EUROGRAPHICS 2002

Tutorial T1: 3D Data Acquisition

R. Scopigno, ISTI-CNR, PISA
C. Andujar, UPC Barcelona
M. Goesele, Max-Planck-Institut für Informatik, Saarbrücken
H. Lensch, Max-Planck-Institut für Informatik, Saarbrücken

Published by
The Eurographics Association
ISSN 1017-4565

The European Association for Computer Graphics
23rd Annual Conference

EUROGRAPHICS 2002

Saarbrücken, Germany
September 2–6, 2002

Organized by

EUROGRAPHICS
THE EUROPEAN ASSOCIATION
FOR COMPUTER GRAPHICS

INFORMATIK
Max-Planck-Institut
für Informatik
Saarbrücken, Germany

UNIVERSITAS SARAVIENSIS
Universität des Saarlandes
Germany

International Programme Committee Chairs
George Drettakis (France)
Hans-Peter Seidel (Germany)

Conference Co-Chairs
Frits Post (The Netherlands)
Dietmar Saupe (Germany)

Tutorial Chairs
Sabine Coquillart (France)
Heinrich Müller (Germany)

Lab Presentation Chairs
Günther Greiner (Germany)
Werner Purgathofer (Austria)

Günter Enderle Award Committee Chair
François Sillion (France)

John Lansdown Award Chair
Huw Jones (UK)

Short/Poster Presentation Chairs
Isabel Navazo (Spain)
Philipp Slusallek (Germany)

Honorary Conference Co-Chairs
Jose Encarnação (Germany)
Wolfgang Straßer (Germany)

STAR Report Chairs
Dieter Fellner (Germany)
Roberto Scopignio (Italy)

Industrial Seminar Chairs
Thomas Ertl (Germany)
Bernd Kehrer (Germany)

Conference Game Chair
Nigel W. John (UK)

Conference Director
Christoph Storb (Germany)

Local Organization
Annette Scheel (Germany)
Hartmut Schirmacher (Germany)

3D Data Acquisition

Tutorial Organizer: Roberto Scopigno

Tutorial Speakers: Carlos Andujar, Michael Goesele, Hendrik P. A. Lensch, Roberto Scopigno

Abstract

3D scanners and image acquisition systems are rapidly becoming more affordable and allow to build highly accurate models of real 3D objects in a cost- and time-effective manner. This tutorial will present the potential of this technology, review the state of the art in model acquisition methods, and will discuss the 3D acquisition pipeline from physical acquisition until the final digital model.

First, different scanning techniques such as time-of-flight or structured light approaches will briefly be presented. Other acquisition related issues including the design of the scanning studio will be discussed and evaluated. In the area of registration, we will consider both the problems of initially aligning individual scans, and of refining this alignment with variations of the Iterative Closest Point method. For scan integration and mesh reconstruction, we will compare various methods for computing interpolating and approximating surfaces. We will then look at various ways in which surface properties such as color and reflectance can be extracted from acquired imagery. Finally, we will examine techniques for the efficient management and rendering of very large, attribute-rich meshes, including methods for the construction of simplified triangle-based representation and sample-based rendering approaches.

1. Tutorial Content

The recent evolution of graphics technology has been impressive, and the management of very complex models is now possible on inexpensive platforms. 3D image acquisition systems (often called 3D scanners) are rapidly becoming more affordable and allow to build highly accurate models of real 3D objects in a cost- and time-effective manner. This talk will present the potential of this technology and review the state of the art in model acquisition methods. The different physical techniques available for acquiring 3D data – including laser-based triangulation, structured light triangulation, and time-of-flight – will be briefly presented, together with the basic pipeline of operations for taking the acquired data and producing a usable numerical model. The design of the scanning studio is a critical step (it can be a simple desk, or a sophisticated photographic lab); alternative technological choices will be discussed and evaluated. We will then look at the fundamental problems of range image registration, line of sight errors, mesh reconstruction, mesh simplification and surface detail (e.g. color) acquisition and mapping. In the area of registration we will consider both the problems of finding an initial global alignment using manual and automatic means, and refining this alignment with variations of the Iterative Closest Point methods. For scan integration and mesh reconstruction, we will compare various methods for computing interpolating and approximating surfaces. We will then look at various ways in which surface properties such as color (more properly, spectral reflectance) can be extracted from acquired imagery. Finally, we will examine techniques for the efficient management and rendering of very large, attribute-rich meshes, including methods for the construction of simplified triangle-based representation and sample-based rendering approaches.

Throughout the tutorial, we will motivate and illustrate the various aspects of the process with examples and results from an important application: the acquisition of Cultural Heritage artifacts.

2. Tutorial Outline and Distribution of Lectures

Lesson 1 (8:30-9:30):
3D Scanning Technology
Roberto Scopigno, CNR

- Welcome and overview
- Fundamentals of 3-D sensing: active 3D sensing (basic optical triangulation; pattern projection; time of flight systems); passive 3D sensing (silhouettes, space carving)

Lesson 2 (9:30-10:00):
Setting up a Scanning Lab
Michael Goesele, MPI

- How to choose scanning studio components (digital cameras, lighting, studio organization, infrastructure, etc.)

Coffee Break (10:00 - 10:30)

Lesson 3 (10:30-11:00):
Basic Acquisition Techniques
Michael Goesele, MPI

- Calibration Techniques
- High Dynamic Range Imaging
- Lab Procedures

Lesson 4 (11:00-12:00):
Range Data Registration and Merging
Roberto Scopigno, CNR

- The scanning pipeline
- Two-scan registration – iterative closest point; variations
- Multi-view registration
- Connect-the-dots – Delaunay sculpting; ball-pivoting
- Volumetric methods for scan integration – estimating a signed distance field, error model;

Lunch Break (12:00 - 14:00)

Lesson 5 (14:00-15:00):
Surface Attributes Acquisition and Management
Hendrik Lensch, MPI

- Using color images directly as texture maps – lighting conditions; problems in blending texture maps; view dependent texture maps
- Estimating diffuse reflectance – computing photo-consistent colors; global color balancing
- Estimated BRDF – assumptions/additional data needed for BRDF

Lesson 6a (15:00-15:30):
Simplification of scanned meshes
Carlos Andujar, UPC

- A brief overview of mesh simplification methods – introduction and classification of methods; overview of clustering, incremental and volumetric methods

Coffee Break (15:30 - 16:00)

Lesson 6b (16:00-16:30):
Simplification of scanned meshes
Carlos Andujar, UPC

- Huge meshes simplification (external memory)
- Techniques for preserving mesh attributes or detail in simplification

Lesson 7 (16:30-17:00):
Rendering scanned meshes
Roberto Scopigno, CNR

- Rendering huge meshes on low-cost computers: triangle-based approaches vs. sample-based approaches
- Using 3D scanned data in Cultural Heritage applications

Conclusion: Questions and Answers, Discussion

3. Speakers Biographies

Carlos Andujar, Technical University of Catalonia (UPC), Spain

Carlos received his PhD in Software Engineering from the Technical University of Catalonia, Spain, where he is currently an Associate Professor in the Software Department and a Senior Researcher at the Barcelona's Virtual Reality Center. His current research is focused on virtual reality applications, large model visualization and geometry simplification.

Michael Goesele, Max-Planck-Institut für Informatik (MPI), Germany

Michael is a Research Assistant in the computer graphics group at the Max-Planck-Institut fuer Informatik, Germany. He is currently a PhD candidate under the guidance of Prof. Hans-Peter Seidel. He studied computer science at the University of Ulm, Germany, and at the University of North Carolina at Chapel Hill, USA, and received his diploma in computer science from the University of Ulm in 1999. His research is focused on image-based acquisition techniques including various calibration aspects. He established a measurement lab for reflection properties.

Hendrik P. A. Lensch, Max-Planck-Institut für Informatik (MPI), Germany

Hendrik is a Research Assistant in the computer graphics group at the Max-Planck-Institut fuer Informatik in Saarbruecken, Germany. He is currently a PhD candidate under the direction of Prof. Hans-Peter Seidel. In 1999 he received his diploma in computer science from the Universitaet Erlangen-Nuernberg. His experience in computer graphics spans the fields of image-based rendering, 3D scanning and BRDF measurement. Current research focuses on the acquisition of real world objects including both their geometry and their surface properties.

Roberto Scopigno, ISTI - CNR, Italy

Roberto is a Senior Research Scientist at ISTI-CNR, an Institute of the Italian National Research Council (CNR). He graduated in Computer Science at the University of Pisa in 1984, joined CNR in '86 and has been involved in Computer Graphics since then. He is currently engaged in research projects concerned with scientific visualization, volume rendering, multiresolution data modeling and rendering, 3D scanning and surface reconstruction. He published more than sixty papers in international refereed journals/conferences and served in the programme committees of several Eurographics workshops and conferences. Since 2001 he is Joint Chief Editor of the Computer Graphics Forum Journal and Member of the Editorial Board of The Visual Computer Journal. He is member of Eurographics, ACM Siggraph and IEEE.

4. Contact Information

Carlos Andujar
Universitat Politecnica de Catalunya
Dept. Llenguatges i Sistemes Informatics
Av. Diagonal, 647, 8 planta
08028 Barcelona
Spain
Phone: +34 934 015 718
Fax: +34 934 012 591
Email: andujar@lsi.upc.es
WWW: http://www.lsi.upc.es/∼andujar

Michael Goesele
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken
Germany
Phone: +49 681 9325 426
Fax: +49 681 9325 499
Email: goesele@mpi-sb.mpg.de
WWW: http://www.mpi-sb.mpg.de/∼goesele

Hendrik P. A. Lensch
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken
Germany
Phone: +49 681 9325 428
Fax: +49 681 9325 499
Email: lensch@mpi-sb.mpg.de
WWW: http://www.mpi-sb.mpg.de/∼lensch

Roberto Scopigno
Istituto di Scienza e Tecnologie dell'Informazione (ISTI)
Consiglio Nazionale delle Ricerche (CNR)
v. Moruzzi 1, 56100 Pisa
Italy
Phone: +39 050 3152929
Fax: +39 050 3158091
Email: roberto.scopigno@cnuce.cnr.it
WWW: http://vcg.iei.pi.cnr.it/∼scopigno

Tutorial T1
"3D Data Acquisition"

Eurographics Conference 2002,
Saarbrucken

R. Scopigno, ISTI-CNR
M. Goesele, H. Lensch, MPI
C. Andujar, UPC

Acquiring Visually Rich 3D Models

- Standard CAD modeling
- Image-based Rendering
  - *Panoramic images: a 2D model!*
- Image-based Modeling
  - *"Blocky" 3D models*
- 3D scanning

Using CAD tools ?

Modeling tools developed for CAD applications:

- complex -- require skilled users
- not adequate for the **accurate** reproduction of highly complex, free form surfaces (e.g. works of art):

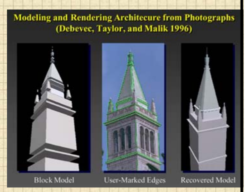
CAD modeling → accuracy of the model often unknown (with respect to the original)

Raffaello's Apartments and
S. Peter Basilica
by InfoByte - Italy

Image-based Rendering / Modeling

Using **images** of the real world/object to get:

- **Interactive Image-based Rendering (I-BR)**
  (panoramic images, QTVR, etc.)
  - from images to images: 2D model + re-projection capabilities
- **Image-based Modeling (I-BM)**
  - 3D structure is derived from a **small** set of uncalibrated images
  - User-assisted construction
  - Ex.: Debevec et al. Sig.'96, Metacreation's Canoma, etc.



Modeling Complex Shapes

Neither CAD nor I-BM can manage the construction of an accurate 3D model of a really complex artifact

➔ **3D scanning**

Obviously, not only Cultural Heritage applications

Tutorial Outline

8:30 - 9:30 Lesson 1: 3D Scanning Technology
[60 min, Roberto Scopigno, CNR]

9:30 - 10:00 Lesson 2: Setting up a scanning lab
[30 min, Michael Goesele, MPI]

10:00 - 10:30 Coffee Break

10:30 -11:00 Lesson 3: High Dynamic Range Imaging
[30 min, Michael Goesele, MPI]

11:00 -12:00 Lesson 4: Range Data Registration and Merging
[60 mins, Roberto Scopigno, CNR]

12:00 - 14:00 Lunch Break

Tutorial Outline [2]

14:00-15:00 Lesson 5: Surface Attributes Acquisition and Management
[60 mins,Hendrik Lensch, MPI]

15:00-15:30 Lesson 6(a): Simplification of scanned meshes
[30 mins, Carlos Andujar, UPC]

15:30 - 16:00 Coffee Break

16:00-16:30 Lesson 6(b): Simplification of scanned meshes
[30 mins, Carlos Andujar, UPC]

16:30-17:00 Lesson 7: Rendering scanned meshes
[30 mins, Roberto Scopigno, CNR]

Conclusion: Q&A, discussion

Tutorial T1: 3D Data Acquisition

"3D Scanning Technology"

Roberto Scopigno
Visual Computing Group
ISTI – C.N.R.
Pisa, Italy

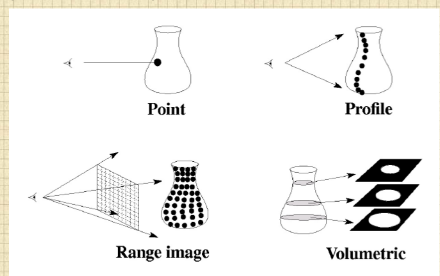
3D Scanning

Characteristics of an ***optimal*** 3D scanner:

- truly 3D
- accurate
- fast
- easy to use and to move in the acquisition space
- safe, both for the user and the reconstructed object
- capable of capturing object appearance (color or radiance)
- low price

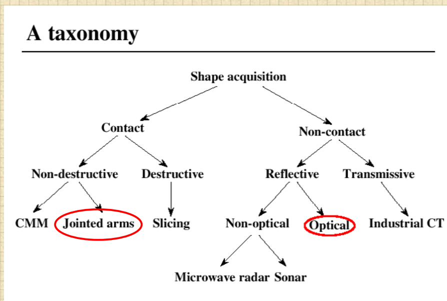
3D Scanning - Output

- Data produced in output:



[Image by Brian Curless, Sig2000 CourseNotes]

3D scanning Taxonomy [1]



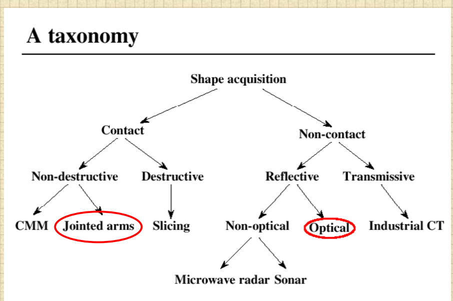
[By Brian Curless, Sig2000 CourseNotes]

Contact Techniques -- Probing

- Contact probe acquisition:
  - Hand-held, manually assisted e.g. Immersion-MicroScribe3D (low cost)
  - Robotic, industrial systems (high cost, very high precision)

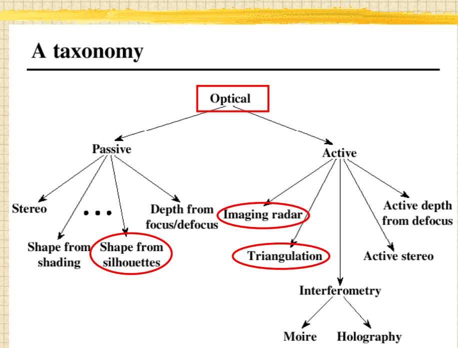
- Disadvantages:
  - **Very long** acquisition time (manual positioning)
  - Sampling accuracy (how do we choose the points to be sampled?)
  - No data on **appearance**

3D scanning Taxonomy [1]



[By Brian Curless, Sig2000 CourseNotes]

3D scanning Taxonomy [2]



[By Brian Curless, Sig2000 CourseNotes]

3D automatic acquisition technologies

Overview of [some] optical technologies

- Passive:
  - reconstruction from silhouettes
- Active:
  - triangulation-based devices
    - laser-based
    - structured light
  - time-of-flight devices

Passive Techniques

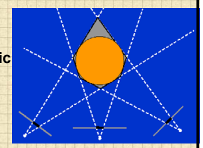
3D geometry can be reconstructed by taking into account:

- Stereo images (photogrammetry)
- Motion (multiple calibrated images)
- **Silhouettes** (multiple calibrated images)
- Focus/defocus
- Shading

PassiveTech. – Silhouettes

Reconstruction from silhouette:

- **acquire *n*** images of the object from different views
- extract the object **silhouette** in each image
- each silhouette + camera center defines a **~conic region** of space which encloses the object
- **intersection** of cones gives a bounding volume
- reconstruct the 3D shape from this bounding volume.



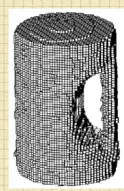
But:

- The bounding volume is the **Visual Hull** of the real 3D object! (no concavities)

PassiveTech. – Silhouettes [2]

View specification:

- Accurate rotating platform (computer-controlled)
- Reconstructed from images (graycodes below)



PassiveTech. – Silhouettes [3]

Reconstruction:

1. By **intersecting polyhedra** in "continuous" 3D space
2. By compositing conic regions in a discrete **voxel space**, and then fitting an **isosurface**
   - Faster and easier to implement, but less accurate

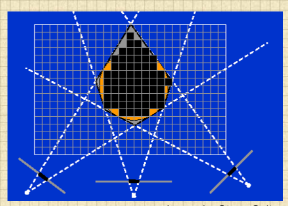


Image by Steve Seitz

PassiveTech. – Silhouettes [4]

- Improving reconstruction of **concave regions**
  - Shape from **shading**, under **controlled lighting** and (carve concave regions according to information inferred by the field of surface normal vectors)
  - Shape from **stereo** (extract a corresponding point pair from two consecutive views, determine geometry via triangulation)
  - Shape from **self-shadowing** (carving concave regions according to self-shadows) [Savarese+02]

PassiveTech. – Silhouettes [5]

How do we manage color ?

1. Stitch [sections of] the acquired images to the reconstructed mesh (texture-mapped output)
2. Assign the corresponding [weighted] color to each voxel, and extract an isosurface with color-per-vertex data
   - Take into account **occlusions**!
   - Potential **unfocusing** and **blurring** of the color data (weighted composition)

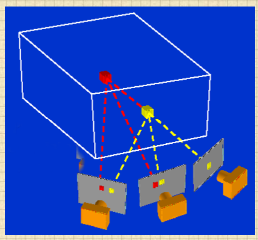


Image by Steve Seitz

PassiveTech. – Silhouettes [6]

Pros:

- Very fast and easy to use (no registration of intermediate results needed)
- Low price

3D ScanBook (5K USD)

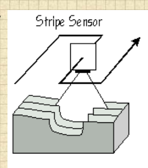
Cons:

- We cannot put all objects on a rotating platform
- Data accuracy is low:
  - Depends on the **resolution** of the sensing device (dig. camera)
  - Problems with **concave regions** (output is the visual hull)

OK for visual presentation (web)
Not sufficiently accurate for other uses

Active Tech. -- Optical Technologies

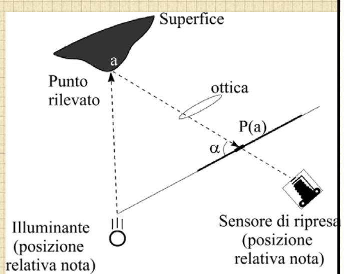
- Using light is much faster than using a physical probe
- Allows also scanning of soft or fragile objects which would be threatened by probing
- Three types of **optical** sensing:
  - **Point**, similar to a physical probe:
    - uses a single point of reference, repeated many times
    - slow approach, as it involves lots of physical movement by the sensor.
  - **Stripe**
    - faster than point probing, a band of many points passes over the object at once
    - it matches the twin demands for **speed** and **precision**.
  - Other **patterns** …



Optical Technologies - Triangulation [3]

How do we compute the 3D coordinates of each sampled point?

- By **triangulation**, known:
  - emitting point of the light source + direction (***illuminant*** or ***emitter***)
  - the focus point of the acquisition camera (***sensor***)
  - the center of the imaged reflection on the acquisition sensor plane (***P(a)***)

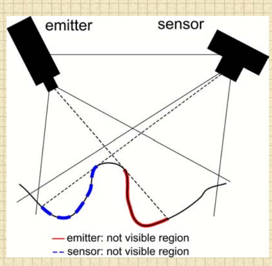


Triangulation is an old, simple approach (Thales-Talete)
Issues: precision and price of the system

Triangulation-based systems

An inherent limitation of the triangulation approach: non-visible regions

- Some surface regions can be ***visible*** to the **emitter** and ***not-visible*** to the **receiver**, and vice-versa
- In all these regions we miss sampled points ➔ integration of multiple scans



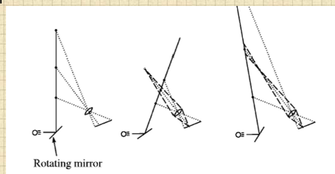
Optical -- Laser Scan

Why are lasers a good idea?
- Compact
- Low power
- Tight focus over long distances
- Single wavelength is easy to isolate in images (filter out background illumination)
- No chromatic aberration

But
- Commercial laser scanners appeared 10-15 years ago
- Product evolution is rather slow, prices did not drop down as fast as other IT commodities

Optical -- Laser Scan [2]

Triangulation scanning configurations:
- A scene can be scanned by sweeping the illuminant
- Disadvantages:
  - Loss of resolution due to defocus
  - Large variation in field of view
  - Large variation in resolution

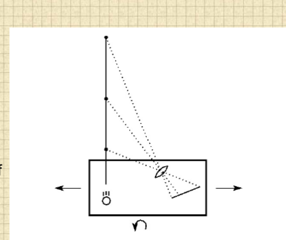


[Image by Brian Curless, Sig2000 CourseNotes]

Optical -- Laser Scan [3]

Triangulation scanning configurations:
- The laser illuminant and the camera can be designed as a **rigid subsystem** (scanning unit)
- The **scanning unit** is translated (comp.-controlled linear motion)

  or the **object** can be **rotated** in front of the scanning unit



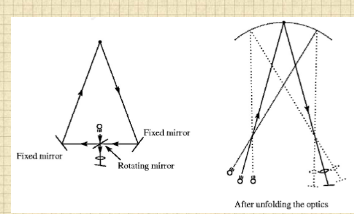
[Image by Brian Curless, Sig2000 CourseNotes]

Optical -- Laser Scan [4]

Triangulation scanning configurations:
- Or we may sweep the laser and the sensor *simultaneously*

  [eg. Patent of NRC of Canada]
- Pro's:

  Increased precision and uniform resolution



[Image by Brian Curless, Sig2000 CourseNotes]

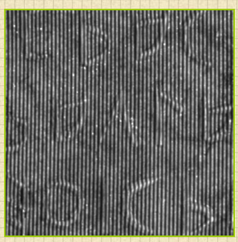
Laser Scan - Systems [6]

3D SCANNERS	http://www.3dscanners.com/
3DM Devices	http://www.3dm.com/
ABANTE Automation	http://www.abante.ca/index.htm
Arius3D	http://www.arius3d.com
Cyberware Home Page	http://www.cyberware.com/ → **15K - 100K $**
Cyberoptics	http://www.cyberoptics.com/flashIndex.htm
Digibotics Home Page	http://www.digibotics.com/
Geometrix, Inc.	http://www.geometrixinc.com/
GIE Menu - English	http://www.gietech.com/
Hamamatsu	http://www.hamamatsu.com/hp2e/products/SYSE/Mesure.html
Imagine Optic France	http://www.imagine-optic.com/
Integrated Vision Products	http://www.ivp.se/products/ranger.htm
Kreon Industrie - Welcome	http://www.kreon3d.com/
Laser Design (LDI)	http://www.laserdesign.com/
MENSI	http://www.mensi.com/index.html
Minolta Corporation, ISD	http://www.minolta3d.com/specs.html → **15K - 40K $**
Nextec	http://www.nextec-wiz.com/index.html
Northern Digital	http://www.ndigital.com/
Polhemus	http://www.polhemus.com/
Rangefinder Cubicscope	http://hilbert.elcom.nitech.ac.jp/CubicscopeHP/index.html
Real 3D Inc. Graphics	http://www.real3d.com/staging/default.htm
Servo-Robot Inc.	http://www.servorobot.com/
ShapeGrabber - Vitana	http://www.shapegrabber.com/
Steintek Optical	http://www.steintek.de/indexe.htm
Virtual 3D Home Page	http://www.virtual3dtech.com/
VITRONIC	http://www.vitronic.com/
Wolf & Beck	http://www.wolfbeck.com/english/index.htm

[List courtesy of J.A. Beraldin, NRC Canada]

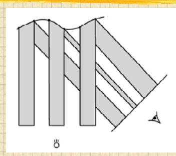
Scanning via Structured Light

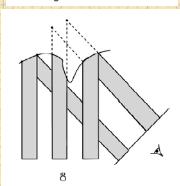
- A diffuse approach:
  - Simple to implement
  - Fast and cheap
  - From low to high accuracy
- Many **multi-stripe pattern** systems designed for human face acquisition (clinical or media applications) :
  - Use a regular stripe pattern (e.g. a simple slide projected on the face)
  - Take just 1 or 2 photos to acquire the 3D shape
  - Reconstructs geometry by **triangulation**



Scanning via structured light [2]

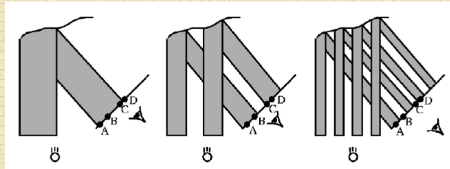
- **Laser:** single stripe of coherent light
- **Incoherent light:** many different patterns can be used (e.g. multiple lines, stripes, etc.)
  - **Pro's:**
    - Simpler design, no sweeping/translating devices needed
    - Faster acquisition (a single image for each multi-stripe pattern)
  - **Con's:**
    - Trade off depth-of-field for speed
- Problem: ***ambiguity*** in single multi-stripe pattern





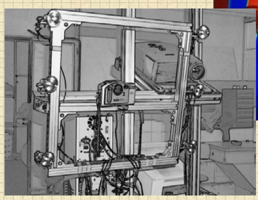
Scanning via structured light [3]

- Reconstruction ambiguity can be prevented by adopting a set of patterns based on **hierarchical** subdivision (Gray code)
  - Given ***K*** the number of pixels on the sensor image plane
  - Project ***Log k*** images (colored stripes with recursively decreasing width)
  - A **binary code** is associated to each pixel (eg. A=111, C=110, D=011)



A low-cost solution – CNR scanner

- Cheap system
  - Emitter: video-projector
  - Sensor: digital camera
- Mixed hierarchical RGB pattern:
  - **Green lines** for the triangulation
  - **Red/blue stripes** for spatial indexing



A low-cost solution – CNR scanner [2]

Projection of a hierarchical multi-image pattern

One image acquired for each pattern

The hierarchical sequence of red-blue stripes allows to assign an univocal code to each green line

A low-cost solution – CNR scanner[3]

The Minerva of Arezzo (bronze statue, Greek or Roman)

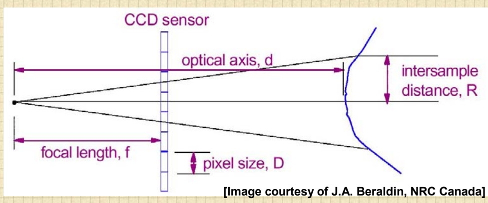
Structured light -- Systems

3 Dimensional Body Scanning	http://www.tc2.com/RD/RDBody.htm
Minolta 3D 1500	http://www.minolta.com/dp/3d1500/
3D Scanner Montech	http://www.montech.com.au/index.html
Breuckmann	http://www.breuckmann.com/english/index.html
EOIS High-Speed 3D Digitizers	http://www.eois.com/
Genex Technologies INC.	http://www.genextech.com/
GOM	http://www.gom.com/
InSpeck inc.	http://www.InSpeck.com
Medic-3D-Rugle	http://www4.justnet.ne.jp/~otoyosan/
OPTONET S.r.l.	http://www.optonet.it/
PPT Vision - Digital Machine Vision Systems	http://www.pptvision.com/smitutor.cfm
Steinbichler Optical Technologies	http://www.steinbichler.com/index2.htm
SYMCAD World © TELMAT Industrie	http://www.symcad.com/eng/ENGindex.htm
Wicks and Wilson Limited	http://www.wwl.co.uk/triform.htm

[List courtesy of J.A. Beraldin, NRC Canada]

Quality of sampling

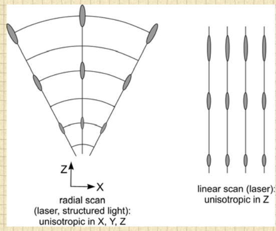
- Resolution
  - smaller **measure variation** that can be measured (XY: inter-sample distance at a given depth **d**; Z: smallest measurable variation in depth)
  - but in some cases, also CCD sensor resolution…
- Accuracy
  - measured location ***vs.*** real point location



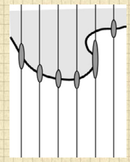
[Image courtesy of J.A. Beraldin, NRC Canada]

Acquisition accuracy [1]

- Depends on sweeping approach …



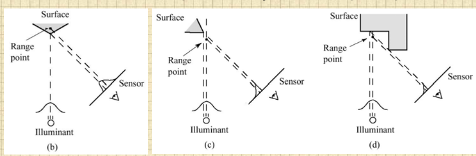
- … on surface curvature w.r.t. light direction …



- **Laser syst.**: the **reflected intensity** can be used as an estimate of the accuracy of the measure

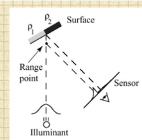
Acquisition accuracy [2]

- … on the surface shape nearby the sampled point



- … and on surface reflectance

[see Curless Levoy "…Space Time Analysis", '95]

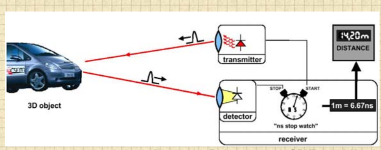


Optical Tech. – Time of Flight

Measure the **time** a light impulse needs to travel from the emitter to the target point (and back)

- Source: emits a **light pulse** and starts a **nanosecond watch**
- Sensor: detects the reflected light, stops the watch (**roundtrip time**)
- Distance = ½ time * lightspeed [e.g. 6.67 ns ➔ 1 m]

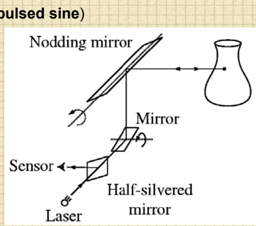
- **Advantages:** no triangulation, source and receiver can be on the same axis ➔ smaller footprint (wide distance measures), no shadow effects



[Image by R. Lange et al, SPIE v.3823]

Optical– Time of Flight [2]

- Optical signal:
  - **Pulsed light**: easier to be detected, more complex to be generated at high frequency (short pulses, fast rise and fall times)
  - **Modulated light** (sine waves, intensity): phase difference between sent and received signal ➔ distance (modulo wavelenght)
  - A combination of the previous (**pulsed sine**)
- Scanning:
  - single spot measure
  - range map, by rotating mirrors or motorized 2 DOF head



[Image by Brian Curless, Sig2000 CourseNotes]

Optical– Time of Flight [3]

In principle is an easy approach, **but:**

- **maximum distance range** limited by the amount of light received by the detector (power of the emitter, environment illumination)
- **accuracy** depends on : optical noise, thermal noise, ratio between reflected signal intensity and ambient light intensity
- Accurate and fast systems are costly:

System	Resolution XY	Accuracy Z (mm)	Scanning rate (pts/sec)	Cost US$
Cyrax 2500	0.25 x 0.25mm (@ 50m)	±6.0 (@ 50 m)	1K	125 K
Riegl LPM – 25HA	0.009°	±8	1K	40 K
3rdTech DeltaSphere 3000	0.03°	±8	25K	40 K

Time of Flight -- Systems [4]

3DV Systems - Visionary 3D Solutions	http://www.3dvsystems.com/
3rdTechDeltaSphere 3000	http://www.3rdtech.com/DeltaSphere.htm
Acuity Research Inc.	http://www.acuityresearch.com/
Cyra Technologies	http://www.cyra.com/
Noptel Measuring instruments	http://www.noptel.fi/nop_eng/measure.html
Origin Instruments	http://www.orin.com/
Perceptron Inc. - Sensing the Future	http://www.perceptron.com/
RIEGL	http://www.riegl.co.at/z210.htm
Welcome to UK Robotics	http://www.robotics.co.uk/

[List courtesy of J.A. Beraldin, NRC Canada]

Optical Technologies [5]

Advantages
- Non contact
- Cheap (low quality device)
- Safe (but should prevent object-scanner collision!)
- Fast

Disadvantages
- Expensive (high quality device)
- Acquire **only** the **visible surface properties** (no data on the interior, e.g. cavities)
- **Sensitivity** to surface properties:
  - transparency, shininess, rapid color variations, darkness (no reflected light), subsurface scatter, confused by inter-reflections

Active - CT scanners

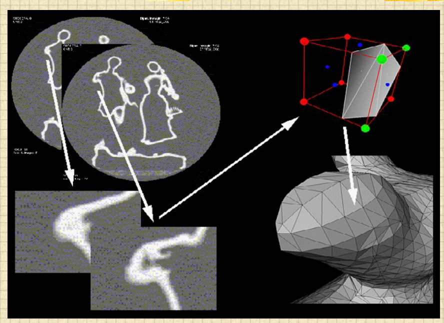
A 3D model can also be acquired by using a computerized tomographic (CT) device, e.g. a medical instrument ➔

Non-contact, transmissive approach

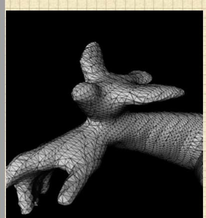
- Not only surface sampling, but real volumetric data (also data on internal structure)
- No data on surface detail (color)

Active - CT scanners [2]

Surface repr. is built by fitting an isosurface (MC)



Active - CT scanners [3]



Active - CT scanner [4]

CT Scanning

- *Advantages:*
  - A complete model is returned in a single shot, **registration and merging not required**
  - Output: **volume data**, much more than exterior surface
- *Disadvantages:*
  - Limitation in the **size** of the scanned object
  - **Cost** of the device
  - Output: no data on **surface attributes** (e.g. color)

Tutorial at Eurographics 2002:

3D Data Acquisition

Lesson 2:

Setting up a Scanning Lab

Michael Goesele
Max-Planck-Institut für Informatik

A Lab for an Off-the-Shelf Scanner

- requirements defined by the acquisition equipment
- often no sophisticated lab required
  - *enough space for the device*
  - *some ambient light (diffuse)*
  - *a suitable computer*
- sometimes even ad-hoc measurements possible
- capturing geometry plus some texture

Measuring more complex Object Properties

- geometry
- texture
- color
- reflection properties
- normals
- transparency
- ...

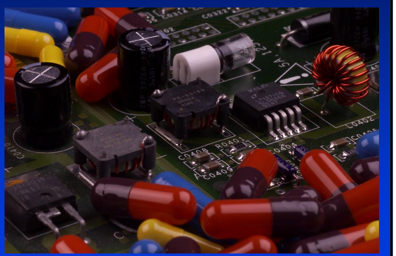
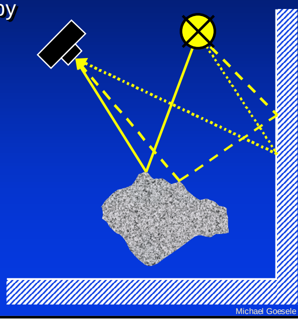


Image-based 3D Data Acquisition

- an object is illuminated by a light source and observed by a camera
- light interacts with
  - *the object*
  - *the environment*
  - *the environment and the object*
- influence of the environment should be small



A Lab for Image-based 3D Data Acquisition

- equipment
  - *cameras*
  - *lights*
  - *environment*
  - *some other useful items*
- experience
  - *building a lab*
  - *using a lab*

A Camera as Measurement Device

how to measure many different surface points?

- massively parallel sensor
- often high quality optical system
- tuned to make good pictures (except for scientific cameras)

⇒ image-based techniques

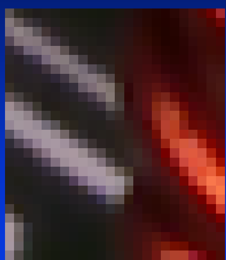
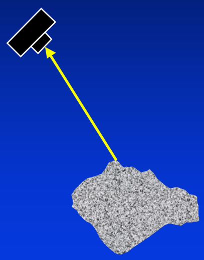


Image Acquisition

- high quality image information, e.g.
  - *high resolution*
  - *high color depth*
  - *high dynamic range*
  - *correct colors*
  - *...*
- known relation between position in space and image coordinates
  - ⇨ *geometric camera calibration*



Michael Goesele

The "Ideal" Camera

Accuracy
- precise optical and mechanical system
- high resolution, high color depth, high dynamic range
- images registered against the lens system
- no distortions
- no lossy compression techniques

Flexibility and Userfriendlyness
- flexible settings
- wide variety of good lenses
- "easy to use"
- remote controllable by a computer

Michael Goesele

Digital Cameras

- fast
- good repeatability
- natural registration of the images against lens system
- remote controllable
- often limited resolution
- artifacts possible due to
  - *lossy compression*
  - *color processing*

Michael Goesele

An Example: Kodak DCS 560

Camera Properties
- 35 mm SLR camera
- exchangeable lenses
- SDK for remote control via IEEE 1394 (FireWire)

Image Properties
- single chip CCD camera
- 12 bit per color channel
- 3040 x 2008 pixel resolution
- lossless compression

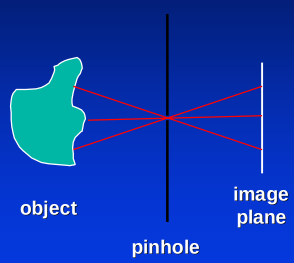
Michael Goesele

An Example: Kodak DCS 560

Limitations
- some parts cannot be controlled remotely (e.g. focus, flash)
  - ⇨ *custom hardware*
- limited dynamic range (about 10^3 - 60 dB)
- image noise

Michael Goesele

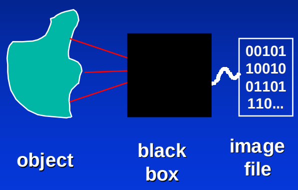
Pinhole Camera Model



- "each pixel corresponds to one ray through the pinhole onto the object"
- not valid for most digital cameras!!!

Michael Goesele

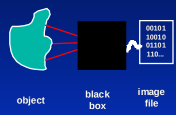
(Pessimistic) Digital Camera Model



- digital camera as a black box
- take only for granted what you measured (or what is given in the manual)

Michael Goesele

(Pessimistic) Digital Camera Model

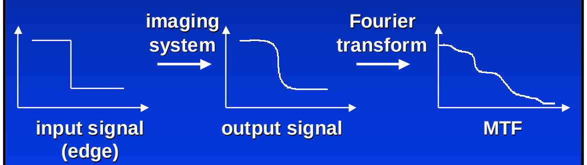


- optical lens system instead of pinhole aperture
- antialiasing filter (blur filter) to bandlimit the optical signal
- CCD/CMOS chip
  - *normally only one color per pixel (e.g. Bayer pattern)*
- camera image processing
  - *color reconstruction, sharpening*
  - *resampling (Nikon D1x: CCD 4024x1324, image 3008x1960)*
  - *noise removal, defect correction, ...*

Michael Goesele

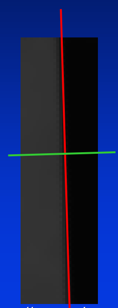
Modulation Transfer Function

- measures the extend to which image detail contrast is maintained by an imaging system
- Fourier transform of the point or line spread function



Michael Goesele

MTF Measurement: Slanted Edge Method



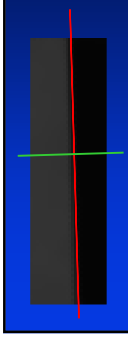
[Burns et al. 2001]

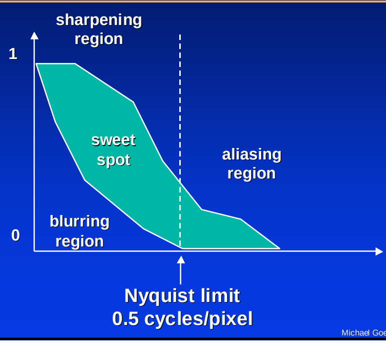
- start with a low contrast, slanted edge
- fit edge paremeters (red)
- project image onto a line perpendicular to the edge (green)
  - ⇒ high resolution edge profile
- calculate Fourier transform of profile
  - ⇒ MTF
- tools available at PIMA web site (search for sfrmat)

http://www.pima.net/standards/iso/tc42/wg18/wg18__sfrmat_matlab_page.htm

Michael Goesele

MTF Measurement: Interpretation

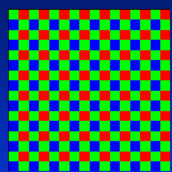




Michael Goesele

Demosaicing

- camera can record only one color per pixel
  - *exemptions: 3-chip cameras, new Foveon chip*
- Bayer pattern
  - *higher sampling rate in green channel*
  - *can be interpreted as luminance channel*
  - *larger sensitivity of the eye to luminance changes than to chrominance (color) changes*
- remaining two color values per pixel must be reconstructed

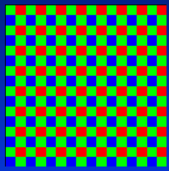


Bayer pattern

Michael Goesele

Demosaicing

- bad reconstruction leads to massive artifacts
- sensible approach:
  - *combining an interpolation and a pattern matching scheme*
  - *groups pixels into regions and makes some continuity assumption within the regions*
- "nice pictures", but no guarantee that two of the R,G,B values per pixel are correct

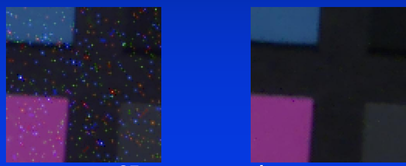


Bayer pattern

Michael Goesele

Noise and Noise Removal

- long exposure times (> 0.1 s) can lead to significant noise in images
- ideal: cooling the chip
- noise removal techniques to separate image data from noise



25 s exposure time

Michael Goesele

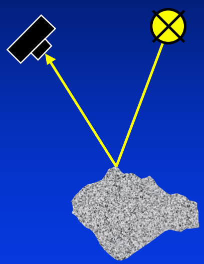
Lighting and Image Acquisition

Goal

- find relation between incoming and outgoing light at a surface point
- derive information from this data

Problems

- knowledge of and control over light sources needed

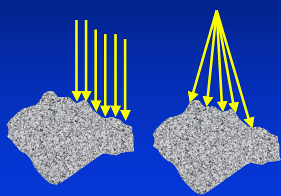


Michael Goesele

Lighting Requirements

Light Source Geometry

- well defined light source
- all incident light on a surface point comes from the same direction
  - parallel light source
  - point light source
- lens or reflector based systems are not ideal

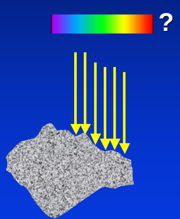


Michael Goesele

Lighting Requirements

Photometric Properties

- uniform distribution
- color constant over time
- even spectral distribution
- very bright
- high efficiency



Michael Goesele

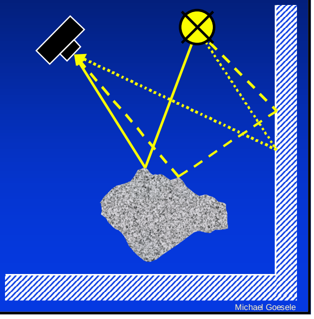
A Point Light Source

- HMI light source
  - *800 W*
  - *very efficient (equals 2500 W tungsten light)*
  - *(almost) daylight spectrum*
  - *constant colors*
  - *point light source*
- additional reflectors and diffusors for standard photographic applications

Michael Goesele

The Environment

- influence of the environment on the measurements should be as low as possible
  - *dark surrounding*
  - *no specular reflections*
  - *open space*



Michael Goesele

The Environment

- our photo studio
  - *walls and ceiling covered with black felt*
  - *black needle fleece carpet*
  - *photographic equipment is often already dark*

Michael Goesele

Other Useful Items

- working area outside the actual measurement lab
  - *computers, ...*
- stands, boxes, turntables for the objects
  - *see physics and chemistry school suppliers*
- various calibration targets
- computer controlled input and output devices
- lots of disk space

⇒ *depends on your application area*

Michael Goesele

Conclusion

lessons learned from our lab

- there is no single acquisition device for all purposes or for all objects
- requirements are often different from standard requirements
- off-the-shelf equipment is hard to find or not available
- new algorithms lead often to new and different requirements

⇒ *lab must be constantly adapted to the new requirements*

Michael Goesele

The End

Questions?

Michael Goesele

Tutorial at Eurographics 2002:

3D Data Acquisition

Lesson 3:

Basic Acquitision Techniques

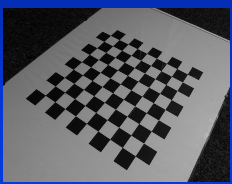
Michael Goesele
Max-Planck-Institut für Informatik

Overview

- calibration techniques
  - *geometric camera calibration*
  - *white balancing, ICC profiling, multispectral imaging*
- interpreting digital counts (pixel values)
  - *OECF*
  - *high dynamic range imaging*
- general lab procedures

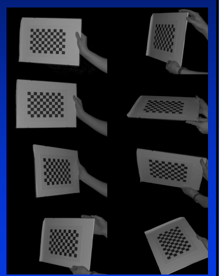
Geometric Camera Calibration

- get transformation between points in space and image coordinates
  - *intrinsic camera parameters (focal length, distortion coefficients)*
  - *extrinsic parameters (position, orientation)*
- several methods, e.g. [Tsai '87, Heikkila '97, Zhang '99]

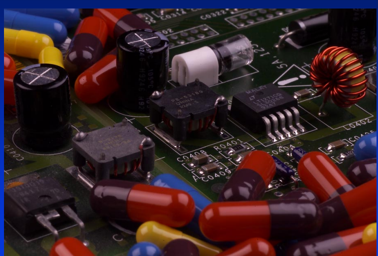


Geometric Camera Calibration

- capture images of calibration target with known dimensions
- extract feature points
- fit a global model for intrinsic camera parameters and local models for extrinsic parameters
  - *not all optimizations are stable*
- various calibration packages are available on the internet



Color Issues



What does it really look like?

White Balance

capture the spectral characteristics of the light source to assure correct color reproduction

scale color channels to assure neutral reproduction of gray/white

tungsten

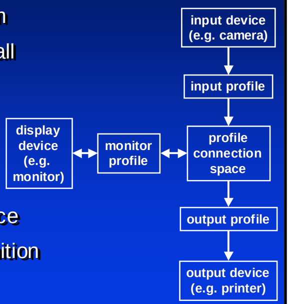
daylight

flourescent

flash

ICC Profiles

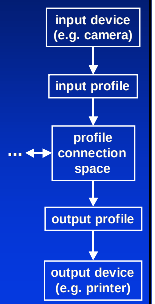
- color management system
- capture the properties of all devices
  - *camera and lighting*
  - *monitor settings*
  - *output properties*
- common interchange space
- sRGB standard as a definition of RGB



Michael Goesele

ICC Profiles

- profile connection spaces
  - *CIELAB (perceptual linear)*
  - *linear CIEXYZ color space*
- can be used to create an high dynamic range image in the profile connection space
- allows for a color calibrated workflow
- more information at http://www.color.org



Michael Goesele

Creating an ICC Profile for a Digital Camera

- put the camera in a defined state (calibration)
- create a a image of a test target under the same illumination conditions as used later on is captured
- the image is analyzed by a profile generation software
- the generated ICC profile can be used in a color management systems

Michael Goesele

Multispectral Imaging

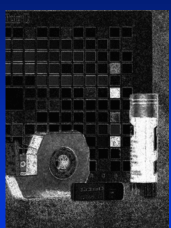
- not only tristimulus values
- 5-8 color channels or even a complete spectrum per pixel
- approaches
  - *dense sampling with narrow band filters*
  - *combining a standard trichromatic camera with absorption filters, PCA analysis of sample color set*
- scene can be reconstructed under arbitrary lighting conditions

Michael Goesele

Multispectral Imaging Example

captured under HMI illumination

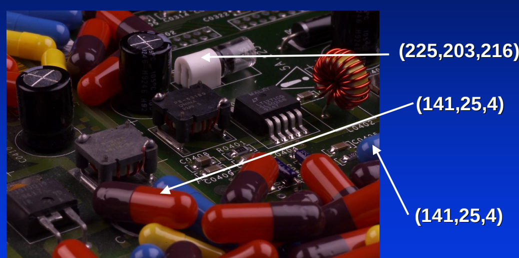
captured under tungsten illumination



color error between both images

Michael Goesele

Interpreting Digital Counts (Pixel Values)



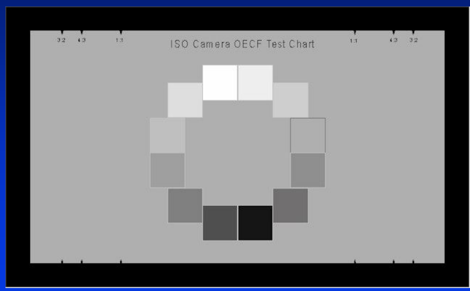
What do these RGB values mean?

Michael Goesele

Camera Response Curve (OECF)

- relationship between digital counts and luminance is unknown (and often non-linear)
  - *gamma correction*
  - *image optimizations*
  - *...*
- can be described by response curve or OECF (Opto-Electronic Conversion Function)
- direct measurement via test chart
  - *uses patches with known gray levels*
  - *patches arranged in a circle*

OECF Test Chart



High Dynamic Range Imaging

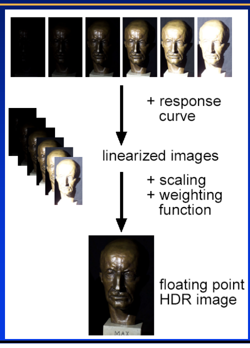
- limited dynamic range of cameras is a problem
  - *shadows are underexposed*
  - *bright areas are overexposed*
  - *sampling density is not sufficient*
- some modern CMOS imagers have a higher and often sufficient dynamic range than most CCD imagers

High Dynamic Range Imaging

general idea of High Dynamic Range (HDR) imaging:

- combine multiple images with different exposure times
  - *pick for each pixel a well exposed image*
  - *response curve needs to be known*
  - *don't change aperture due to different depth-of-field*

High Dynamic Range Imaging



High Dynamic Range (HDR) Imaging

- analog film with several emulsions of different sensitivity levels by Wyckoff in the 1960s
  - *dynamic range of about 10^8*
- commonly used method for digital photography by Debevec and Malik (1997)
  - *selects a small number of pixels from the images*
  - *performs an optimization of the response curve with a smoothness constraint*
- newer method by Robertson et al. (1999)
  - *optimization over all pixels in all images*

HDR Imaging: Algorithm of Robertson et al.

Principle of this approach:

- calculate a HDR image using the response curve
- find a better response curve using the HDR image

(to be iterated until convergence)

HDR Imaging: Algorithm of Robertson et al.

input:

- series of **i** images with exposure times $\mathbf{t_i}$ and pixel values $\mathbf{y_{ij}}$
- a weighting function $\mathbf{w_{ij} = w_{ij}(y_{ij})}$ (bell shaped curve)
- a camera response curve $\mathbf{I_{ij}}$
  - *initial assumption: linear response*

⇒calculate HDR values $\mathbf{x_j}$ from images using

$$x_j = \frac{\sum_i w_{ij} t_i I_{y_{ij}}}{\sum_i w_{ij} t_i^2}$$

HDR Imaging: Algorithm of Robertson et al.

optimizing the response curve $\mathbf{I_{ij}}$ resp. $\mathbf{I_m}$:

- minimization of objective function **O**

$$O = \sum_{i,j} w_{ij} (I_{y_{ij}} - t_i x_j)^2$$

using Gauss-Seidel relaxation yields

$$I_m = \frac{1}{\mathrm{Card}(E_m)} \sum_{i,j \in E_m} t_i x_j$$
$$E_m = \{(i,j) : y_{ij} = m\}$$

- normalization of **I** so that $\mathbf{I_{128}}$=1.0

HDR Imaging: Algorithm of Robertson et al.

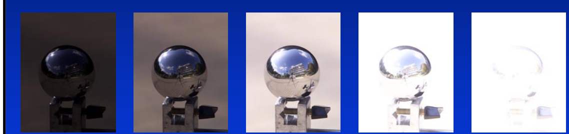
both steps

- calculation of a HDR image using **I**
- optimization of **I** using the HDR image

are now iterated until convergence

- *criterion: decrease of **O** below some threshold*
- *usually about 5 iterations*

HDR Imaging Example: Capturing Environment Maps



1/2000s 1/500s 1/125s 1/30s 1/8s

series of input images

HDR Imaging Example: Capturing Environment Maps

General Lab Procedures

MPI INFORMATIK

- a lot depends on experience
- some rules for doing experiments
  - *similar to rules in Physics, Chemistry labs*
- important to ensure
  - *safety*
  - *efficiency*
- here: some hints but not a final, perfect, ... list

Michael Goesele

1. Safety I

MPI INFORMATIK

Know how the equipment can hurt you!

- safety rules for exist for all potential dangerous devices
- read the manual
- in addition: use common sense!
- examples:
  - *lasers*
  - *lamps*
  - *heavy objects*
  - *...*

Michael Goesele

2. Safety II

MPI INFORMATIK

Know how you can hurt the equipment!

- read the manual
- in addition: use common sense!
- ask your local lab guru
- if necessary: talk to the engineers
  - *they know best what can be done and what shouldn't be done with their products*

Michael Goesele

2. Safety II

MPI INFORMATIK

Some (odd) examples from our experience:

- replaceing batteries in a flash with an AC adapter
  - *the current must be limited to avoid destruction of capacitors*
- taking long exposed images with one of our digital cameras
  - *destroys the electronic system due to overheating*
  - *cooling the camera with a fan helps somewhat*
  - *additional benefit: image noise is reduced*

Michael Goesele

3. Original Data

MPI INFORMATIK

Very (most?) important non-safety rule:

Save all original data!

- Sysadmin's rule:
  - *Delete everything that can be created by a simple "make" command!*
- from this follows:
  - *Don't delete anything else!*

Michael Goesele

3. Original Data

MPI INFORMATIK

Very (most?) important non-safety rule:

Save all original data!

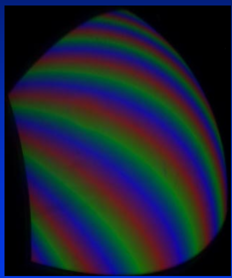
- original data often really hard or impossible to recreate
- storage space is cheap
- you never know which errors you made
- you might be able to apply better techniques to your original data later on

Michael Goesele

3. Original Data

An example:

- an error was discovered
- the original was lost
- only a processed version was available
- the experiment had to be repeated (2 working days)



Michael Goesele

4. Metadata

Another important rule:

Create, save, and preserve all metadata that is possibly needed!

What is metadata?

- data about data
- giving additional information not contained in the actual data set

Michael Goesele

4. Metadata

An example for an image:

```
...
# @@image_type =  DCRGBImage24
# @@temperature = 23
# @@camera_determined_illum = DC5xx_CAM_ILLUM_DAYLIGHT
# @@photographer_balance_mode = DC5xx_BAL_MODE_PRESET
# @@exposure_index = 80
# @@f_number = 16.0
# @@exposure_time = 1/8
# @@focal_length = 50.0
# @@noise_filter = 1
...
```

Michael Goesele

5. Equipment I

Use the right equipment!

- our experience: we always have different requirements than the “average user” such as
  - *getting measurements and not “good images” from a digital camera*
  - *black carpets*
- equipment quality should be consistent
  - *high-end camera with cheap lenses doesn’t make sense*
  - *buy a good camera and good lenses instead*

Michael Goesele

6. Equipment II

Know your equipment!

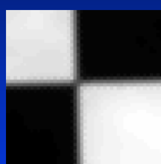
- its properties
- its strengths
- its weaknesses

- many assumptions about cameras, ... are approximations that are only valid under certain conditions
  - *our applications often violate these assumptions*

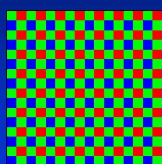
Michael Goesele

6. Equipment II

Some examples:



blurring



Bayer pattern for sampling

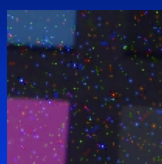


image noise (CCD chip)

Michael Goesele

The End

MPI INFORMATIK

Questions?

Please visit us at
http://www.mpi-sb.mpg.de

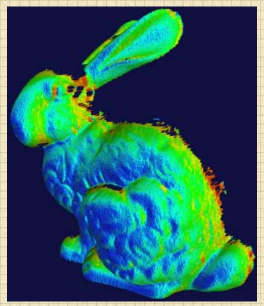
Tutorial T1: 3D Data Acquisition

"3D Scanning Pipeline - *Registration and Merging*"

Roberto Scopigno
Visual Computing Group
ISTI – C.N.R.
Pisa, Italy

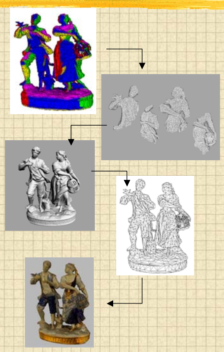
3D Scanning

The acquisition of a single **range map** is only an intermediate single step of the overall acquisition session



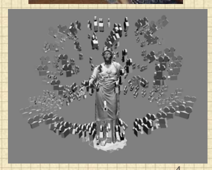
The 3D scanning pipeline [1]

- **3D Scanning**:
  - *[**Acquisition planning**]*
  - **Acquisition** of multiple range maps
  - Range maps **Editing**
  - **Registration** of range maps
  - **Merge** of range maps
  - Mesh **Editing**
  - Geometry **simplification**
  - **Capturing appearance** (color acquisition, registration, mapping on surface, color-preserving simplification)
  - Archival and data conversion



Acquisition Planning [1]

- Selecting the set of views is not easy
- An example: **Scanning the Minerva**
  - Bronze statue, Archeological Museum Florence (under restoration), 155 cm
  - 3 acquisitions with different scanners (2000-2002)
  - Last: scanned with Minolta laser scanner (03/2002)
    - No. range scans: **297**
    - Sampling resolution: **~0.3 mm**
    - Scanning time: **1,5 days**



Acquisition Planning [2]

Definition of the **optimal acquisition patchwork**:

- Given: scanner & object characteristics
- Obtain an **optimal** coverage (all object surface covered):
  - **Minimal number** of scans
  - Sufficient **inter-scan overlap**
  - With each scan:
    - shot from a view direction **nearly orthogonal** to the surface
    - **physically feasible** (consider potential collisions with the object/environment, self-occlusions)
- Neglected phase – ***any supporting tool available***

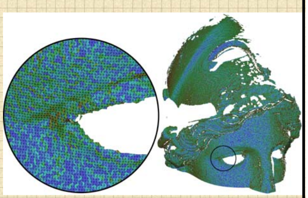
Acquisition Planning [3]

- Recent research results:
  - Given CAD model of the object, compute an optimal sensing plan (to improve acquisition accuracy) [Prieto+99]
  - Given incomplete scanned model, get an optimal selection of the next_best_viewpoint [Morooka+99]
  - A proposal for optimal texture acquisition (given a known shape) [Matsushita+99]
- Open problem:
  - Few research proposal (usually, find next best view)
  - Any supporting tool available

The input data

- Sampling **density / distribution** depends on the scanner characteristics
- Often, range data are given in a regular form (**range images)**
- We can manage range data as:
  - Unstructured point set, **point clouds**
  - Multiple **range surfaces**

- Converting a **range image** into a **range surface:** easy because points are regularly distributed, topology is implicit



The input data [2]

- Triangulation of range images: a **tessellation threshold** is introduced to avoid erroneous connection of "adjacent" samples which corresponds to non-adjacent pieces of surface

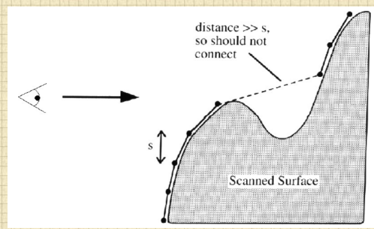
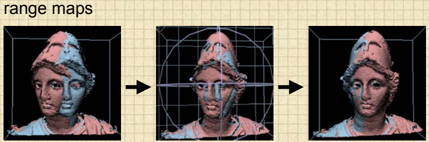


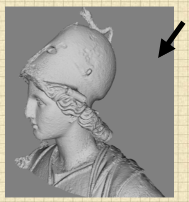
Image by Brian Curless, Sig'2000 Course Notes

Registration and Merging

First: **Register** all range maps

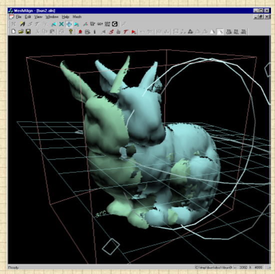


Second: **Merge** in a single triangulated surface with no redundancy



Registration [1]

- Independent scans are defined in coordinate spaces which depend on the spatial locations of the **scanning unit** and the **object** at acquisition time
- They have to be **registered** (roto-translation) to lie in the same space
- Standard approach:
  1. initial **manual** placement
  2. **Iterative Closest Point** (ICP) [Besl92,CheMed92]



In the figure:
MeshAlign 1.0
(C) Visual Computing Group

Registration [3]

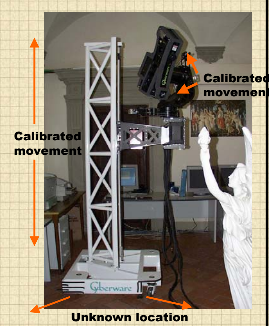
- Accuracy of the registration is a critical issue
- It can be based on:
  - Calibrated scanner positioning
  - Software optimization
  - Both Calibrated + Software opt.
- *"Ideal word":*
  - The scanner positioning system gives sufficient information to directly compute the registration matrix

BUT

  - Calibration system has to be very accurate, and this increases substantially the cost of the scanner (e.g. Michelangelo project)
  - It is not easy to provide an accurate system which detects any possible movement of the scanner w.r.t. the object

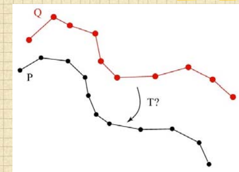
Registration [4]

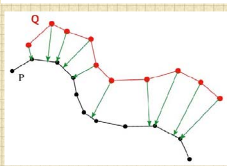
- An example: the Digital Michelangelo scanner
  - Calibrated scanner head vertical movement (on the gantry) and orientation
  - Does not detect the gantry location on the floor
  - It uses a mixed registration approach



Pairwise Registration [1]

- **Registration as an optimization:**
  - Start with an initial **approximate registration**, obtained:
    - Manually
    - Via sensor tracking
    - Automatic (possible only for objects with well defined and easy to detect features)
  - Given two overlapping range scans, we wish to find the rigid transformation **T** that **minimizes** the **distance** between them



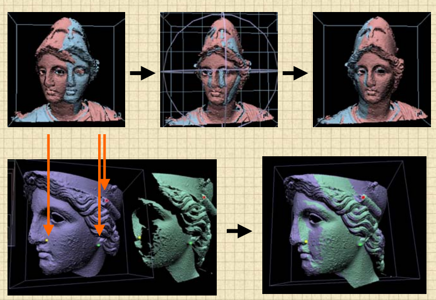


Pairwise Registration [2]

Initial registration with **user intervention:**

Mode 1) The user manually places a range map over another (interactive manipulation)

Mode 2) Selection of multiple pairs of matching points

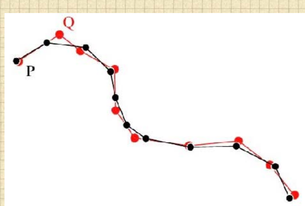


Pairwise Registration [3]

- An approximation to the distance between range scans is:

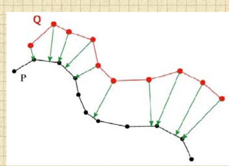
$$E = \sum || T\, q_i - p_i ||^2$$

where the $\mathbf{q_i}$ are samples from scan **Q** and the $\mathbf{p_i}$ are the *corresponding* points of scan **P**.



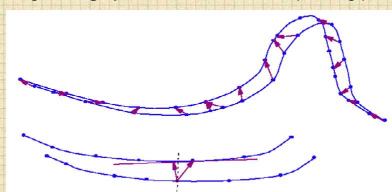
Pairwise Registration [4]

- If the correspondences are known a priori, then there is a closed form solution for ***T***. However, the correspondences are not known in advance.
- **Iterative closest point** (ICP) [Besl+92]
  - Start from an approximate registration
  - Repeat
    - Identify corresponding points (minimal distance)
    - Compute and apply the optimal rigid motion ***T***
  - Until registration error **E** is small



Pairwise Registration [5]

- This approach is troubled by slow convergence when surfaces need to slide along each other.
- Chen and Medioni [Chen+92] describe a method that does not penalize sliding motions, by constraining points to stay in proximity of corresponding **average planes** instead of corresponding points



Pair-wise vs Global Registration

- We have described an approach that allows to register **pairs** of range scan → ***local registration***
- A set of range map can be registered pairwise, but pairwise alignment leads to **accumulation of errors** when walking across the surface of an object
- An optimal solution should minimize distances between all range scans ***simultaneously***

  This is sometimes called the ***global registration*** problem

Global Registration [1]

Global Registration

- A simple approach: add another loop to ICP

```
while (NOT DONE) {
    for each range map {
        for each point {
            find closest points in all other range maps
            }
        move the current surface closer
    }
```

→ ***Very slow!!***
→ ***Lot of memory!!***

Global Registration [2]

- Many different global registration approaches [Bernardini00]
  - Global optimization
    - Simulated annealing *[Blais, Levine 95]*
    - Springs *[Stoddart, Hilton 96]*
    - Hierarchical linearized least squares *[Neugebauer 97]*
  - Incremental
    - Iterate moving one scan, keep others fixed *[Bergevin et al 96]*
    - Speed up using hardware *[Benjemaa, Schmitt 97]*
    - **Keep pairwise matches** *[Pulli 99]* ←

Global Registration [3]

Global Registration [Pulli99]

- Uses output of pairwise registration (final set of corresponding points) as constraints in the multi-range_maps registration phase
- Registration error are spread among many range maps, rather than accumulated
- Efficient (time+space) :
  - The global step works on sets of corresponding points, not on the entire dataset

Global Registration [4]

Pulli's method:

- **Local** – Pairwise registration
  - Builds a **graph**, with a **node** for each range map and an **arc** for each pair of overlapping range maps
  - For each arc, iterate **ICP** until registration accuracy < ε
    - Select set of matching points (point-plane approach) ← **Most expensive step**
    - Minimize distances
  - Assign the last matching point set (point to point) to the arc
- **Global** – Multi-rangemap registration:
  - For all arcs in the graph
    - Build the union of the matching point sets (point to point)
    - Minimize globally the sum of squared distances

Using Color in Registration [1]

Color-- enhanced registration

- **If** the scanner acquires also an RGB texture (calibrated with the range map)

  Then available color info can be used to improve/simplify the registration

- Improves results on range maps with few geometric features

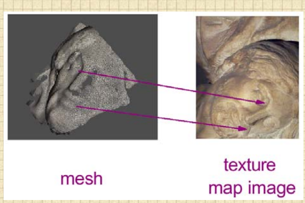


Image by F. Bernardini

Using Color in Registration [2]

Using Color in registration **[Bernardini00]**

- **Initial Registration**
  - **Manual** (user selects of matching point, with automatic improvement)
  - **Automatic** (by identification of texture feature points)
- **Refinement of Registration**
  - **Instead of ICP** (align textures projected in a common view, by image cross-correlation)
  - **During ICP** (consider each point in 6D space XYZRGB)
  - **After ICP** (to further refine the ICP result, aligning feature texels)

Using Color in Registration [3]

- An example of **color-refined registration** [Bernardini00]
  - Shape: a vase, lacking sufficient geometric features
  - After IPC, the rigid transformation is improved by aligning textures feature points

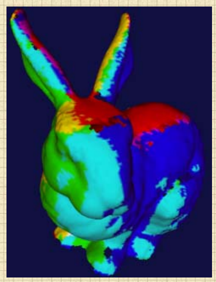
before after

Image by F. Bernardini

Merging [1]

- **Merge (or fusion):**
  - Once registered, all scan have to be fused in a single, continuous, hole-free mesh

Output of the **Marching Intersection** code
(C) Visual Computing Group



Merging [2]

Desirable properties for surface reconstruction:

- No restriction on topological type
- Representation of **range uncertainty**
- Utilization of all range data (integrate over overlapped regions)
- Incremental and order independent updating
- Time and space efficiency
- Robustness
- Ability to fill holes in the reconstruction

Merging [3]

Many reconstruction methods proposed:

- Reconstruction from point clouds
  - any previous info on samples adjacency (unorganized points)
  - very general
  - often slower
- Reconstruction from range maps
  - use all available information (adjacency in range maps, orientation, weighted evaluation of samples)
  - less general
  - more robust and accurate (weighted blending)

Merging [4]

Reconstruction from point clouds:

- Methods that construct **triangle meshes directly**:
  - *Local Delaunay triangulations* [Boissonat84]
  - *Alpha shapes* [Edelsbrunner+92]
  - *Crust algorithm* [Amenta+98]
  - *Delaunay-based sculpturing* [Attene+00]

  (Global Delaunay)

  - ***Ball Pivoting*** [Bernardini+99] ←
  - *Localized Delaunay* [Gopi+00]

  (Local Methods)
- Methods that construct **implicit functions**:
  - *Voxel-based signed distance functions* [Hoppe+92]
  - *Bezier-Bernstein polynomials* [Baja+95]
  - ***Radial Basis Functions*** [Carr+01]

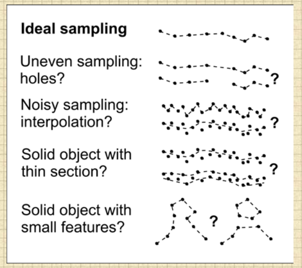
Merging [5]

Reconstruction from range maps:

- Methods that construct **triangle meshes** directly:
  - Re-triangulation in projection plane [Soucy+92]
  - **Zippering** in 3D [Turk+94] ←
- Methods that construct **implicit functions**:
  - Signed distances to nearest surface [Hilton+96]
  - Signed distances to sensor + space carving [Curless+96] ←
  - Marching Intersections [Rocchini+00] ←

Merging [6]

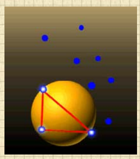
Sampling quality and reconstruction issues:



Merging – Ball-Pivoting

- Region growing paradigm: [Bernardini99]

  Build an initial seed face
  LOOP
  Pop next open edge (v_i, v_j)
  Pivot a ball around boundary edge, until it touches a sample v_k
  Create new triangle (v_i, v_j, v_k)
  Push new open edges (v_i, v_k) (v_j, v_k)
  UNTIL no more open edges



- Efficient: 15M triangles in 30 mins, 100MB memory on Pentium II 450Mhz

Merging - Zippering

Mesh Zippering [Turk94]

- One of the methods which combine range surfaces by stitching polygon meshes together.
- Overview:
  - Tessellate range images and assign **weights** to vertices
  - Detect **overlapping sections** and remove redundant triangles
  - **Zipper** meshes together
  - Extract a **consensus geometry**

Merging – Zippering [2]

Weight assignment

- Final surface will be **weighted combination** of range images
- Weights are assigned at each vertex to:
  - Favor views with higher sampling rates (view \\ to surface normal)
  - Encourage smooth blends between range images (low weight in the vicinity of mesh boundary)

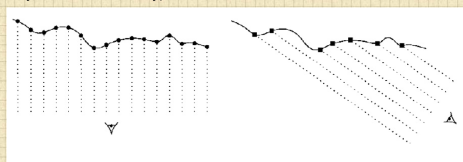
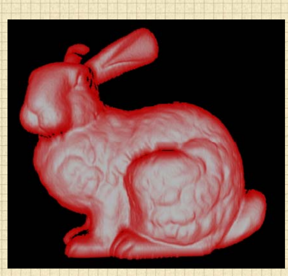


Image by Brian Curless, Sig'2000 Course Notes

Merging – Zippering [3]

- An example of **weight visualization** on a single range mesh
  - Weight value mapped with color (red=low weight)



Merging – Zippering [4]

- Redundancy removal and zippering

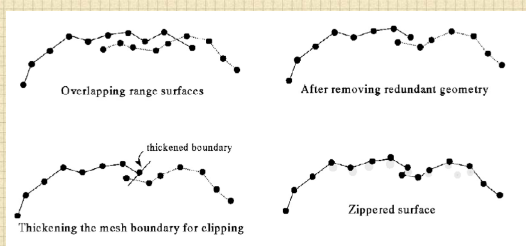


Image by Brian Curless, Sig'2000 Course Notes

Merging – Zippering [4]

- Finding a **consensus** geometry

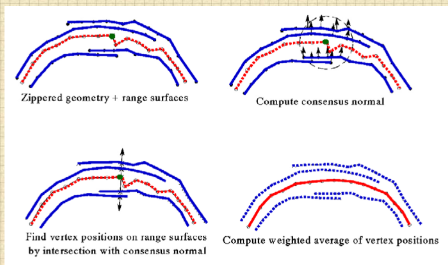


Image by Brian Curless, Sig'2000 Course Notes

Merging – Zippering [5]

- Example of quality improvements introduced by the search of the **consensus** geometry

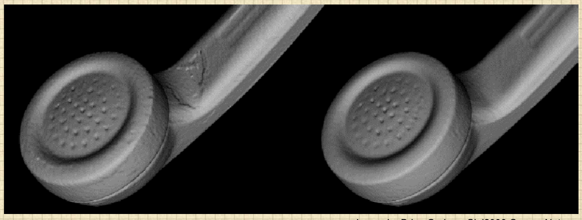
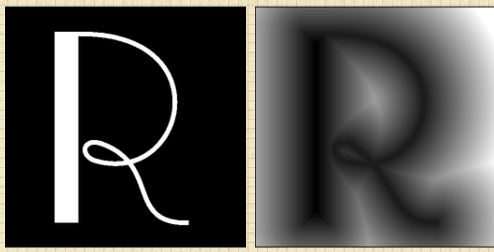


Image by Brian Curless, Sig'2000 Course Notes

Merging – Volume-based

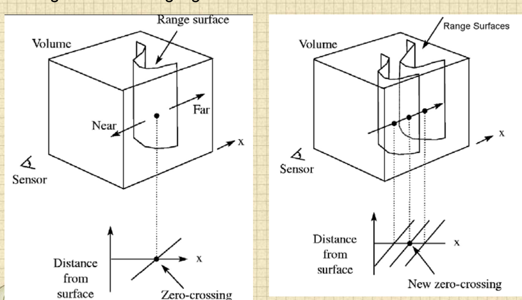
- Combining the meshes **volumetrically** can overcome difficulties of stitching polygon meshes
  - Fast and robust
  - Sensitive to range scan characteristics
  - User-defined voxel size
- Overview [Curless+96]
  - Convert range images to **signed distance functions**
  - Combine signed distance functions
  - Carve away empty space
  - Extract hole-free isosurface (isosurface fitting by MC)

Distance Fields



Merging – Volume-based [2]

Evaluating and combining signed distances



The Marching Cube Algorithm

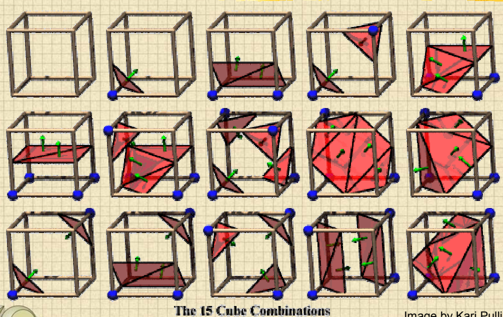


Image by Kari Pulli

Merging – Volume-based [3]

Hole filling

- Unseen portions appear as holes in the reconstruction
- A hole-free mesh is useful for:
  - Fitting surfaces to meshes
  - Manufacturing replicas (e.g., stereolithography)
  - Aesthetic renderings
  - Processing (eg. evaluation of measures)

Merging – Volume-based [4]

Holes can be removed:

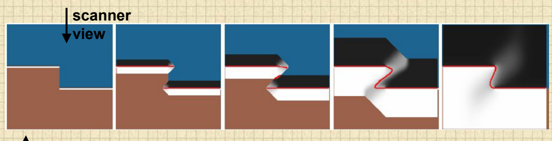
❶ Merging time:
- construct an hole-free distance volume (**space carving**), by taking into account the known visual hull and the scanner view direction
- easy to implement **but** increased reconstruction time

❷ Postprocessing:
- explicit triangulation of holes
- not easy to implement, **robustness** is an open issue (complex hole shape)
- slow (usually, user-controlled)
- patches can take into account surface curvature in the surrounding frontier region (e.g. spline fitting, subdivision surfaces)

Image by Levoy et al.

Merging – Volume-based [5]

Space Carving [Curless+96, Davis+02]
- building a distance volume which allows automatic hole filling at fitting time



Space Carving [Curless+96]

plus volumetric diffusion of the distance field [Davis+02]

Merging – Volume-based [6]

- Voxel space resolution: don't use too small cubes
  - Size >= max of
    - registration error
    - sampling error
    - sampling density
- Use of directional information allows reconstructing thin objects

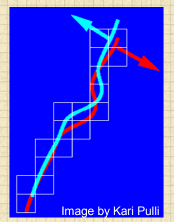


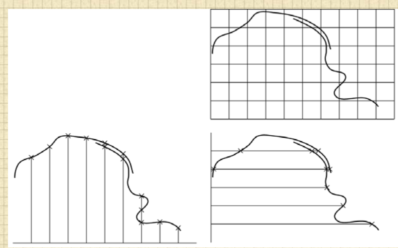
Image by Kari Pulli

Merging – Marching Intersections

- Yet another volumetric method… [Rocchini+00]
  - Methods based on Distance Volume → **voxel-based**
  - Marching Intersection → **cell-based**
- Marching Intersections
  - Instead of building a distance volume, evaluates the **intersections** of the range surfaces with the edges of the discrete space
  - Merge nearly coincident intersections, according to a threshold ε
  - Reconstruct the surface on each cell, from the intersection pattern (inverse Marching Cubes)

Merging – Marching Intersections [2]

- Compute the intersection between the range surfaces and the grid lines of a cell-based volume:
  - Done independently on the 3 set of parallel grid lines (X, Y, Z)
  - Stored on 3 sets of lists (compact repr., no explicit repr. of the volume grid)

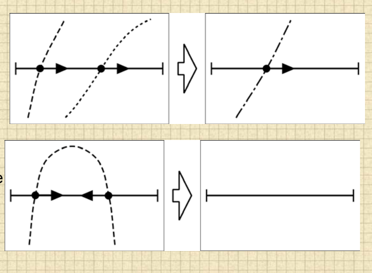


Merging – Marching Intersections [3]

- For each **grid line**, all intersections at distance lower than a given threshold are:
  - **Fused** if concordant normal (weighted fusion)

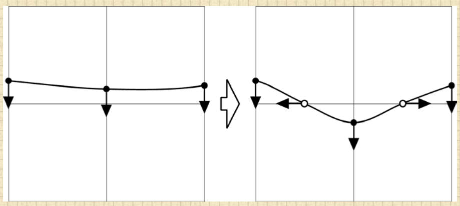
  or

  - **Removed** if lie on the same range map and normals are discordant



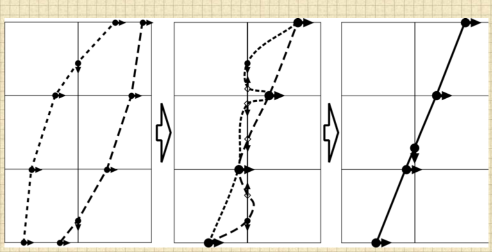
Merging – Marching Intersections [4]

- **Fusion** implies moving intersections.
- When an intersection is moved from a cell to an adjacent one, new "virtual" intersections have to be introduced to maintain data integrity



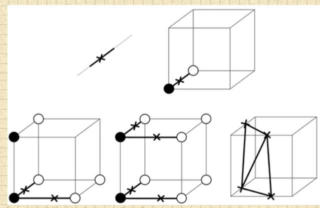
Merging – Marching Intersections [5]

- A complete merging sequence:



Merging – Marching Intersections [6]

- Once all possible fusion/removal steps have been performed on the grid lines, we reconstruct the corresponding isosurface:
  - For each **cell**:
    - Find intersections on its edges (patch geometry)
    - Compute the corresponding MC vertex configuration
    - Assign to the corresponding MC patch topology the known geometry

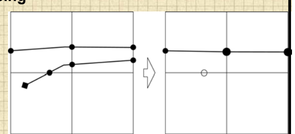


Merging – Marching Intersections [6]

Anomalies and holes

- Non-correct configurations can be obtained on some cells, due to **multiple** or **missing** intersections, e.g. due to:
  - **boundary faces** of the range maps
  - **holes** in the range maps

  Most of them are disambiguated by considering adjacent cells



- Holes are detected (open boundaries) and can be filled on the fly

Merging – Marching Intersections [8]

- Results
  - PentiumII 350MHz, 512MB RAM
  - Times in sec., include I/O

Dataset	# Scans	# Input Triangles	Grid Size	# Output Triangles	Time (mm:ss.d)	Memory Usage (Mb)
Bunny	10	693,807	109x107x84	77,461	00:04.6	3
			338x335x263	774,069	00:36.9	30.5
Dragon	71	3,497,403	244x209x195	138,949	00:34.2	20.5
			712x501x322	890,199	16:30.0	151
Happy Buddha	58	5,826,721	190x280x187	178,229	01:23.3	19
			407x957x407	1,352,872	10:33.2	140

[10]

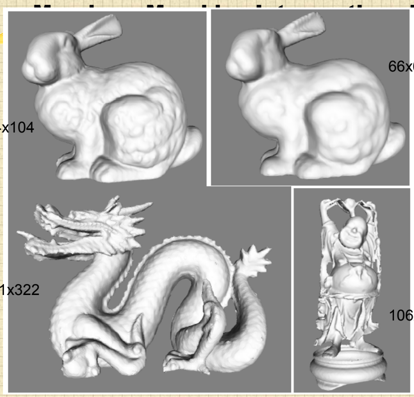
- Results:

142x134x104

66x62x48

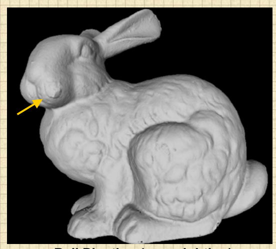
712x501x322

106x264x176

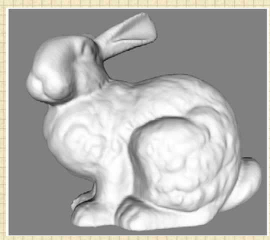


Merging

- A "visual" comparison



Ball Pivoting (no weighting): **ρ=0.7mm, 452K tr., 92MB, 92 sec.** (P450MHz)



March.Inters.(weighted composition): **338x335x263 (vox=0.45mm), 774K tr., 30MB, 36 sec.** (P350MHz)

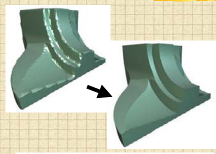
Merging – Marching Intersections [11]

Merging range surfaces via Marching Intersection

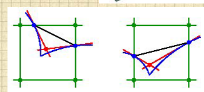
- Pro's
  - Geometric **accuracy:** output mesh geometry is ON range surface, ***vs.*** range→ dist.voxels → MC interpolat.
  - Time & space **efficient**
  - **No** staircase **discontinuity** on overlapping regions (weighted fusion)
- Con's
  - Not straightforward to **implement**
  - Sensible to alignment inaccuracies

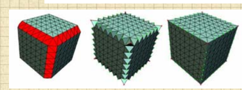
Merging

Improved reconstruction of **feature edges** [Kobbelt+01]



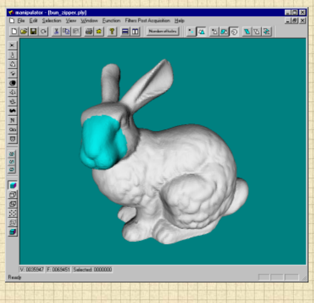
- Replaces scalar distances with **directed distances** (in the x,y,z directions)
- Works in three steps:
  - cells that contain a **feature** are identified
  - then, **one new sample** is included per cell
  - one round of edge flipping reconstructs the feature edges





Mesh Editing

- Geo-topological editing needed on:
  - range maps
  - merged mesh
- Apply smoothing filters
- Correct the mesh (hole filling, topology checks and edits, noise removal, etc)



In the figure:
Manipulator
(C) Visual Computing Group

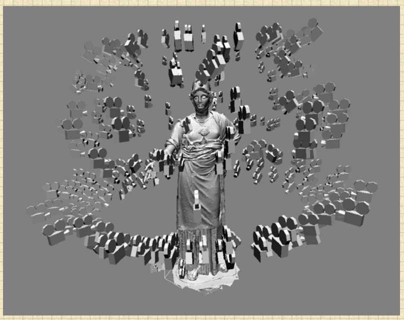
Scanning the Minerva [1]

Minerva of Arezzo

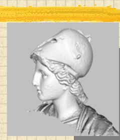
- Bronze statue, Archeological Museum Florence (under restoration), height 1.55 m
- 3 acquisitions with different scanners (2000-2002)
- Last: scanned with Minolta laser scanner (03/2002)
  - No. range scans: **297**
  - Sampling resolution: **~0.3 mm**
  - Scanning time: **1,5 days**

Scanning the Minerva [2]

- Scanner positions during the acquisition



Scanning the Minerva [3]



Post-processing

- Registration of range maps
  - Registration Time: **8 days (>80%)**
- Model after align/merging:
  - Voxel size: **0.5 mm**
  - Output mesh: **25 M faces**
- Models after simplification:
  - From **15 M** down to **20K faces**
  - Simplification time: **~ 30 min**

Lessons learned

- Scanning:
  - Producing a complete sampling is often impossible
  - Tools for on-line selection of best views needed
- Registration:
  - Most time-consuming step in the scanning process
  - Requires an operator:
    - sufficiently expert
    - patient (it is a boring phase)
    - precise
  - Any improvement in the **automation** of the process is welcomed
- Merging:
  - Should blend range maps on overlapping regions

Bibliography [1]

[Bernardini+00] Bernardini F., Rushmeier H., "The 3D Model Acquisition Pipeline", EG2000 STAR, pp.41-62.

[Curless+00] Curless, Seitz et al., "3D Photography", Course 19, Course Notes for SIGGRAPH 2000

[Amenta+98] Amenta, N., et al. "A new voronoi-based surface reconstruction algorithm," SIGGRAPH '98. p. 415-421.

[Attene+00] Attene M., Spagnolo M., "Automaatic surface reconstruction from point set in space", Computer Graphics Forum, vol.19(3), pp.457-466.

[Bajaj+95] Bajaj, C.L., Bernardini, F. and Xu, G, "Automatic reconstruction of surfaces and scalar fields from 3D scans," SIGGRAPH '95, pp. 109--118.

[Bernardini+99] Bernardini, Mittleman, Rushmeier, Silva, Taubim, "Ball-pivoting algorithm for surface reconstruction", IEEE TVCG vol.5(4), 1999, pp.349-359.

[Besl+92] Besl, P.J. and McKay, H.D., "A method for registration of 3-D shapes," IEEE Transactions on Pattern Analysis and Machine Intelligence, Feb. 1992, 2(4), pp. 239--256.

[Boissonnat 84] Boissonnat, J.-D, "Geometric Structures for Three-Dimensional Shape Representation," ACM TOG, October, 1984, 3(4), pp. 266-286.

[Carr+01] Carr et al, "Reconstruction and Representation of 3D Objects With Radial Basis Functions", Siggraph 2001, pp. 67-76.

[Curless+96] Curless, B. and Levoy, M., "A volumetric method for building complex models from range images." In Proceedings of SIGGRAPH '96, pp. 303-312.

[Chen+92] Chen, Y. and Medioni, G., "Object modeling by registration of multiple range images," Image and Vision Computing, 10(3), April, 1992, pp. 145-155.

[Davis+02] Davis, Marshner, Garr, Levoy "Filling holes in complex surfaces using volumetric diffusion", Int. Symp. On 3D Data Porcessing, Visualiz. and Transmission, pp.428-438, 2002.

[Kobbelt+01] L. Kobbelt, M. Botsch, U. Schwanecke, H.-P. Seidel, "Feature Sensitive Surface Extraction from Volume Data", Siggraph 2001.

Bibliography [2]

[Gopi+00] Gopi M., Krishnan S., Silva C.,"Surface reconstruction based on lower dimensional localized delaunay triangulation", Computer Graphics Forum, vol.19(3), pp.467-478

[Edelsbrunner+92] Edelsbrunner, H., Mucke, E.P. "Three-dimensional alpha shapes." ACM TOG (Jan. 1994) vol.13, no.1, pp. 43-72.

[Hilton+96] Hilton, A., Stoddart, A.J., Illingworth, J., and Windeatt, T, "Reliable Surface Reconstruction from Multiple Range Images," Fourth European Conference on Computer Vision, April 1996, vol. 1, pp. 117-126.

[Hoppe+92] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W., "Surface reconstruction from unorganized points." SIGGRAPH '92. pp. 71-80.

[Lamb+99] Lamb et al., An automation system for industrial 3D laser digitizing, 3DIM'99 Proc., pp.148-157.

[Matsushita+99] Matsushita, Kaneko, "Efficient and Handy Texture Mapping on 3D Surfaces", Eurographics '99, pp. 349-358.

[Mooroka+99] Mooroka et al., Computation of a spherical view space for efficient planning of viewpoints in 3D object modeling, 3DIM'99 Proc., pp.138-147.

[Prieto+99] Prieto et al, CAD-based range sensor placement for otimum 3D data acquisition, 3DIM'99 Proc., pp.128-137.

[Pulli99] K. Pulli,"Multiview registration for large datasets", Proc 2nd Int.l Conf. on 3D Digital Imaging and Modeling,1999,pp.160-168

[Rocchini+00] Rocchini, Cgnoni, Montani, Scopigno, "The Marching Intersections Algorithm for merging range images", submitted paper, July 2000.

[Soucy+92] Soucy, M. and Laurendeau, D., "Multi-resolution surface modeling from multiple range views," Proceedings 1992 IEEE Conf. on Computer Vision and Pattern Recognition, pp. 348-353.

[Turk+94] Turk, G. and Levoy, M., "Zippered polygon meshes from range images." In Proceedings of SIGGRAPH '94, pp.311-318.

3D Data Acquisition

MPI INFORMATIK

Surface Attributes Acquisition and Management

Hendrik P.A. Lensch

Max-Planck-Institut für Informatik, Saarbrücken (Germany)

Digitizing Real World Objects

MPI INFORMATIK

by 3D geometry

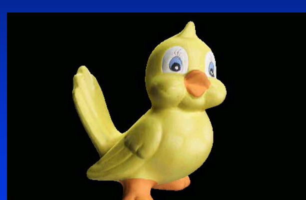
- no color

Digitizing Real World Objects

MPI INFORMATIK

by images

- no interaction

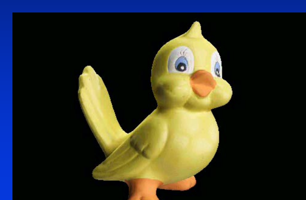


Digitizing Real World Objects

MPI INFORMATIK

by geometry plus texture

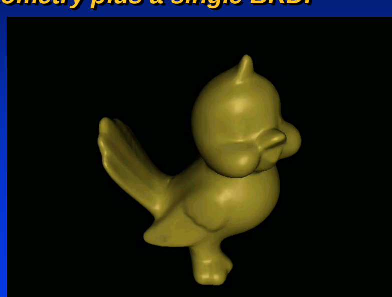
- no relighting



Digitizing Real World Objects

MPI INFORMATIK

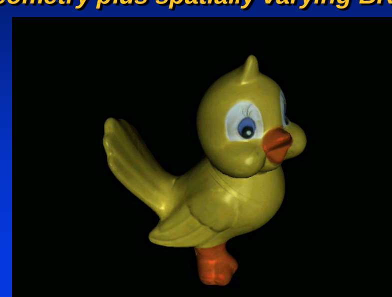
by geometry plus a single BRDF



Digitizing Real World Objects

MPI INFORMATIK

by geometry plus spatially varying BRDFs



Overview

- Introduction
- Texture Acquisition
  - *3D - 2D Registration*
- Consistent Colors
- Image-Based BRDF Measurement
  - *for homogeneous materials*
  - *for spatially varying materials*
- Conclusion

Hendrik Lensch

Overview

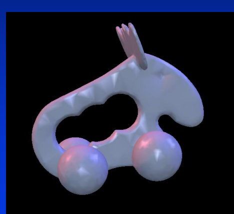
- Introduction
- **Texture Acquisition**
  - *3D - 2D Registration*
- Consistent Colors
- Image-Based BRDF Measurement
  - *for homogeneous materials*
  - *for spatially varying materials*
- Conclusion

Hendrik Lensch

Digital 3D Models of Real World Objects

Geometry

Image data

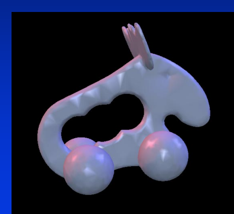


Hendrik Lensch

Acquiring Real World Models

Geometry
- 3D scanner

Texture data
- digital camera

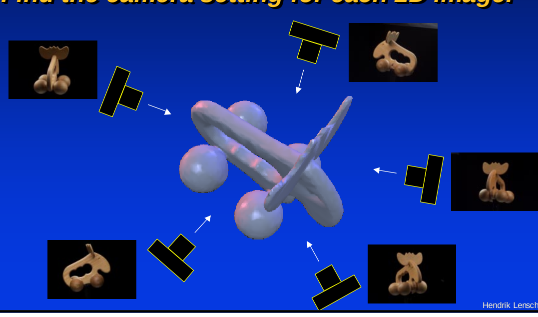


single sensor vs. multiple sensors

Hendrik Lensch

3D – 2D Registration

Find the camera setting for each 2D image.



Hendrik Lensch

Camera Model

Transformations

- to camera coordinates (extrinsic):

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R(x_w - g) + t$$

- to 2D image space (intrinsic):

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} + \frac{f}{z_c} \begin{pmatrix} \frac{1}{\alpha} x_c \\ y_c \end{pmatrix}$$

$\Rightarrow$ determine ***R*, *t*** and *f* (6+1 dimensions)

Hendrik Lensch

Similarity Measure

Which features to investigate?

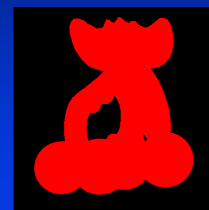
- no color information on the model
- correspondence of geometric features hard to find


Hendrik Lensch

Similarity Measure

Compare silhouettes [Etienne de Silhouette 1709-1767]

- model: render monochrome
- photo: automatic histogram-based segmentation

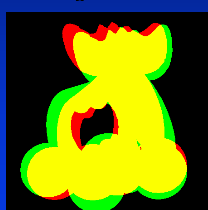



Hendrik Lensch

Similarity Measure

Compare silhouettes [Etienne de Silhouette 1709-1767]

- model: render monochrome
- photo: automatic histogram-based segmentation



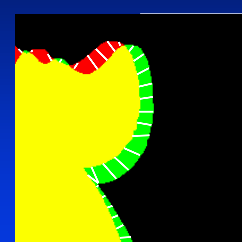
Hendrik Lensch

Distance Measure for Silhouettes

[Neugebauer & Klein '99, Matsushita & Kaneko '99]

Point-to-outline distances

- slow because points on the outline must be determined
- speedup by distance maps

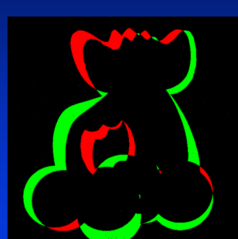


Hendrik Lensch

Pixel-based Distance Measure

Count the number of pixels covered by just one silhouette.

- XOR the images
- compute histogram (hardware)



Hendrik Lensch

Non-linear Optimization

Downhill Simplex Method [Press 1992]

- works for N dimensions
- no derivatives
- easy to control

Hendrik Lensch

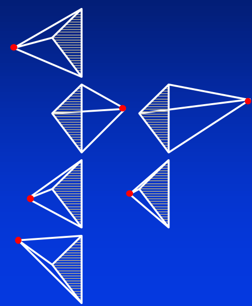
Simplex Method in 3D

original simplex

reflection and/or expansion

shrinking

random perturbation



Hierarchical Optimization

- optimize on low resolution first
- restart optimization to avoid local minima
- switch to higher resolution
- mesh resolution can be adapted

Texture Stitching

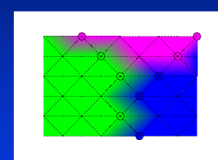
projective texture mapping

assign one image to each triangle

- triangle visible in image? (test every vertex)
- select best viewing angle
- discard data near depth discontinuities
- blend textures at assignment boundaries

Blending Across Assignment Borders

- find border vertices
- release all triangles around them
- assign boundary vertices to best region
- assign alpha-values for each region
  - *1 to vertices included in the region*
  - *0 to all others.*



Texture Processing

problems

- complicated rendering
- multiple textures for one object
- multiple texture coordinates per vertex
- triangles will be drawn up to three times

Texture Processing

solution [Rocchini et al. 1999]

- pack all relevant texture parts into one texture
- adapt texture coordinates
- duplicate vertices

Packed Texture

Textured 3D Model

Overview

- Introduction
- Texture Acquisition
  - *3D - 2D Registration*
- **Consistent Colors**
- Image-Based BRDF Measurement
  - *for homogeneous materials*
  - *for spatially varying materials*
- Conclusion

Consistent Colors

current problems:

- visible artifacts at assignment boundaries
- no view-dependent effects
- highlights do not move
- no relighting possible

Removing Artifacts

- use almost diffuse area light sources to avoid sharp highlights.
- still the texture captures the current lighting situation

Diffuse Color

MPI INFORMATIK

[Rushmeier '97]

- assume lambertian surfaces, outgoing radiance:

$$L_o = \rho L_i \cdot \langle \hat{\omega}_i \mid \hat{n} \rangle$$

- take a number of pictures with different but known light source positions (point-light source)
- discard highest and lowest intensities

Hendrik Lensch

Diffuse Color

MPI INFORMATIK

- solve the following system for ρ (and $\hat{n}$)

$$\rho L_i \begin{pmatrix} \omega_{1,1} & \omega_{1,2} & \omega_{1,3} \\ \omega_{2,1} & \omega_{2,2} & \omega_{2,3} \\ \omega_{3,1} & \omega_{3,2} & \omega_{3,3} \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \begin{pmatrix} L_{o,1} \\ L_{o,2} \\ L_{o,3} \end{pmatrix}$$

- yields consistent colors
- removes all highlights

Hendrik Lensch

Surface Light Fields

MPI INFORMATIK

[Miller '98, Wood '00]

- store radiance values for multiple viewing directions for each surface point
- reconstruct the by interpolation of the closest views
- yields almost perfect results including highlights
- huge acquisition effort (several hundred images)
- huge amount of data (requires compression)
- no relighting possible

Hendrik Lensch

Reflectance Fields

MPI INFORMATIK

[Debevec '00, Matusik '02, Masselus '02, Furukava '02]

- captures (also) lighting dependent effects
- requires even more acquisition effort
- produces more data
- allows relighting!

Hendrik Lensch

Overview

MPI INFORMATIK

- Introduction
- Texture Acquisition
  - *3D - 2D Registration*
- Consistent Colors
- **Image-Based BRDF Measurement**
  - *for homogeneous materials*
  - *for spatially varying materials*
- Conclusion

Hendrik Lensch

Overview
BRDF Measurement

MPI INFORMATIK

- image-based BRDF measurement (homogenous materials)
- data acquisition
- resampling
- material separation
- projection (spatially varying behavior)

Hendrik Lensch

Overview BRDF Measurement

MPI INFORMATIK

- **image-based BRDF measurement** (homogenous materials)
- data acquisition
- resampling
- material separation
- projection (spatially varying behavior)

Hendrik Lensch

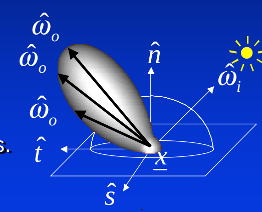
Reflection Properties

MPI INFORMATIK

- a BRDF (bi-directional reflectance distribution function)

$$f(\hat{\omega}_o, \underline{x}, \hat{\omega}_i)$$

yields the fraction of reflected to incident radiance at one point for any pair of directions.

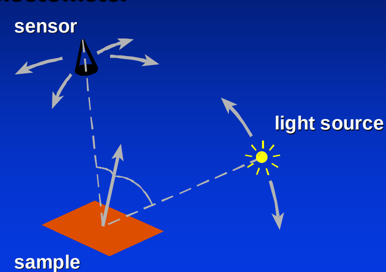


Hendrik Lensch

BRDF Measurement

MPI INFORMATIK

Gonioreflectometer

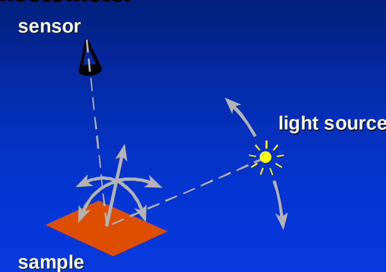


Hendrik Lensch

BRDF Measurement

MPI INFORMATIK

Gonioreflectometer



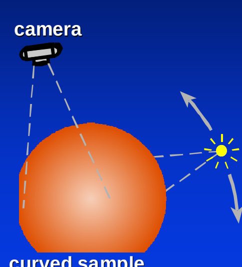
Hendrik Lensch

Image-Based BRDF Measurement

MPI INFORMATIK

[Marschner 1999]

- capture lots of BRDF samples at one shot by a sensor array / camera.



Hendrik Lensch

Overview BRDF Measurement

MPI INFORMATIK

- image-based BRDF measurement (homogenous materials)
- **data acquisition**
- resampling
- material separation
- projection (spatially varying behavior)

Hendrik Lensch

Acquisition Equipment

MPI INFORMATIK

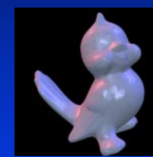
- 3D scanner
  (structured light, CT)
- digital camera
  (high dynamic range)
- point-light source
- dark room
- calibration targets
  (checkerboard, metal spheres)

Hendrik Lensch

Acquisition

MPI INFORMATIK

3D scan



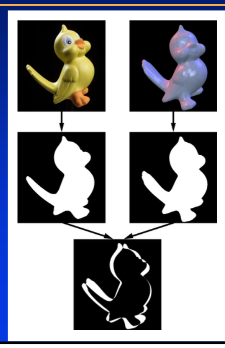
HDR images, calibrated camera/light source position

Hendrik Lensch

3D-2D Registration

MPI INFORMATIK

- calibrated gantry
- corresponding points
- we use a silhouette-based method

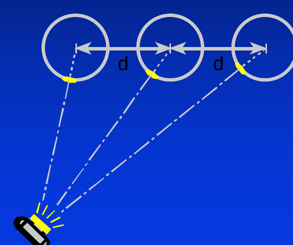


Hendrik Lensch

Light Source Position

MPI INFORMATIK

- detect highlights of ring flash reflections
- determine the position of the spheres

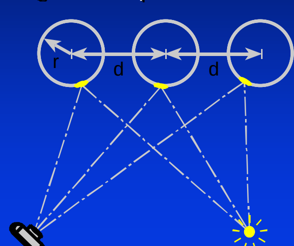


Hendrik Lensch

Light Source Position

MPI INFORMATIK

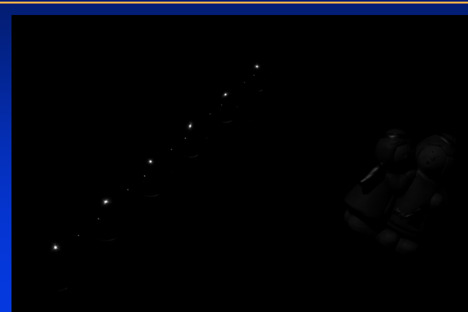
- detect highlights of light source reflections
- reconstruct light source position



Hendrik Lensch

Light Source Position

MPI INFORMATIK



Hendrik Lensch

Overview BRDF Measurement

MPI INFORMATIK

- image-based BRDF measurement (homogenous materials)
- data acquisition
- **resampling**
- material separation
- projection (spatially varying behavior)

Hendrik Lensch

Lumitexels

MPI INFORMATIK

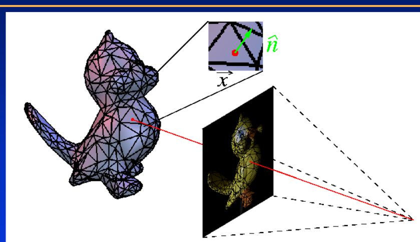
A lumitexel L collects all data available for a point on the surface:

- 3D position $\vec{x}$ } **from geometry**
- normal $\vec{n}$
- list of radiance samples R_i, one for every image where $\vec{x}$ is visible and lit:
  - radiance value r_i } **from images**
  - light source direction $\hat{u}_i$
  - viewing direction $\hat{v}_i$

Hendrik Lensch

Assembling Lumitexels

MPI INFORMATIK

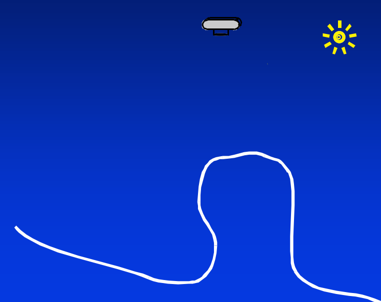


- for each triangle:
  find the image with the highest sampling rate
  generate one lumitexel for each pixel

Hendrik Lensch

Depth Discontinuities

MPI INFORMATIK



Hendrik Lensch

Depth Discontinuities

MPI INFORMATIK

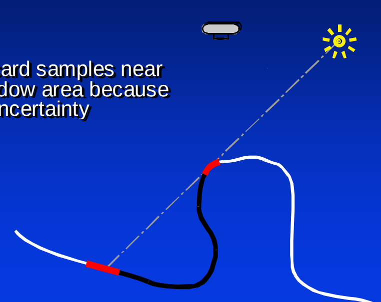
- discard shadow area

Hendrik Lensch

Depth Discontinuities

MPI INFORMATIK

- discard samples near shadow area because of uncertainty

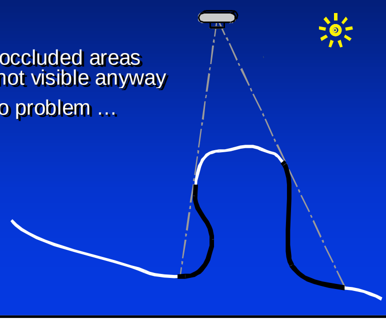


Hendrik Lensch

Depth Discontinuities

MPI INFORMATIK

- self-occluded areas are not visible anyway
  ⇒ no problem …

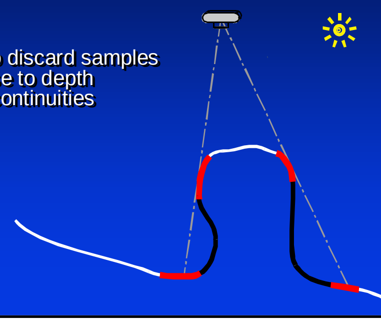


Hendrik Lensch

Depth Discontinuities

MPI INFORMATIK

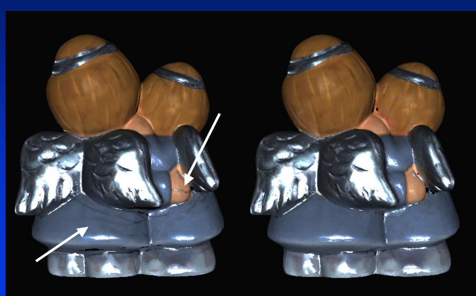
- also discard samples close to depth discontinuities



Hendrik Lensch

Removing Discontinuities

MPI INFORMATIK



Hendrik Lensch

Overview BRDF Measurement

MPI INFORMATIK

- image-based BRDF measurement (homogenous materials)
- data acquisition
- resampling
- **material separation**
- projection (spatially varying behavior)

Hendrik Lensch

Problem

MPI INFORMATIK

- describe the reflection properties for the basic materials
- too few radiance samples
  ⇒ no dense sampling of the BRDF
  ⇒ fit a BRDF model

Hendrik Lensch

The Lafortune Model

MPI INFORMATIK

$$f_r(\hat{u},\hat{v}) = \rho_d + \sum_i \left(C_{x,i}(u_x v_x + u_y v_y) + C_{z,i} u_z v_z \right)^{N_i}$$

- physically plausible
- diffuse component plus a number of lobes
- 3*(1+ i*3) parameters (12 for a single lobe model)
- fit parameters to samples

Hendrik Lensch

Fitting BRDFs to Lumitexels

- define error measure between a BRDF and a lumitexel:

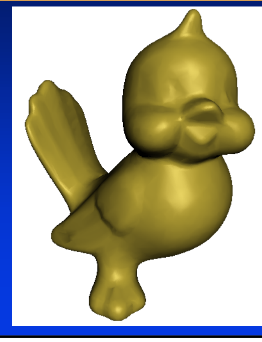
$$E_{f_r}(L) = \frac{1}{|L|} \sum_{R_i \in L} \Delta\left(f_r(\hat{u}_i, \hat{v}_i) u_{i,z}, r_i\right)^2$$

= average error over all radiance samples

- perform non-linear least square optimization for a set of lumitexels using Levenberg-Marquardt
- yields a single BRDF (i.e. its parameters) per set of lumitexels

Hendrik Lensch

Fitting Result



Hendrik Lensch

Clustering

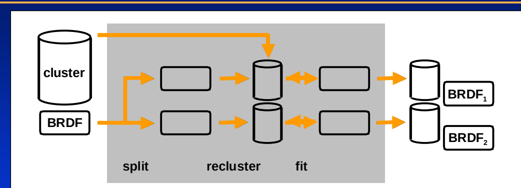
Goal: separate the different materials

- similar to Lloyd iteration

- start with a single cluster containing all lumitexels
- split cluster along direction of largest variance
- stop after *n* clusters have been constructed

Hendrik Lensch

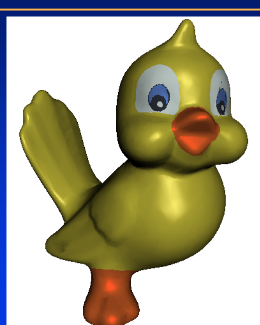
Split-Recluster-Fit Cycle



- split into two BRDFs along direction of largest variance of parameters (covariance matrix)
- distribute initial lumitexels forming two new clusters
- refit new BRDFs
- repeat reclustering and fitting until clusters are stable

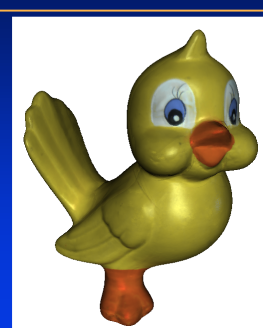
Hendrik Lensch

Clustering Results



Hendrik Lensch

Spatially Varying Materials



Hendrik Lensch

Overview BRDF Measurement

MPI INFORMATIK

- image-based BRDF measurement (homogenous materials)
- data acquisition
- resampling
- material separation
- **projection (spatially varying behavior)**

Hendrik Lensch

Projection

MPI INFORMATIK

Goal: assign a separate BRDF to each lumitexel

- too few radiance samples for a reliable fit
- represent the BRDF f_π of every lumitexel by a linear combination of already determined BRDFs $f_1, f_2, \ldots, f_m$:

$$f_\pi = t_1 f_1 + t_2 f_2 + \ldots + t_m f_m$$

- determine linear weights $t_1, t_2, \ldots, t_m$

Hendrik Lensch

Projection

MPI INFORMATIK

- compute the pseudo-inverse using SVD to get a least square solution for

$$\begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{|L|} \end{pmatrix} = \begin{pmatrix} f_1(\hat{u}_1,\hat{v}_1)u_{1,z} & f_2(\hat{u}_1,\hat{v}_1)u_{1,z} & \cdots & f_m(\hat{u}_1,\hat{v}_1)u_{1,z} \\ f_1(\hat{u}_2,\hat{v}_2)u_{2,z} & f_2(\hat{u}_2,\hat{v}_2)u_{2,z} & \cdots & f_m(\hat{u}_2,\hat{v}_2)u_{2,z} \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\hat{u}_{|L|},\hat{v}_{|L|})u_{|L|,z} & f_2(\hat{u}_{|L|},\hat{v}_{|L|})u_{|L|,z} & \cdots & f_m(\hat{u}_{|L|},\hat{v}_{|L|})u_{|L|,z} \end{pmatrix} \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_m \end{pmatrix}$$

- avoid negative t_i

Hendrik Lensch

Initial Basis

MPI INFORMATIK

for each cluster take

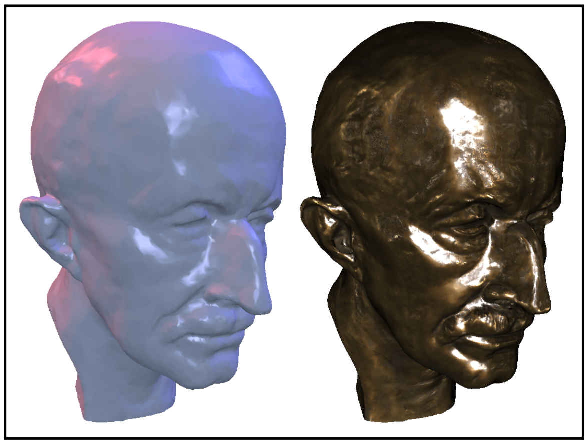
- the fitted BRDF f_C
- the BRDFs of spatially neighboring cluster
- the BRDFs of similar material clusters
- two slightly modified versions of f_C

Hendrik Lensch

Reconstruction Results

MPI INFORMATIK

Hendrik Lensch



Results

- truly spatially varying BRDFs
- small number of input images
- high quality compact object representation
- 200 MB image data (26 views) ⇒ 20 MB output (angels)
- reasonable acquisition effort
- normal maps even for specular objects

Future Work

improve algorithms:

- consider interreflections
- level-of-detail representation (mip maps)
- anisotropic materials

Conclusion

- it requires a long pipeline also for the acquisition of surface attributes:
- taking pictures with/without special light sources
- 2D-3D registration / calibration
- resampling
- transformation into a reasonable representation
- model fitting

Thank You

Questions?

www.mpi-sb.mpg.de

Lesson 6: Simplification of scanned meshes

Carlos Andújar
Universitat Politècnica de Catalunya

Organization

- Part a:
  - Overview of mesh simplification methods
- Part b:
  - External memory simplification
  - Techniques for appearance preservation

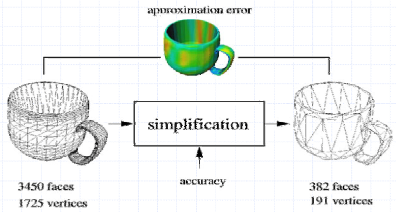
6.1 Mesh simplification

Contents

- Introduction to mesh simplification
- Automatic simplification
- Classification of methods
- Clustering methods
- Incremental methods
- Volume-based methods

Introduction

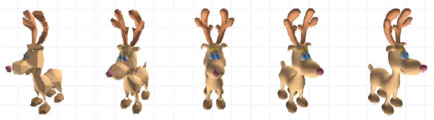
- ***Geometry simplification*** deals with the generation of geometric models that resemble the input model but involve less faces:



Applications

Mesh simplification is used to:

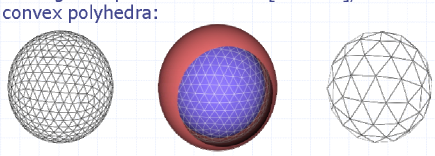
- Accelerate the handling of complex models by omitting unnecessary or unessential computation steps:
  - **Real-time visualization,** Photo-realistic rendering
  - Collision detection, Visibility analysis
- Reduce storage space and transmission times



376t 1,300t 4,700t 8,200t 100,648t

Automatic simplification

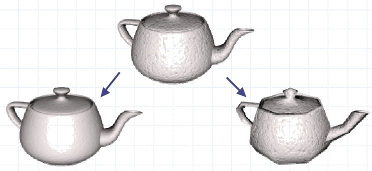
- Finding the optimal is NP-hard [Das+90], even with convex polyhedra:



- A simplification approach has two main components:
  - Transformation operation
  - Criteria for guiding the process

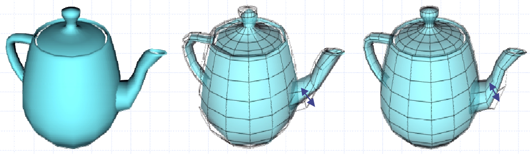
Approximation error

- Quantifies the notion of "similarity"
- Two kinds of similarity:
  - Geometric similarity (surface deviation)
  - Appearance similarity (material, normal…)



Geometric similarity

- Two main components:
  - Distance function
  - Function Norm:
    - L_2: average deviation
    - L_{inf} : maximum deviation - Hausdorff distance

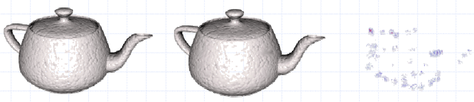


Appearance similarity

- Difference between two images:

$$\|I_1 - I_2\| = \frac{1}{n^2}\sum_x \sum_y d(I_1(x,y) - I_2(x,y))$$

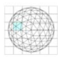
- Difference between two objects:
  - Integrate the above over all possible views

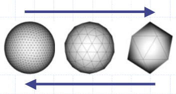


I_1 I_2 $I_1 - I_2$

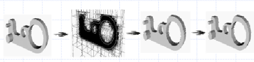
Simplification approaches

Surface-based:

- Clustering 
- Fine-to-coarse (decimation)
- Coarse-to-fine (refinement)



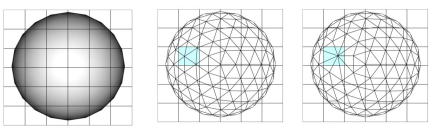
- Others (superfaces, wavelets)

Volume-based 

Clustering methods

Approach

- Group nearby vertices into groups (clusters).
- Vertices inside a cluster are replaced by a single vertex. These replacements cause many faces to collapse.



Clustering methods (ii)

Main deals

- Grouping criteria
- New vertex placement

Examples

- [Rossignac+93, Lindstrom00] (uniform grid)
- [Luebke+97] (octree)
- [Low+97] (arbitrary floating cells)

Incremental methods

Approach:

- Top-down strategy: Iterative elimination of geometric entities through reduction operators chosen upon geometric optimality criteria.

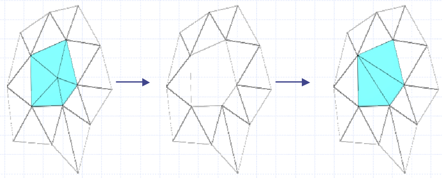
Main deals:

- Reduction operator adopted
- Topology reduction/preservation
- Criteria for guiding the simplification

Reduction operators

Vertex removal

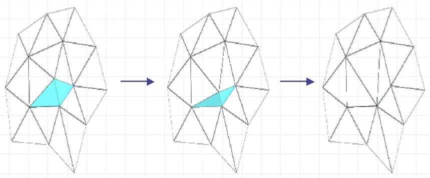
- Adopted in [Schroeder92, Soucy+96...]



Reduction operators (ii)

Edge collapse

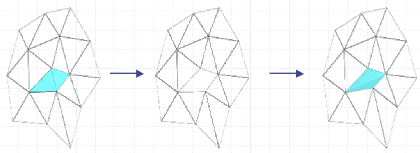
- Adopted in [Hoppe96, Ronfard+96, Gueziec96...]



Reduction operators (iii)

Edge flip

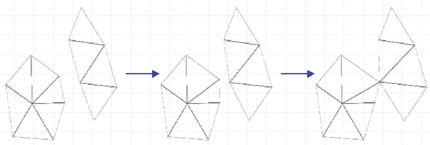
- Adopted in [Hoppe93]



Reduction operators (iv)

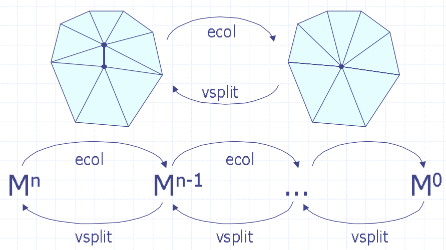
Pair contraction

- Does not preserve topology
- Adopted in [Garland+97, Popovic+97, Erikson+98, El-Sana+99...]



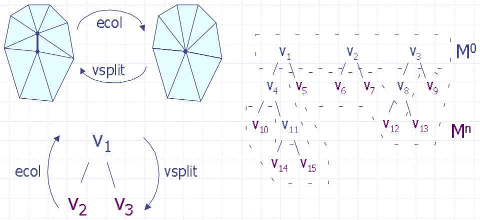
Progressive Meshes [Hoppe96]

Idea: describe a mesh as a coarse mesh along with a sequence of refinement operations.



Vertex hierarchies [Hoppe97, Xia+96]

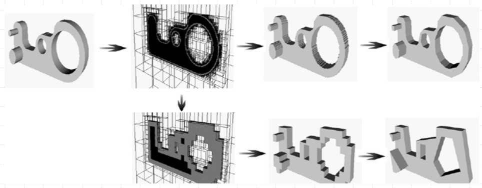
Idea: use vertex hierarchies defined by vsplit transformations for selective refinement.



Volume-based methods

Approach:

- Topology simplification is achieved by converting a 3D model to and from a volumetric representation.



Volume-based methods

Main deals:

- Intermediate representation
- Reconstruction algorithm
- Further decimation

Examples:

- [He+95, He+96] (volume buffers)
- [Andujar+02] (octree)

6.2 Ext. memory simplification

Contents:

- Introduction
- External memory approaches
- Clustering methods
- Hierarchical subdivision
- Spanned meshes
- Octree subdivision + tagging

Introduction

- Acquired meshes often exceed the main memory size of current graphics hardware.
- Almost all simplification algorithms discussed so far require the entire mesh to be loaded in main memory:
  - Applicability limited to models that fit in main memory, or otherwise...
  - Significant decrease of performance due to page faults.
- The adoption of an external memory (EM) approach is required whenever we want to handle a huge mesh on limited core memory.

EM approaches

- Clustering methods
  - Uniform subdivision [Lindstrom00]
  - Adaptive subdivision [Shaffer+01]
- Mesh partition techniques
  - Hierarchical subdivision [Hoppe98, Prince00]
  - Spanned meshes [El-Sana+00]
  - Octree subdivision + tagging [Cignoni+02]

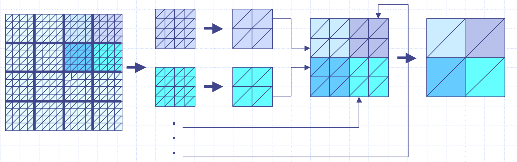
Clustering methods

Approach:
- Extend classic clustering methods so that the original mesh is not required to be in core memory.
- Small or no use of connectivity information (triangle soups).

Examples:
- [Lindstrom00] (uniform grid, quadric error metrics)
- [Shaffer+01] (adaptive subdivision)

Hierarchical subdivision

Approach:
- Partition the original mesh into smaller blocks which fit into main memory.
- Simplify each block until some threshold (boundary vertices are preserved).
- Merge neighboring blocks and simplify recursively.



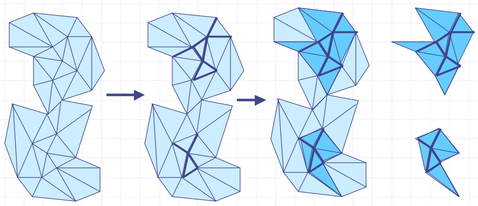
Hierarchical subdivision (ii)

Main deals:
- Spatial subdivision (mesh partition)
- Choice of block size
- Error metric

Examples:
- [Hoppe98] (for terrains)
- [Prince02] (for arbitrary surfaces)

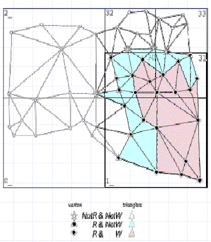
Spanned meshes [El-Sana+00]

Approach:
- Partition the mesh using spanning submeshes build by adding edges in shortest-first order (with respect to a given error metric).
- Correct collapsing order is preserved.

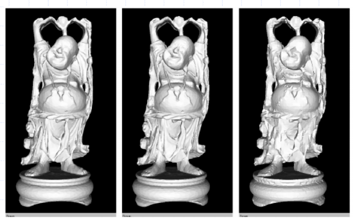


Octree subdivision [Cignoni+02]

Approach
- Partition the mesh using an octree (OEMM) with no replication of elements.
- Use R/W flags on faces and vertices for consistent *loading, modify, save* cycle of any mesh portion.
- Only the octree structure and active loaded fragments are loaded in core memory.

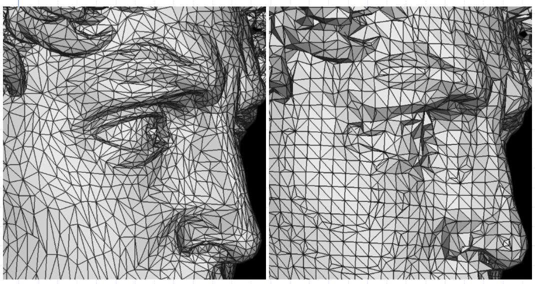


Octree subdivision [Cignoni+02]



RAM-QEM simplif. 18,338 faces | OEMM-QEM simplif. 18,338 faces | OOCC simplif. 19,071 faces

OEMM & OoCS (from Cignoni+02)



6.3 Appearance preservation

Contents

- Introduction
- Detail-preserving techniques
- Attribute-aware simplification
- Texture-enhanced simplification
- Sampling-based texture generation

Introduction

- Most of the methods described so far only guarantee *geometric similarity*; no support is provided for controlling the *visual degradation* of the simplified mesh.
- ***Appearance preservation*** deals with the generation of simplified meshes that preserve visual attributes (color, high frequency shape details) in order to minimize the perceptual error.

Detail-preserving techniques

- Attribute-aware simplification

  Attribute values are used during simplification to avoid collapsing primitives with sharp discontinuities.
- Texture-enhanced simplification

  A new texture map encoding the original detail is generated during the simplification process.
- Sampling-based texture generation

  After simplification, detail is encoded through texture and bump maps by sampling the simplified mesh.

Attribute-aware simplification

Approach:

- These methods evaluate the impact of each transformation on visual degradation.
- A primitive transformation (e.g. edge-collapse) is applied only when the visual error is less than a user-provided threshold.

Examples:

- [Erikson+99], [Garland+98], [Hoppe98] (based on quadric error metrics)
- [Bajaj+98] (attribute topology preservation)

Texture-enhanced simplification

Approach:

- A new texture and/or bump map is build to encode all the detail that is lost during the simplification process.

Examples:

- [Soucy+96]
- [Cohen+98]

Sampling-based map generation

Approach:

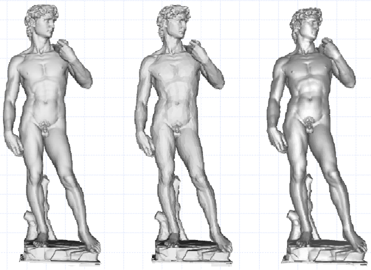
- After simplification, appearance attributes are encoded through texture and bump maps.
- These methods generate triangular texture patches by sampling the simplified mesh. These patches are then packed into a single texture map.

Examples:

- [Cignoni+99]

Sampling-based map generation

Results (from [Cignoni+02]):



1,683K faces — 10K faces — 10K faces, bump map

Concluding remarks

- Recent research has produced many effective techniques for generating approximations and multiresolution models.
- Available algorithms offer several tradeoffs between efficiency and accuracy (from fast clustering methods to HQ, slow optimization methods).
- To improve:
  - Topology simplification
  - Detail preservation
  - External memory adaptive simplification

References

Clustering methods

[Lindstrom00] Lindstrom (2000) Out-of-Core Simplification of Large Polygonal Models, In Proceedings of ACM SIGGRAPH'00, 259-262

[Low+97] Low, Tan (1997). Model simplification using vertex-clustering. In Proc. of the Symposium on Interactive 3D Graphics. ACM Press, New York.

[Luebke+97] Luebke, Erikson (1997) View-dependent simplification of arbitrary polygonal environments. In Proceedings of ACM SIGGRAPH'97, 199–208

[Rossignac+93] Rossignac, Borrel (1993) Multiresolution 3D approximations for rendering complex scenes. In Modeling in Computer Graphics, Springer-Verlag

Incremental methods

[El-Sana+99] El-Sana, Varshney (1999) Generalized view-dependent simplification. Computer Graphics Forum 18, 83–94.

References

[Erikson+98] Erikson, Manocha (1998) Simplification culling of static and dynamic scene graphs. Tech. Rep. TR98-009, Dep. of Computer Science, University of North Carolina-Chapel Hill.

[Floriani+00] De Floriani, Magillo,Morando, Puppo (2000). Dynamic View-Dependent Multiresolution on a Client-Server Architecture, Computer-Aided Design 32(13)

[Garland+97] Garland, Heckbert (1997) Surface simplification using quadric error metrics. In ACM SIGGRAPH'97. 209–216.

[Gueziec96] Gueziec (1996) Surface simplification inside a tolerance volume. Tech. report. Yorktown Heights, IBM Research Report RC 20440.

[Hoppe96] Hoope (1996) Progressive meshes. Computer Graphics 30, Annual Conference Series, 99–108.

[Hoppe97] Hoppe (1997) View-Dependent Refinement of Progressive Meshes, In ACM SIGGRAPH'97, 189-198.

References

[Popovic+97] Popovic , Hoppe (1997). Progressive simplicial complexes. In ACM SIGGRAPH 97, 217–224.

[Ronfard+96] Ronfard, Rossignac (1996) Full-range approximation of triangulated polyhedra. Computer Graphics Forum 15 (3)

[Schroeder+92] Schroeder, Zarge, Lorenzen (1992) Decimation of triangle meshes. In Proceedings of ACM SIGGRAPH'92

[Soucy+96] Soucy, Laurendeau (1996). Multiresolution surface modeling based on hierarchical triangulation. Computer Vission and Image Understanding, 63 (1) 1–14.

[Xia+97] Xia, Varshney. Dynamic Viewdependent simplification for polygonal models. In proc. Visualization'96, pp 327-334

Volume-based methods

[Andujar+02] Andujar, Brunet, Ayala (2002) Topology-Reducing Surface Simplification Using a Discrete Solid Representation, ACM Transactions on Graphics 22(2)

References

[He+95] He, Hong, Kaufman, Varshney, Wang (1995) Voxel based object simplification. In Proc. of Visualization'95, Atlanta, GA, 296–303.

[He+96] He, Hong, Varshney, Wang (1996) Controlled topology simplification. IEEE Trans. Vis. Comput. Graph. 2(2), 171–184

External memory simplification

[Cignoni+02] Cignoni, Rocchini, Montani, Scopigno (2002) External Memory Management and Simplification of Huge Meshes, to appear in IEEE Trans. on Visualization and Comp. Graph. , Vol.8

[El-Sana+00] El-Sana, Chiang (2000). External Memory View-Dependent Simplification, Computer Graphics Forum 19(3).

[Hoppe98] Hoppe (1998) Smooth View-Dependent Level-of-Detail Control and its Application to Terrain Rendering, IEEE Visualization '98, pp 35-42

[Lindstrom00] Lindstrom (2000) Out-of-Core Simplification of Large Polygonal Models, In Proceedings of ACM SIGGRAPH'00, 259-262

References

[Prince00] Prince (2000) Progressive Meshes for Large Models of Arbitrary Topology, Master Thesis

[Shaffer+01] Shaffer, Garland (2001). Efficient Adaptive Simplification of Massive Meshes, in IEEE Visualization'01

Attribute-aware simplification

[Bajaj+98] Bajaj, Schikore (1998) Topology preserving data simplification with error bounds. Computer Graphics 22:3-12

[Erikson+99] Erikson, Manocha (1999) GAPS: General and automatic polygonal simplification. In ACM Symposium on Interactive 3D Graphics, ACM press, pp 79-88

[Garland+98] Garland, Heckbert (1998) Simplifying surfaces with color and textures using quadric error metrics. In IEEE Visualization'98, ACM Press,New York, pp 264-270

[Hoppe99] Hoppe (1999) New quadric error metric for simplifying meshes with appearance attributes. In IEEE Visualization'99, ACM Press,New York, pp 59-66

References

Texture-enhanced simplification

[Cohen+98] Cohen, Olano, Manocha (1998) Apperance preserving simplification. In ACM SIGGRAPH'98 proceedings, pp 115-122

[Hoppe97] Hoppe (1997) View-dependent refinement of progressive meshes. In ACM SIGGRAPH'97 proceedings, pp 189-198

[Soucy+96] Soucy, Godin, Rioux (1996) A texture-mapping approach for the compression of 3D triangulations. The Visual Computer 12(10)

Sampling-based texture generation

[Cignoni+99] Cignoni, Montani, Rocchini, Scopigno, Tarini (1999) Preserving attribute values on simplified meshes by re-sampling detail textures, The Visual Computer 15 (10), 519-539.

Computational geometry

[Das+90] Das, Joseph (1990) The complexity of minimum nested convex polyhedra. In Proc 2nd Canadian Conf. Computational Geometry, pp 296-301

Tutorial T1: 3D Data Acquisition

"Rendering Scanned Data"

Roberto Scopigno
Visual Computing Group
ISTI – C.N.R.
Pisa, Italy

3D Scanning – Using Data

- Focus: **Cultural Heritage** applications (but techniques presented are general)
- Rendering-related applications
  - Triangle meshes or splat-based rendering?
  - Non-photorealistic rendering
- Beyond rendering: other applications of the data

Rendering Data

- Which are potential uses of the 3D data, in the context of Cultural Heritage?
- **Rendering-oriented:**
  - Pre-computed animations (didactic appl., turism, etc.)

    Demo of videos: Dav_Rotate, Dav_view, Quark
  - Interactive rendering (VR, virtual museums, multimedia, etc.)

    Demo of Inspector

Visualization

- Huge data not easy to manage and render
  - Scanned models: 10M → Giga samples/faces
- Two different approaches:
  - **Sample-based**
    - Organize sampled points in efficient data structure (**hierarchical**)
    - Render using **splatting**
  - **Triangle-based**
    - **Simplify** surfaces, build LOD repr. (very concise)
    - Organize data using **hierarchical** repr.
    - Enhance visual quality of low resolution models with **textures**
    - Visualize using standard **triangle-based hardware graphics**

Visualization - Sample-based [1]

Qsplat [Rusinkiewicz+00]

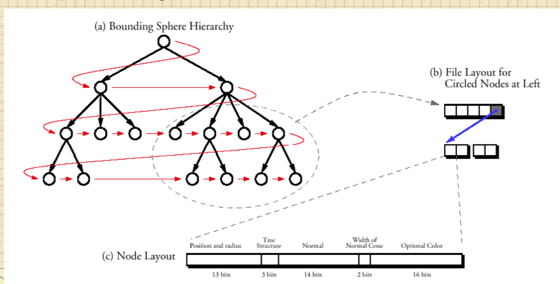
A multiresolution point rendering system for large meshes

- octree-based encoding of scanned samples
- supports view frustum culling and level-of-detail rendering
- OT traversed by considering nodes bounding sphere w.r.t current view and pixels size

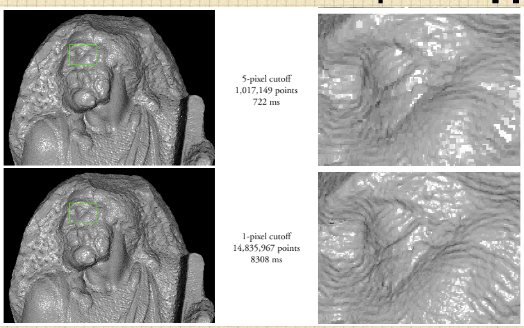
```
TraverseHierarchy(node)
{
if (node not visible)
    skip this branch of the tree
else if (node is a leaf node)
    draw a splat
else if (benefit of recursing further is too low)
    draw a splat
else
    for each child in children(node)
        TraverseHierarchy(child)
}
```


Visualization - Sample-based [2]

- Qsplat tree organization:



Visualization - Sample-based [3]



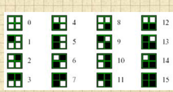
SGI Onyx2 with InfiniteReality graphics, screen resolution of 1280x1024

Visualization - Sample-based [4]

Point-based rendering [Bosch+02]

Another point rendering system for large meshes:

- Hierarchical repr. (octree-based)
- Compressed encoding of scanned samples:
  - point coordinate quantization (8 OT nodes rer. with 4 bit code)
  - less than 2 bits per point position



- Rendering:
  - SW-based rendering (optimized view transformation)
  - Speed on a commodity PC: **14M/sec** Phong shaded + textured samples, **4M/sec** anti-aliased splats.

Triangle-based Rendering

Inspector tool

- **Main goals:**
  - design a **user-oriented** interface, to allow naïve users to easily manipulate and inspect a complex 3D model

    assuming: ***single object, accurate digital model***
  - design an internal architecture supporting inspection of a **huge mesh** coupled with a **real time behaviour** on standard OpenGL accelerated platforms

    real-time behavior vs. visual accuracy

Interactive Viewers

- A number of **interactive viewers** exist:
  - Usually, data represented at a single resolution
  - Interface is by direct manipulation or navigation → not easy to control/drive
  - Non-expert users often get lost in ***sideral space*** (e.g. most of the VRML applications showing a single object)
  - An example: inspecting the **Minerva** model with a standard viewer

Demo of Plyview …

Interactive Response → Efficient Data Management

Application Context:

single object of limited extension in space

Therefore, our **design choices** have been:

- Different resolutions managed through a **Level of Detail** (LoD) approach
- Static LoD's organized in a **forest of Octrees** built off-line (frustum & visibility culling, choice of LOD level)
- Selection of current LOD should be **transparent to the user** and interaction as **intuitive** as possible

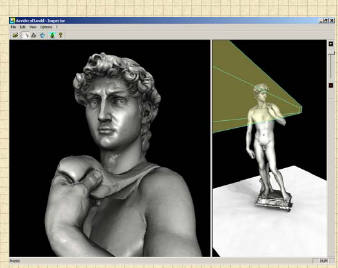
Let us start from the GUI…

Naïve Users: an ad-hoc GUI

Inspector layout:

- Two visualization windows:

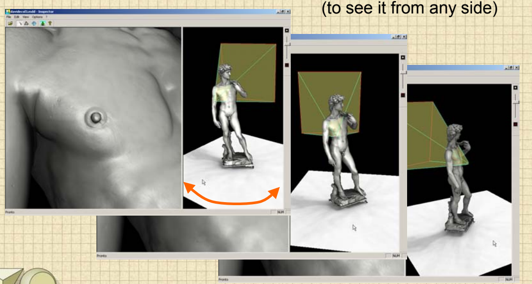
  The statue **dummy** on the right is used to select the specifications of the detailed view (left frame)



- Interaction approach:
  - mostly **point&click**
  - constrained by buttons and sliders

...GUI: dummy Rotation

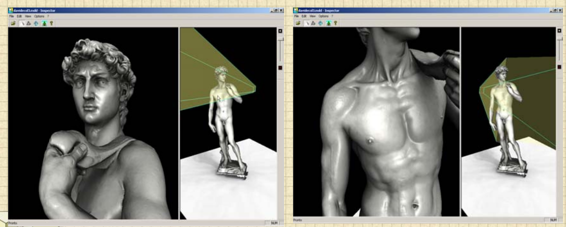
Rotation of the dummy
(to see it from any side)



...GUI: selection of a focus

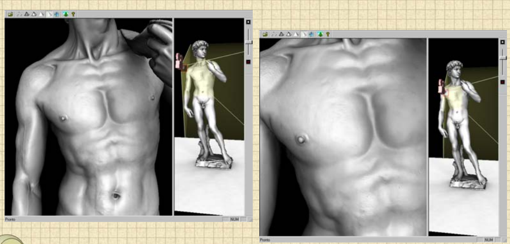
To select a **detailed view**, simply **point&click** the preferred focus point on the **dummy**

The corresponding view is shown in the left-most frame



...GUI: Pointing and Zooming

... To see the focus region more in detail, user simply increases [decreases] the **zooming factor** (see the slider on the extreme right)



Internals: data management

- Original David model: 56 million faces, **simplified** progressively to different sizes:
  17M t, 9M t, 4M t, 1M t, 270K t (LOD repr.)
  using our **out-of-core simplifier** (edge collapse)
- All these models are stored on disk, and the system chooses dynamically the one to be used in each rendering
- Criterion:
  - **If** graphic board can render interactively $\mathbf{K_{rend}}$ triangles per frame **and** given current **view_volume**
  - **Then** render the LOD such that *geometry **in** view_volume* <= K_{rend}

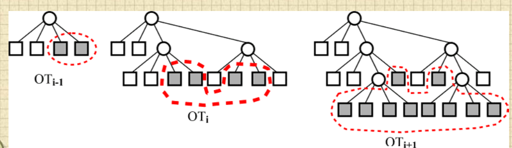
LoD: Octree Structure

- **Key point:** compute efficiently the size of the **visible geometry**, for each view specs
- Each simplified model is represented with an **octree (OT):** a pre-computed spatial index on the mesh faces
  - OT construction: subdivide recursively all nodes with > K_{OT} triangles
  - Any leaf node stores:
    - a **pointer** to a **rendering-oriented** representation of the geometry (three vectors containing: vertex positions, normals and **triangle strips**)
    - **size** (# triangles) and **cone of normals** of the mesh section contained in the node
- Data stored:
  - Core Mem: **forest of octrees,** with pointers to geometry on disk
  - Disk: geometry (loaded on demand on RAM)

LOD: view-dependent selection

At run time, for each new view_volume:

- Compute for the current **OT**$_i$ the **#triang** contained in the view_volume (**frustum culling**)
- IF **#triang** ~= $\mathbf{K_{rend}}$ THEN render OT_i ELSE check $\{OT_{i-1} \mid OT_{i+1}\}$
- Take into account **visibility culling** (using cones of normals)



LoD: Performance

- **Space:**
  - A mesh of 15M-30M triangles produces an octree of **2.5K- 5K nodes** (at maximal accuracy and with a threshold K_{OT} of 16K tr.)
  - Overhead: 140KB for a huge mesh (single OT structure repr. in RAM)
  - Disk: ~= 390MB, David LoD geometry (loaded on demand)
- **Time:** search and loading of a partial mesh in 0.01-0.05 sec.
- **High resolution background rendering:**
  - Run in spare times (no user interaction), to show the model at maximal accuracy/resolution
  - Few seconds for the higher resolution model (17 million triangles) Renders at **7-12 Mtr/sec** (on a GeForce2)

Visualization: point- *vs.* triangle-based

Both can be accelerated using **hierarchical structures** (frustum culling)

- Point-based
  - LOD is easy
  - Low quality if just points → render **splats**
  - Antialiasing mandatory
- Triangle-based
  - Simplification technology can produce **high-quality LOD repr.** → triangle distribution **adapts** to shape detail, interpolation fills the gaps between samples
  - Use all the power of current **graphics boards** (and all features/tricks)

Faster drawing many splats or fewer [bump textured] triangles?

Beyond Rendering [1]

- **Non-Rendering uses:**
  - Introducing 3D digital models in CH **catalogues**
  - Support **physical reproduction** (avoiding to cast endangered artifacts)
  - Using 3D data in **restoration** …

Beyond Rendering [2]

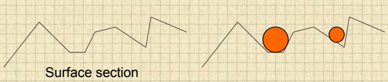
Comp. Aided Restoration?

- Restoration planning
  - Simulate planned modifications to **appearance** and **shape** before acting on the real object
- Restoration journaling and **documentation**
  - Take trace of all actions operated on the object (annotations on the 3D mesh, assign attributes to mesh sections → **~time-varying 3D GIS**)
  - Map and cross-correlate different imaging results on 3D geometry (e.g. UV photos, X-ray, thermographs, etc.)
- Reproduce missing parts (modeling + rapid prototyping)
- Reassemble fragmented [incomplete] artifacts
- Produce feature-oriented **non-photorealistic** output (mimicking classical pen&ink drawing style)
- …

Non-photorealistic Rendering

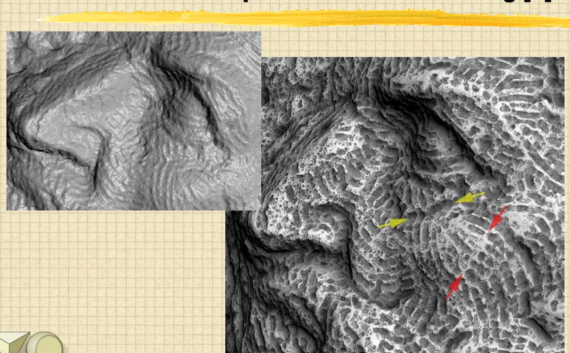
Improving insight via visualization tech.

- Goal: study the sculpting style of Michelangelo
- Evaluate in an objective and comparable manner the chisel mark (size, depth, direction)
- Not easy to do on a real or a photorealistic image
- Can be easily done in an automatic manner: accessibility computation by rolling virtual spheres on the surface



Surface section

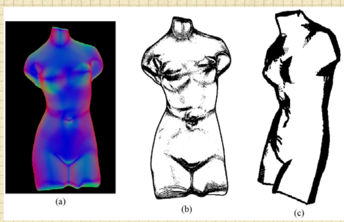
Non-photorealistic Rendering [2]



[Images by M. Levoy]

Non-photorealistic Rendering [3]

- Producing virtual “hand-made” drawing
  - Restorers experts are used to hand-made drawings ot the artifact
  - Simulate **pen-based** drawing of the 3D model (silhouettes, curvature, features …)



Non-photorealistic Rendering [4]

- Another example:

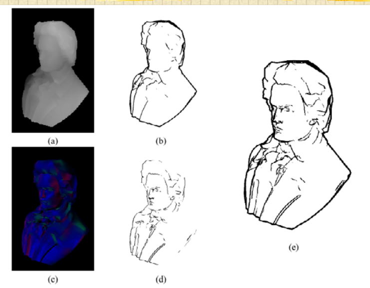


Figure 3: Outline detection of a more complex model. (a) Depth map. (b) Depth map edges. (c) Normal map. (d) Normal map edges. (e) Combined depth and normal map edges. (See also the Color Plates section of the course notes.)

Bibliography

[Rusinkiewicz +00] S. Rusinkiewicz, M. Levoy, “QSplat: A Multiresolution Point Rendering System for Large Meshes” Siggraph 2000.

[Bosch+02] M. Botsch, A. Wiratanaya, L. Kobbelt, “Efficient High Quality Rendering of Point Sampled Geometry”, Eg Rendering WS 2002.

[Borgo+01] Borgo, P. Cignoni, R. Scopigno, “An easy to use visualization system for huge cultural heritage meshes”, *VAST'01 Conference*, Greece, Nov. 28-30, 2001.

How to contact

- ***Visual Computing Group of ISTI/CNR***
  http://vcg.iei.pi.cnr.it
- Roberto Scopigno
  roberto.scopigno@cnuce.cnr.it