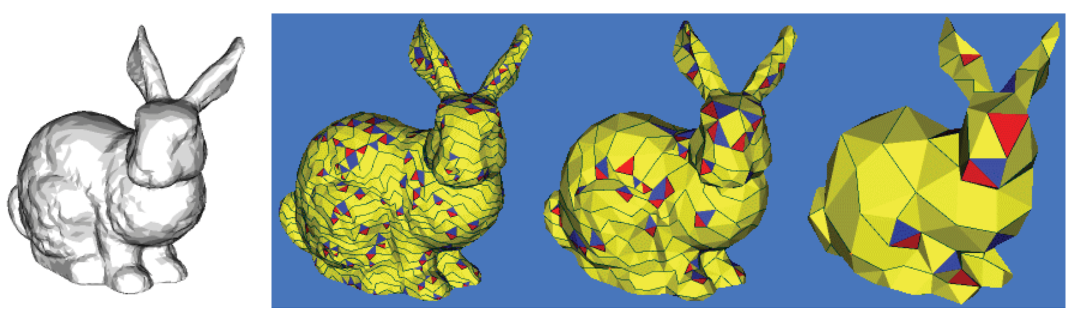


- Laplacian smoothing is the simplest smoothing/denoising algorithm
- Fix to shrinkage problem: toggle parameter sign at each iteration !

G. Taubin. A Signal Processing Approach To Fair Surface Design.
Siggraph, 1995

Poligon Mesh Simplification / Decimation

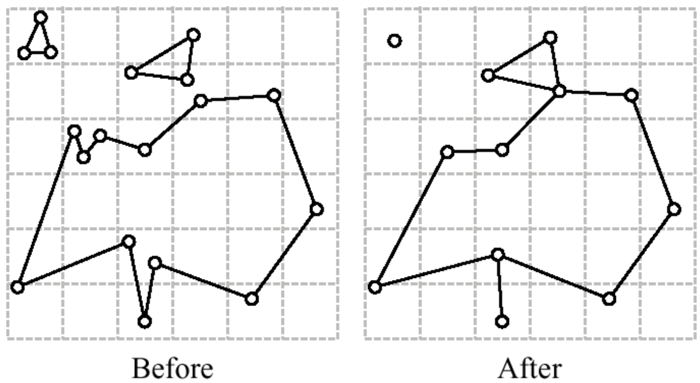
- Algorithms to reduce the number of vertices and faces while preserving geometric approximation to original shape



- Vertex clustering (Rossignac & Borrel, 1993)
- Edge Collapse (Garand Heckbert 1997; many others)

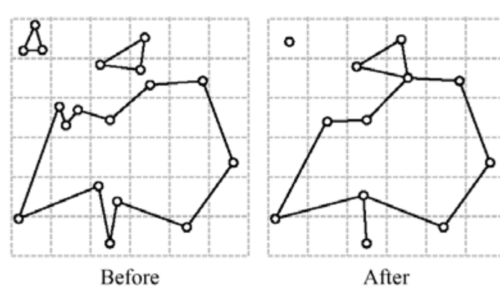
Vertex Clustering

- Quantize coordinates with respect to a bounding box
- Identify vertices with same coordinates
- Remove empty triangles



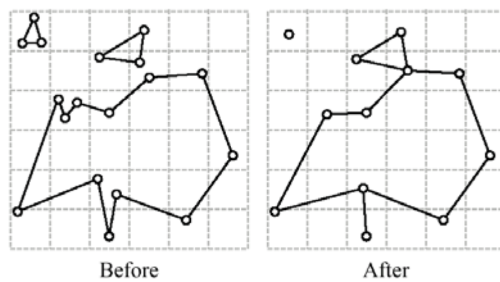
Vertex Clustering Algorithm

- Quantize coordinates with respect to bounding box
- Assign a new vertex index to each occupied cell
- Determine coordinates of new vertices
- Construct new vertex index look-up table
- Replace vertex indices in faces
- Remove empty triangles from list of faces

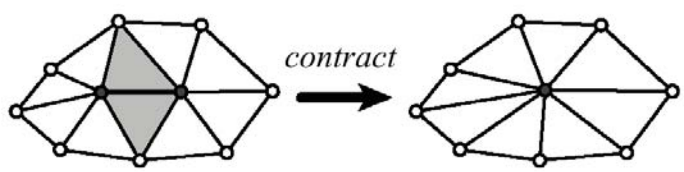


Vertex Clustering

- Advantages
  - Simple to implement
  - Works on large scenes with multiple objects
- No manifold restriction
- Disadvantages
  - Produces non-manifold meshes
  - Quality of simplified model is often not very good

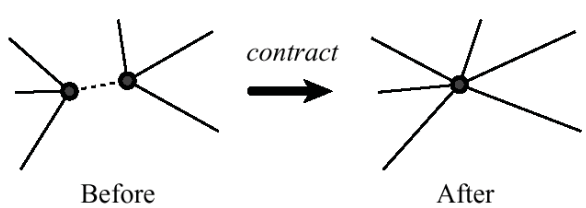


Edge Collapse

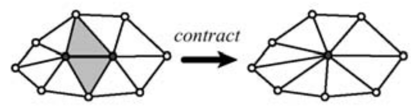


- Identify endpoints
- Determine vertex position
- Remove incident triangles
- Which edges to collapse ?
- In which order ?

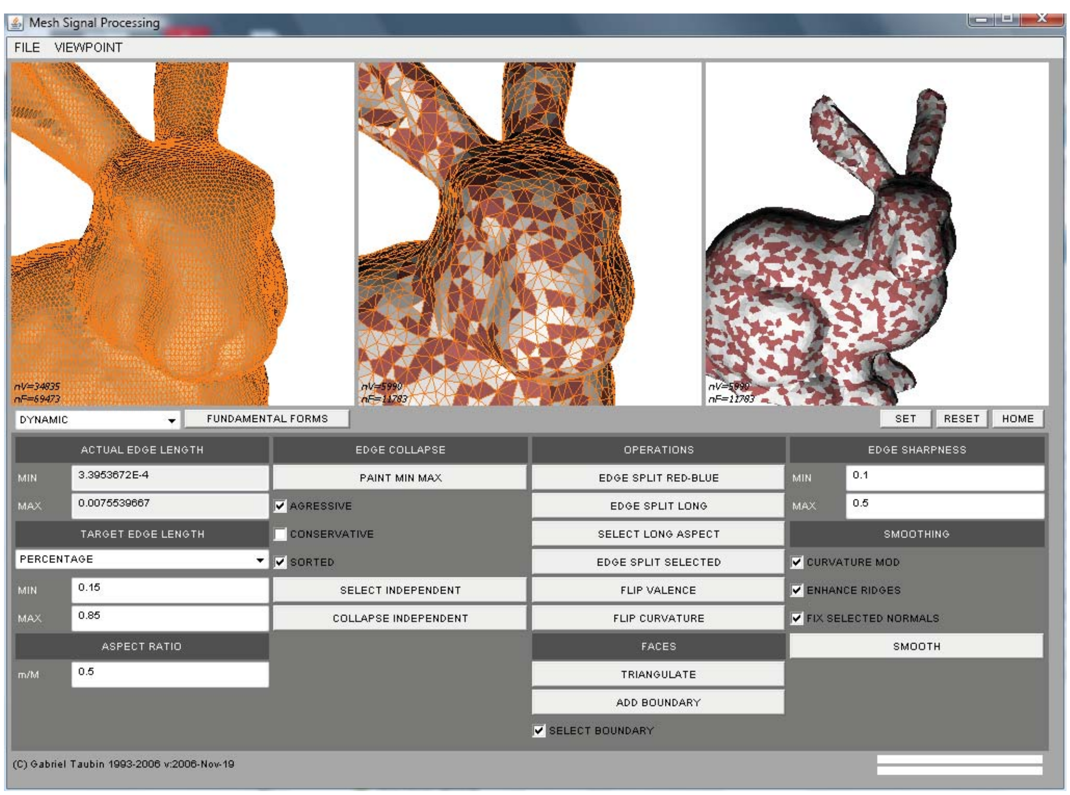
Non-existing edges



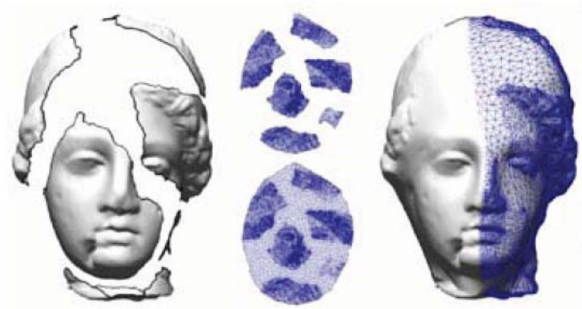
Basic Edge Collapse Algorithm



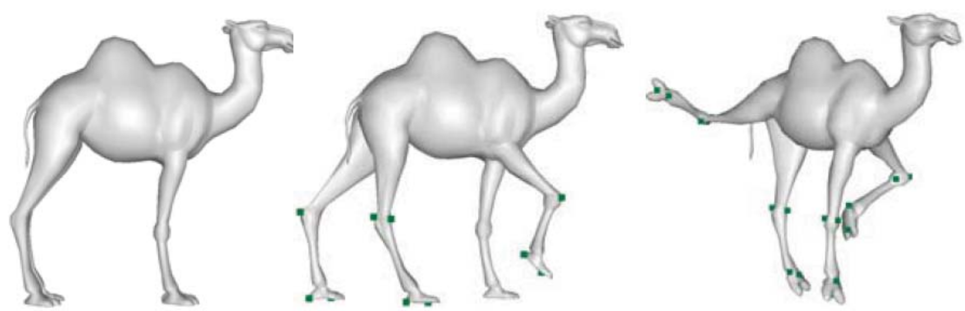
- Put all **collapsible** edges in a priority queue according to **removal error**
- While queue is not empty
  - – Delete minimum edge from queue
  - – Collapse edge
  - – Identify vertices
  - – Remove all incident edges from the queue, determine if collapsible, recompute removal error, re-insert in queue
- Need dynamic data structures



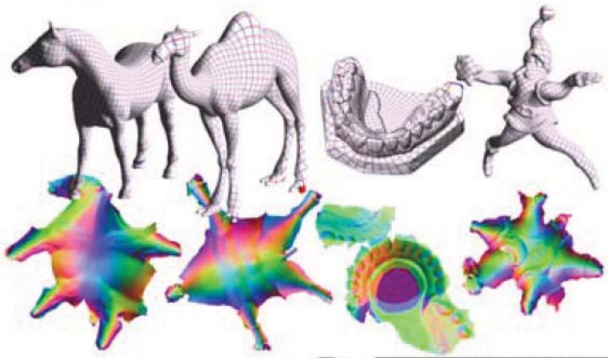
Completion

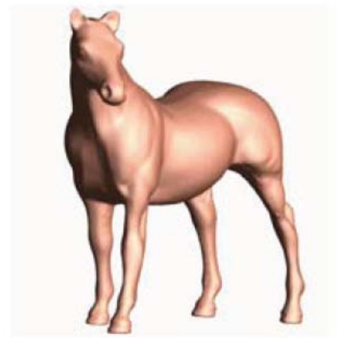


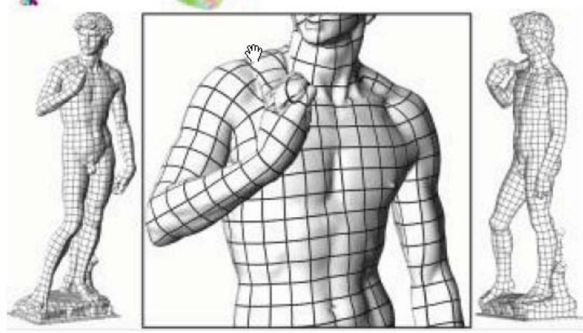
Deformation



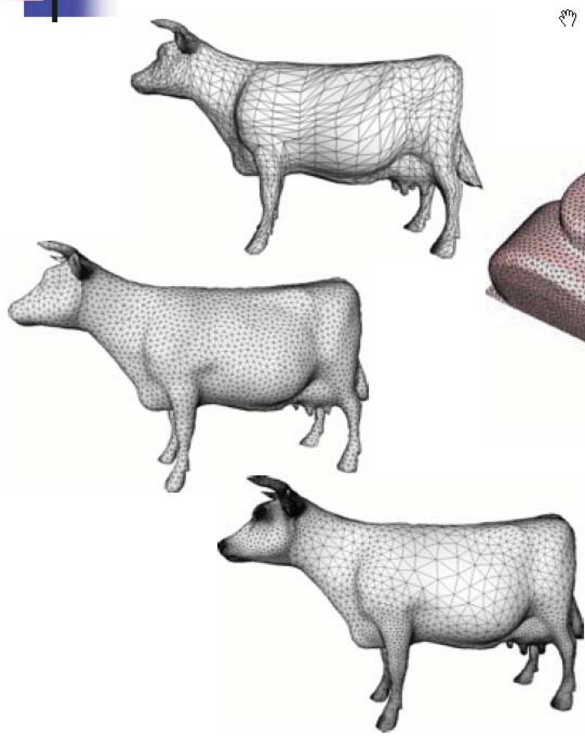
Parameterization

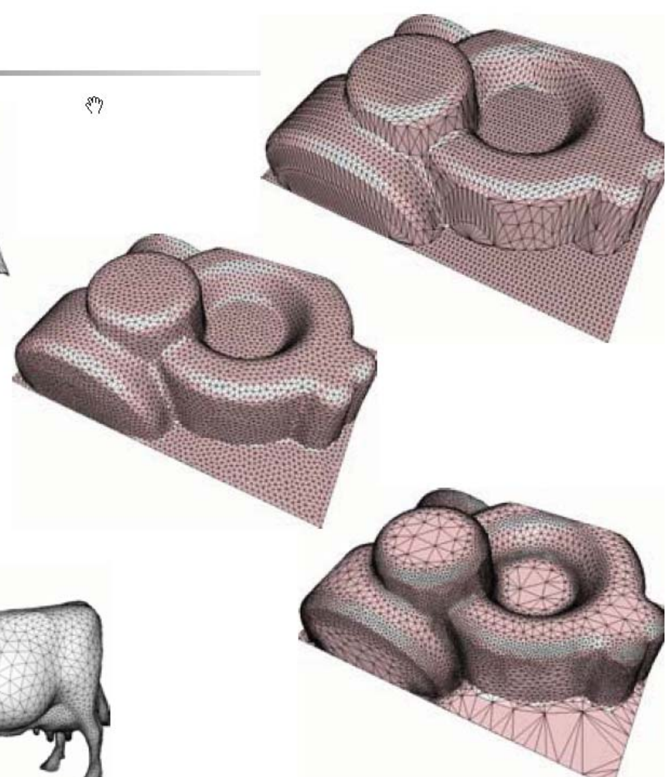




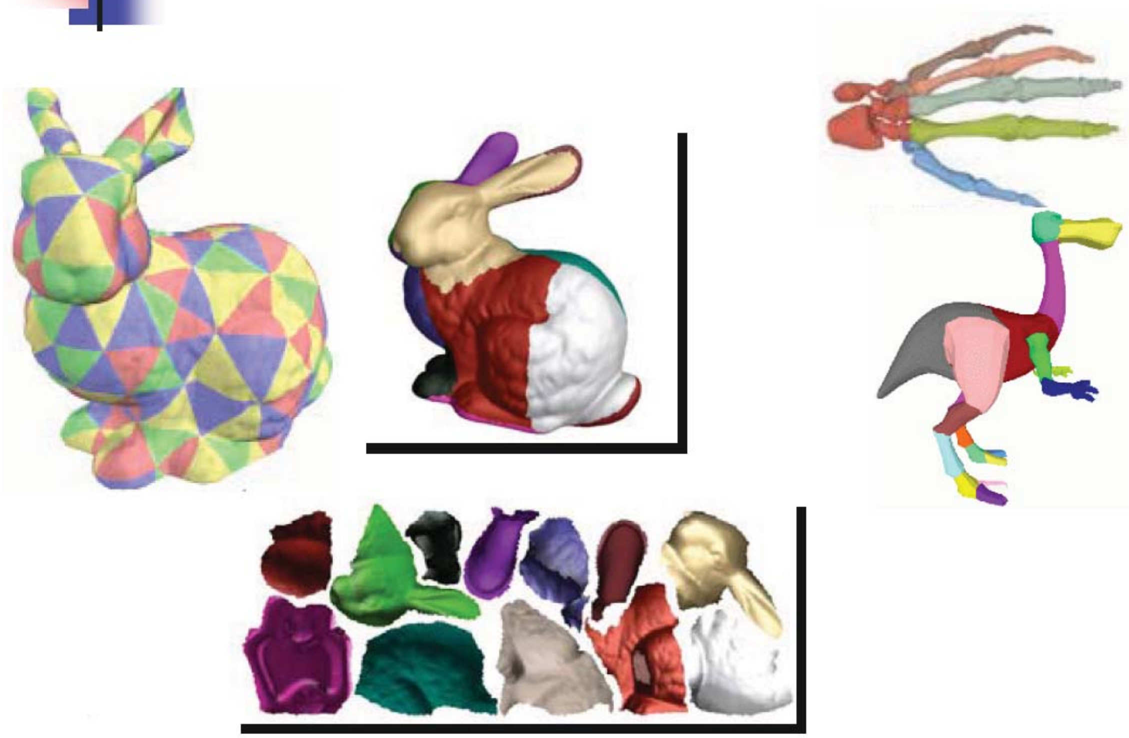


Remeshing





Segmentation



Course Schedule

Session II

- Structured Lighting
- Projector Calibration / Structured Light Reconstruction
- Combining Point Clouds Recovered from Multiple Views
- Surface Reconstruction from Point Clouds
- Elementary Mesh Processing
- ***Related Projects***
- Conclusion / Q&A

REVEAL Digital Archaeology Project

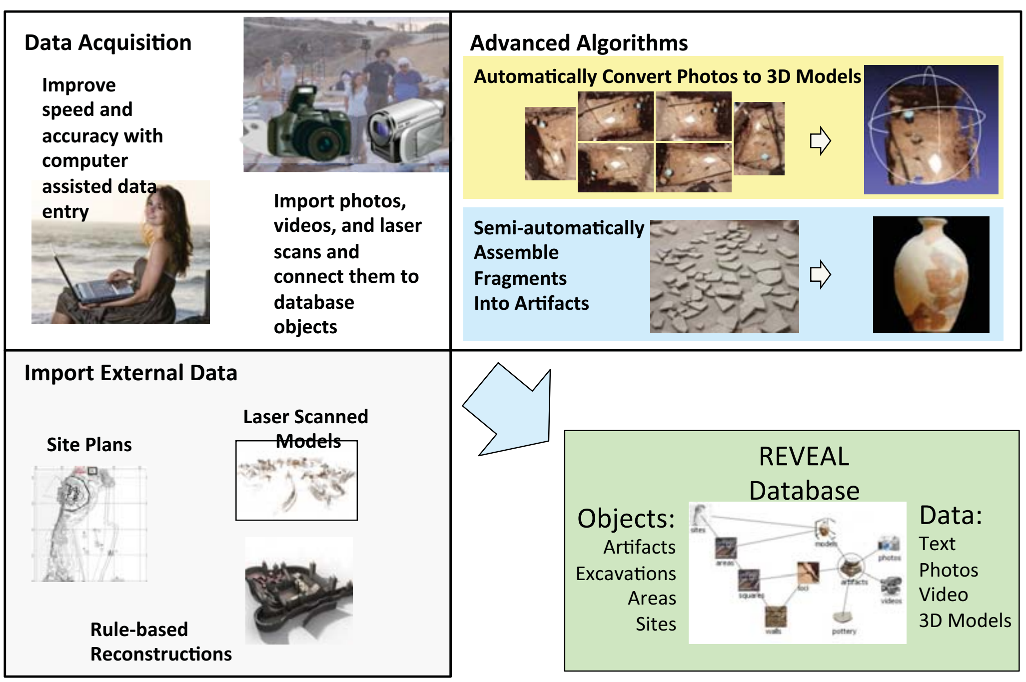
Cooper, Kimia, Taubin, Galor, Sanders, Willis

- Automates the tedious processes of data collection and documentation at the excavation site
- Provides visualization tools to explore the data collected in the database
- Solves specific problems in Archaeology using computer vision techniques
- Integrated Multi-View-Stereo (MVS) pipeline reconstructs 3D shapes from photos captured with hand-held cameras
- MVS pipeline can be installed independently of the rest of the system
- Software distributed as Open Source

http://sourceforge.net/projects/revealanalyze/

REVEAL Archaeological Data Acquisition

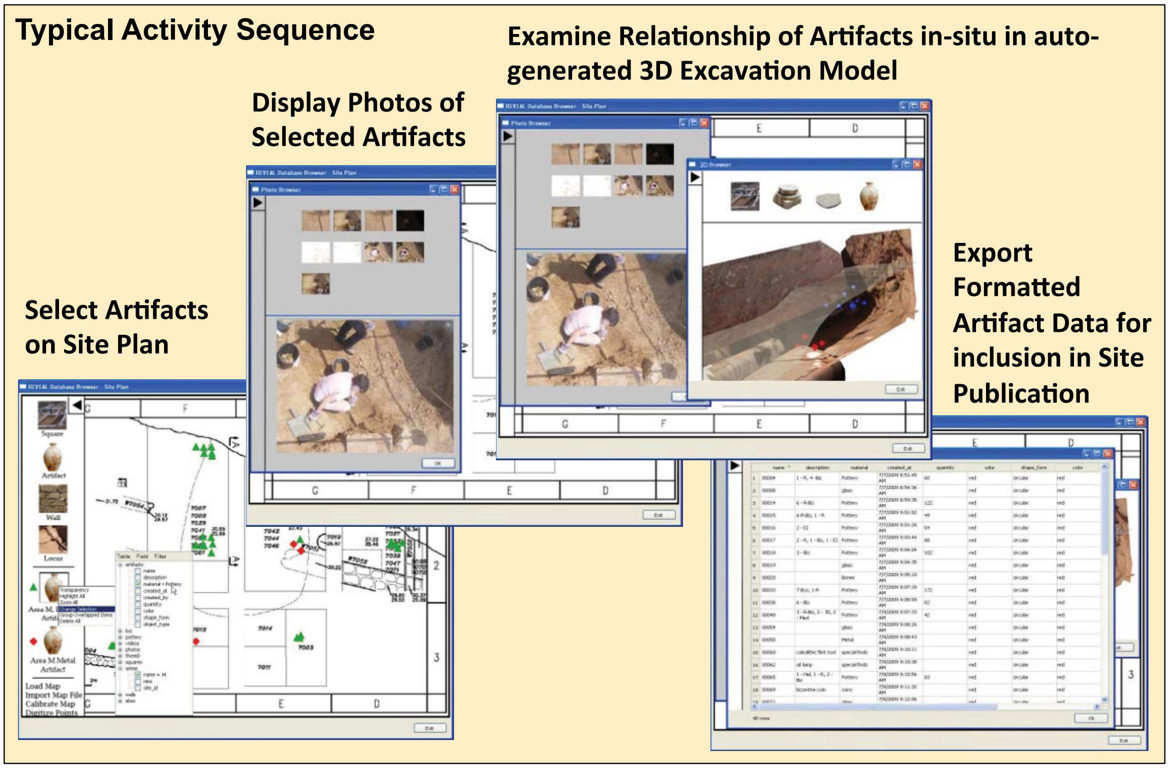
Assisted Data Acquisition, Algorithmic Reconstruction, Integrated multi-format analysis



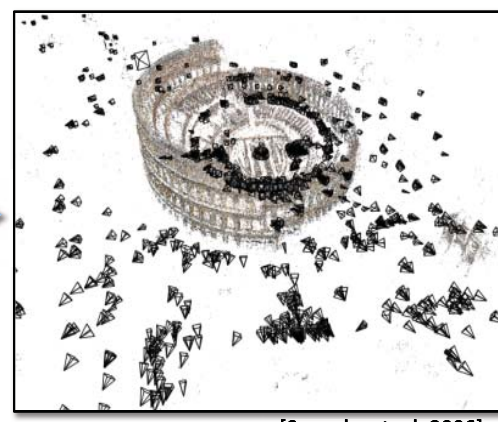
http://sourceforge.net/projects/revealanalyze

REVEAL Archaeological Analysis

Data integrated and synchronized in tabular, plan drawing, 3D spatial, image, and video formats

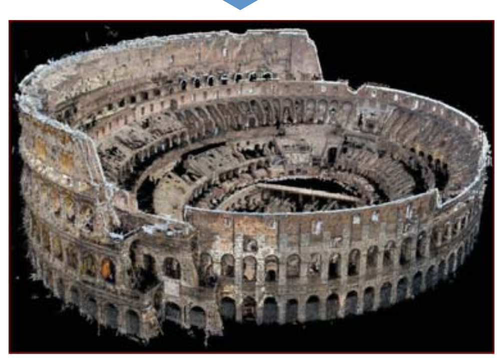


http://phototour.cs.washington.edu/bundler/



[Snavely et. al. 2006]

Patch-based **M**ulti-**V**iew **S**tereo (PMVS)
http://grail.cs.washington.edu/software/pmvs/



[Furukawa and Ponce 2008]

Reconstruction of colored meshes

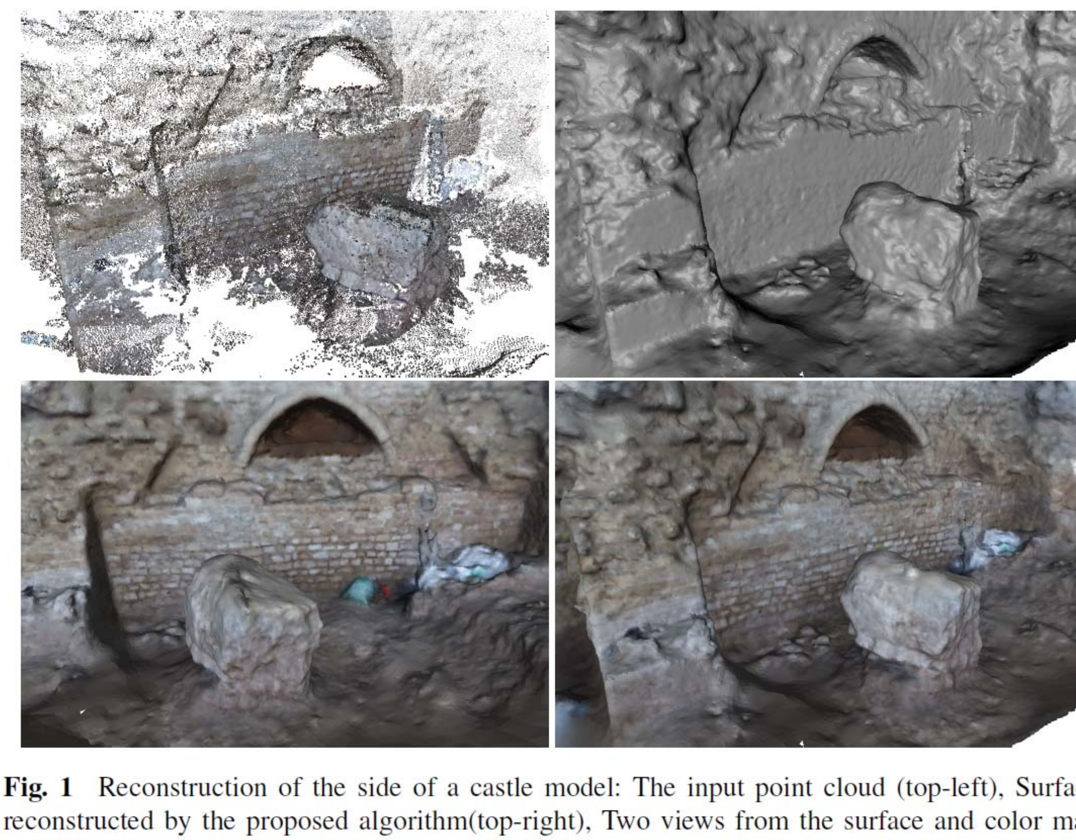


Fig. 1 Reconstruction of the side of a castle model: The input point cloud (top-left), Surface reconstructed by the proposed algorithm(top-right), Two views from the surface and color map reconstructed by the proposed algorithm (bottom).

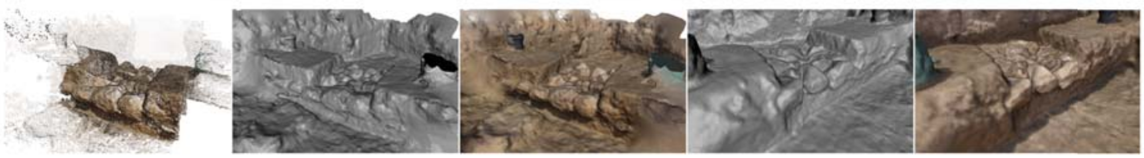
BROWN

High Resolution Surface and Appearance Modeling for Multi-view Stereo

Fatih Calakli and Gabriel Taubin

Reconstruction of a brick with cuneiform, Mesopotamian, 859-840 BCE, Williams College Museum of Art, Williamstown, MA

Reconstruction of Tel es-Safi wall section



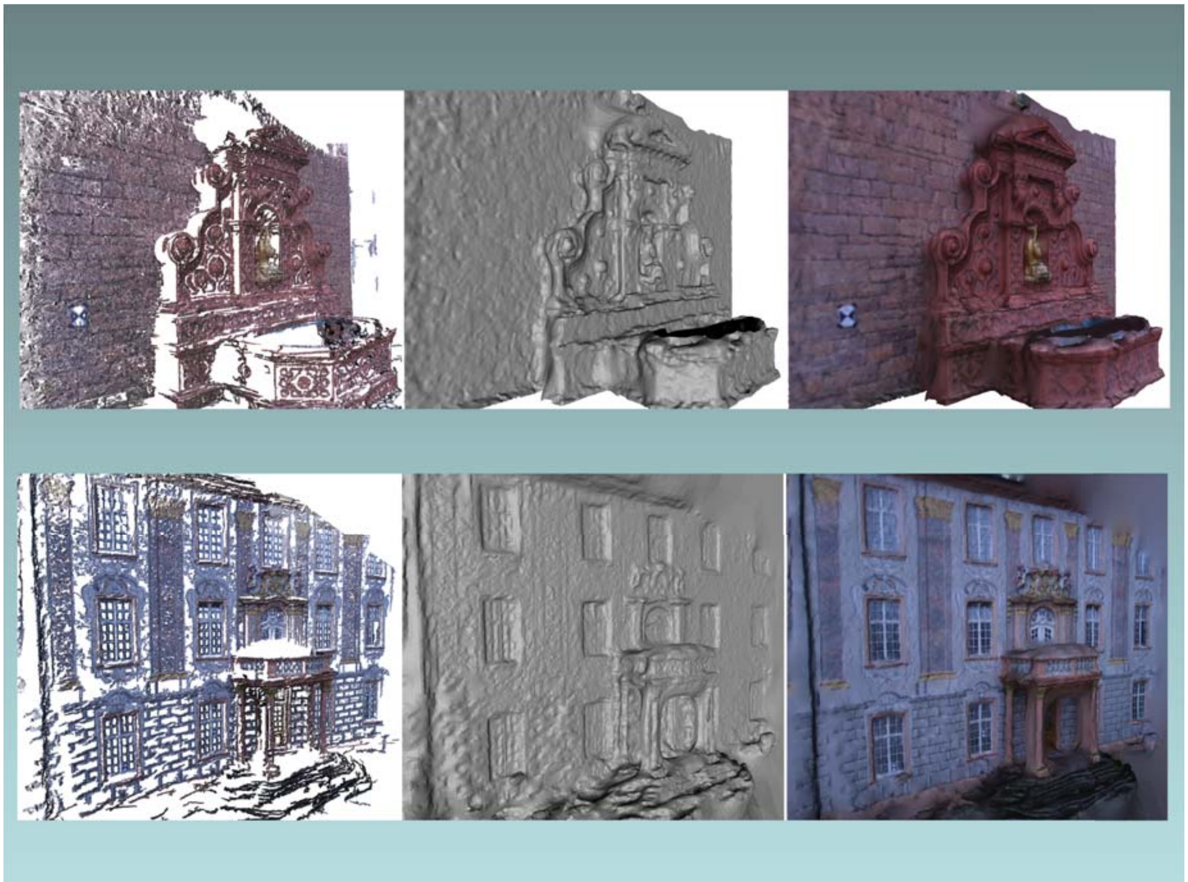
BROWN

High Resolution Surface and Appearance Modeling for Multi-view Stereo

Fatih Calakli and Gabriel Taubin

Reconstruction of Fountain P-11, from EPFL Benchmark

Reconstruction of Castle-Entry P-10, from EPFL Benchmark



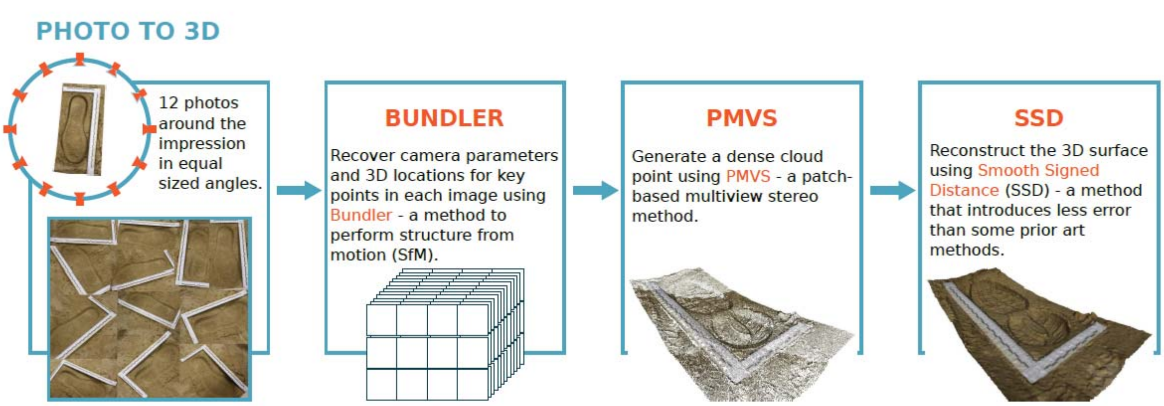
From Multi-View Video Cameras

View Interpolation From Multi-View Video Cameras

With Background Segmentation

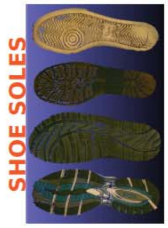
Ongoing work

Accurate 3D Footwear Impression Recovery From Photographs,
F. A. Andalo, F. Calakli, G. Taubin, and S. Goldenstein,
International Conference on Imaging for Crime Detection and Prevention (ICDP-2011).



Comparable to 3D Laser Scanner

EXPERIMENTAL RESULTS



d_g : Haursdoff distance map between the scanned shoe print and the scanned shoe sole.

d_m : Haursdoff distance map between our 3D model and the scanned shoe sole.

Shoeprint #	$\overline{d_g}$	$\overline{d_m}$	$\overline{d_m} - \overline{d_g}$
1	9.996	10.002	0.006
2	8.157	8.660	0.503
3	8.715	9.480	0.765
4	8.816	9.114	0.298

(mm)

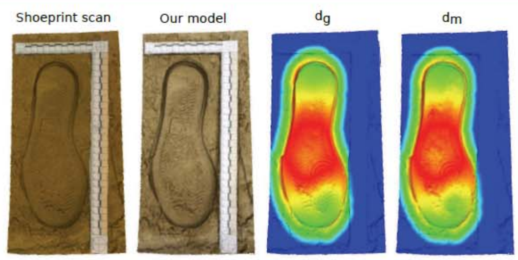
CONCLUSIONS

We presented a pipeline to recover footwear impressions from crime scenes using multiview stereo, which has not been considered for this kind of application until now.

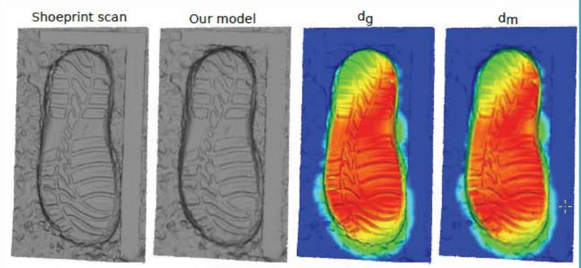
Despite the simplicity, the obtained surfaces are comparable with 3D scanning.

Future work: more experiments - number of images, angle between images, comparison with casting.

EXAMPLES

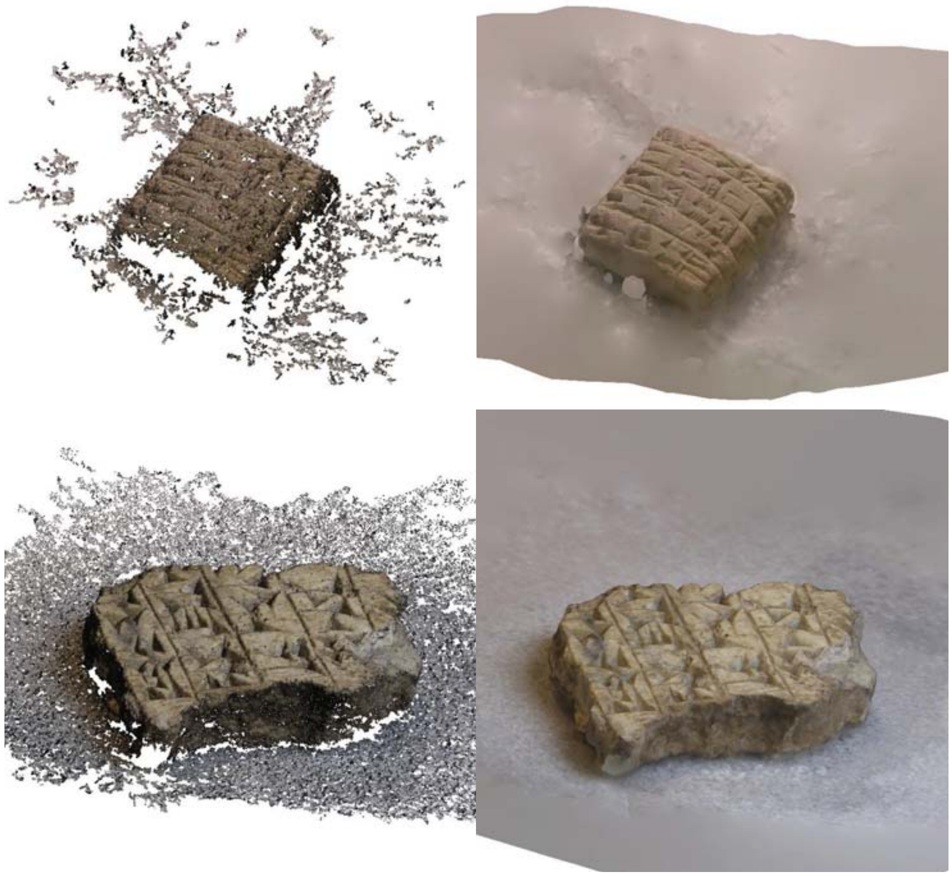


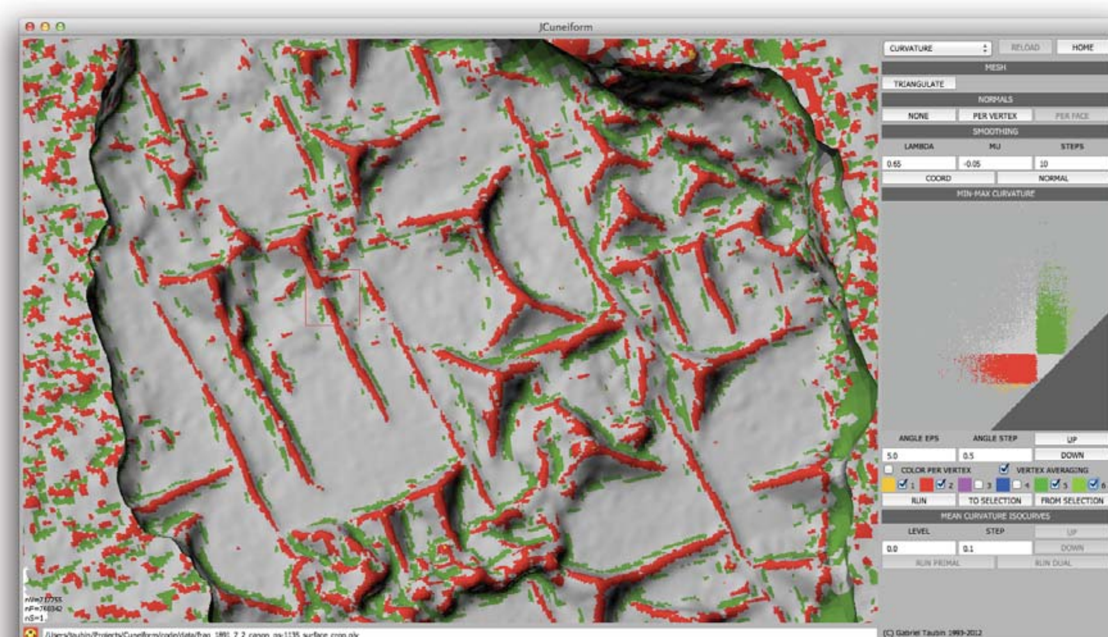
1st Shoeprint



3rd Shoeprint

Fig. 6 Reconstruction of a brick with cuneiform, Mesopotamian, 859-840 BCE, clay, Overall 35 x 35 x 11 cm, Williams College Museum of Art, Williamstown, MA, Gift of Professor Edgar J. Banks and Dr. John Henry Haynes, Class of 1876,(20.1.33.A). Top row: the input point cloud (left), and surface geometry (right) reconstructed by the proposed algorithm. Middle row: Two views from surface and color map reconstructed by the proposed algorithm. Bottom row: 6 examples from the set of 21 images that are used for shape acquisition.

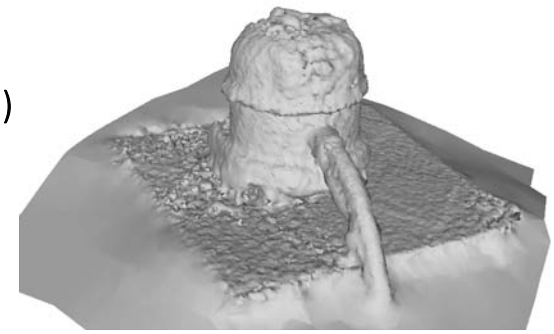


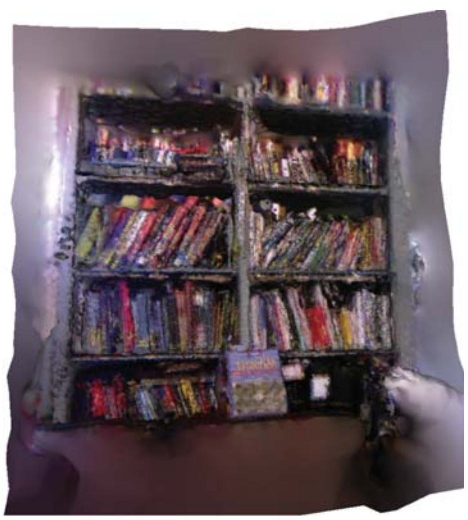

JCuneiform
CURVATURE
RELOAD
HOME
MESH
TRIANGULATE
NORMALS
NONE
PER VERTEX
PER FACE
SMOOTHING
LAMBDA
MU
STEPS
0.65
-0.05
10
COORD
NORMAL
MIN-MAX CURVATURE
ANGLE EPS
ANGLE STEP
UP
5.0
0.5
DOWN
COLOR PER VERTEX
VERTEX AVERAGING
1 2 3 4 5 6
RUN
TO SELECTION
FROM SELECTION
MEAN CURVATURE ISOCURVES
LEVEL
STEP
UP
0.0
0.1
DOWN
RUN PRIMAL
RUN DUAL
nS=1
/Users/taubin/Projects/Cuneiform/code/data/frag_1891_7_2_canon_ps-1135_surface_crop.ply
(C) Gabriel Taubin 1993-2012

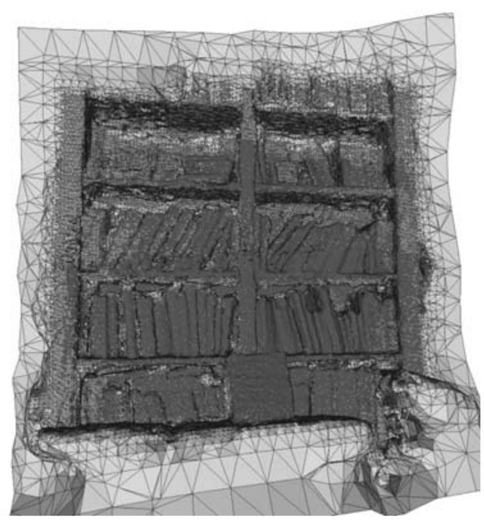
Handheld Interactive, Incremental 3D Object Scanning

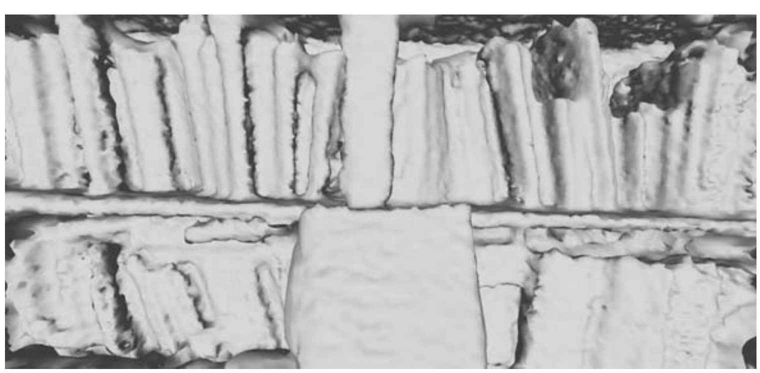
J. Kim & G. Taubin

- Based on MS Kinect sensor
- Continuous coarse pose estimation from color camera using PTAM
- 3D snapshots captured from different points of view using depth camera
- Alignment improced with Iterative Closest Points (ICP) algorithm
- SSD is run on aligned 3D snapshots





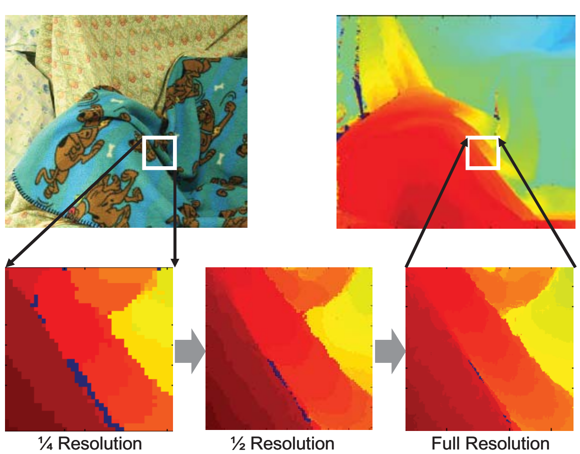




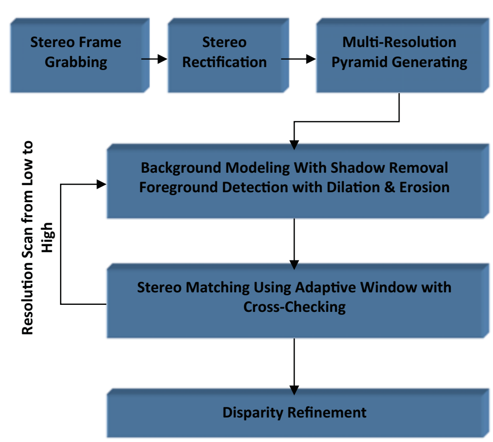
Real-Time High-Definition Stereo on GPGPU
using Progressive Multi-Resolution Adaptive Windows
Y. Zhao, and G. Taubin, Image and Vision Computing 2011.

Screen shots of our real-time stereo system working on the field

Real-Time High-Definition Stereo on GPGPU
using Progressive Multi-Resolution Adaptive Windows
Y. Zhao, and G. Taubin, Image and Vision Computing 2011.

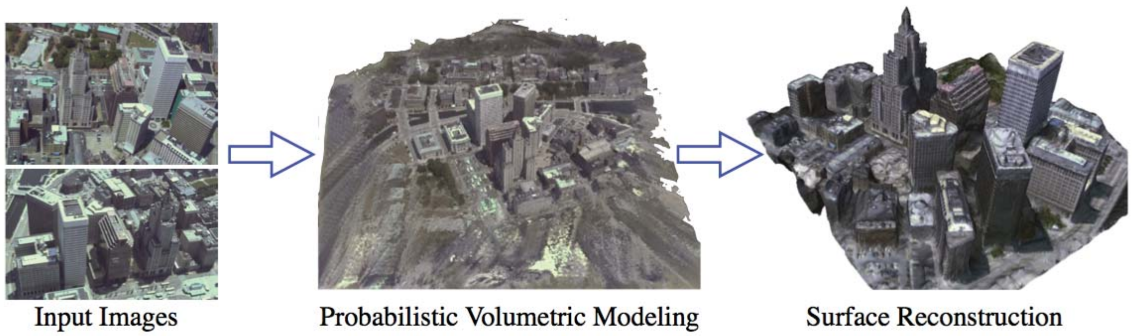


Coarse-to-fine matching on multiple resolutions

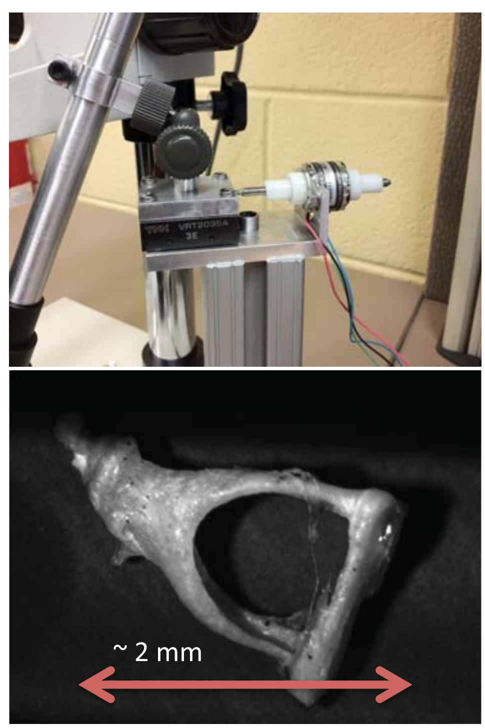


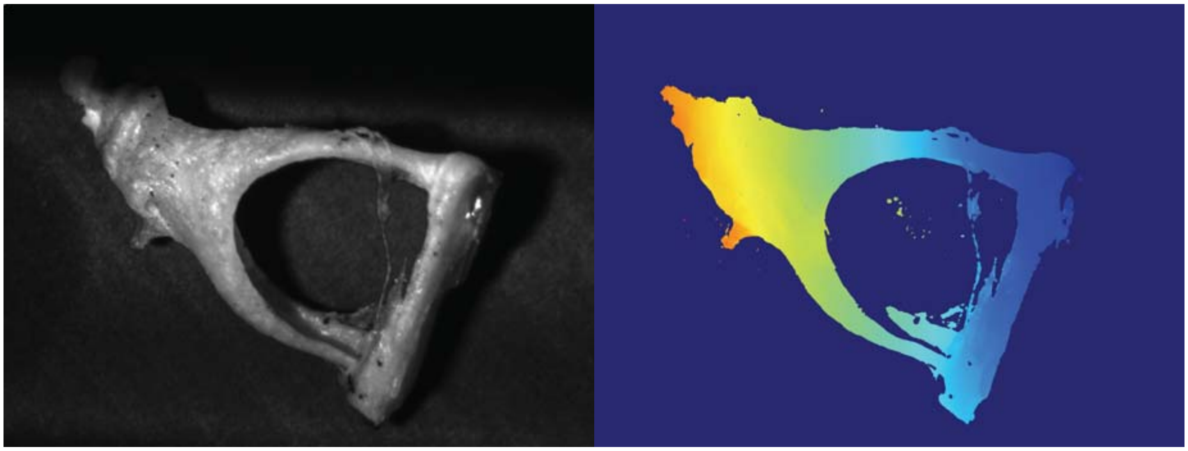
Processing Pipeline

High Resolution Surface Reconstruction from Multi-view Aerial Imagery
by Calakli, Ulusoy, Restrepo, Mundy & Taubin, 3DIMPVT 2012

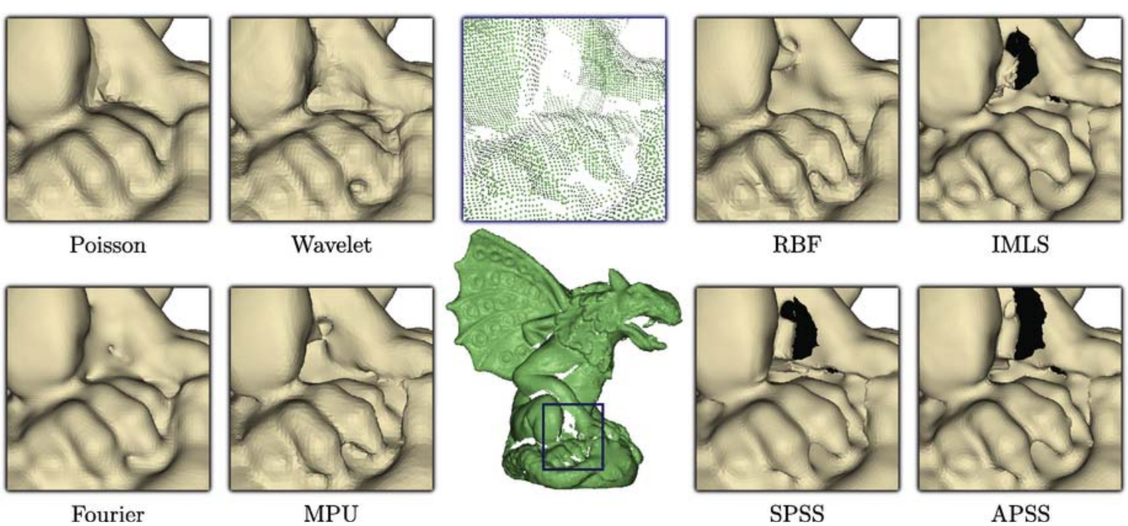


Microscopic 3D Shape Capture
Liberman & Taubin (work in progress)





An Evaluation and Comparison of Surface Reconstruction
M. Berger, J. Levine, L. Nonato, C. Silva, and G.Taubin
ACM Transactions on Graphics 2013 (Siggraph 2013)



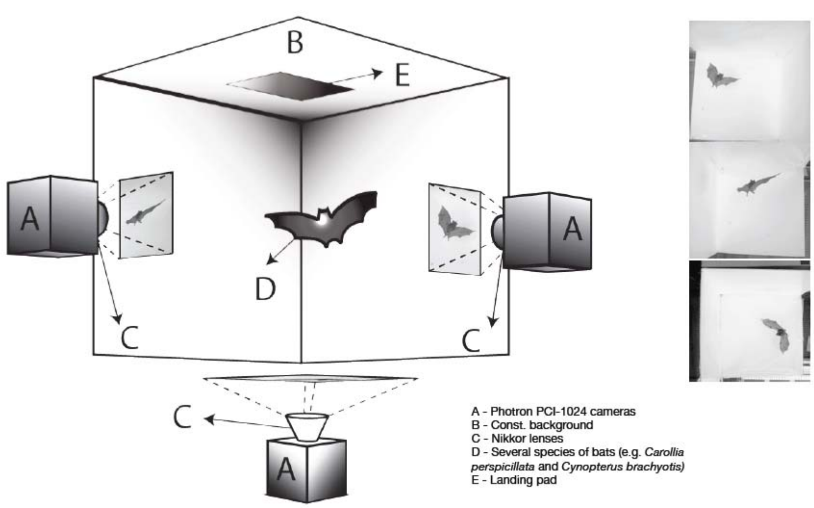
http://www.cs.utah.edu/~bergerm/recon_bench/

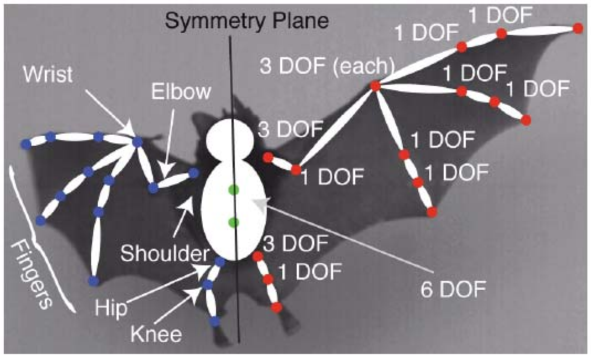
3D Reconstruction & Analysis of Bat Flight Maneuvers

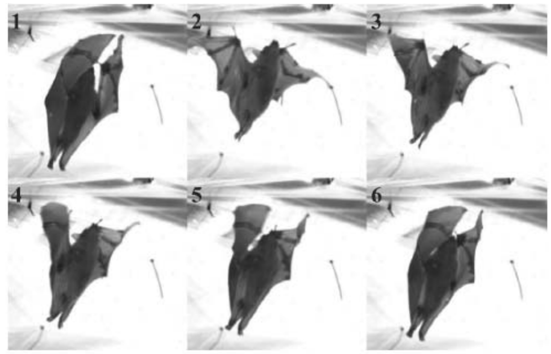
- 3D Reconstruction of Bat Flight Kinematics from Sparse Multiple Views, by A. Bergou, S. Swartz, and G. Taubin, and K. Breuer, 4DMOD, 2011.
- 3D Reconstruction and Analysis of Bat Flight Maneuvers from Sparse Multiple View Video, by A. Bergou, S. Swartz, S. K. Breuer, G. Taubin, BioVis, 2011.
- Falling with Style - The Role of Wing Inertia in Bat Flight Maneuvers, by A. Bergou, D. Riskin, G. Taubin, S. Swartz, and K. Breuer, Annual Meeting, Society for Integrative and Comparative Biology, 2011.
- Falling with Style-Bat Flight Maneuvers, by A. Bergou, D. Riskin, G. Taubin, S. Swartz, and K. Breuer, Bulletin of the American Physical Society, Vol. 55, 2010.

How do we measure bats ?

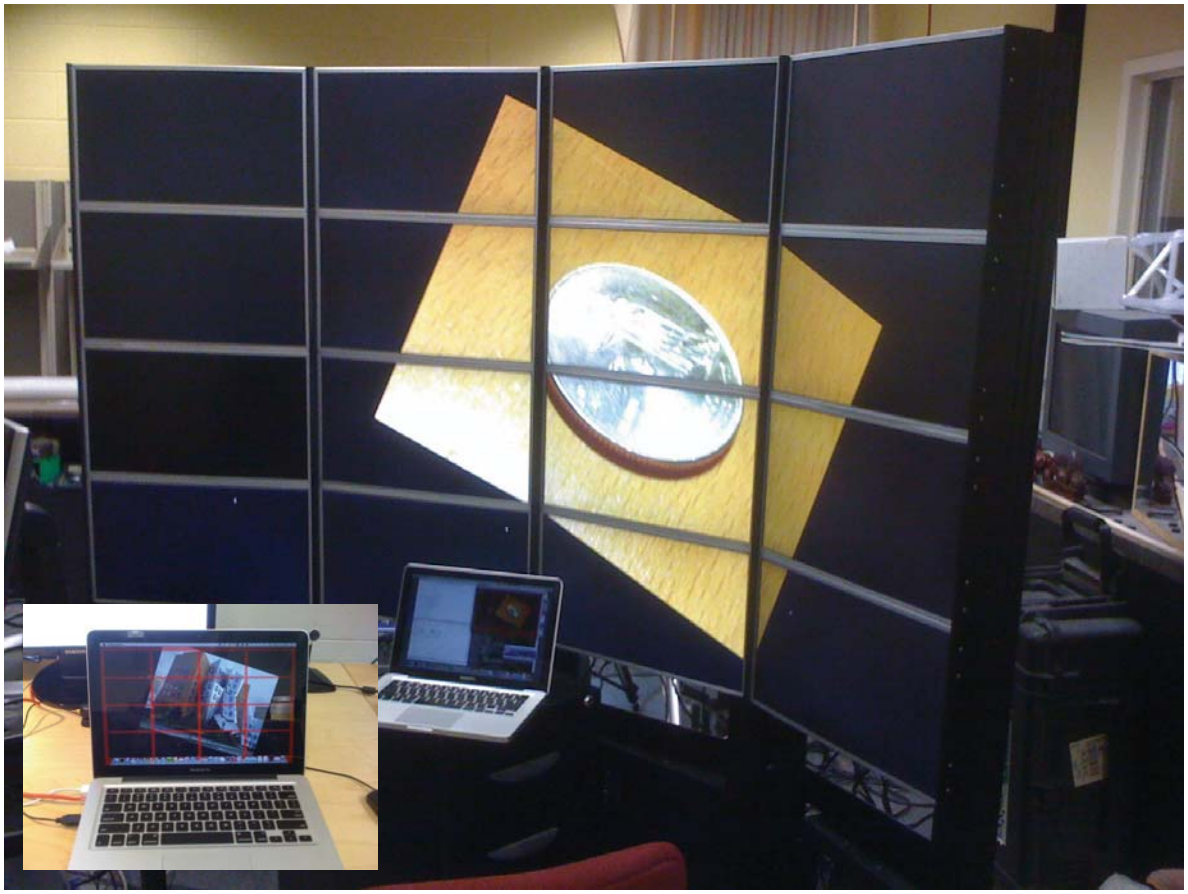
- Multiple synchronized 1000fps+ cameras
- Controlled environment (backdrop & illumination)
- Bats trained to land on landing pad
- Experiments with several species







- Bats have highly articulated wings
- Very complex wing motion
- Current goal: Detailed reconstruction of wing and body kinematics and derivatives from visual data
- Skeleton model with 52 degrees of freedom
- Geometry parameterized by 37 constants
- **Future Goal: Model-less Dynamic Shape Reconstruction**



Course Schedule

Session II

- Structured Lighting
- Projector Calibration / Structured Light Reconstruction
- Combining Point Clouds Recovered from Multiple Views
- Surface Reconstruction from Point Clouds
- Elementary Mesh Processing
- Related Projects
- ***Conclusion / Q & A***