NSF Engineering Research Center
for Computer Integrated Surgical
Systems and Technology

LABORATORY FOR Computational Sensing + Robotics
THE JOHNS HOPKINS UNIVERSITY

Segmentation and Modeling

CIS I – 600.455/655

Russell H. Taylor
John C. Malone Professor of Computer Science,
with joint appointments in Mechanical Engineering, Radiology & Surgery
Director, Laboratory for Computational Sensing and Robotics
The Johns Hopkins University
rht@jhu.edu

100 YEARS
JOHNS HOPKINS ENGINEERING

WHITING SCHOOL OF ENGINEERING
THE JOHNS HOPKINS UNIVERSITY

Note: This lecture contains many slides from colleagues, including Jerry Prince, Eric Grimson, and Ayushi Sinha.

I have tried to make appropriate acknowledgments on the sides

1

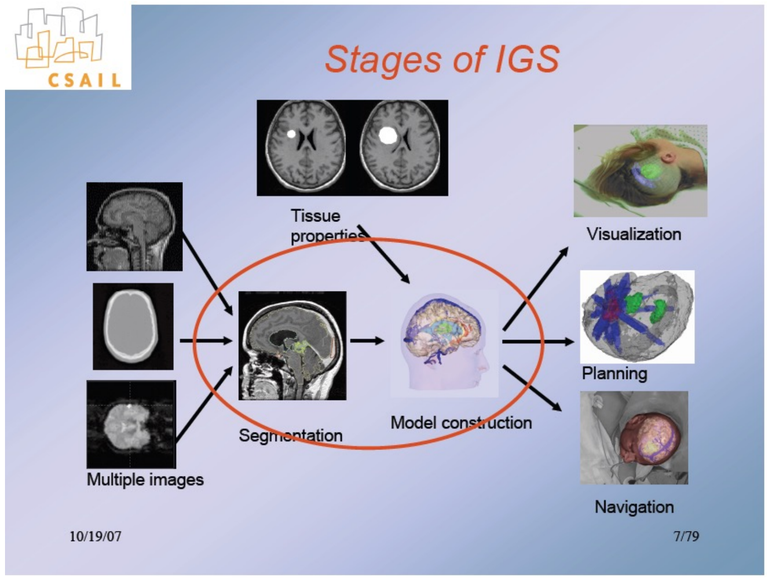
Segmentation & Modeling

Images

Segmented Images

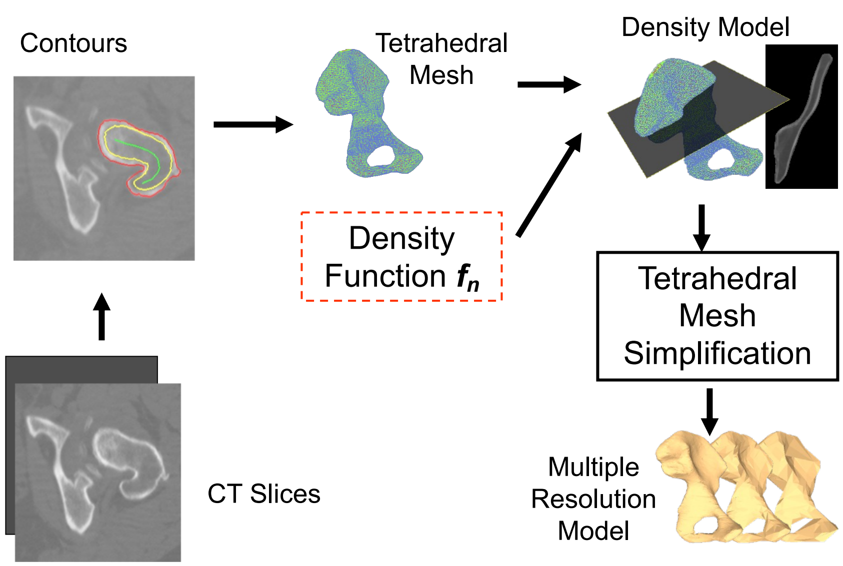
Models

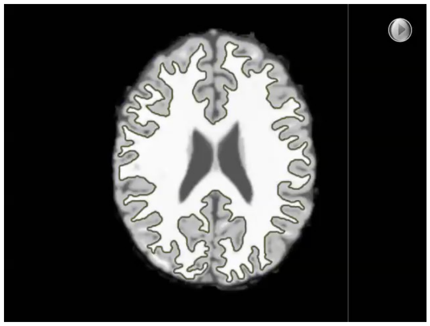
2

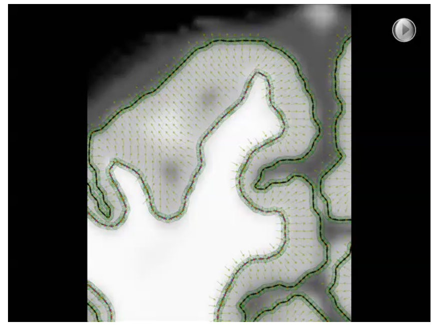


Credit: Eric Grimson

Example: Bone Modeling from CT







Brain Examples: Blake Lucas

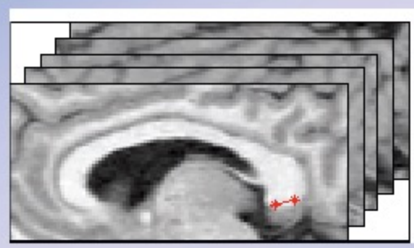
5

Image Segmentation

- Process of identifying structure in 2D & 3D images
- Output may be
  - labeled pixels
  - edge map
  - set of contours

6

Manual Segmentation
(Outlining)



- Extremely time-consuming (~6 hours per case)
- 3D Imagery – Performed slice at a time
- Some structures near impossible (blood vessels)

10/19/07 10/79

Credit: Eric Grimson

7

Automation Approaches

- Pixel-based
  - Thresholding
  - Region growing
  - Machine learning approaches
- Edge/Boundary based
  - Contours/boundary surface
  - Deformable warping
  - Deformable registration to atlases

8

Thresholding

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

Thresholding

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

Thresholding

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

11

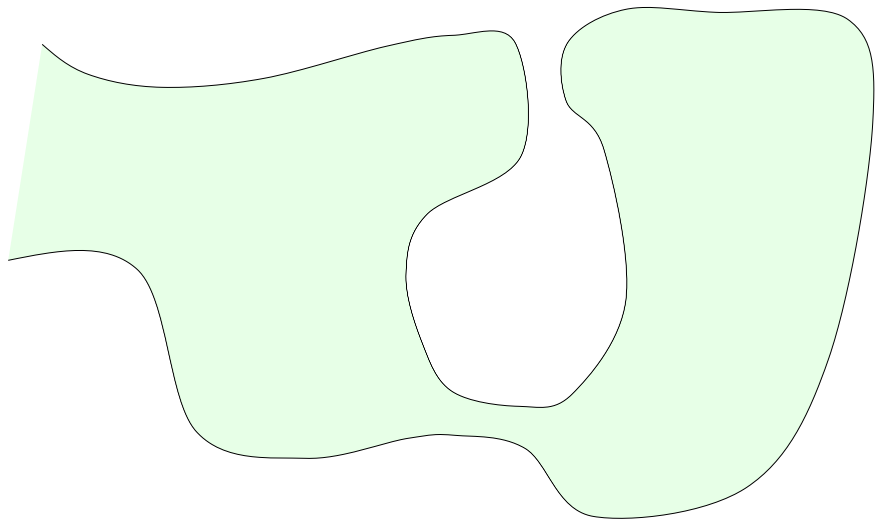
Thresholding

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

Copyright © 1999-2022 R. H. Taylor
Computer Integrated Surgery 600.445/645

12

“Partial volume” effects



Computer Integrated Surgery 600.445/645

13

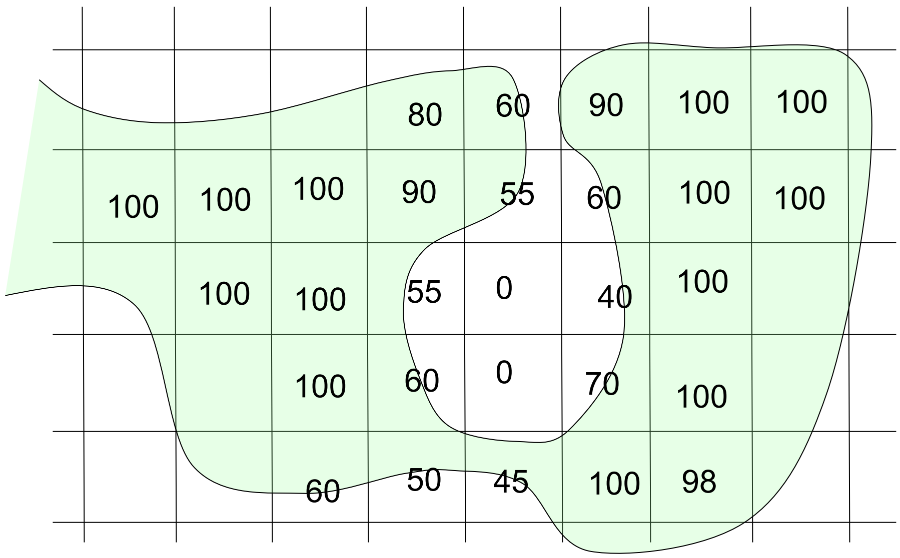
“Partial volume” effects

			80	60	90	100	100
100	100	100	90	55	60	100	100
	100	100	55	0	40	100	
		100	60	0	70	100	
		60	50	45	100	98	

Computer Integrated Surgery 600.445/645

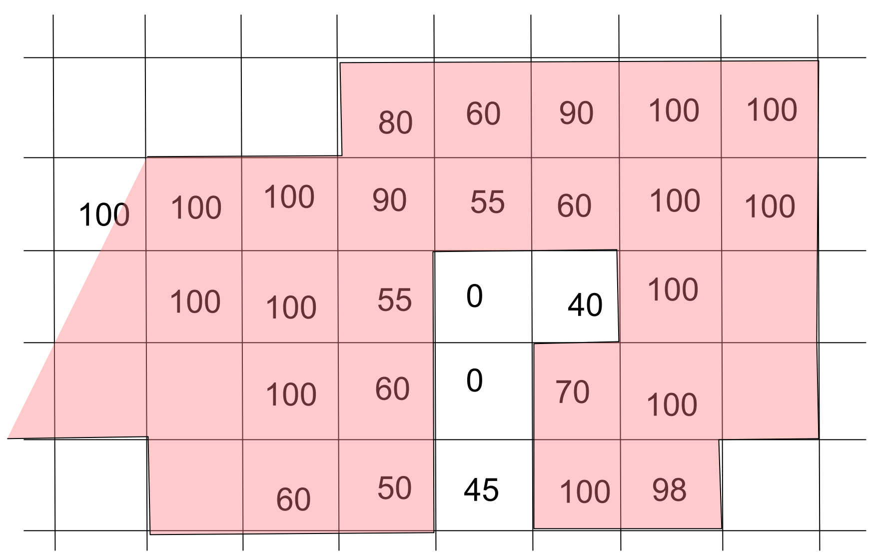
14

“Partial volume” effects



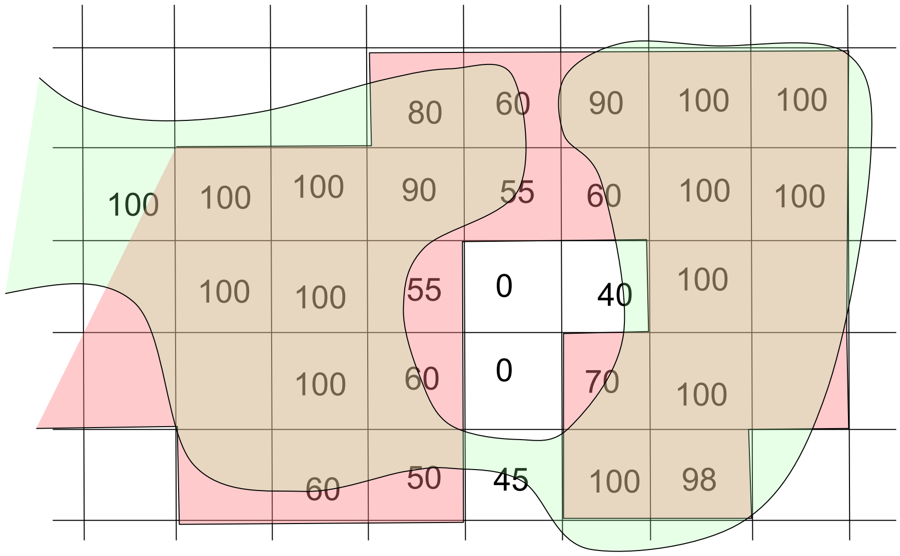
15

“Partial volume” effects



16

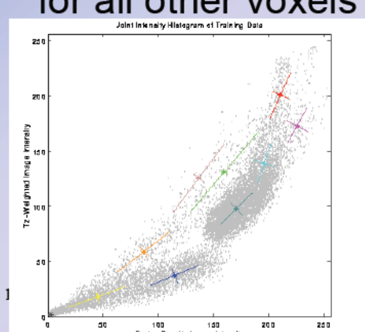
“Partial volume” effects

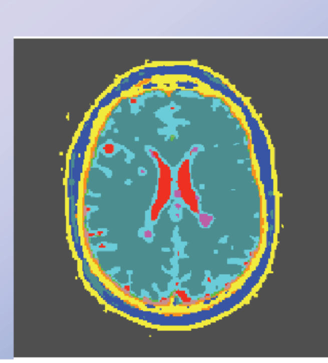


17

Segment statistically

- Measure distribution of intensities at known tissue locations
- Use nearest neighbor style classifiers for all other voxels

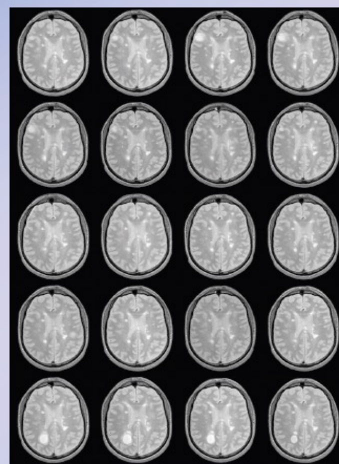




Credit: Eric Grimson

23

Standard Scans

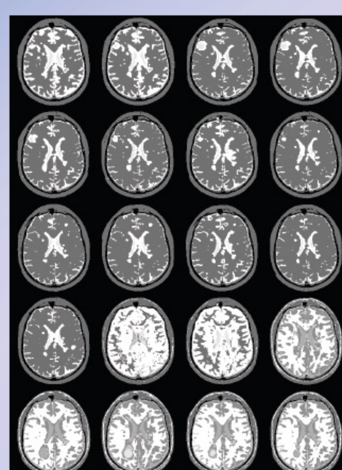


10/19/07 13/79

Credit: Eric Grimson

24

Statistical segmentation



10/19/07 14/79

Credit: Eric Grimson

25

Between Scylla and Charybdis

- Problem: imagery contains non-linear gain artifacts that shift the intensity values in a non-stationary way
- If one knew the gain field, could correct image and use standard statistical method
- If one knew the tissue types, could predict the image and find the gain field correction
- Solution: Use Expectation/Maximization method to iteratively solve for gain field and tissue class, using probabilistic models

10/19/07 15/79

Credit: Eric Grimson

26

EM-Segmentation [Wells 1994]

E-Step

Compute tissue posteriors using current intensity correction.

Estimate intensity correction using residuals based on current posteriors.

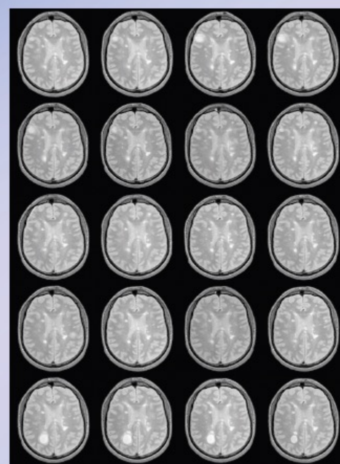
M-Step

10/19/07 16/79

Credit: Eric Grimson

27

Standard Scans

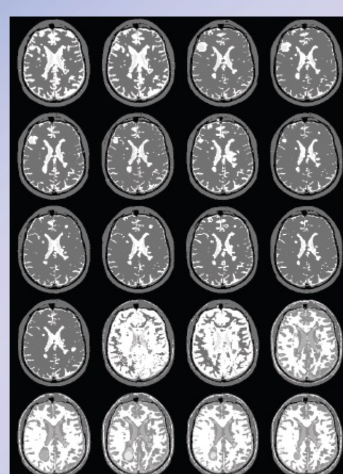


10/19/07

13/79

Credit: Eric Grimson

Statistical segmentation

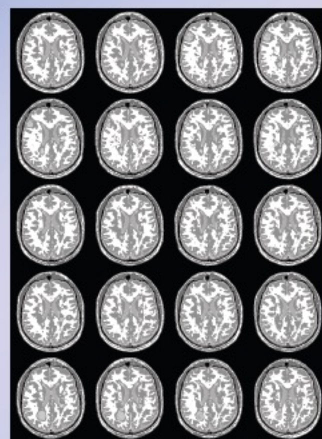


10/19/07

18/79

Credit: Eric Grimson

Gain Corrected Scans



10/19/07 19/79

Credit: Eric Grimson

30

Deformable Surfaces

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

31

Deformable Surfaces

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

Deformable Surfaces

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

Deformable Surfaces

3	5	7	3	4	2	1
2	4	9	10	22	9	3
3	5	12	11	15	10	3
5	6	11	9	17	19	1
2	3	11	12	18	16	2
3	6	8	10	18	9	5
4	6	7	8	3	3	1

- Basic concepts proposed by Demetri Terzopoulis
  - M. Kass, A. Witkin, and D. Terzopoulos, "Snakes:Active Contour Models", *Intl Journal of Computer Vision*, pp. 321-331, 1988.
- Many refinements since then

34

Traditional Active Contour

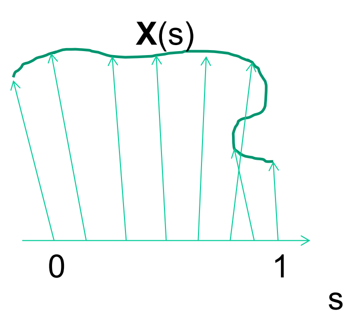
- Initialize a curve **X**(s) around or near the object boundary
- Find **X**(s) that minimizes:

$$E = \int_0^1 \left[\frac{1}{2} \left\{ \alpha |\mathbf{X}'(s)|^2 + \beta |\mathbf{X}''(s)|^2 \right\} + E_{\text{ext}}\{\mathbf{X}(s)\} \right] ds$$

- Where α = 0.001, β = 0.09 and

$$E_{\text{ext}}(x,y) = -\|\nabla f(x,y)\|^2$$

- How to find **X**(s)?



35

Dynamic Equation From E-L Equation

- Euler-Lagrange equation

$$\frac{\partial}{\partial s}\left(\alpha\frac{\partial \mathbf{X}}{\partial s}\right)-\frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 \mathbf{X}}{\partial s^2}\right)-\nabla P(\mathbf{X})=0$$

- Make $\mathbf{X}$ dynamic: $X(s) \rightarrow X(s,t)$

$$\mathbf{X}(s,t) = [X(s,t),Y(s,t)] \quad \text{where } s \in [0,1]$$

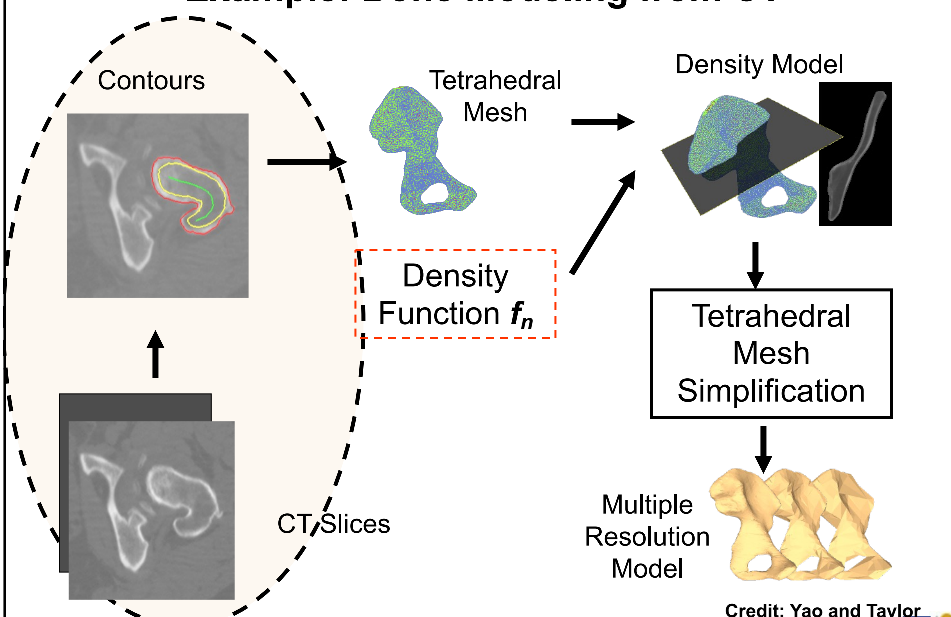
- Now set "in motion" – gradient descent

$$\gamma\frac{\partial \mathbf{X}}{\partial t}=\frac{\partial}{\partial s}\left(\alpha\frac{\partial \mathbf{X}}{\partial s}\right)-\frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 \mathbf{X}}{\partial s^2}\right)-\nabla P(\mathbf{X})$$

- General dynamical equation for snake:

$$\gamma\mathbf{X}_t=\mathbf{F}_{\text{int}}+\mathbf{F}_{\text{ext}}$$

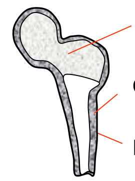
Example: Bone Modeling from CT



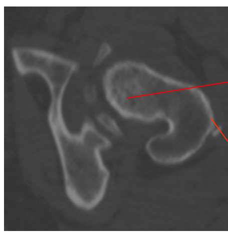
Credit: Yao and Taylor

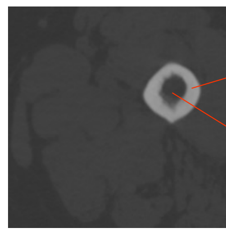
Bone Structure

- Compact bone
- Spongy bone
- Medullary Cavity



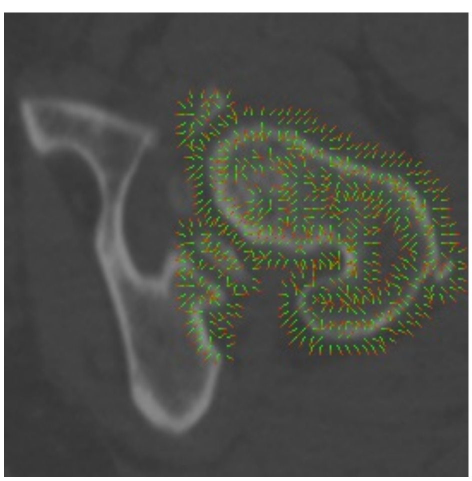
Bone Structure



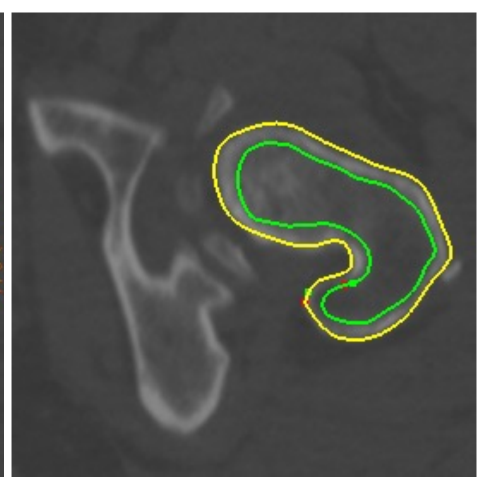


Credit: Yao and Taylor

Bone Contour Extraction



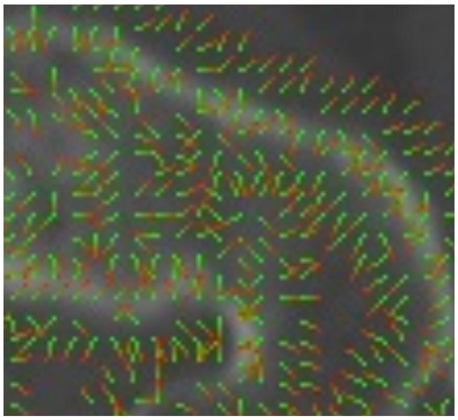
Needle graph of Image force



Bone Contours

Credit: Yao and Taylor

Bone Contour Extraction
Closer-up view



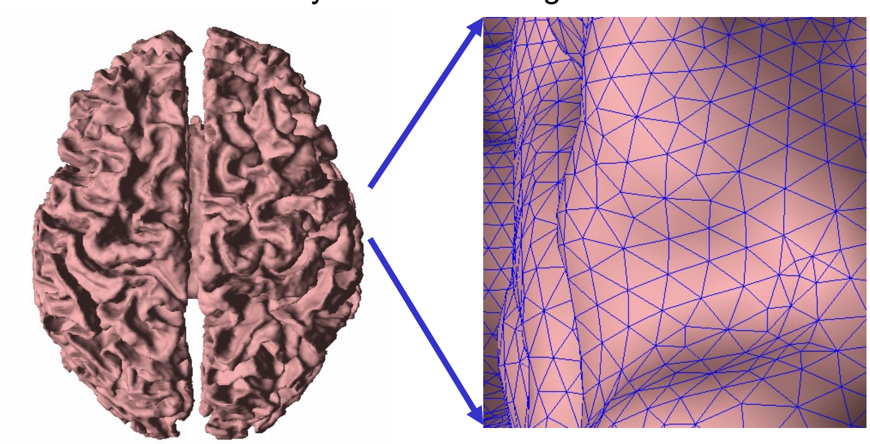
Needle graph of Image force

Bone Contours

Credit: Yao and Taylor

3D Deformable Surface Model

Commonly done with triangle mesh



- Added complexity, time, especially to avoid self-intersection

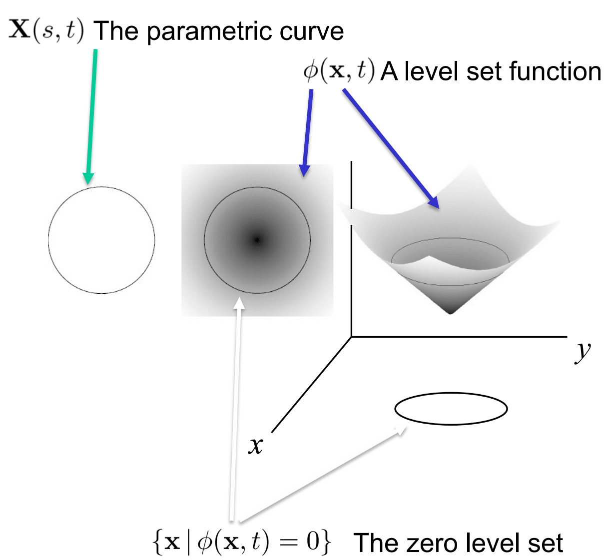
© Jerry L. Prince

Critique of Parametric Models

- Advantages:
  - explicit equations, direct implementation
  - automatic topology control
- Disadvantes:
  - costly to prevent overlaps
  - requires reparameterization to space out triangles

44

Basic Idea of Geometric Active Contours

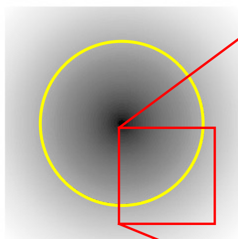


- The level set function is usually a signed distance function
- Convention:
  - positive on outside
  - negative on inside

45

GDM: Geometric Deformable Model

- Conventional level set function $\phi(x,t)$
  - signed distance function
- Change the values of ϕ ➔ move the contour



-5.50	-4.50	-3.50	-2.50	-1.50	-0.50	0.50	1.50
-4.50	-4.08	-3.26	-2.34	-1.37	-0.40	0.58	1.57
-3.50	-3.26	-2.67	-1.89	-1.03	-0.11	0.82	1.78
-2.50	-2.34	-1.89	-1.25	-0.50	0.33	1.20	2.12
-1.50	-1.38	-1.02	-0.50	0.16	0.90	1.71	2.56
-0.50	-0.40	-0.11	0.33	0.90	1.57	2.31	3.10
0.50	0.58	0.82	1.21	1.71	2.31	2.98	3.72
1.50	1.57	1.78	2.11	2.56	3.10	3.72	4.40

46

Philosophy of GDMs

- Curve is not parameterized until the end of evolution
  - tangential forces are meaningless
  - forces must be derived from “spatial position” and “time” because location on the curve is meaningless
  - Final contour is an “isocurve” (2D) or “isosurface” (3D)
  - It has a “Eulerian” rather than “Lagrangian” framework
- Speed function incorporates internal and external forces
  - Design of geometric model is accomplished by selection of F(x), the speed function
  - curvature terms takes the place of internal forces
- “Action” is near the zero level set
  - “narrowband” methods are computationally more efficient

47

Ventricle Segmentation

© Jerry L. Prince

50

Cortical Surface Segmentation

© Jerry L. Prince

51

Critique of Geometric Deformable Models

- Advantages:
  - Produce closed, non-self-intersecting contours
  - Independent of contour parameterization
  - Easy to implement: numerical solution of PDEs on regular computational grid
  - Stable computations
- Disadvantages:
  - topologically flexible
  - some numerical difficulties with narrowband and level set function reinitialization

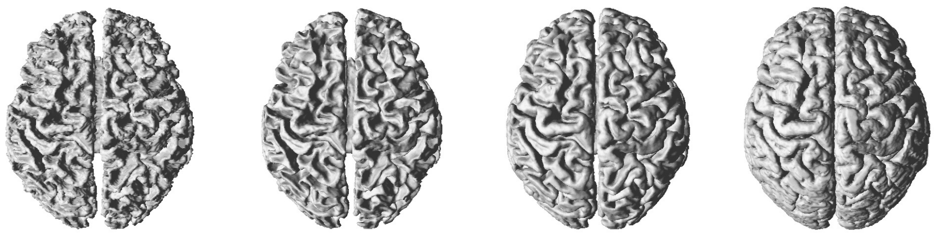
Topology Preserving Geometric Deformable Model (TGDM)

- Evolve level set function according to GDM PDE
- If level set function is going to change sign, check whether the point is a simple point
  - If simple, permit the sign-change
  - If not simple, prohibit the sign-change
  - (replace the grid value by epsilon with same sign)
  - (Roughly, this step adds 7% computation time.)
- Extract the final contour using a *connectivity consistent isocontour algorithm*

X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25- 6, pp. 755-768, 2003.

Nested Deformable Surfaces

TGDM-1 TGDM-2 TGDM-3

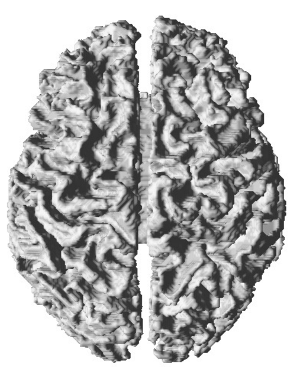


Initial WM Isosurface | Inner Surface | Central Surface | Pial Surface

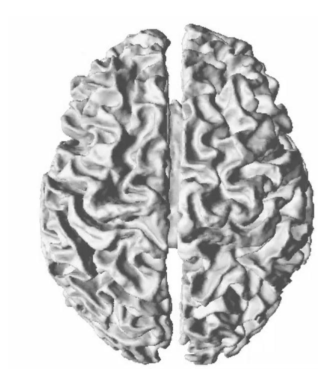
54

TGDM for Inner Surface

[Han et al., NeuroImage, 2004]



Initial WM Isosurface

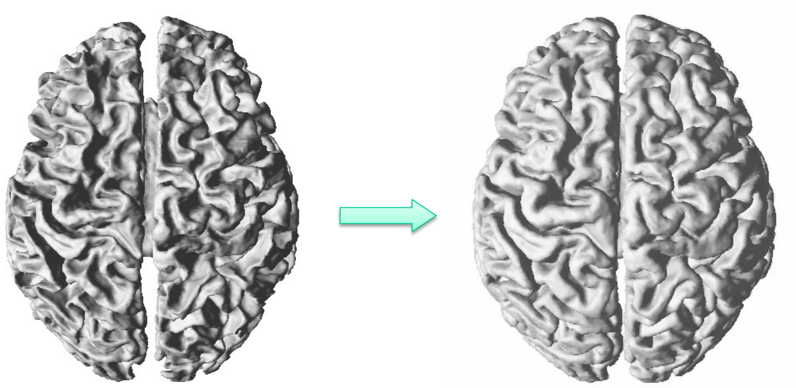


Evolving GM/WM Interface

$$\Phi_t = (\omega_1 R(\bar{x}) + \omega_2 \kappa(\bar{x})) \|\nabla \Phi\|$$

55

TGDM for Central Surface

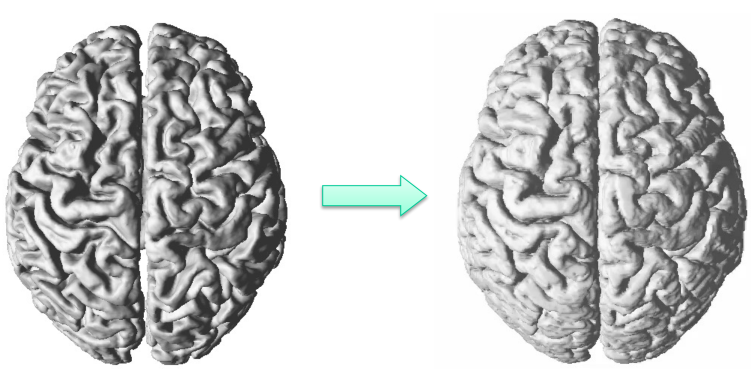


Initialize with GM/WM surface

Evolving toward Central Surface

$$\Phi_t = (\omega_1 R(\bar{x}) + \omega_2 \kappa(\bar{x}))\|\nabla\Phi\| + \omega_3 F_{\mathsf{GVF}}(\bar{x}) \cdot \nabla\Phi$$

TGDM for Outer Surface



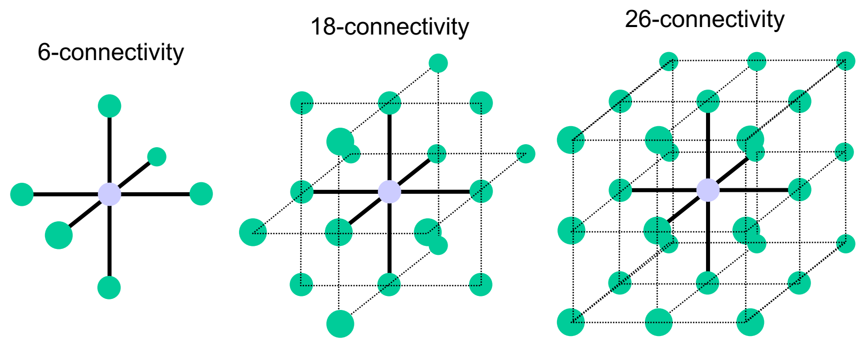
Start from Central Surface

Evolving toward Outer Surface

$$\Phi_t = (\omega_1 R(\bar{x}) + \omega_2 \kappa(\bar{x}))\|\nabla\Phi\| + \omega_3 F_{\mathsf{GVF}}(\bar{x}) \cdot \nabla\Phi$$

3D Digital Connectivity

- In 3D there are three connectivities: 6, 18, and 26
- Four consistent connectivity pairs:
  (foreground, background) → (6,18), (6,26), (18,6), (26,6)



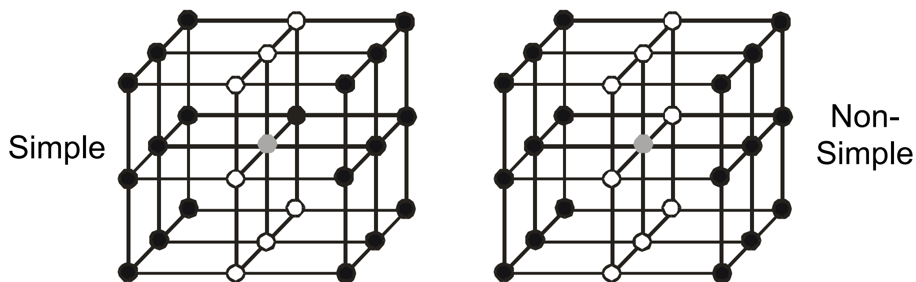
Topology Preservation Principle

[Han et al., PAMI, 2003]

- Preserving topology is equivalent to maintaining the topology of the digital object
- The digital object can only change topology when the level set function changes sign at a grid point
- To prevent the digital object from changing topology, the level set function should only be allowed to change sign at ***simple*** points

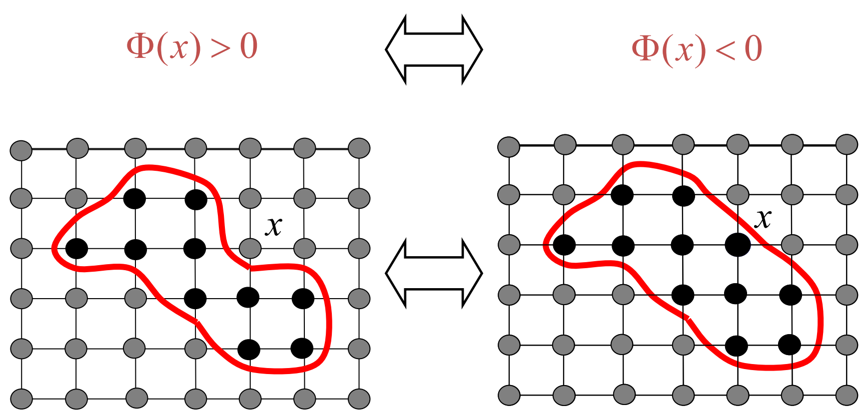
Simple Point

- Definition: a point is simple if adding or removing the point from a binary object will not change the digital object's topology
- Determination: can be characterized locally by the configuration of its neighborhood (8- in 2D, 26- in 3D) [Bertrand & Malandain 1994]



60

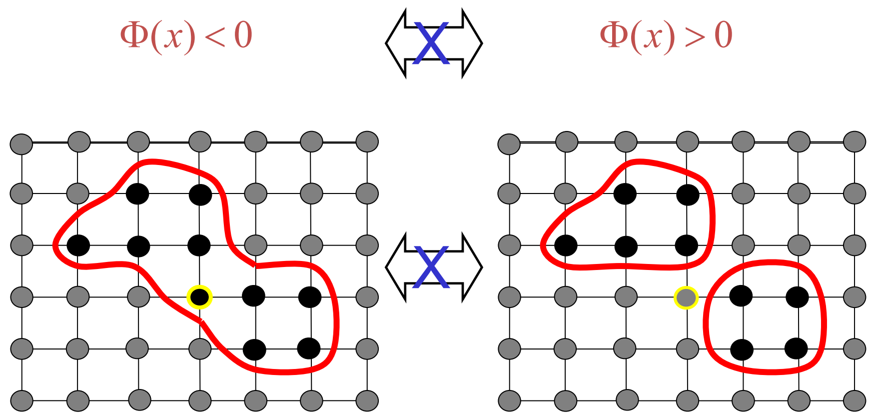
x is a Simple Point



(Connectivity happens to be irrelevant in this case)

61

x is Not a Simple Point (if n=4)

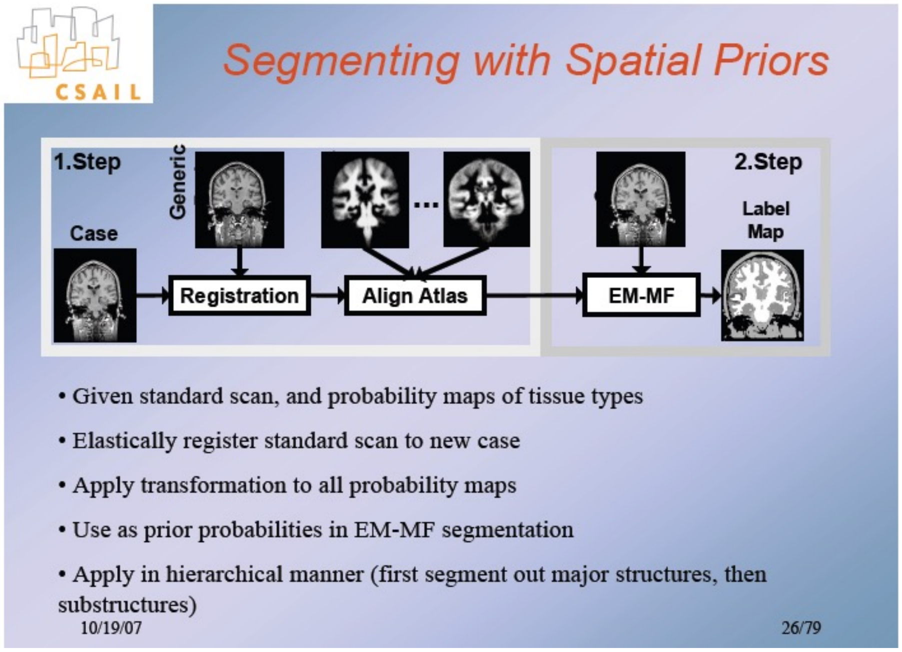


Digital connectivity assumption is crucial in this case

Topology Preserving Geometric Deformable Model (TGDM)

- Evolve level set function according to GDM PDE
- If level set function is going to change sign, check whether the point is a simple point
  - If simple, permit the sign-change
  - If not simple, prohibit the sign-change
  - (replace the grid value by epsilon with same sign)
  - (Roughly, this step adds 7% computation time.)
- Extract the final contour using a *connectivity consistent isocontour algorithm*

X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25- 6, pp. 755-768, 2003.

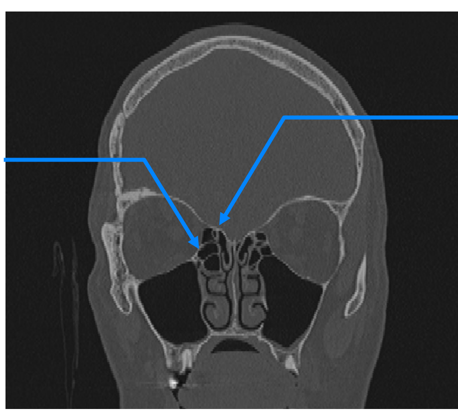


Credit: Eric Grimson

Example: Sinuses & Nasal Airway

Complex structures with thin boundaries

Boundary between the sinuses and the orbit
Thickness: ~ **0.91 mm**[4]

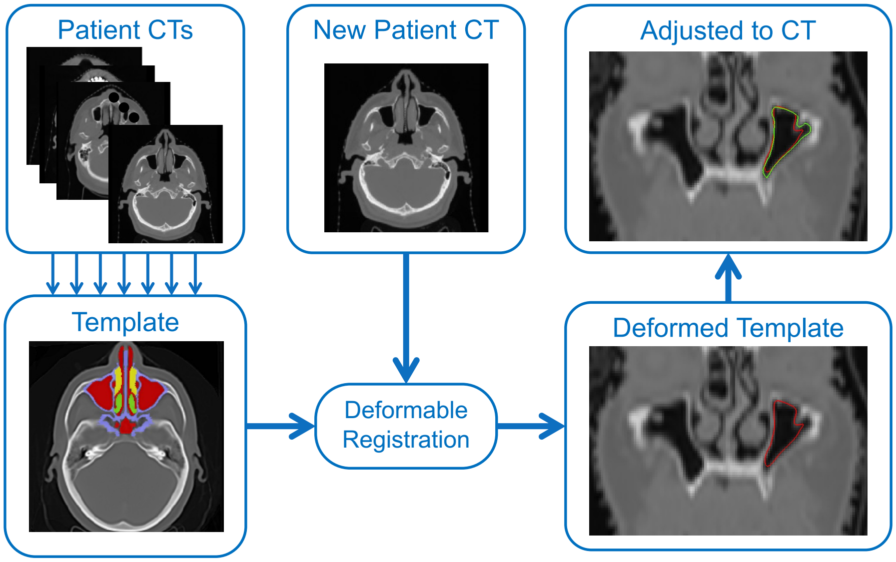


Fovea ethmoidalis: separates the ethmoid cells from the anterior cranial fossa
Thickness: ~ **0.5 mm**[3]

[3] Kainz, J. and Stammberger, H., "The roof of the anterior ethmoid: A place of least resistance in the skull base," American Journal of Rhinology 3(4), 191-199 (1989).
[4] Tao, H., Ma, Z., Dai, P., and Jiang, L., "Computer-aided three-dimensional reconstruction and measurement of the optic canal and intracanalicular structures," The Laryngoscope 109(9), 1499-1502 (1999).

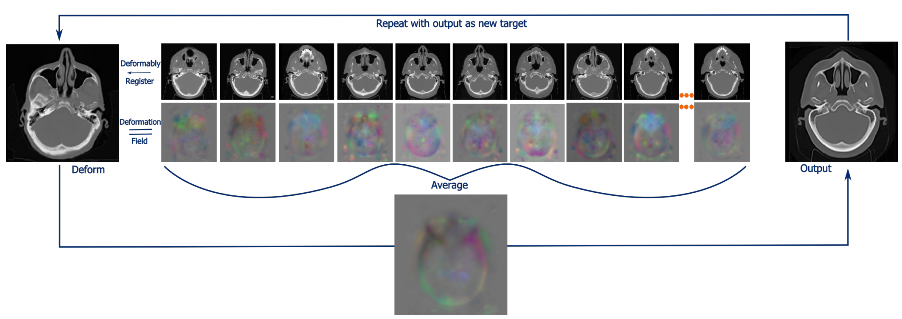
Slide Credit: Ayushi Sinha

Automated Segmentation Workflow



A. Sinha, S. Leonard, A. Reiter, M. Ishii, R. H. Taylor, and G. D. Hager, "Automatic segmentation and statistical shape modeling of the paranasal sinuses to estimate natural variations", in Proc. SPIE 9784, San Diego, Feb. 27, 2016. pp. 97840D.1-8.

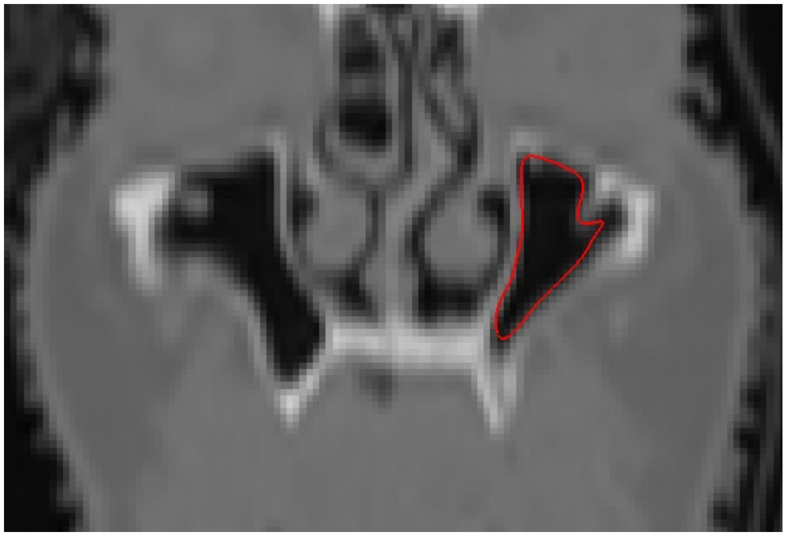
Template Creation



BB Avants, P Yushkevich, J Pluta, D Minko, M Korczykowski, J Detre, JC Gee, "The optimal template effect in hippocampus studies of diseased populations," NeuroImage 49(3), p. 2457, 2010.

Slide Credit: Ayushi Sinha

Deformable Registration of Template to Image

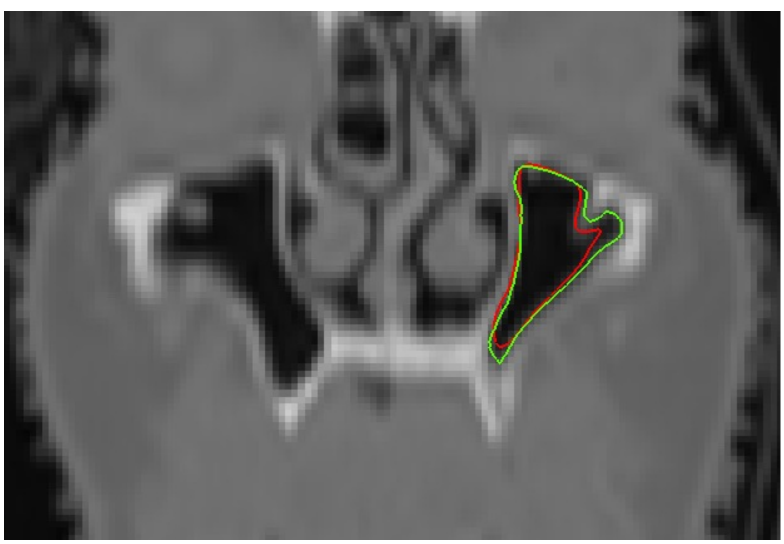


BB Avants, NJ Tustison, . Song, PA Cook, A Klein, and JC Gee, "A reproducible evaluation of ANTs similarity metric performance in brain image registration," NeuroImage 54(3), pp. 2033-2044, 2011.

 Slide Credit: Ayushi Sinha

68

Adjustment of Template to Patient CT

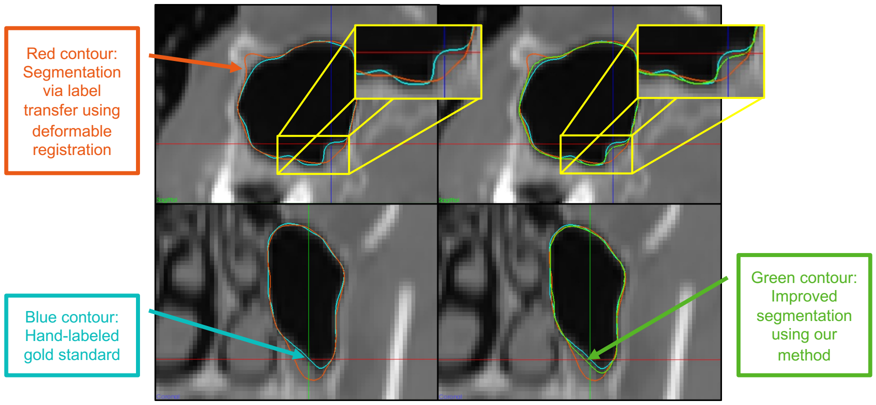


[10] C. Xu and J. L. Prince, "Gradient vector flow: A new external force for snakes," in IEEE Computer Vision and Pattern Recognition, pp. 66-71, 1997.
[11] C. Xu and J. Prince, "Snakes, shapes, and gradient vector flow,", IEEE Transactions on Image Processing, 7, pp. 359-369, March 1998.

 Slide Credit: Ayushi Sinha

69

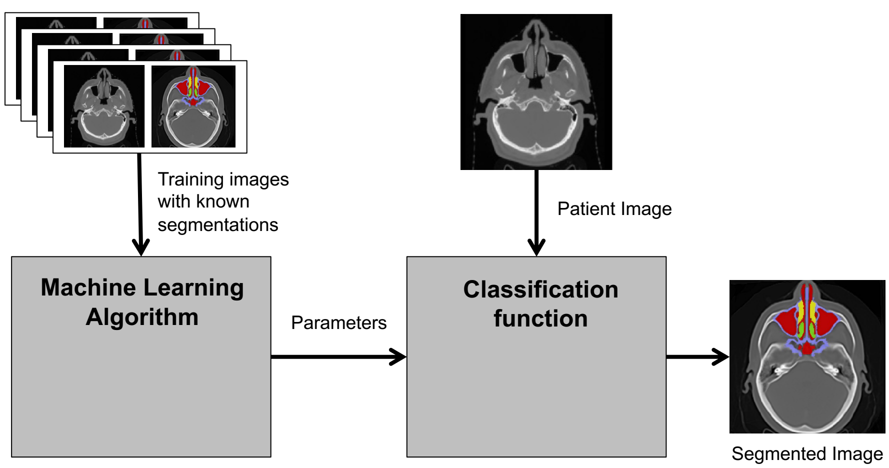
Results



A. Sinha, A. Reiter, S. Leonard, M. Ishii, G. D. Hager, and R. H. Taylor, "Simultaneous segmentation and correspondence improvement using statistical modes", in SPIE Medical Imaging, Orlando, 2017.

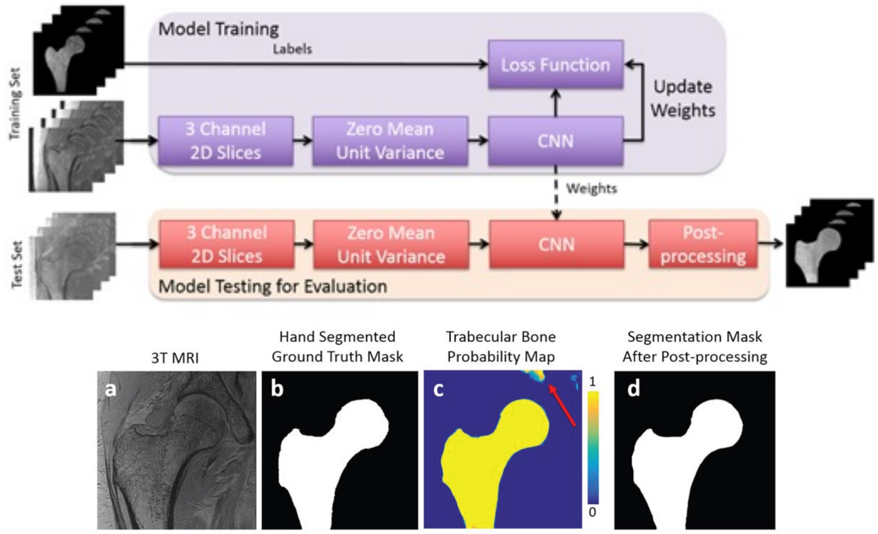
Slide Credit: Ayushi Sinha

Machine Learning Methods



- Basic approach has been used in one form or another for many years
- Emergence of modern convolutional neural nets with GPUs has made these approaches extremely successful recently
- However, require large amounts of training data

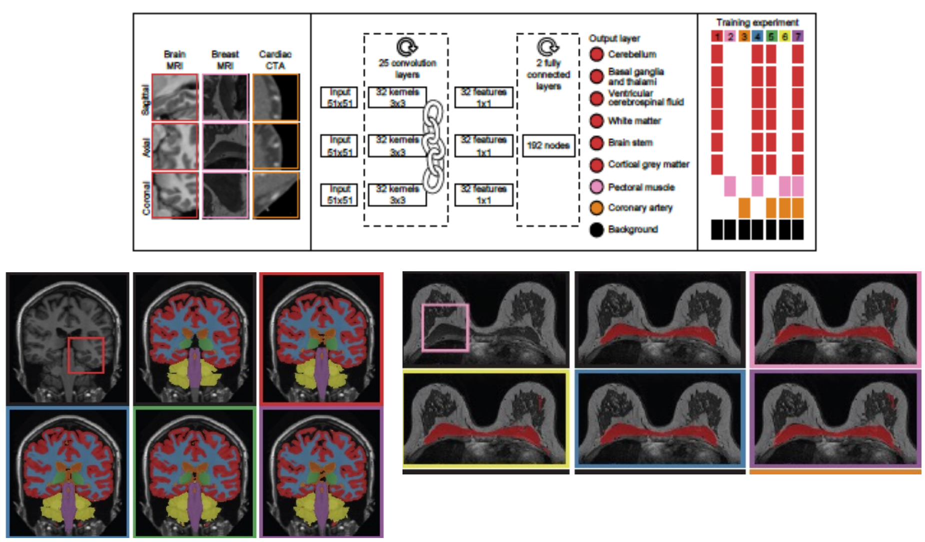
Example: Segmentation of Femur in MRI



Cem M. Deniz, Spencer Hallyburton, Arakua Welbeck, Stephen Honig, Kyunghyun Cho, Gregory Chang," Segmentation of the Proximal Femur from MR Images using Deep Convolutional Neural Networks', https://arxiv.org/abs/1704.06176, 2017.

72

Example: Deep Learning in Multi-Modality Segmentation



This paper has been published in October 2016 as: Moeskops, P., Wolterink, J.M., van der Velden, B.H.M., Gilhuijs, K.G.A., Leiner, T., Viergever, M.A., and Isgum, I. (2016). Deep learning for multi-task medical image segmentation in multiple modalities. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI; 2016, Part II, LNCS 9901, pp. 478-486

73

Modeling

- Representation of anatomical structures
- Models can be
  – Images
  – Labeled images
  – Boundary representations

75

FROM VOXELS TO SURFACES

Representing solids:

- B-REP - surface representation,
  d/s of vertices, edges, faces.
- CSG- composition of primirive solids

binary image → **B-REP representation**

Surface construction algorithms:

- 2D-based algrorithms
- 3D-based algorithms

76

Surface Representations

- Implicit Representations
  $\{\bar{x} \mid f(\bar{x}) = 0\}$
- Explicit Representations
  - Polyhedra
  - Interpolated patches
  - Spline surfaces
  - ...

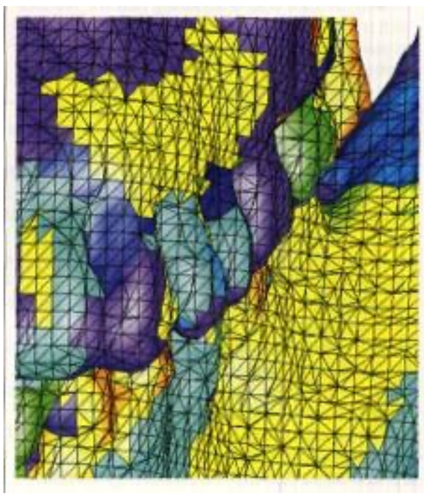
FIGURE 4.7 Segmentation of vertebra defined by a set of CT slices. Four steps of the deformation of a roughly spherical snake spline toward the vertebra are shown.

Source: CIS p 73 (Lavallee image)

77

Polyhedral Boundary Reps

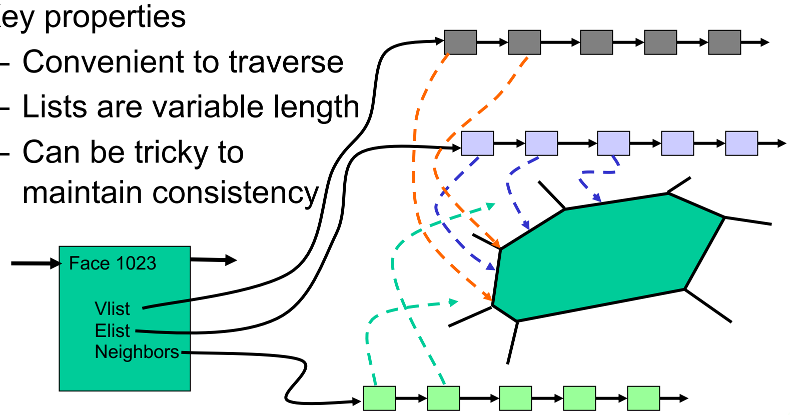
- Common in computer graphics
- Many data structures.
  - FEV lists
  - Winged edge
  - Connected triangles
  - etc.



78

FEV lists

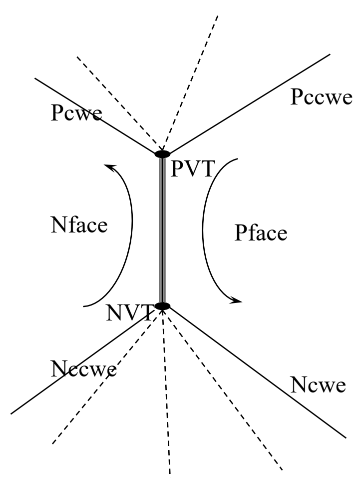
- Explicit linked lists of faces, edges, vertices
- Many variations
- Key properties
  - Convenient to traverse
  - Lists are variable length
  - Can be tricky to maintain consistency



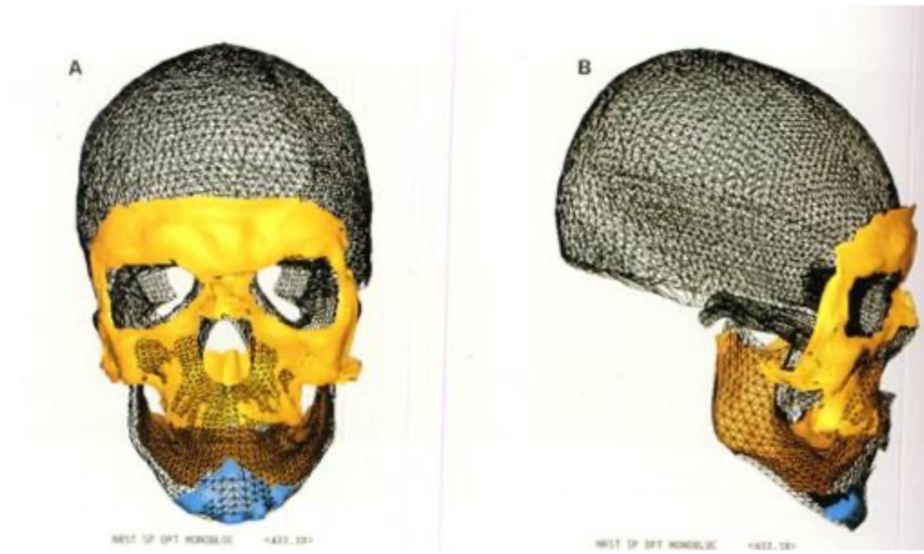
79

Winged Edge

- Baumgart 1974
- Basic data structures
  - winged edge (topology)
  - vertex (geometry)
  - face (surfaces)
- Key properties
  - constant element size
  - topological consistency



80

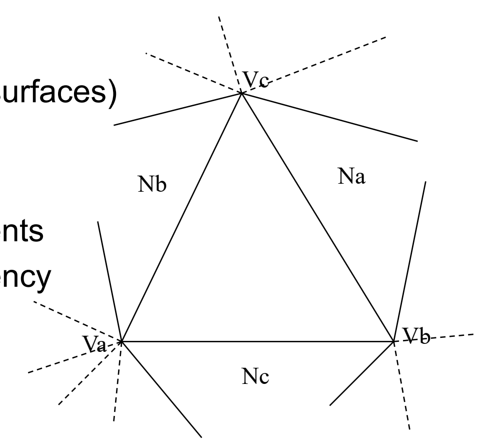


Source: C. Cutting, CIS Book

81

Connected Triangles

- Basic data structures
  - Triangle (topology, surfaces)
  - Vertex (geometry)
- Properties
  - Constant size elements
  - Topological consistency

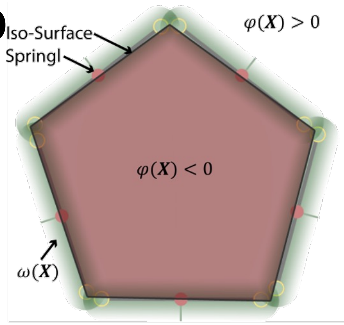


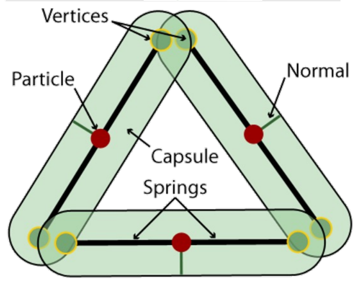
82

Anatomy of a SpringI

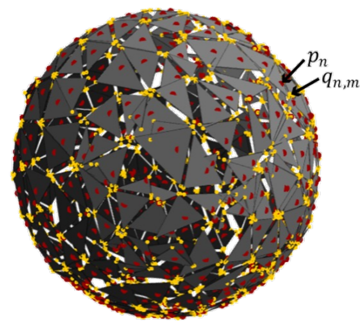
Blake Lucas

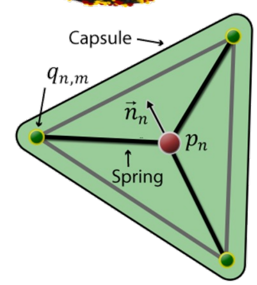
2D





3D

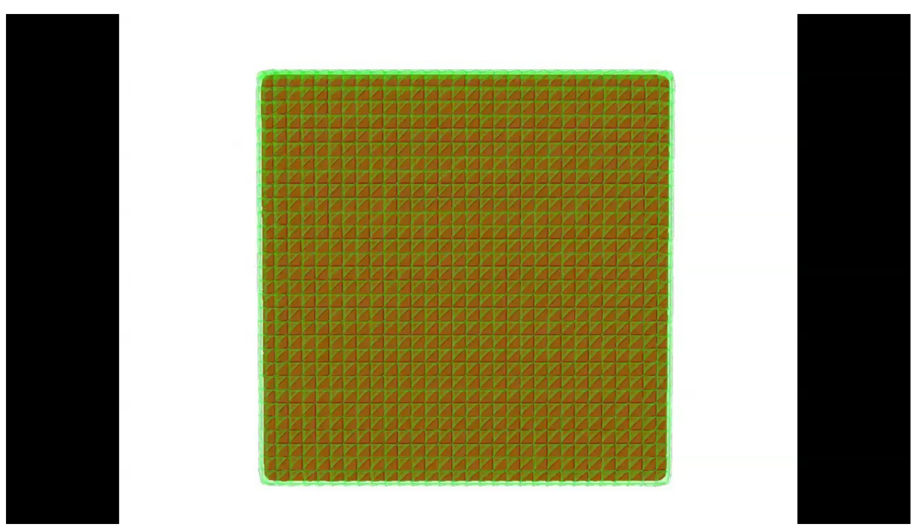




Slide credit: Blake Lucas' PhD Thesis Defense, 2012

83

Deformable Surfaces & Level Sets

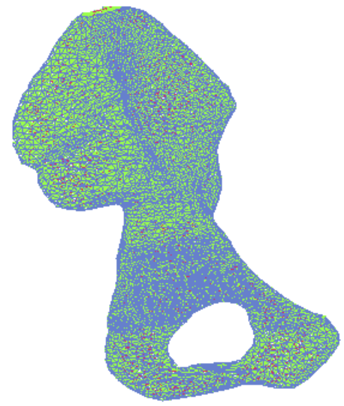


Blake Lucas – "SpringIs" (October 2010)

84

Tetrahedral Mesh Data Structure

- Vertex list
  - x, y, z coordinates
  - reference to one tetrahedron
- Tetrahedron list
  - references to four vertices
  - references to four face neighbors
- Properties such as density functions



Credit: Yao and Taylor

85

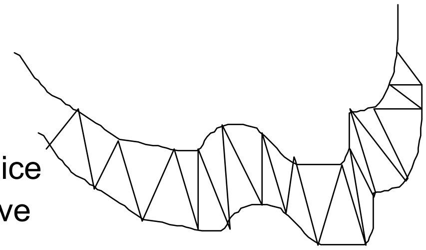
Advantages of Tetrahedral Mesh

- Greatest degree of flexibility
- Data structure, data traversal, and data rendering are more involved
- Ability to better adapt to local structures
- Computational steps such as interpolation, integration, and differentiation can be done in closed form
- Finite element analysis
- Hierarchical structure of multiple resolution meshes

Credit: Yao and Taylor

86

2D-based Methods for Shape Reconstruction



- Treat 3D volume as a stack of slices
- Outline
  - Find contours in each 2D slice
  - Match contours in successive slices
  - Connect contours to create tiled surfaces (for boundary representation)
  - Use contours to guide subdivision of space between slices into tetrahedra (for volumes)

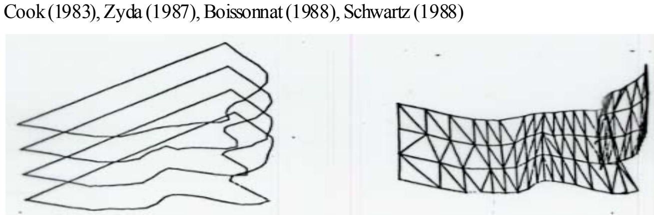
89

SURFACE CONSTRUCTION ALGORITHMS

2D-based algorithms

1. 2D contour extraction
2. tiling of counours

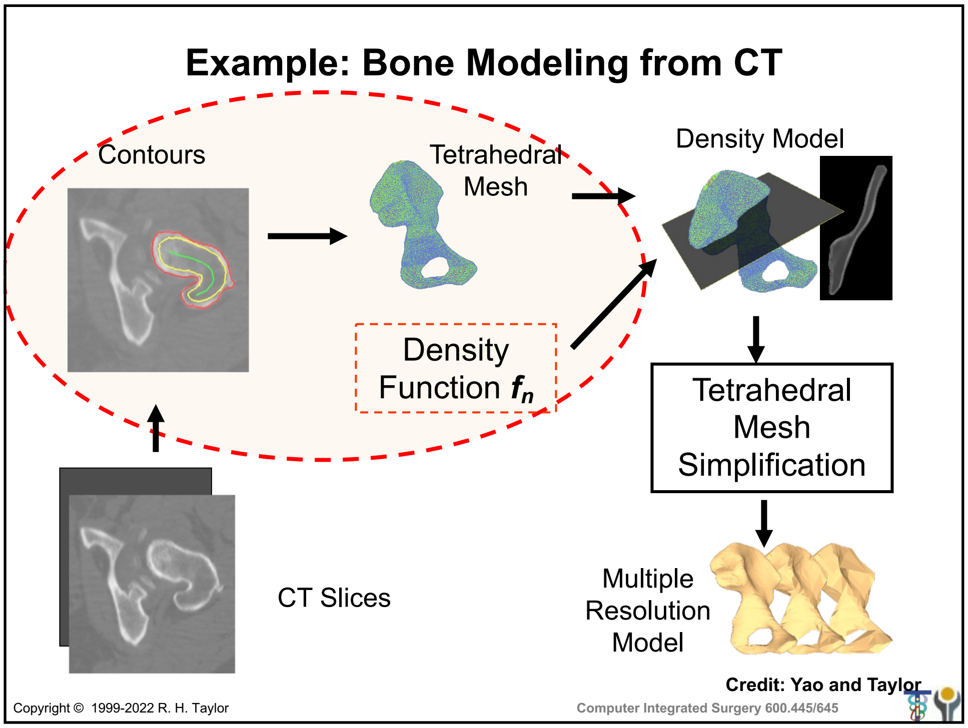
Keppel (1975), Fuchs (1978), Christiansen (1981), Shantz (1981), Ganapathy (1982), Cook (1983), Zyda (1987), Boissonnat (1988), Schwartz (1988)



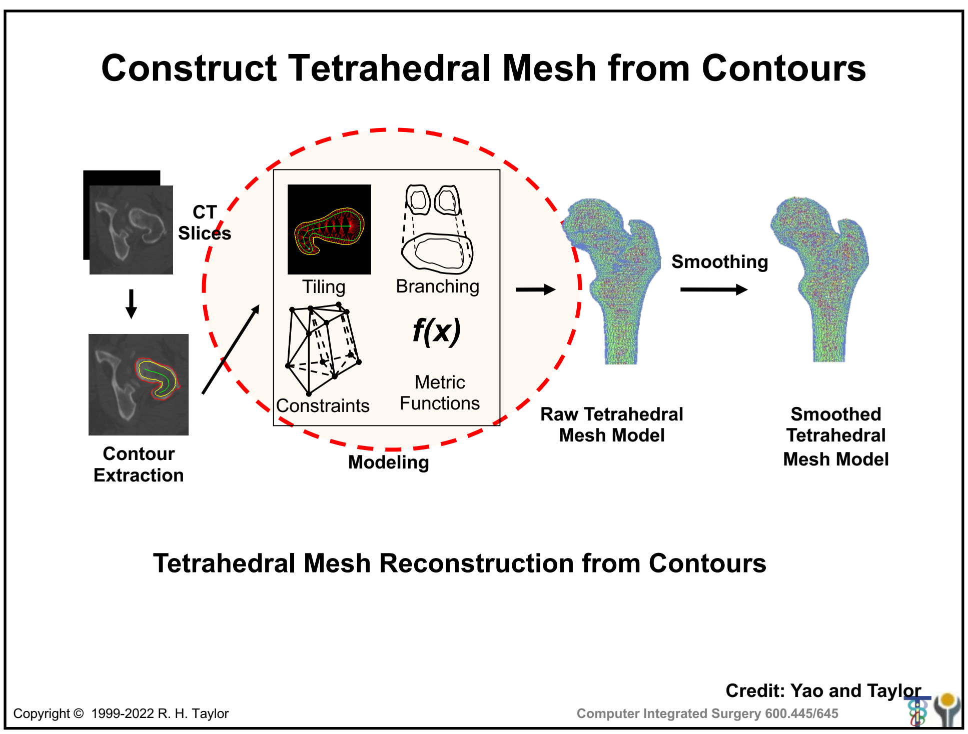
Contour extraction

- Sequential scanning
- boundary following (random access to pixels)

90


Example: Bone Modeling from CT
Contours
Tetrahedral Mesh
Density Model
Density Function f_n
Tetrahedral Mesh Simplification
CT Slices
Multiple Resolution Model
Credit: Yao and Taylor
Copyright © 1999-2022 R. H. Taylor
Computer Integrated Surgery 600.445/645

91


Construct Tetrahedral Mesh from Contours
CT Slices
Contour Extraction
Tiling
Branching
Constraints
f(x)
Metric Functions
Modeling
Raw Tetrahedral Mesh Model
Smoothing
Smoothed Tetrahedral Mesh Model
Tetrahedral Mesh Reconstruction from Contours
Credit: Yao and Taylor
Copyright © 1999-2022 R. H. Taylor
Computer Integrated Surgery 600.445/645

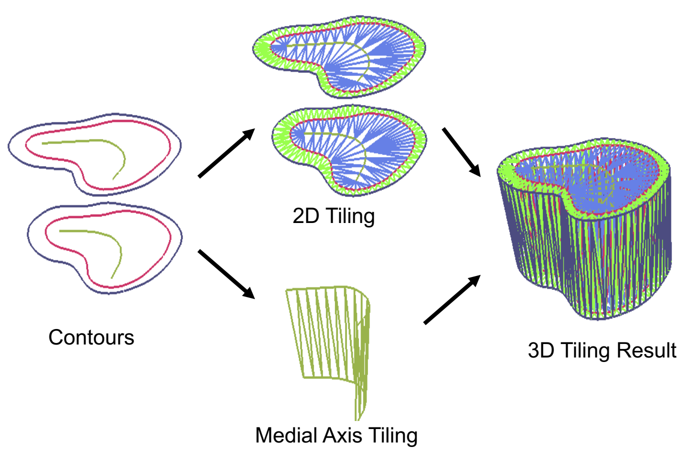
92

Tetrahedral Mesh Tiling

- Objectives
  - Subdivide the space between adjacent slices into tetrahedra, slice by slice
- Method
  - Two-steps tiling strategy
    - 2D tiling and medial axis tiling
    - 3D tiling

Credit: Yao and Taylor

Tiling Strategy



Credit: Yao and Taylor

Metric Functions

- Maximize Volume, f_v
- Minimize Area, f_a
- Minimize Density Deviation, f_d
- Minimize Span Length, f_s

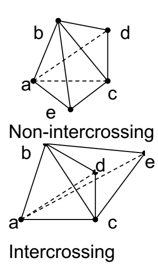
Current Metric Function:

- Combination of minimizing density deviation and span length
- Minimize $F = w_1 * f_d + w_2 * f_s$

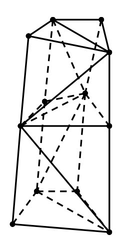
Credit: Yao and Taylor

Tiling Constraints

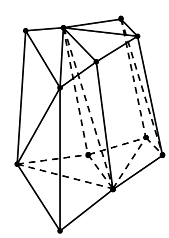
- Non-intersection between tetrahedra
- Continuity between slices
- Continuity between layers



Intercrossing between tetrahedra



Continuity constraint between slices

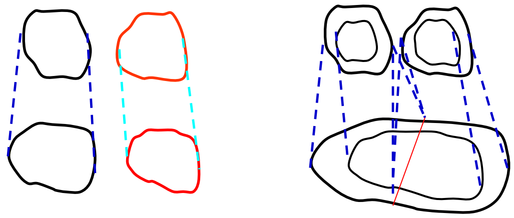


Continuity constraint between layers

Credit: Yao and Taylor

Correspondence Problem

- Examining the overlap and distance between contours on adjacent slices
- Graph based method



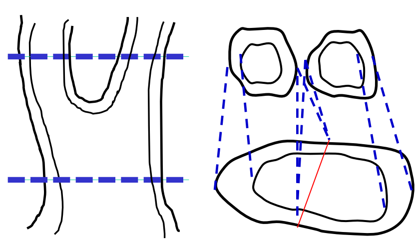
Contour Correspondence

Credit: Yao and Taylor

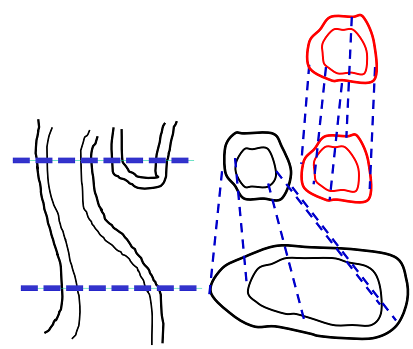
97

Branching Problem

- Branching Between layers
  - Convert to tiling of 3 contours
- Branching Between contours
  - Composite contour
  - Split contour



Split Contour



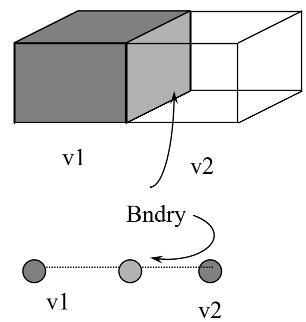
Composite Contour

Credit: Yao and Taylor

98

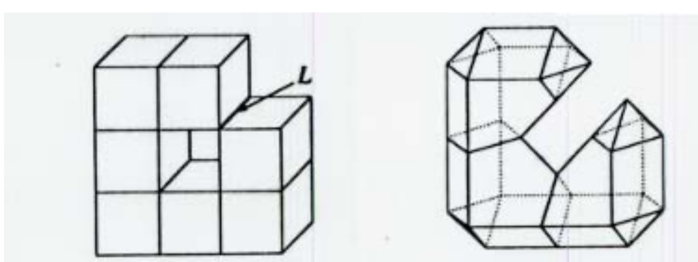
3D-based methods for Surface Reconstruction

- Segment image into labeled voxels
- Define surface and connectivity structure
- Can treat boundary element between voxels as a face or a vertex



3D-BASED ALGORITHMS

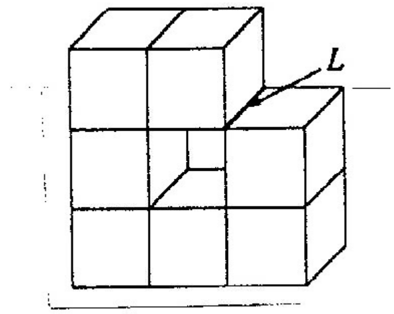
Block-form and Beveled-form representations of surface:



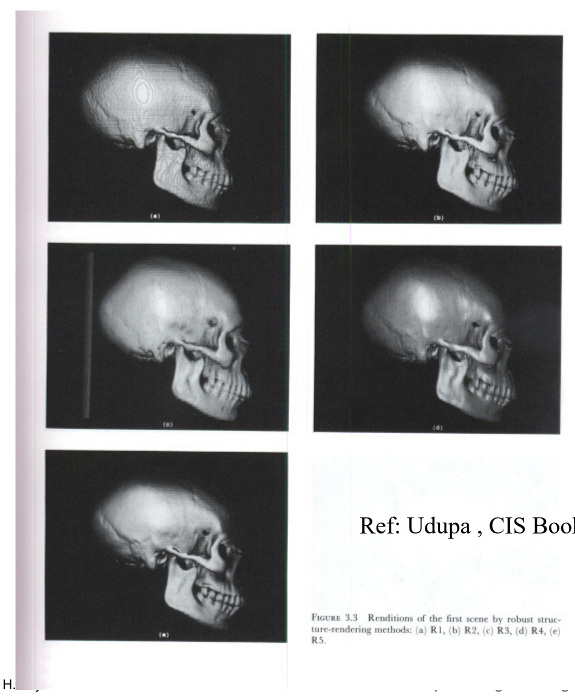
(a) Block-form representation. (b) Beveled-form representation.

Block form methods

- "Cuberille"-type methods
- Treat voxels as little cubes
- May produce self-intersecting volumes
- E.g., Herman, Udupa



101



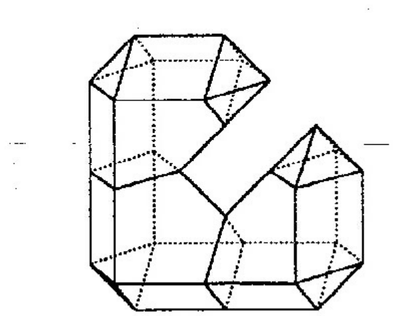
Ref: Udupa , CIS Book, p47

FIGURE 3.3 Renditions of the first scene by robust structure-rendering methods: (a) R1, (b) R2, (c) R3, (d) R4, (e) R5.

102

Beveled form methods

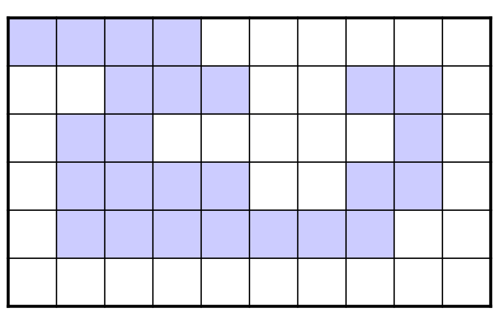
- “Marching cubes” type
- Voxels viewed as 3D grid points
- Vertices are points on line between adjacent grid points
- E.g. Lorensen&Cline, Baker, Kalvin, many others



103

Block form to beveled form

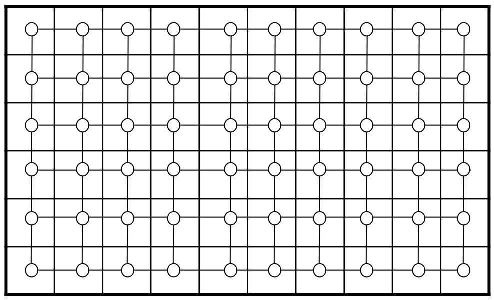
Segmented voxels



104

Block form to beveled form

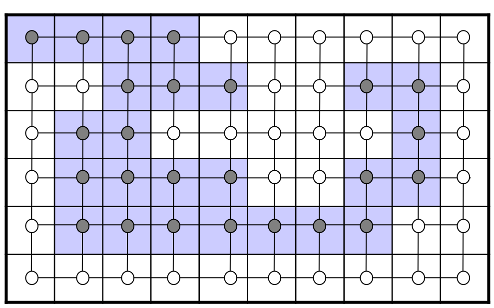
Duality between voxels and vertices on adjacency graph



105

Block form to beveled form

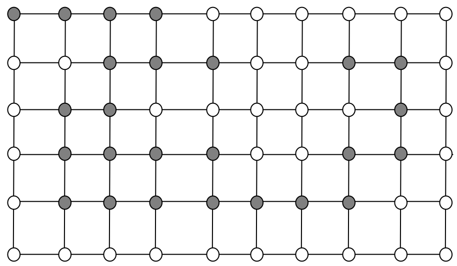
Label vertices based on segmentation labels



106

Block form to beveled form

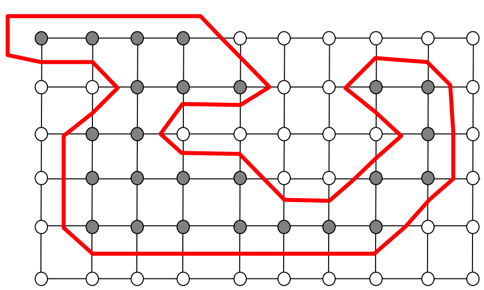
Label vertices based on segmentation labels



107

Block form to beveled form

Boundary crosses edges between occupied and empty vertices

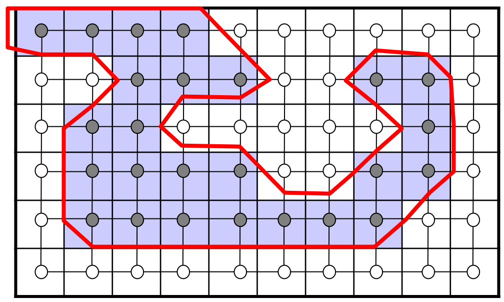


108

Block form to beveled form

Boundary crosses edges between occupied and empty vertices

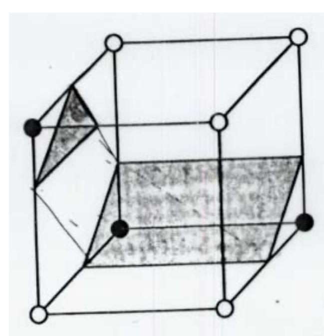
Note: Choice of exact vertex placement is somewhat arbitrary. One choice is linear interpolation along edge based on density.



109

Beveled form basic approach

- Segment the 3D volume
- Scan 3D volume to process "8-cells" sequentially
- Use labels of 8 cells as index in (256 element) lookup table to determine where surfaces pass thru cell
- Connect up topology
- Use various methods to resolve ambiguities

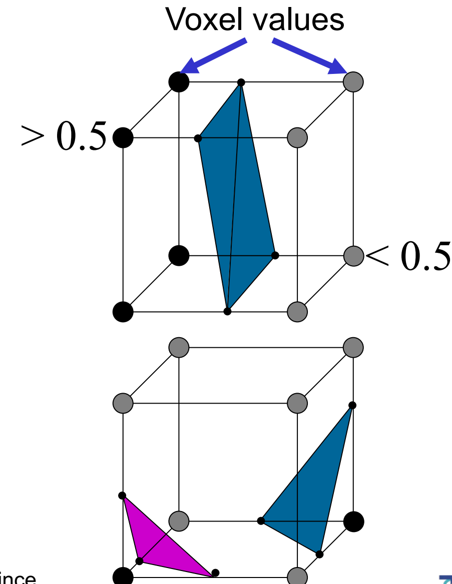


Source: Kalvin survey

111

Marching Cubes Isosurface Algorithm

- How to "tile/triangulate" the zero level set?
- Consider values on corners of voxel (cube)
- Label as
  - above isovalue
  - below isovalue
- Determine the position of a triangular mesh surface passing through the voxel
  - Linear interpolation



112

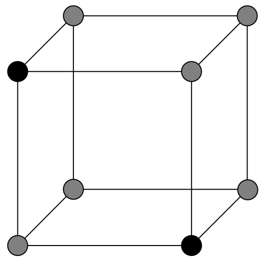
Connectivity Errors

Most isosurface codes use rules that lead to connectivity errors

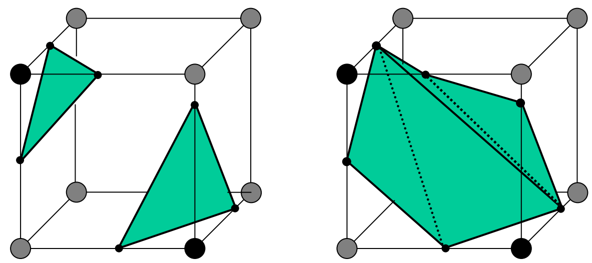
- Multiple meshes
  - typically solved by selecting the largest mesh
- Touching vertices, edges, and faces
  - typically solved isovalue choice
- Ambiguous faces and cubes
  - solved by use of a specially coded *connectivity consistent* MC algorithm

113

Ambiguous Faces

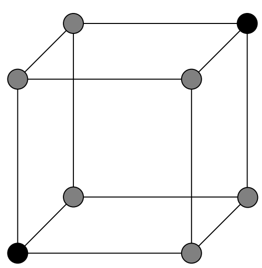


Two possible tilings:

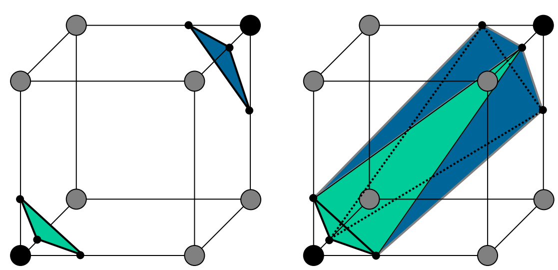


115

Ambiguous Cubes



Two possible tilings:



116

Wyvill, McPheters, Wyvill

Step 1: determine edges on each face of 8 cube

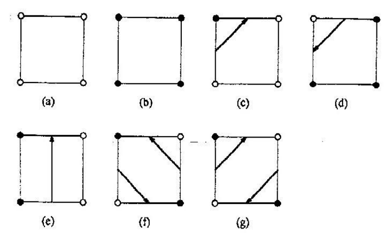
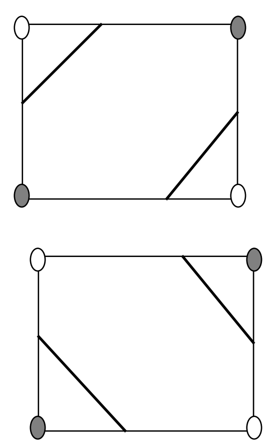


Figure 6: The seven cases for calculating vertices and ec

Step 2: Connect the edges up to make surfaces

Ambiguities

- Arise when alternate corners of a 4-face have different labels
- Ways to resolve:
  – supersampling
  – look at adjacencent cells
  – tetrahedral tessallation



Tetrahedral Tessalation

- Many Authors
- Divide each 8-cube into tetrahedra
- Connect tetrahedra
- No ambiguities

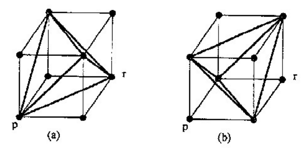


Figure 8: The two tetrahedral partitionings of an 8-cell.

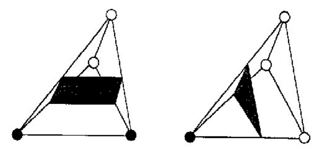
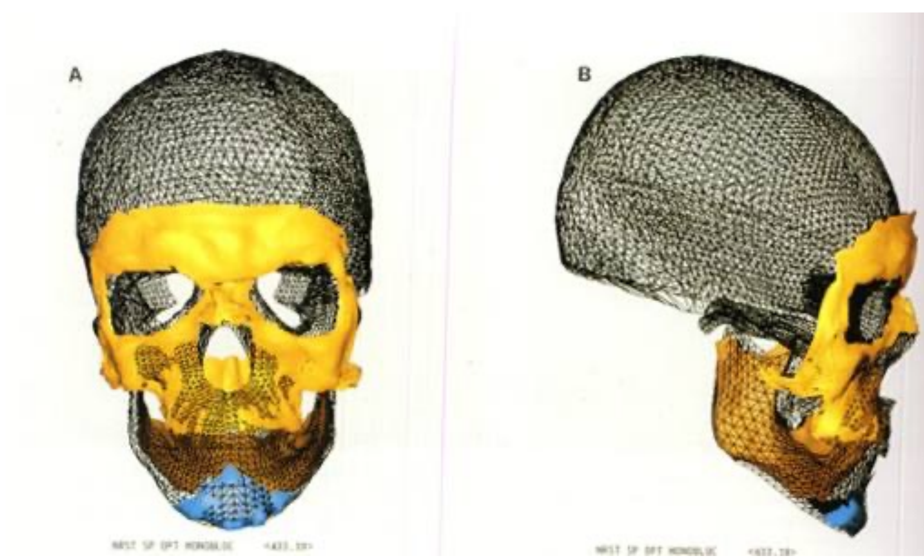


Figure 9: The two cases used for surface construction.

Beveled-form algorithms based on the tetrahedral decomposition of the 3D volume have been developed Payne and Toga [34], Hall and Warren [21], and Nielson *et al.* [29].

While this approach does provide a neat resolution to the ambiguous 8-cell problem, it

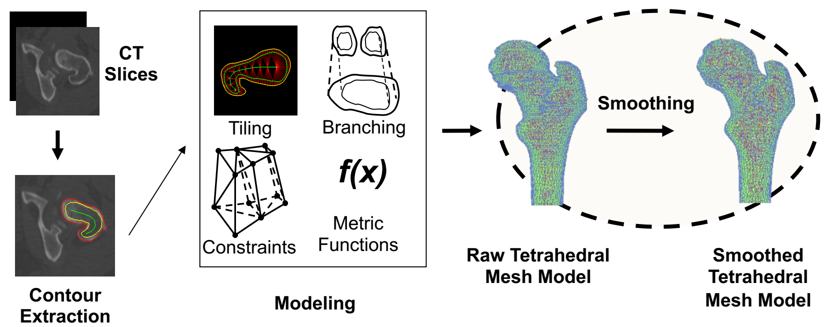
120



Source: C. Cutting, CIS Book

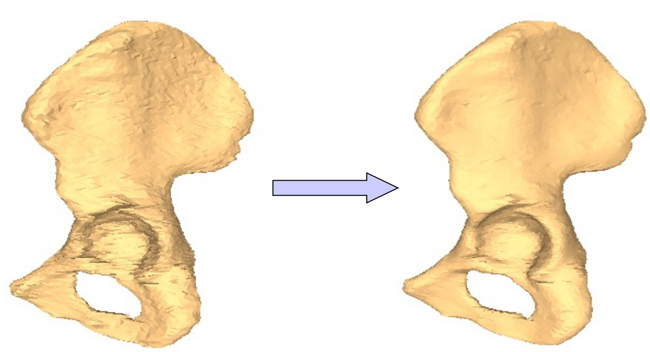
124

Mesh Smoothing



Mesh Smoothing

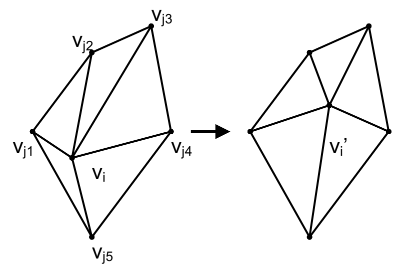
- Motivations
  - Noise/discretazition in CT data set
  - Artifacts during segmentation



Classic Laplacian Smoothing Method

- Equation

$$v_i' = \frac{1}{|N_i|} \sum_{j \in N_i} v_j$$



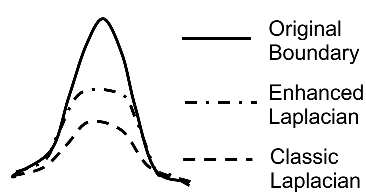
- Advantages
  - Fast and easy
- Drawbacks
  - Shrinkage
  - Invalid elements

127

Enhanced Laplacian Smoothing Method

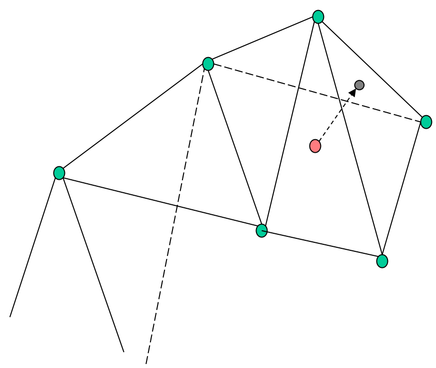
- Objective
  - Reduce shrinkage
- Method
  - Project back to boundary

$$v_i' = proj(\frac{1}{|N_i|} \sum_{j \in N_i} v_j)$$



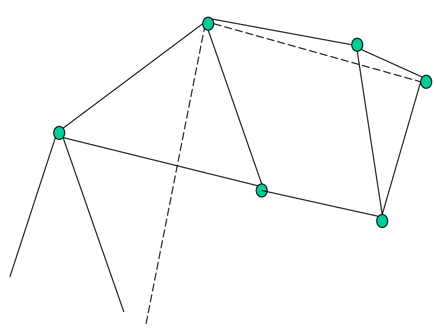
128

Average and reproject



129

Average and reproject



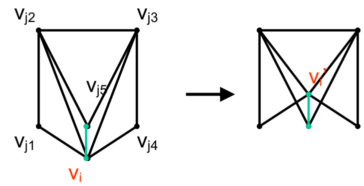
130

Enhanced Laplacian Smoothing Method

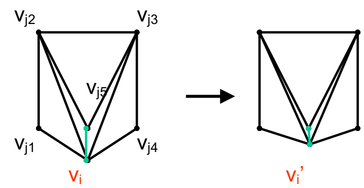
- Objective
  - Prevent invalid element
- Method
  - Iterative assignment

$$v_i^{'(0)} = proj\left(\frac{1}{|N_i|}\sum_{j \in N_i} v_j\right)$$

$$v_i^{'(k)} = \alpha \cdot v_i + (1-\alpha) v_i^{'(k-1)} \quad ,0 \le \alpha \le 1$$



Classic Laplacian



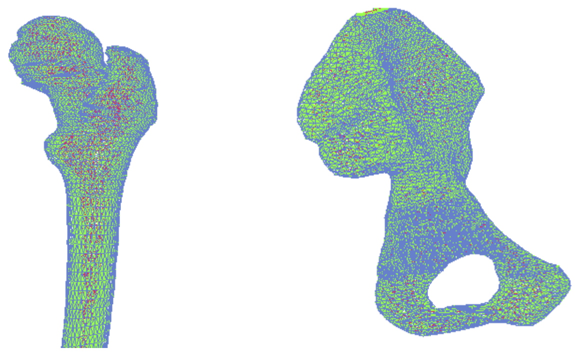
Enhanced Laplacian

Mesh Smoothing Results

a) Before Smoothing

b) After Smoothing

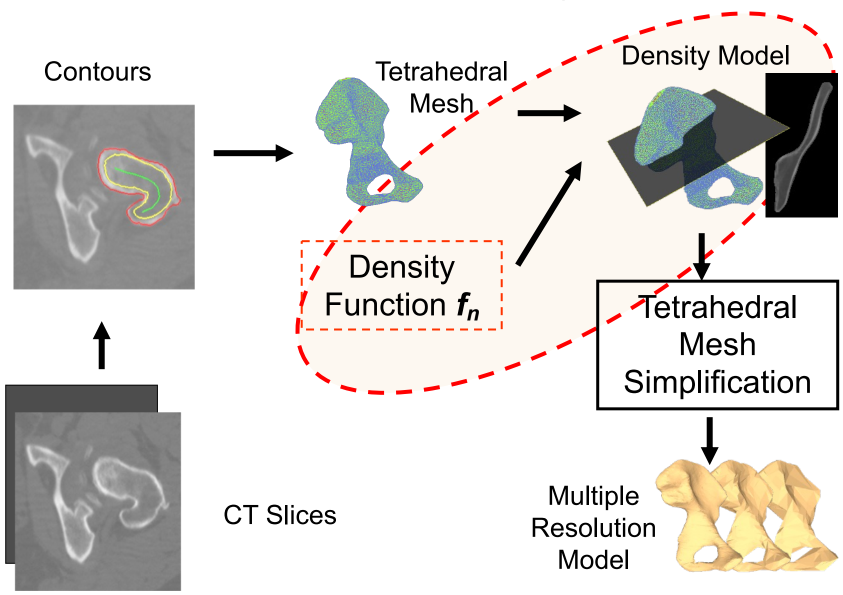
Tetrahedral Mesh Models



Model	Num of Vertices	Num of Tetrahedra	Num of Slices	Total Num of Voxels inside	Avg Num of voxels Per Tetra	Volume (mm³)	Avg Vol. Per Tetra (mm³)
Femur	6163	31,537	83	1,802,978	57.1	312,107	9.9
Pelvis	8219	32,741	110	1,941,998	59.3	347,070	10.6

133

Example: Bone Modeling from CT



134

Density Functions

- n-degree Bernstein polynomial in barycentric coordinate

$$D(\mu) = \sum_{i+j+k+l=n}^{n} C_{i,j,k,l} B^n_{i,j,k,l}(\mu)$$

$C_{i,j,k,l}$ polynomial coefficient

$B^n_{i,j,k,l}(\mu) = \frac{n!}{i!\,j!\,k!\,l!} \mu_x^i \mu_y^j \mu_z^k \mu_w^l$ barycentric Bernstein basis

Barycentric Coordinate of Tetrahedron

- Local coordinate system
- Symmetric and normalized
- Every 3D position can be defined by an unique coordinate (*x, y, z, w*)

  $V = x^*V_a + y^*V_b + z^*V_c + w^*V_d$

  $x+y+z+w=1$, V_a, V_b, V_c, V_d are coordinate of Tetrahedron vertices

  x,y,z,w within[0,1] if *V* is inside the tetrahedron

Density Functions

- Advantages
  - Efficient in storage
  - Continuous function
  - Explicit form
  - Convenient to integrate, to differentiate, and to interpolate

Fitting Density Function

- Minimize the density difference between the density function and CT data set

$$\min \sum_{\rho_i \in \Omega} \left(\left(\sum_{i+j+k+l=n}^{n} C_{i,j,k,l} B^{n}_{i,j,k,l}(\mu_{\rho_i}) \right) - T(\mu_{\rho_i}) \right)^2$$

Ω is the set of sample voxels, $T(\mu_{\rho i})$ is the density value from the CT data set.

$$\begin{bmatrix} B_1(\mu_{\rho 1}) & B_2(\mu_{\rho 1}) & \dots & B_m(\mu_{\rho 1}) \\ B_1(\mu_{\rho 2}) & B_2(\mu_{\rho 2}) & \dots & B_m(\mu_{\rho 2}) \\ \vdots & \vdots & \vdots & \vdots \\ B_1(\mu_{\rho s}) & B_2(\mu_{\rho s}) & \dots & B_m(\mu_{\rho s}) \end{bmatrix} \cdot \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_m \end{bmatrix} = \begin{bmatrix} T(\mu_{\rho 1}) \\ T(\mu_{\rho 2}) \\ \vdots \\ T(\mu_{\rho s}) \end{bmatrix}$$

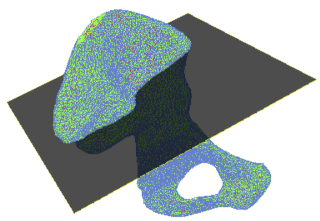
s: number of sample voxels

m: number of density function coefficient,

$s>2m$

Accuracy vs Degree of Density Function

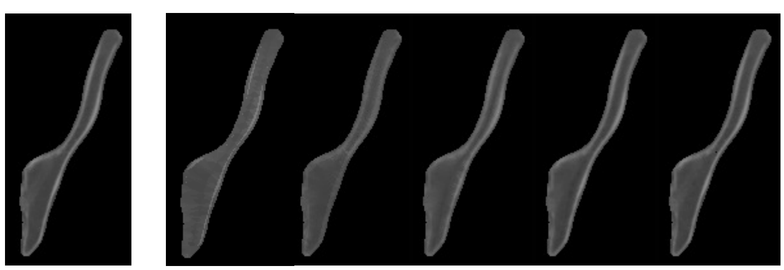
- Use CT data set as ground truth
- Cut an arbitrary plane through the model



Arbitrary Cutting Plane

Partitions by tetrahedra on cutting plane

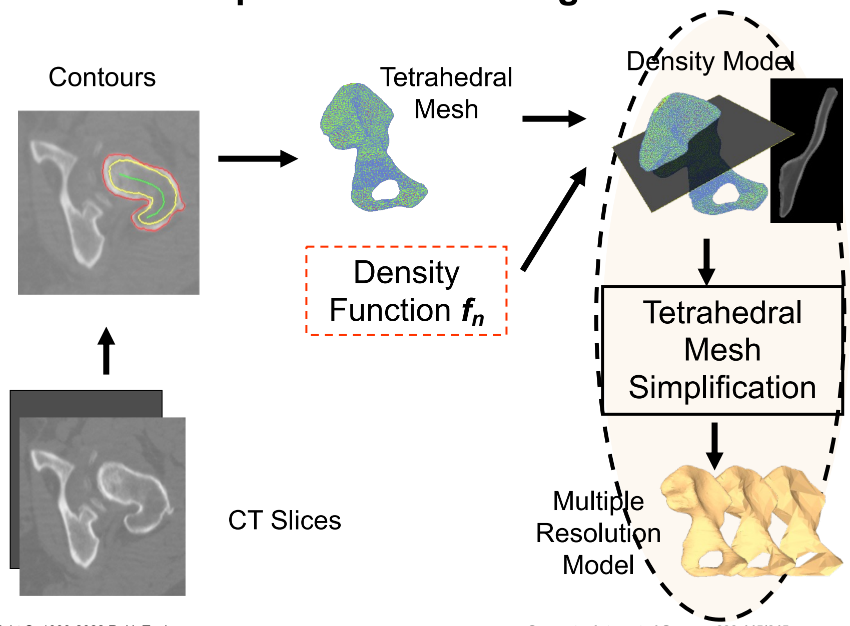
Accuracy vs Degree of Density Function (cont')



Ground Truth n=0 n=1 n=2 n=3 n=4

Degree	0	1	2	3	4	5	6	7	8
Coeff Number	1	4	10	20	35	56	84	120	165
Avg. Density Err (%)	3.291	1.583	0.766	0.442	0.298	0.216	0.167	0.149	0.128

Example: Bone Modeling from CT



141

Model Simplification

- Models used in CIS must be guaranteed to be accurate within known bounds
- But 3D models from medical images often are very complex, and require representations with large data structures.
- Algorithms using models are often computationally expensive, and computation costs go up with model complexity
- **PROBLEM:** reduce model complexity while preserving adequate accuracy

~350,000 triangles!

142

Model simplification

- Problem is also common in computer graphics
  - There is a growing literature
  - **But** many graphics techniques only care about appearance, and do not necessarily preserve accuracy or other properties (like topological connectivity) important for computational analysis

- Broadly, two classes of approaches
  - Top down
  - Bottom-up

Top down

- Active surfaces used in segmentation

- Deformable registration of an atlas to a patient
  - E.g., skull atlas discussed in craniofacial lecture had about 5000 polygons (perhaps 15-20,000 triangles)

- Recursive approximations
  - E.g., Pizer *et al.* "cores"

Bottom up methods

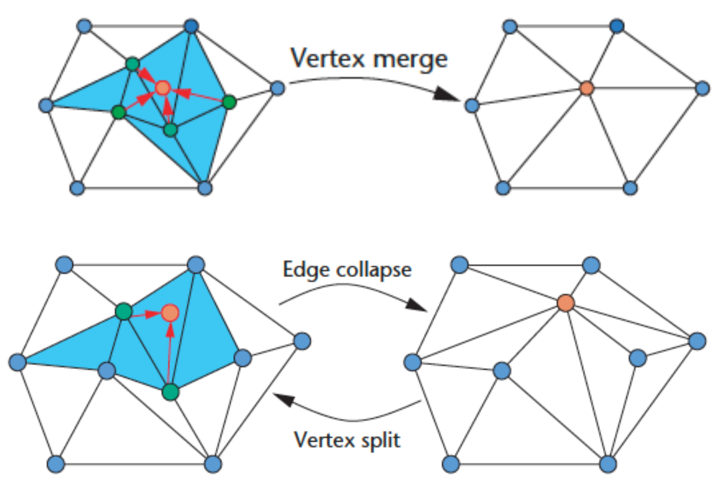
- Typically, start with very high detail model generated from CT images
  - Large number of elements a consequence of small size of pixels & way model is created

- Then merge nearby elements into larger elements
  - E.g., “decimation” (Lorensen, et. al.)
  - E.g., “superfaces” (Kalvin & Taylor)
  - E.g., Gueziec

- An excellent tutorial may be found in:
  - David Luebke; A Developer’s Survey of Polygonal Simplification Algorithms; IEEE Computer Graphics and Application, May 2001

145

Bottom-up merging



Source: David Luebke; A Developer’s Survey of Polygonal Simplification Algorithms; IEEE Computer Graphics and Application, May 2001

146