

Visualization: Volume rendering

000

Visualization

Visualization is any technique for creating images, diagrams or animations to **communicate a message**.

Wikipedia

Scientific data visualization

- Abstract
- Physics (fluids, ...)
- Medical (X-Rays, MRI, Images, ...)
- Technics (Mechanics, ...)
- ...

001

Visualization : Problematic

- Complex Data: non visualizable (density, tensors, ...)
- Large amount of data : 10, 100 To (landscape, connections, scanners, ...)
- Noisy data (medical, ...)

Goal: Being able to visualization what is **significant**, **usefully**, and **efficiently**.

002

Data types

- Scalar field (temp, pression, ...)
- Vectorial field (vitesse, orientation, ...)
- Tensorial field (mechanical constraints, curvature, ...)

Goal: Do we define the data on a surface, on a volume ?

003

Scalar field

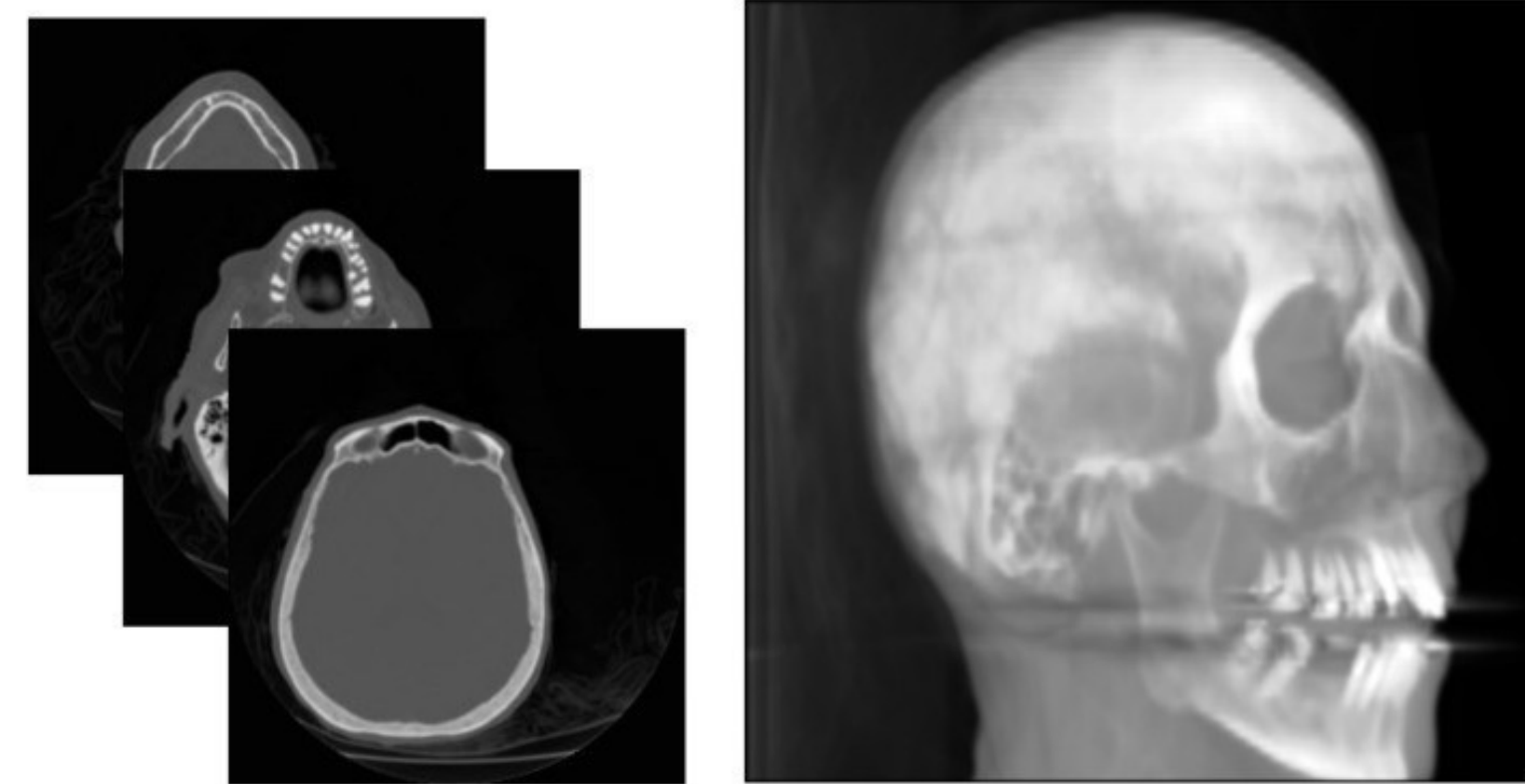
Surface of the domain or internal characteristics



004

Scalar field

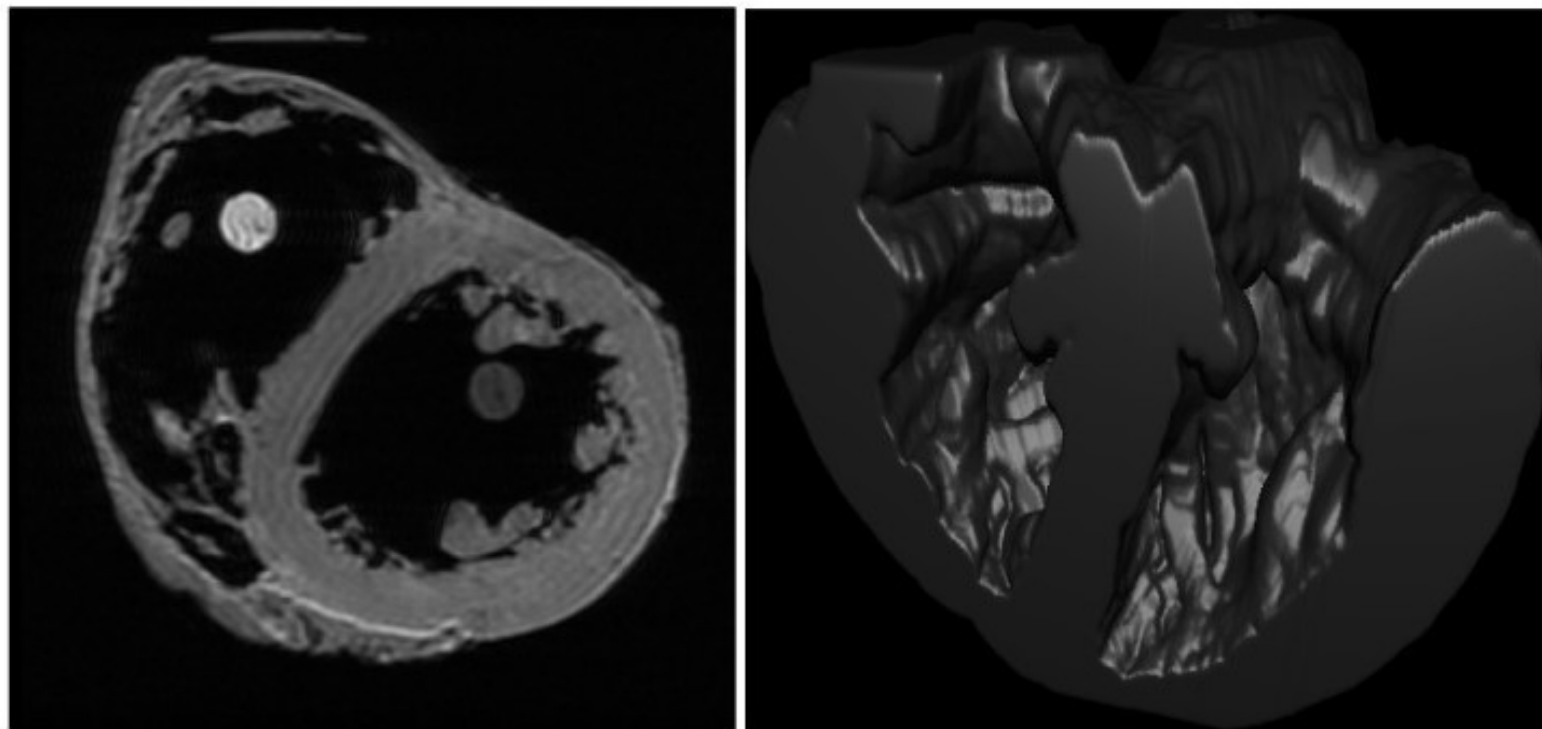
2D section or volume rendering (isosurface, 3D textures, ...)



005

Scalar field

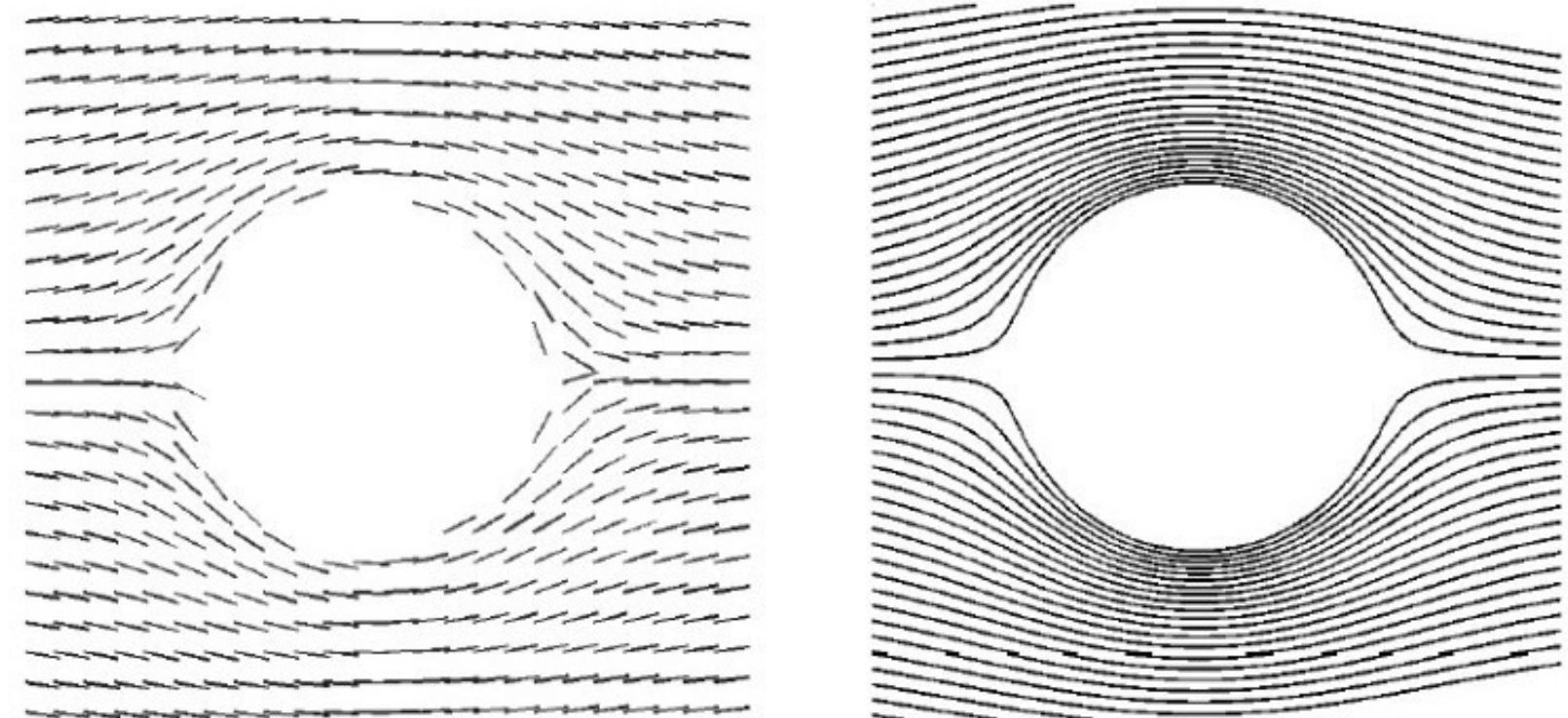
2D Section or 3D Isosurface



006

Vectorial field

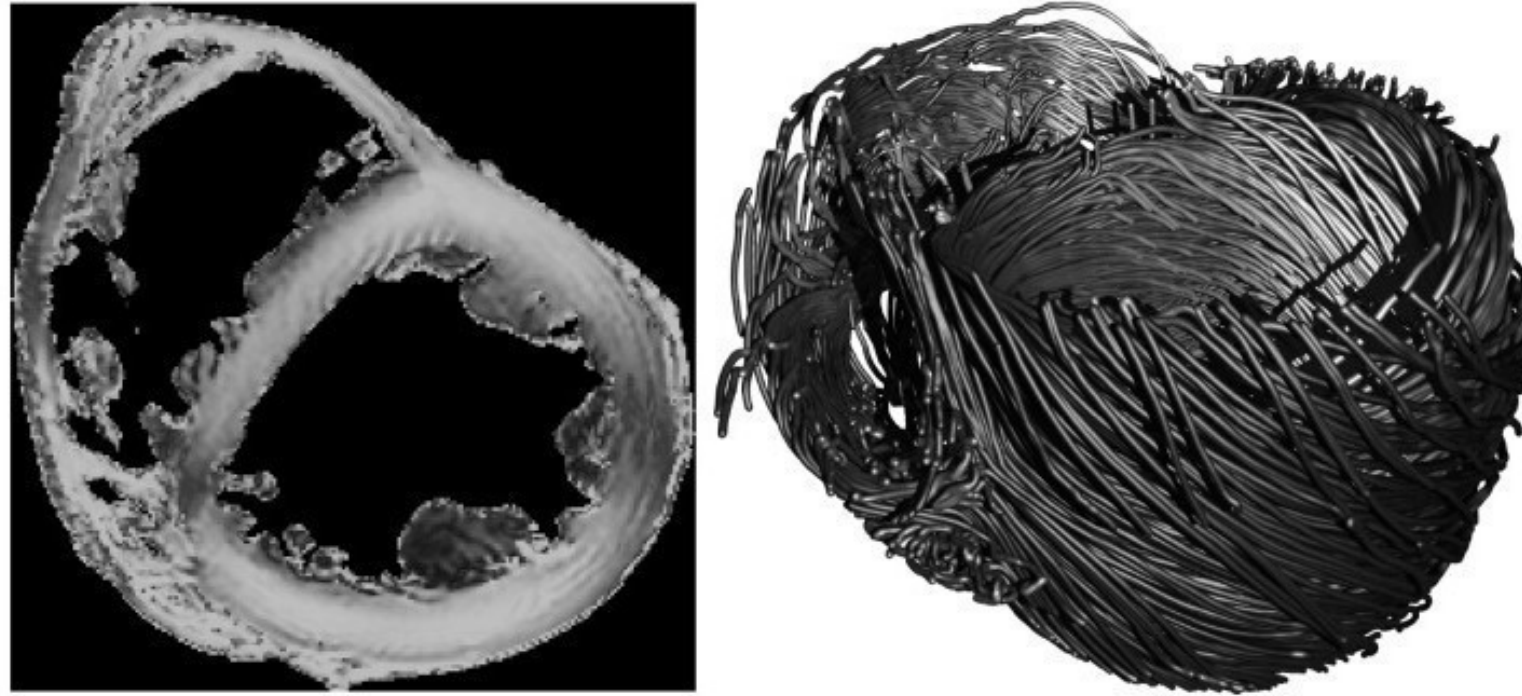
Vectors or trajectories



007

Vectorial field

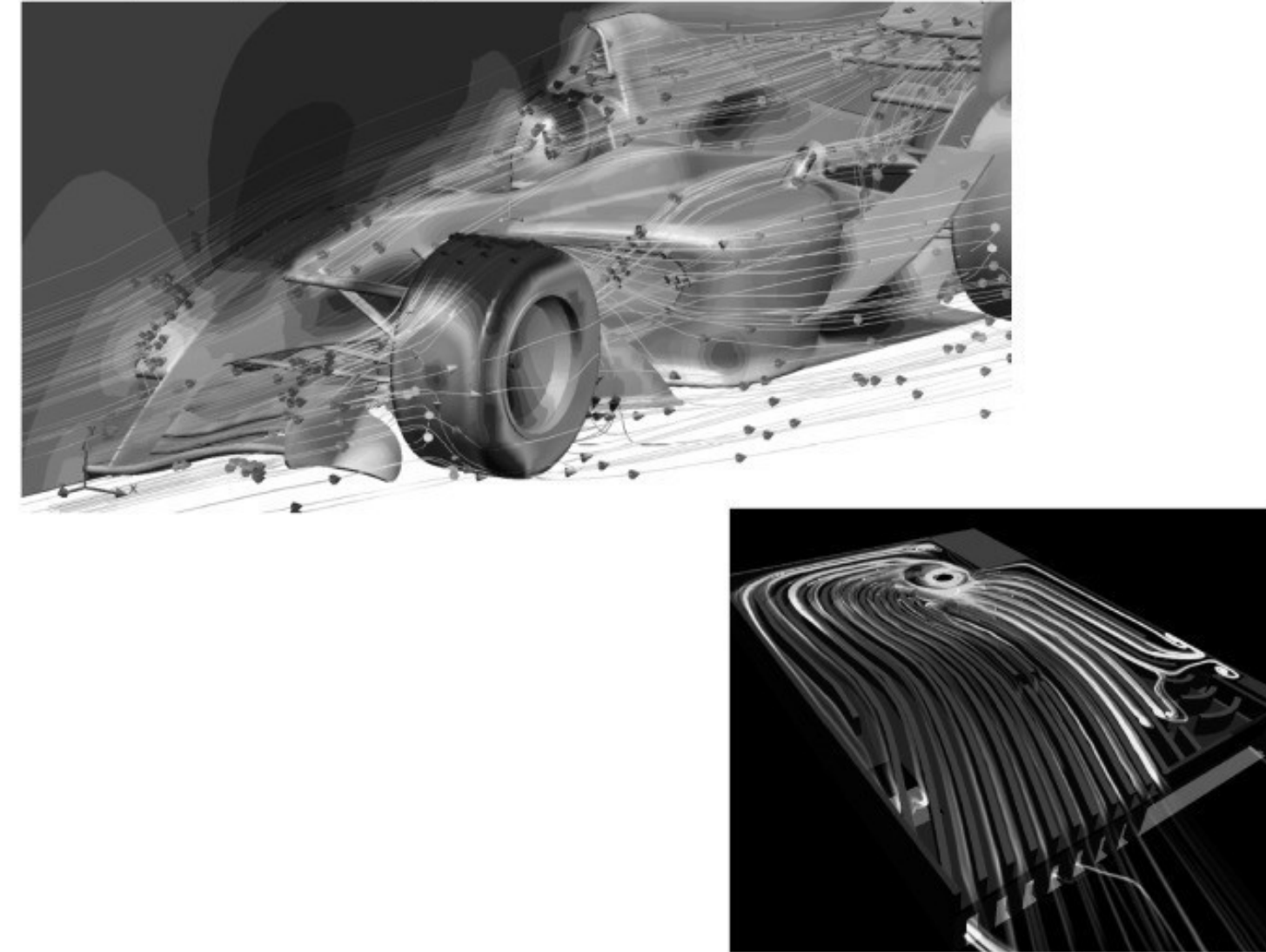
Vectors or trajectories (stream lines can represent real data)



008

Vectorial field

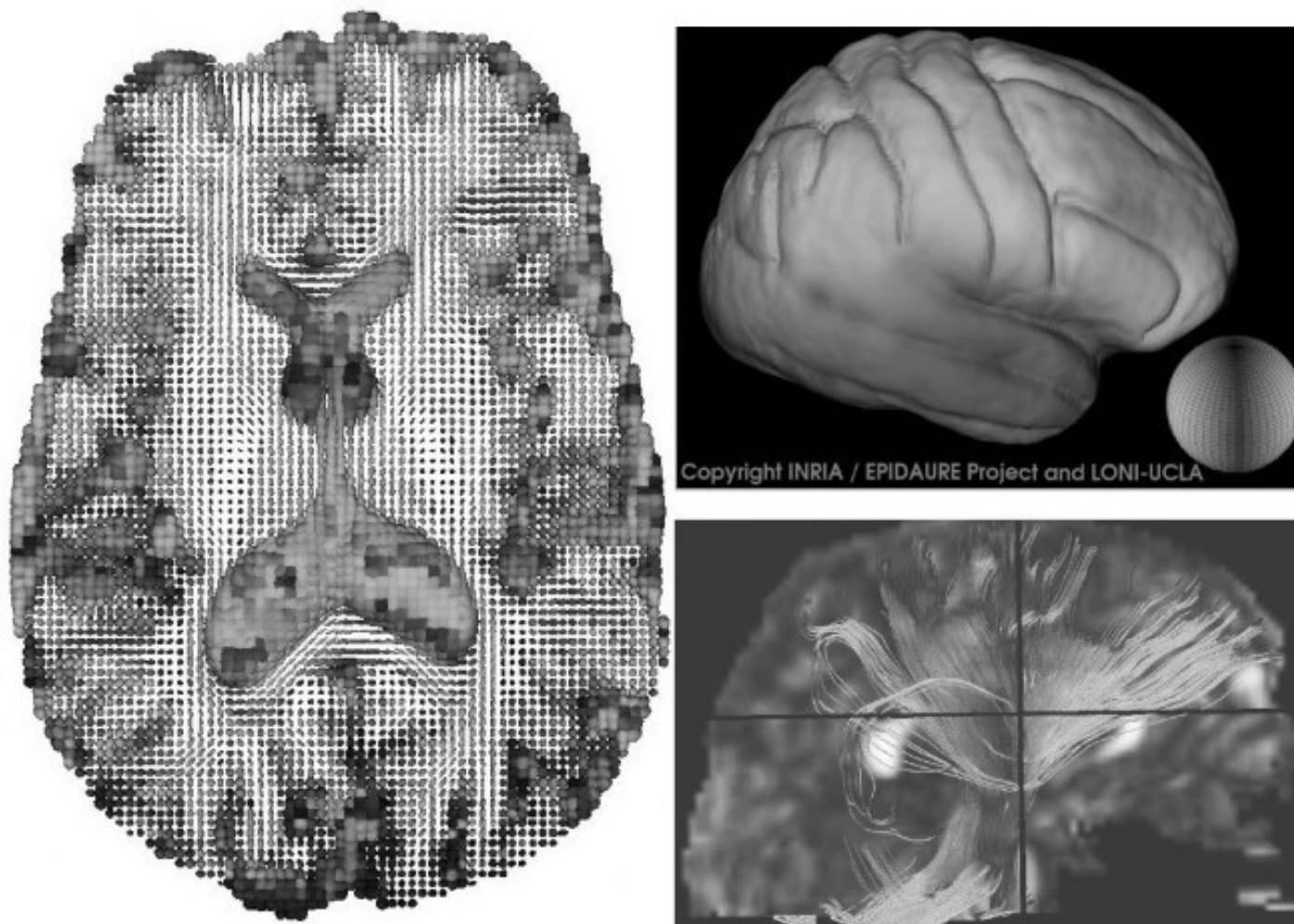
Complex physically based simulation (streamlines, ...)



009

Tensorial field

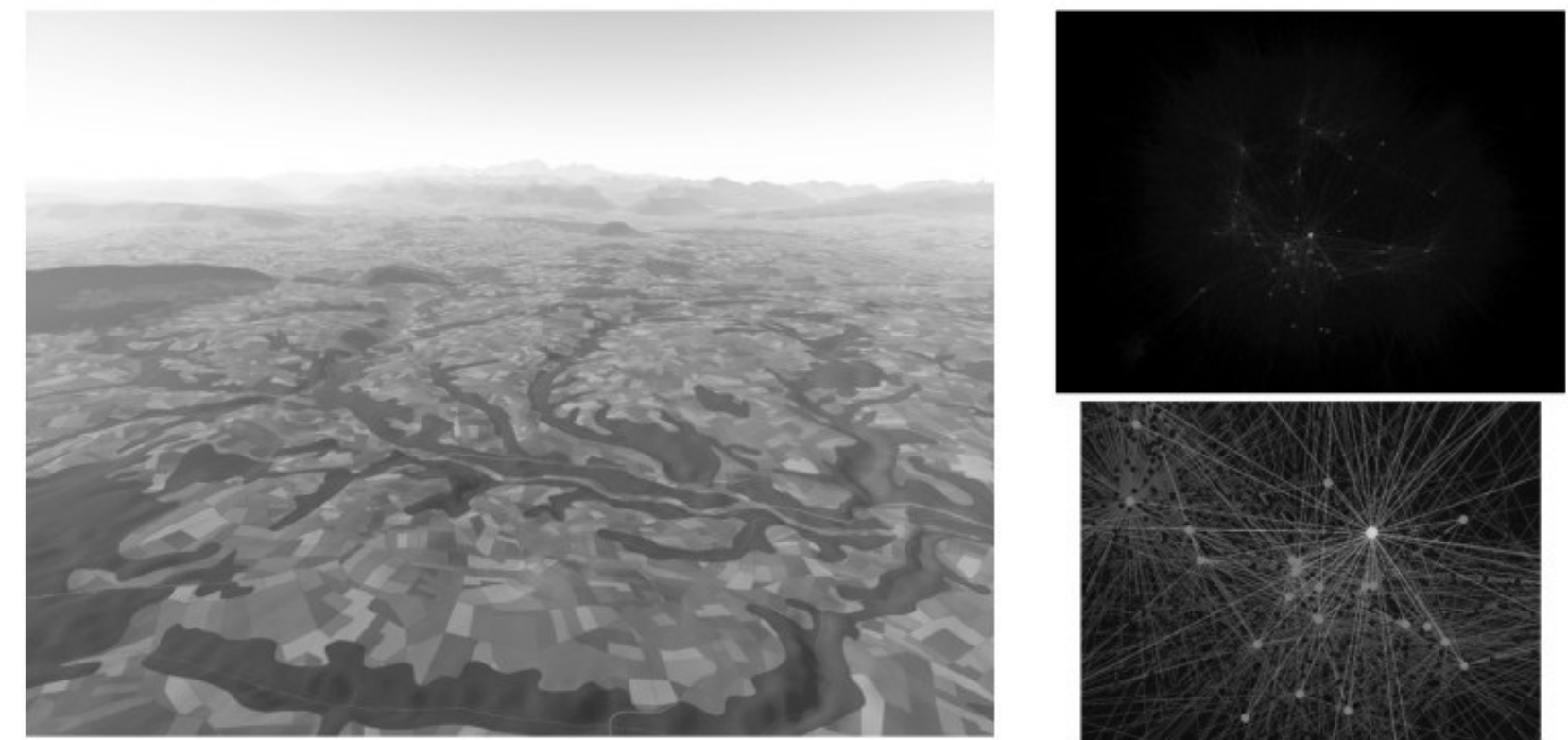
Symetric matrix 3x3 (Ellipsoid, glyphs, orientation, fiber-tracking, ...)



010

Large data

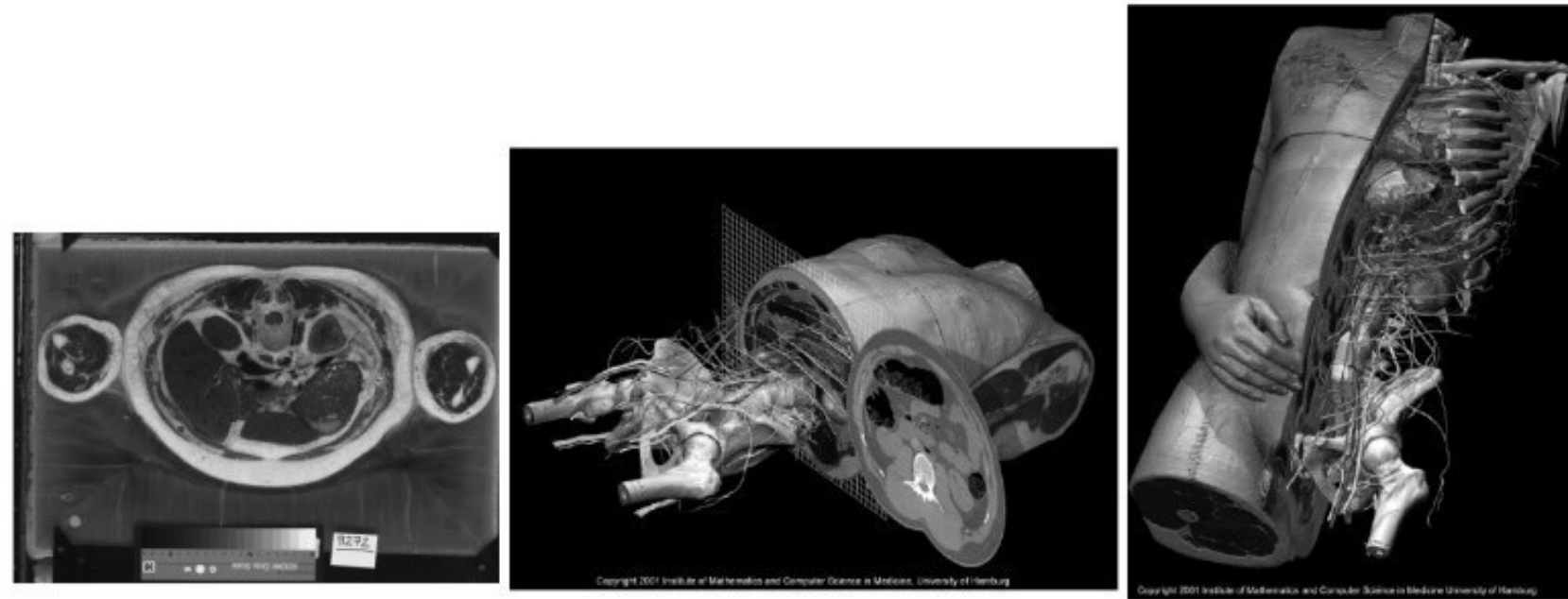
Data acquired from physical scanners are often too large (geography, networks, ...)



011

Large data

Visible Human Project, 40Go (slices of 0.33mm)



012

Classification

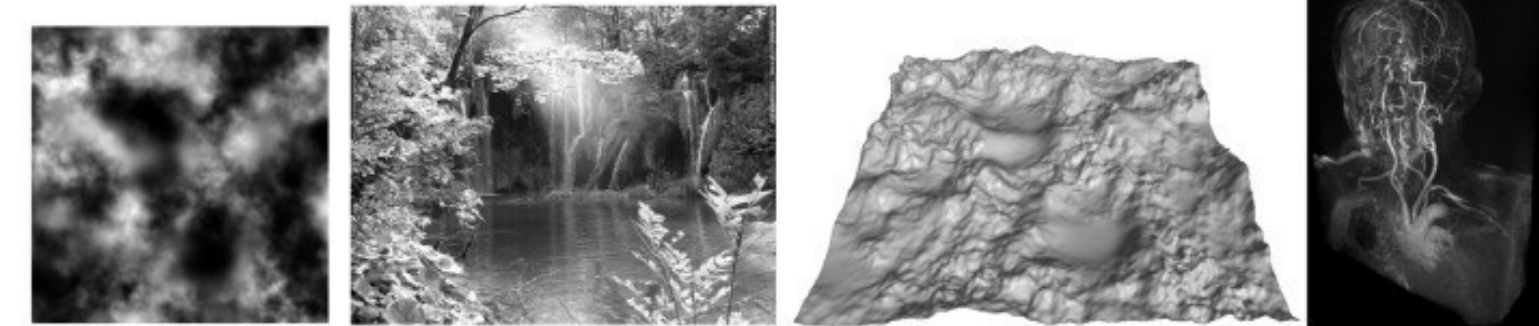
Visualize $f : \begin{cases} \mathbb{R}^v \rightarrow \mathbb{R}^d \text{ embedded in } \mathbb{R}^n \\ u \mapsto f(u) \end{cases}$

$d=1$ scalar field
 $d>1$ vectorial field
 $d=(i \times j)$ matrix field

$v=1$ lineic field
 $v=2$ surfacic field
 $v=3$ volume field

Common special cases

v	d	n	
2	1	2	B&W image
2	3	2	Color image (texture)
2	1	3	Heigh-field (mountains)
3	1	3	Volume density



013

Surfacic scalar data

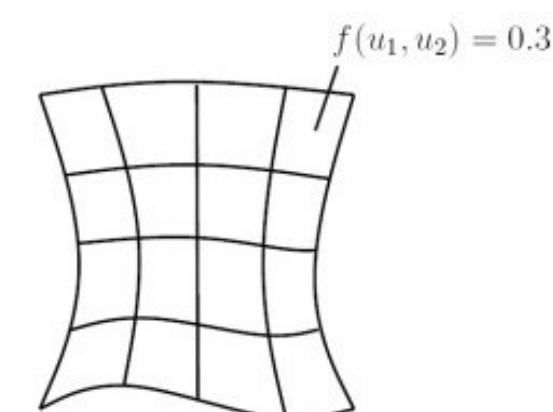
014

Surfacic scalar data : Notation

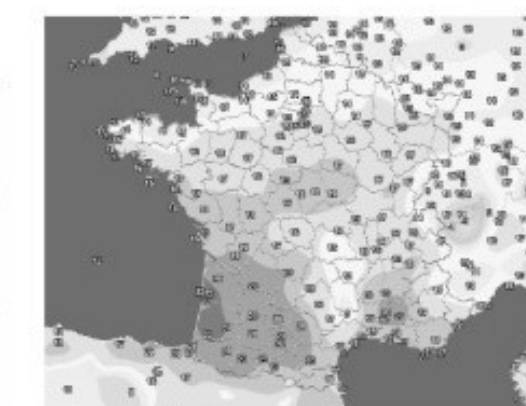
We call a density $f(u_1, u_2) = I \in \mathbb{R}$

Very often: $f(x, y) = I$

After discretization: $f(k_x \Delta x, k_y \Delta y) = I_{k_x, k_y}$

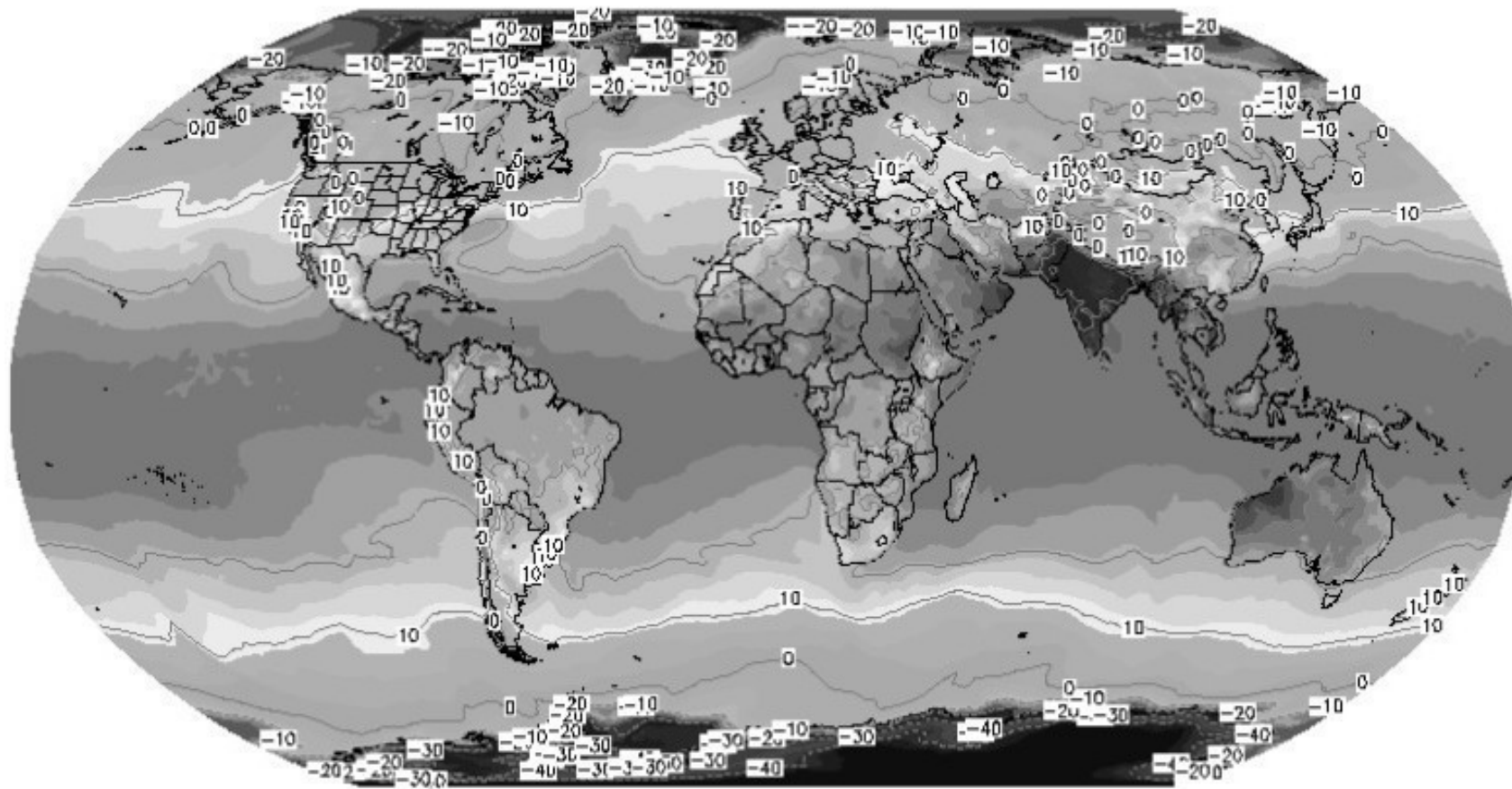


0.5	-0.2	1.1
1.5	0.5	0.9
-0.1	0.0	0.7



015

Surfacic scalar data : Example



016

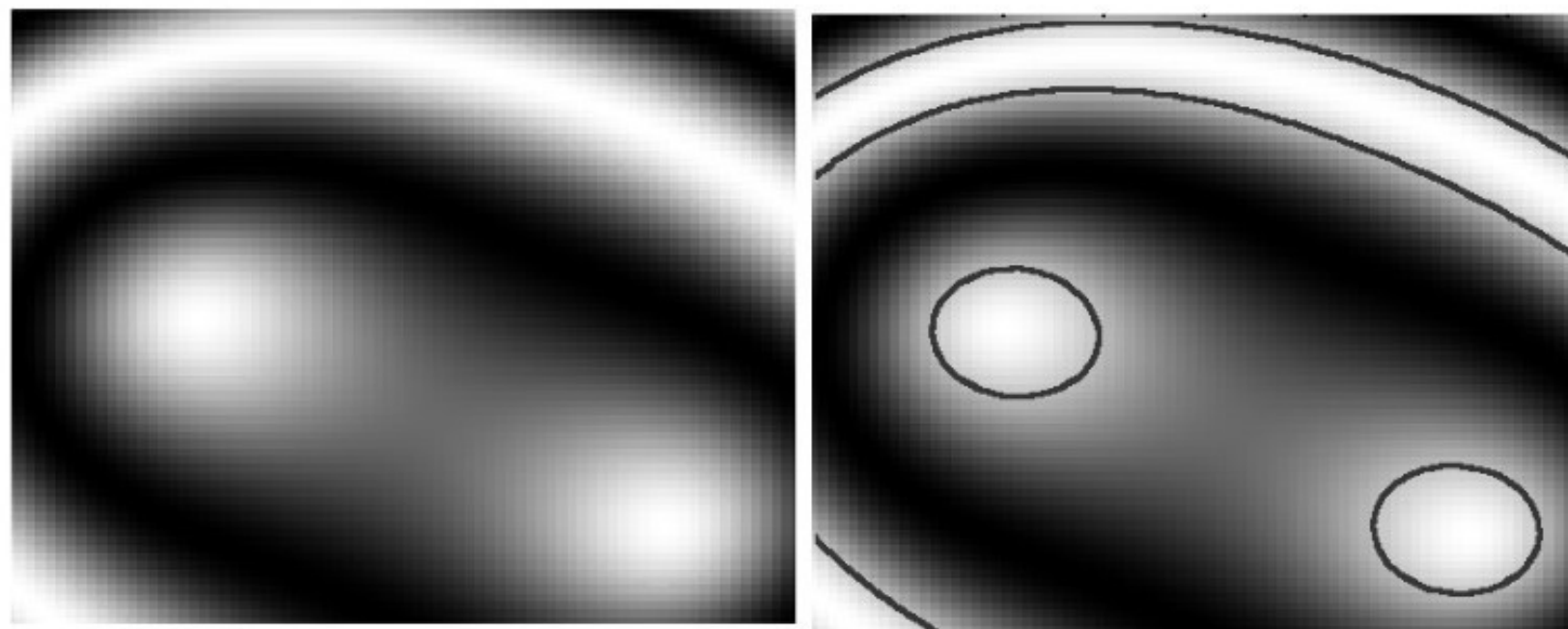
Surfacic scalar data : Example



017

Isolines

Goal: Trace curves on a specific value
Called: isolines, isocurves, level sets, ...

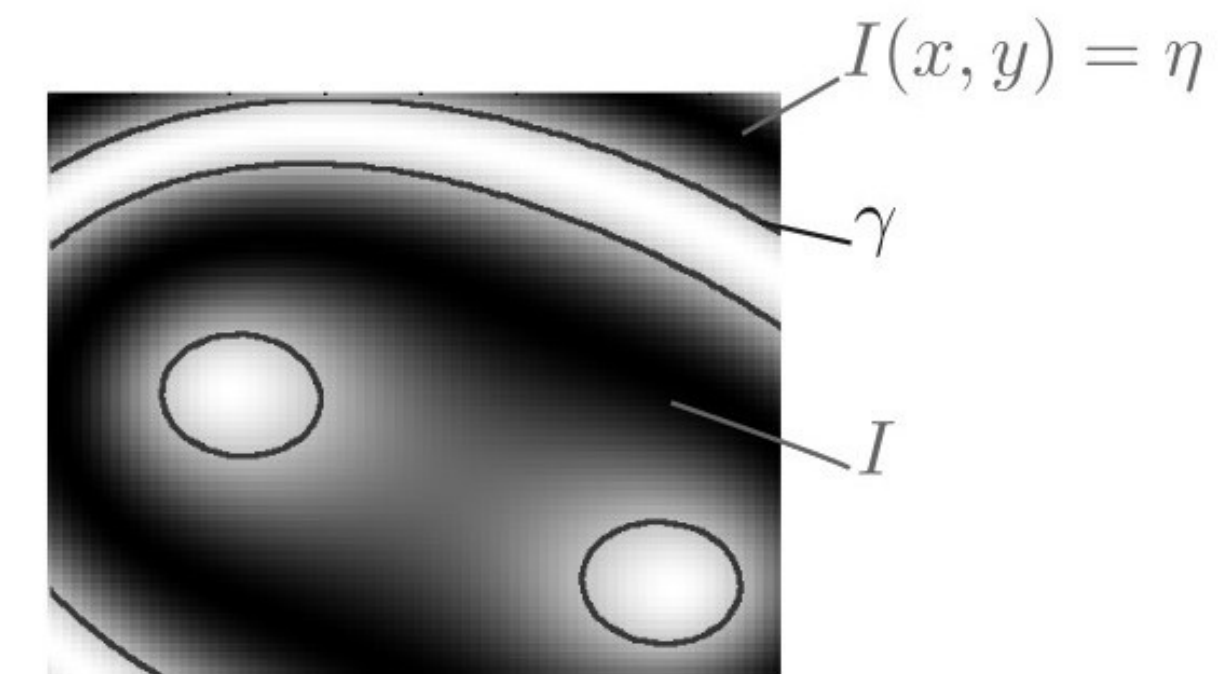


018

Isolines : input/output

Input: Scalar values on a regular discrete grid + isovalue η

Output: Set of curves $\{\gamma = (x, y) \in \mathbb{R}^2 | I(x, y) = \eta\}$
(degenerated cases: points, regions)



019

Example: continuous cases

For $\eta = 0$

$$F_1 = 1$$

$$F_2 = 1$$

$$F_3 = (x - x_0)^2 + (y - y_0)^2 - r_0^2$$

$$F_4 = F_3(x_0, y_0, r_0) + F_3(x_1, y_1, r_1)$$

$$F_5 = F_3(x_0, y_0, r_0) \times F_3(x_1, y_1, r_1)$$

We can define a curve by its implicit equation
+ Arbitrary topology

020

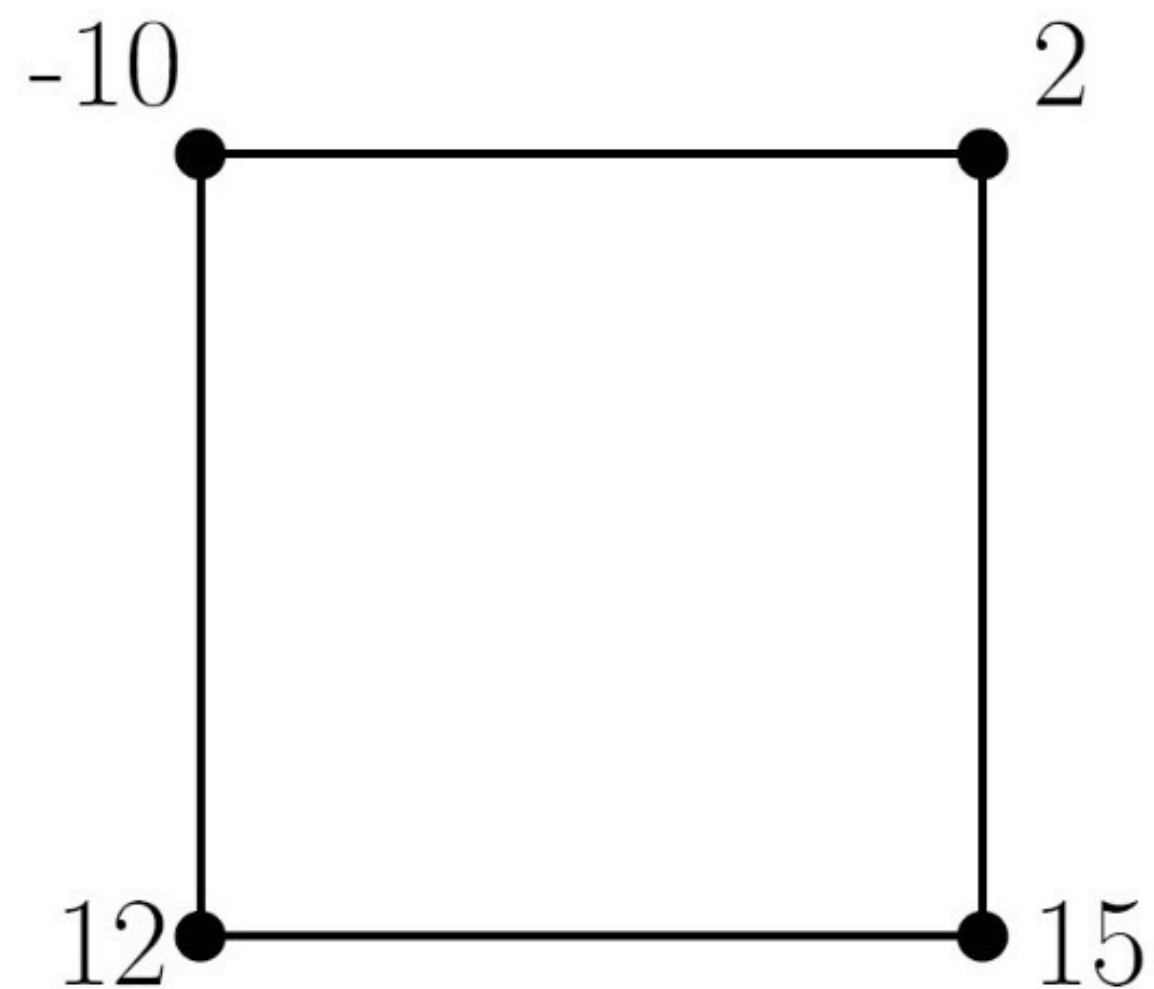
Marching squares

-61	-45	-42	-52	-72	-91	-99	-89
-17	8	13	-2	-34	-69	-94	-98
25	57	64	43	2	-45	-84	-99
51	87	94	71	25	-30	-76	-99
51	87	94	71	25	-30	-76	-99
25	57	64	43	2	-45	-84	-99
-17	7	13	-2	-34	-69	-94	-98
-61	-45	-42	-52	-72	-91	-99	-89

021

Marching squares

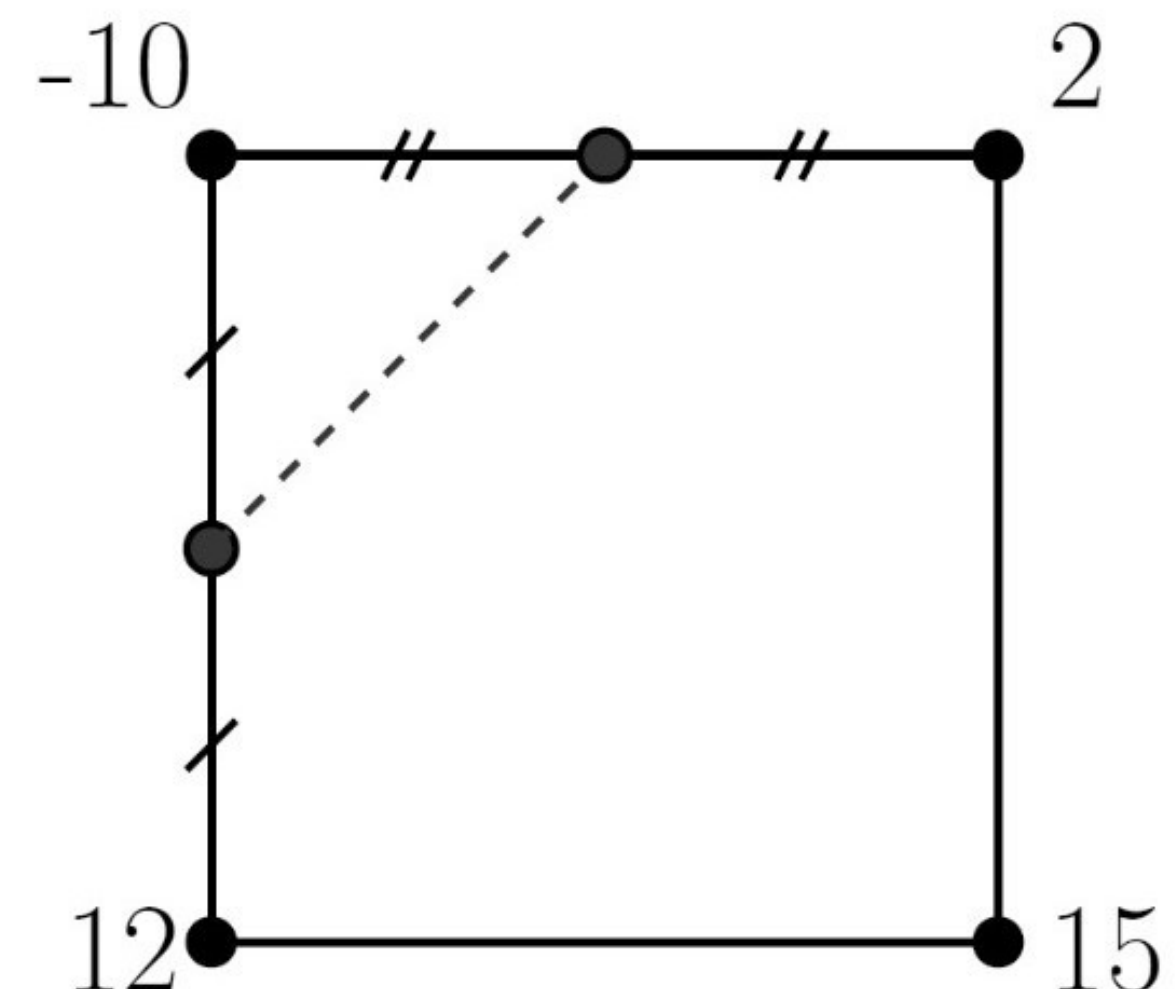
Which curve should we consider?



022

Marching squares

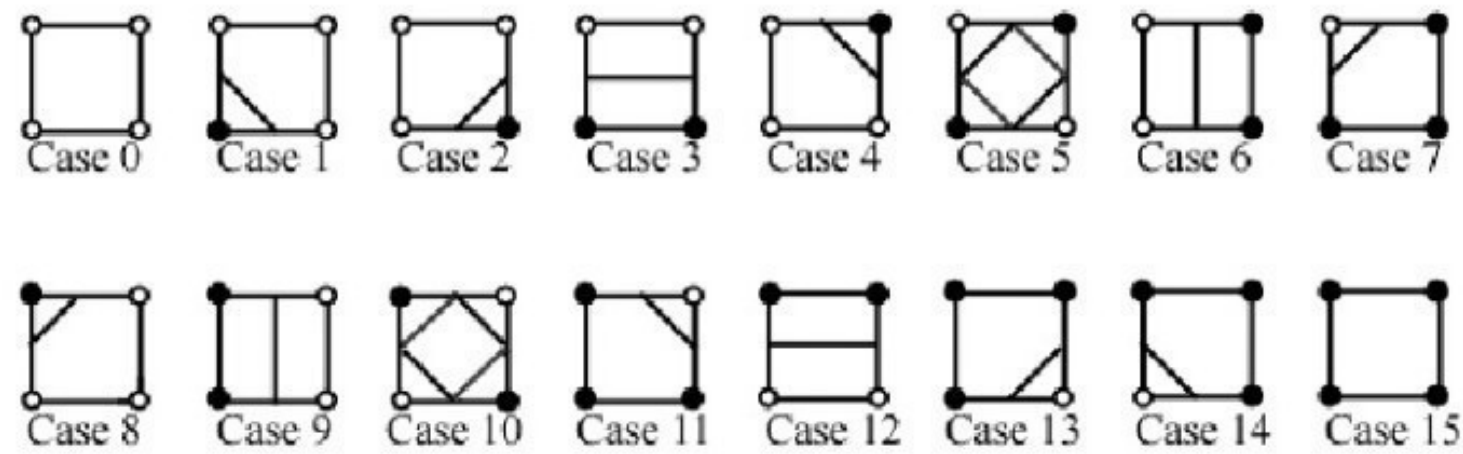
Middle of the edges



023

Marching square

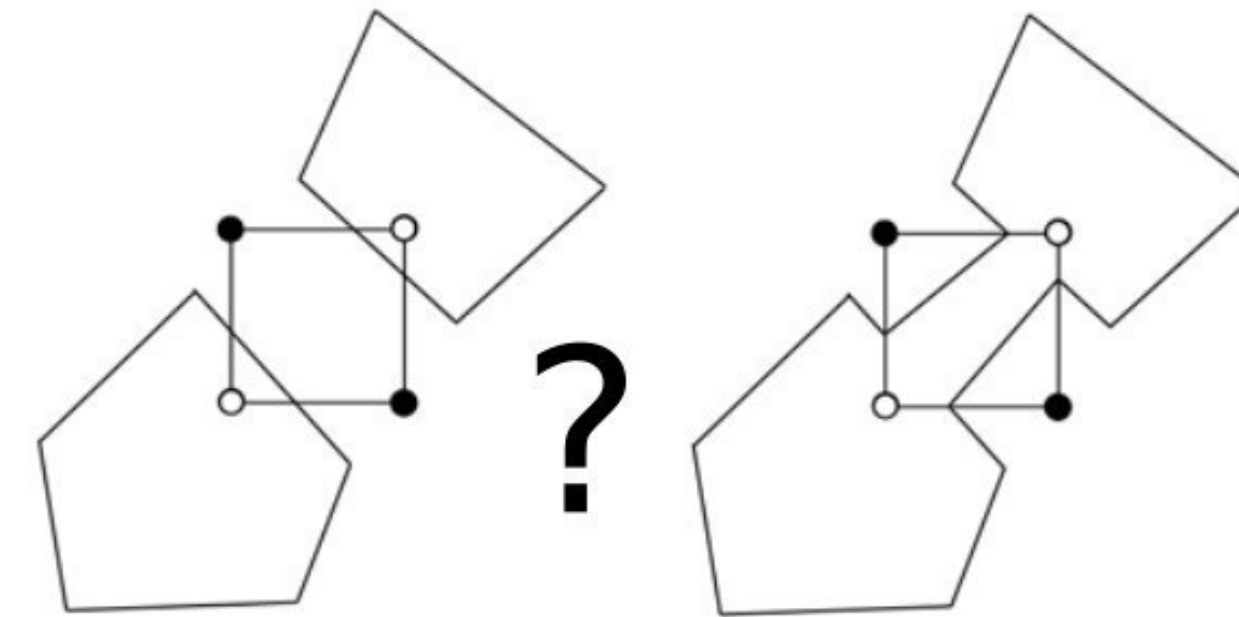
For a single cell: 16 different possibilities



028

Marching square

Some undetermined case



029

Volume data

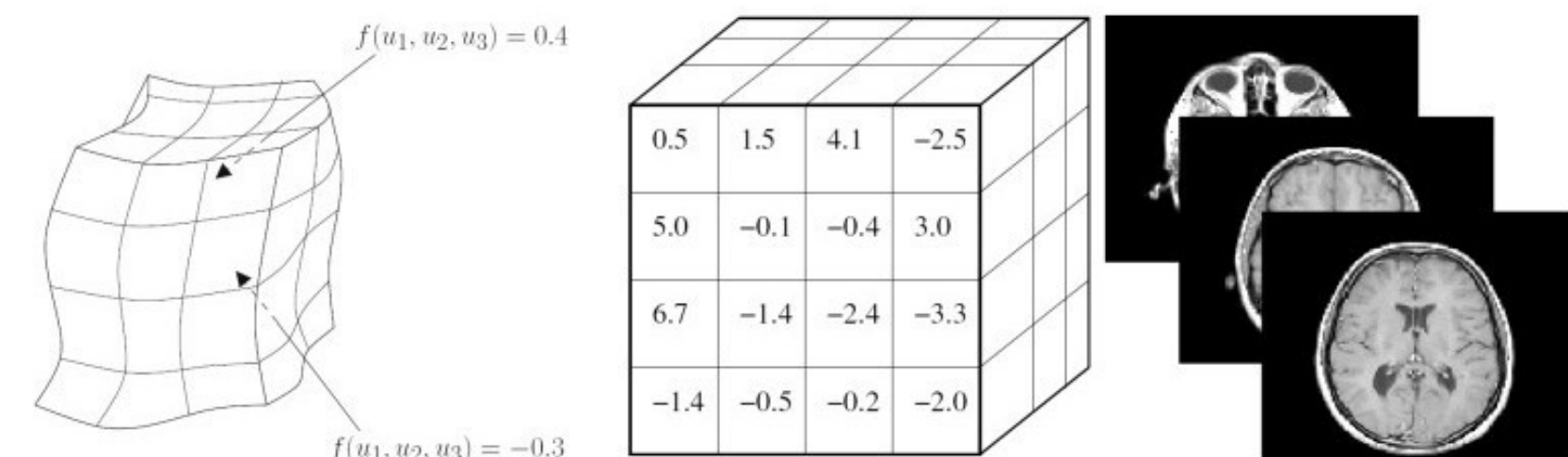
030

Volume data : Notation

For volume field: $f(u_1, u_2, u_3) = I \in \mathbb{R}$

Very often: $f(x, y, z) = I$

After discretization: $f(k_x \Delta x, k_y \Delta y, k_z \Delta z) = I_{k_x, k_y, k_z}$

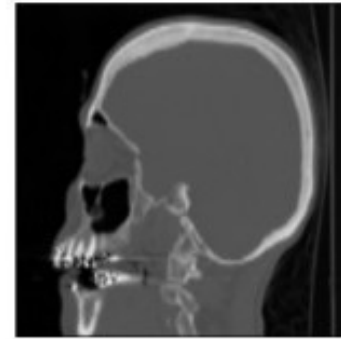


031

Medical Imaging Modalities

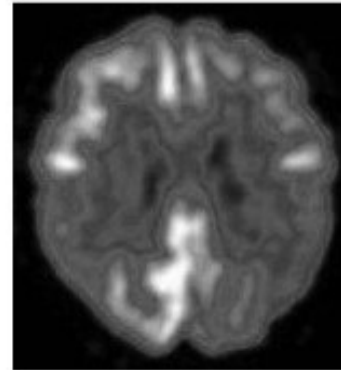
X-Ray

Anatomical
Absorption measurement
(inverse problem)



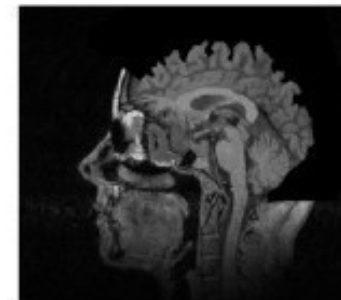
Nuclear (PET, SPECT)

Functional
Attenuated emission
(inverse problem, noise)



MRI

Anatomical (MRI, Angiography)
or Functional
Density measurement (direct)



032

Slicing visualization

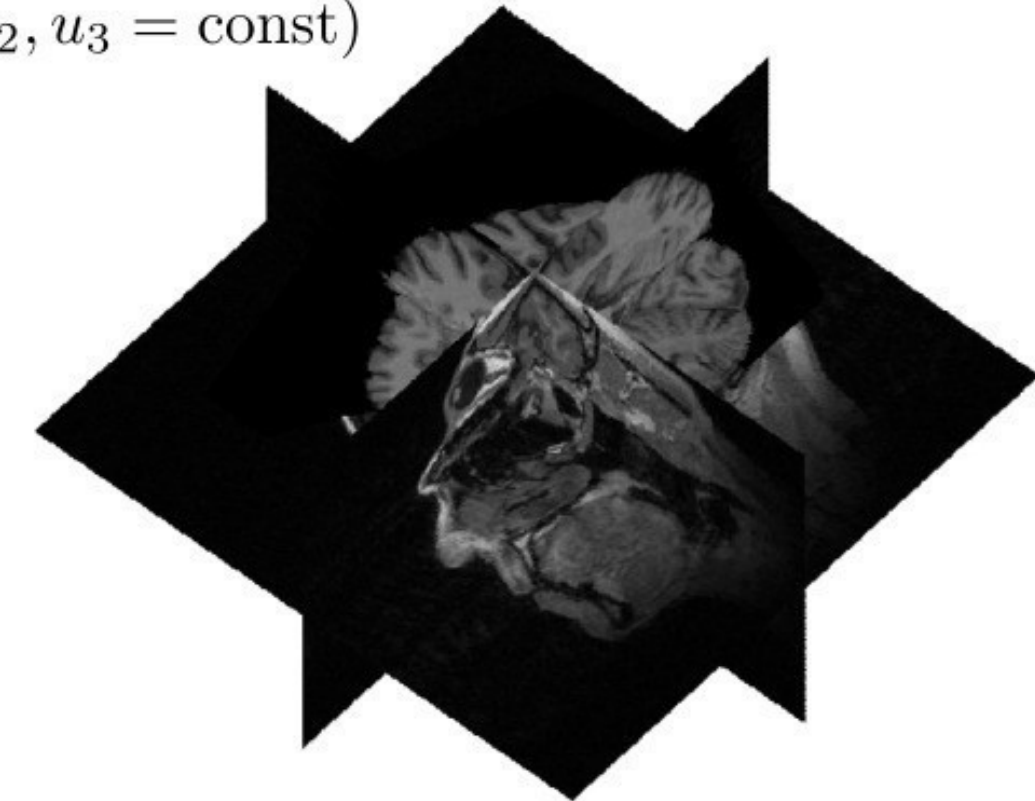
Idea: Slice some surfaces on the volume

Encode the field as a color (gray level, texture, etc)

Draw $I(u_1 = \text{const}, u_2, u_3)$

$I(u_1, u_2 = \text{const}, u_3)$

$I(u_1, u_2, u_3 = \text{const})$



033

Slicing visualization

We can use more general surfaces
Which is the best surface ?



034

Marching cubes

A common surface is the iso-surface

The isosurface of isovalue η of the I function is the set

$$\{(x, y, z) \in \mathbb{R}^3 | I(x, y, z) = \eta\}$$

In making η evolving, we obtain different surfaces

How to triangulate such implicit surface ?

035

Marching cubes : Examples

For $\eta = 0$

$$F_1 = 1$$

$$F_2 = 0$$

$$F_3 = (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - r_0^2$$

$$F_4 = F_3(x_0, y_0, z_0, r_0) + F_3(x_1, y_1, z_1, r_1)$$

$$F_5 = F_3(x_0, y_0, z_0, r_0) \times F_3(x_1, y_1, z_1, r_1)$$

A surface can be defined by its equation
+ Arbitrary topology

036

Marching cubes : Intro

Goal: Build a triangulated surface from a discrete volumetric scalar field given by $I(x, y, z) - \eta$

First software patent in CG in 1985 from Lorensen & Cline.

Input data: 3D Grid in (x,y,z) of (Ni,Nj,Nk) voxels.

0.5	1.5	4.1	-2.5
5.0	-0.1	-0.4	3.0
6.7	-1.4	-2.4	-3.3
-1.4	-0.5	-0.2	-2.0

037

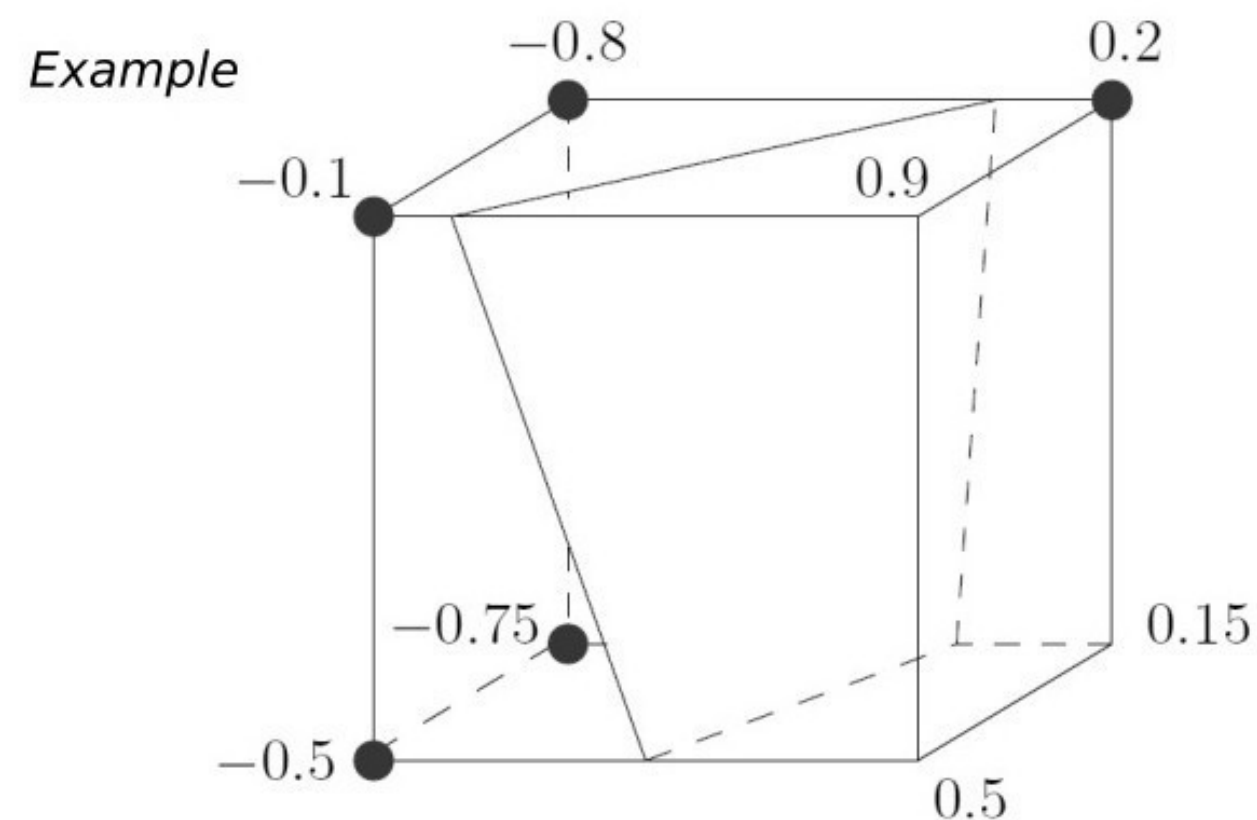
Marching cubes : Principle

Traversal of the grid "cube by cube"

Compute the sign of $I(x, y, z) - \eta$

Check the possible cases

The 0 value is obtain by interpolation

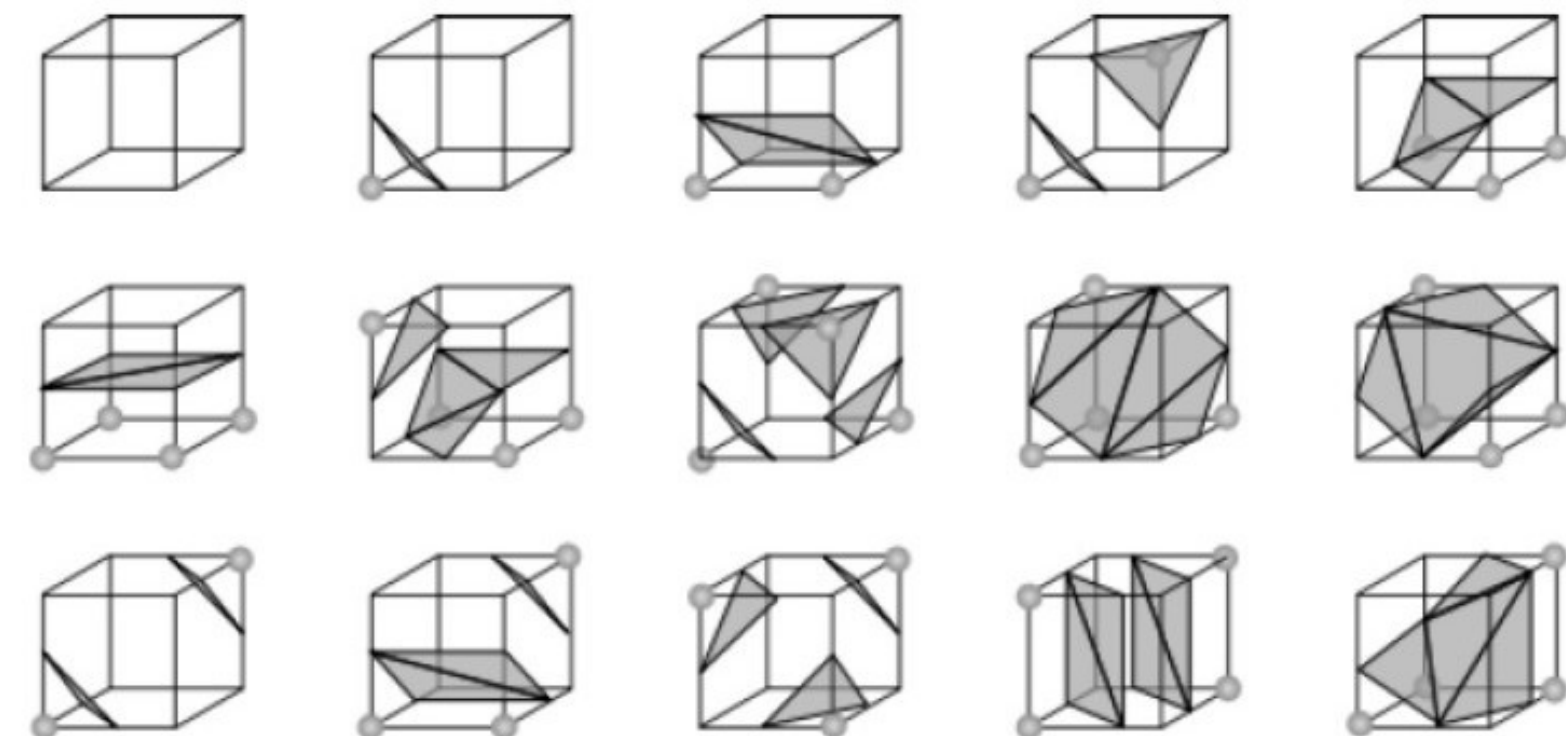


038

Marching cubes : Different cases

A total of 256 possible cases

Only 15 basic cases



039

Marching cubes : Usage

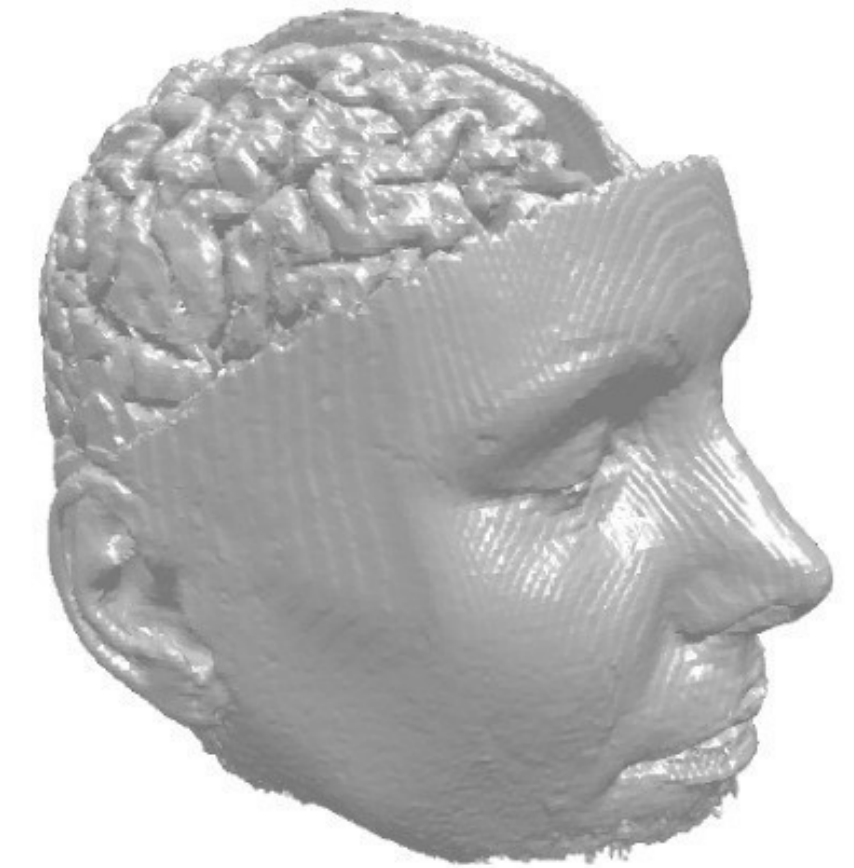
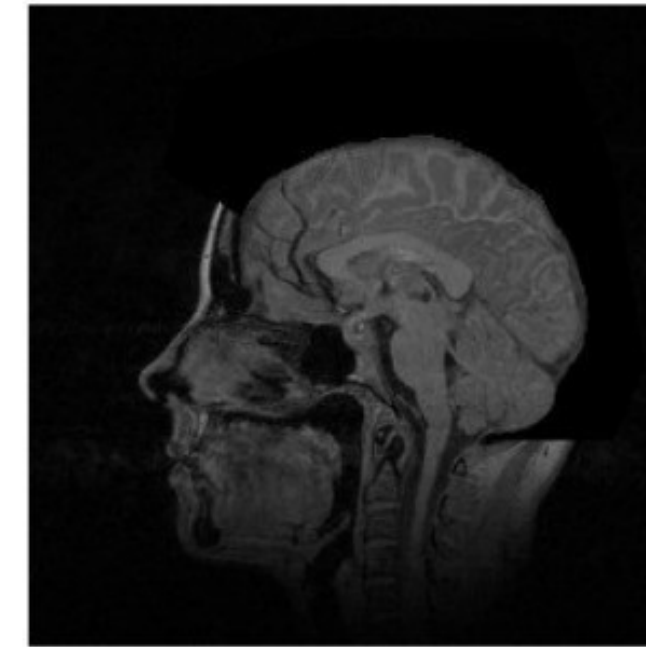
+ Efficient

- Cubic aspect
 - Smooth volume
 - Smooth surface
 - Medical correctness
- Undetermined cases

040

Isosurface example: MRI

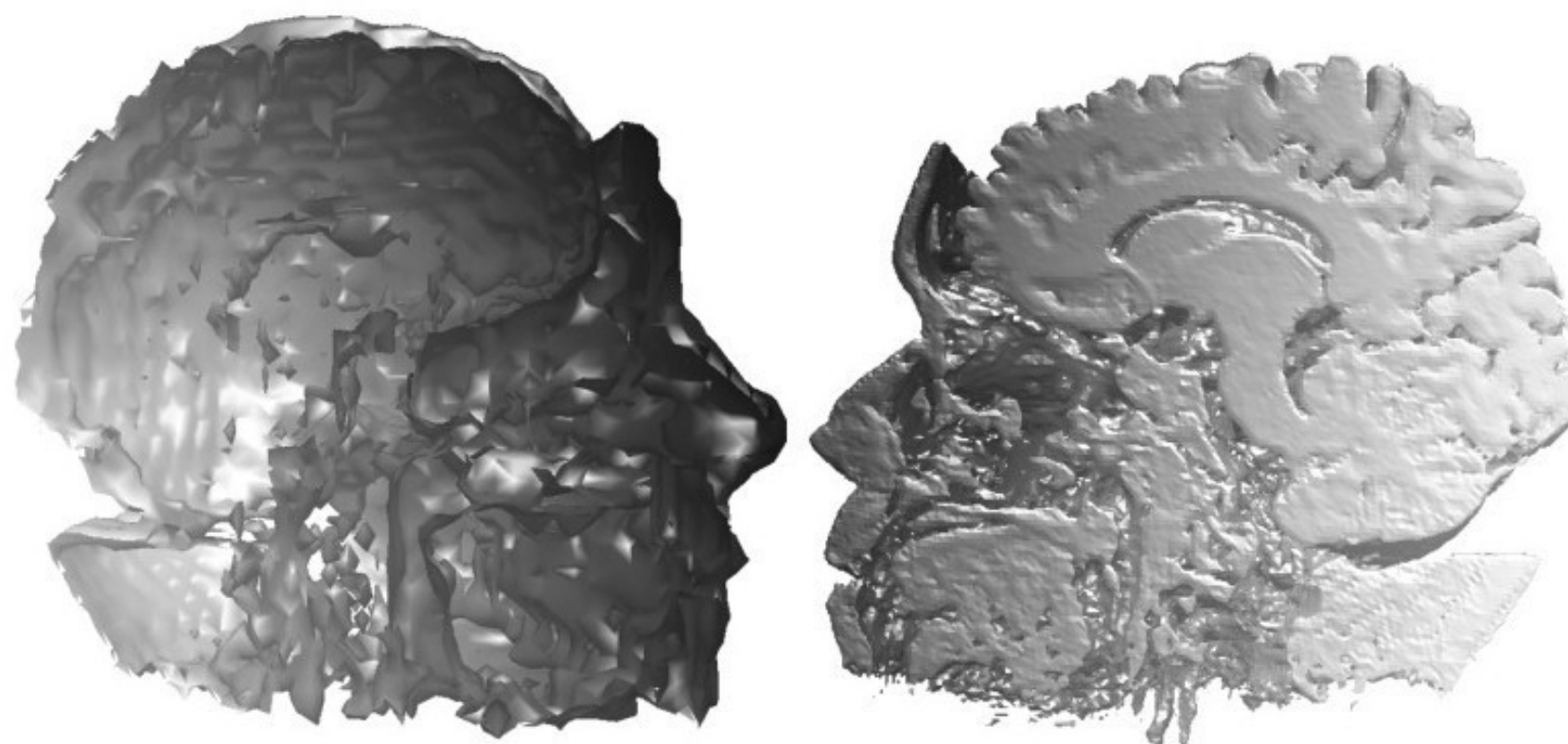
MRI Data (256 x 256 x 99)



041

Isosurface example: MRI

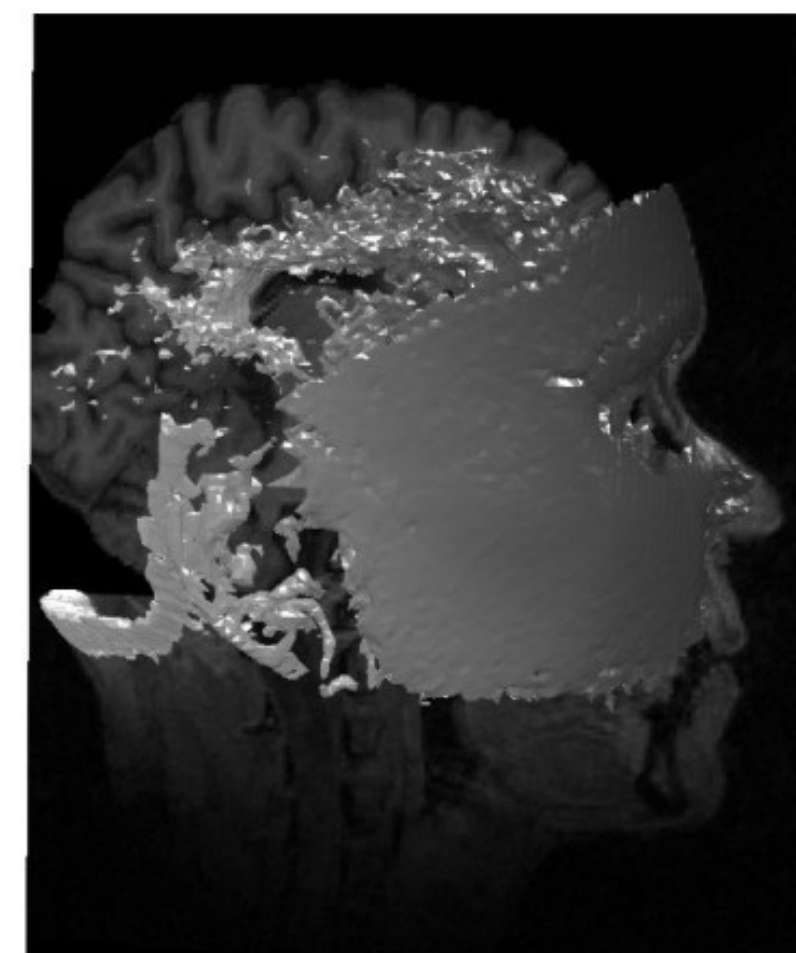
Can observe internal structure



042

Isosurface example: MRI

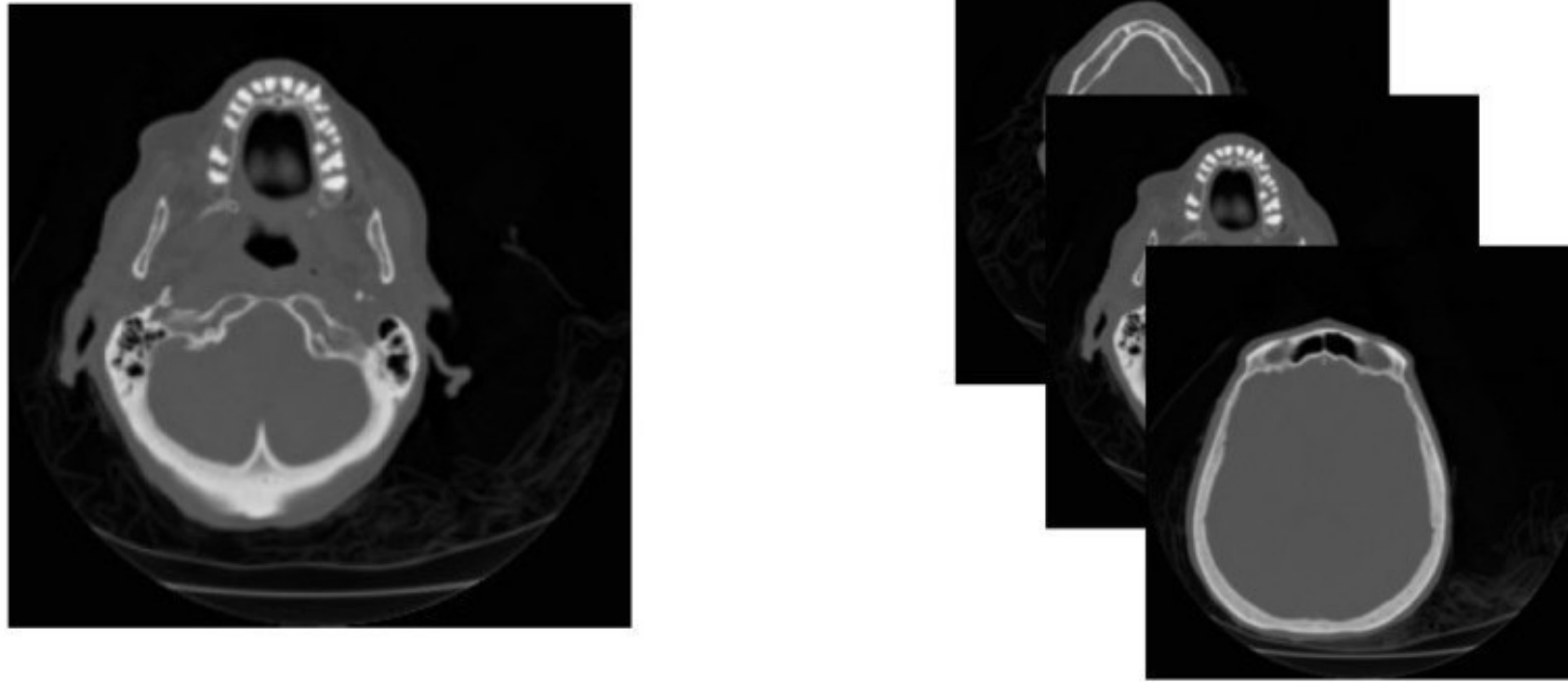
Combining Slicing + isosurface



043

Isosurface example: CT

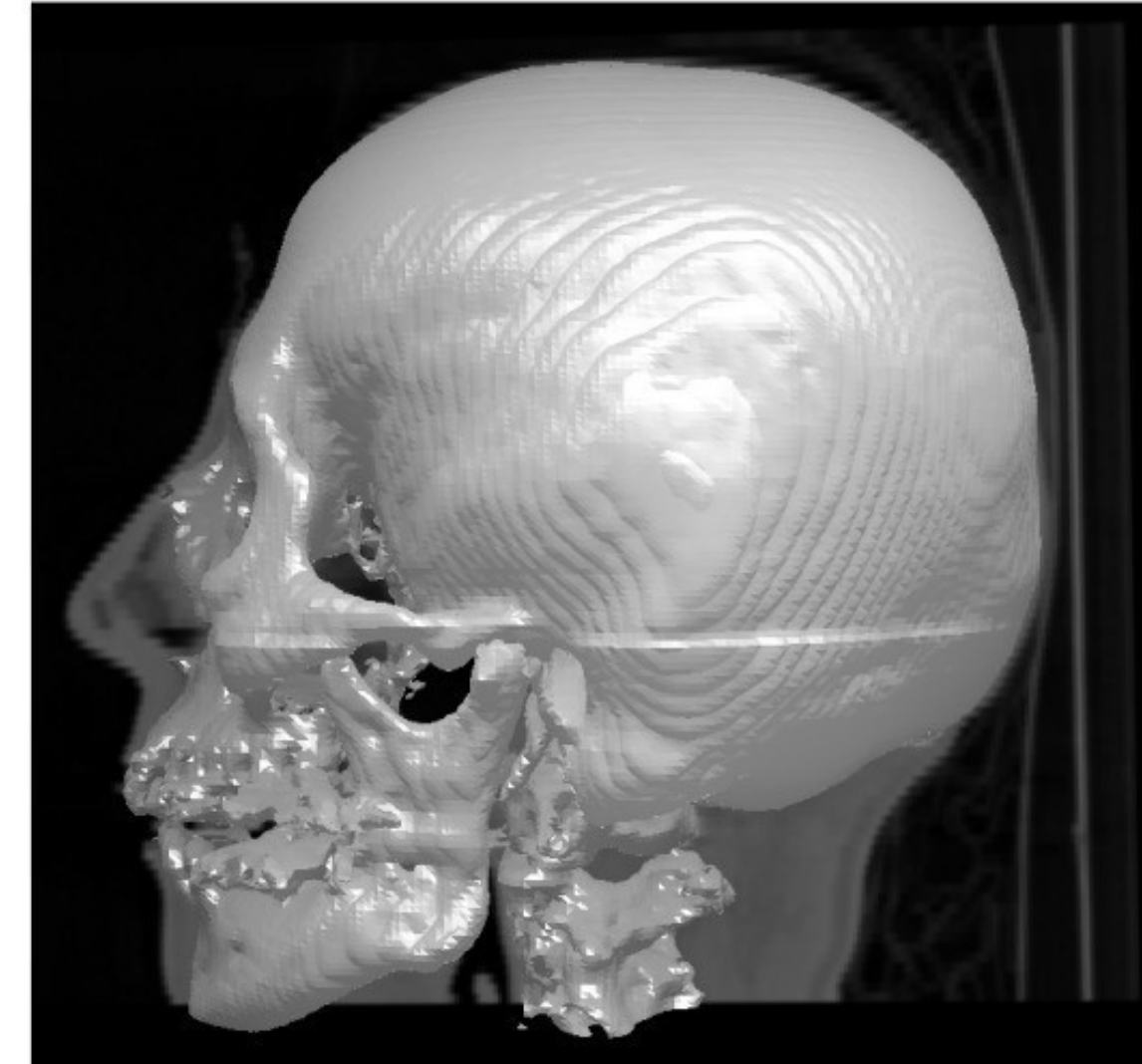
CT (X-Ray) data
Morphological data (skin, bone)
256 x 256 x 99



044

Isosurface example: CT

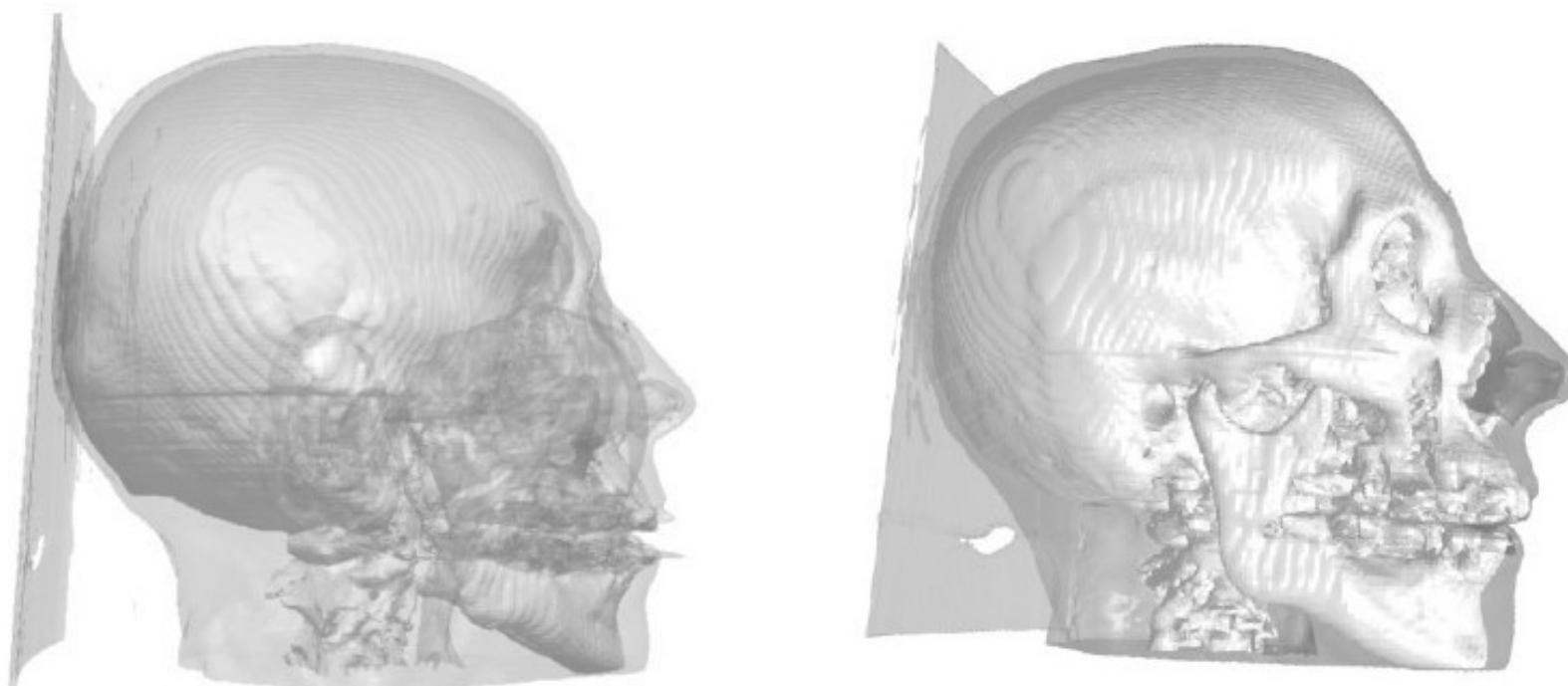
Combination slice + isosurface



045

Isosurface example: CT

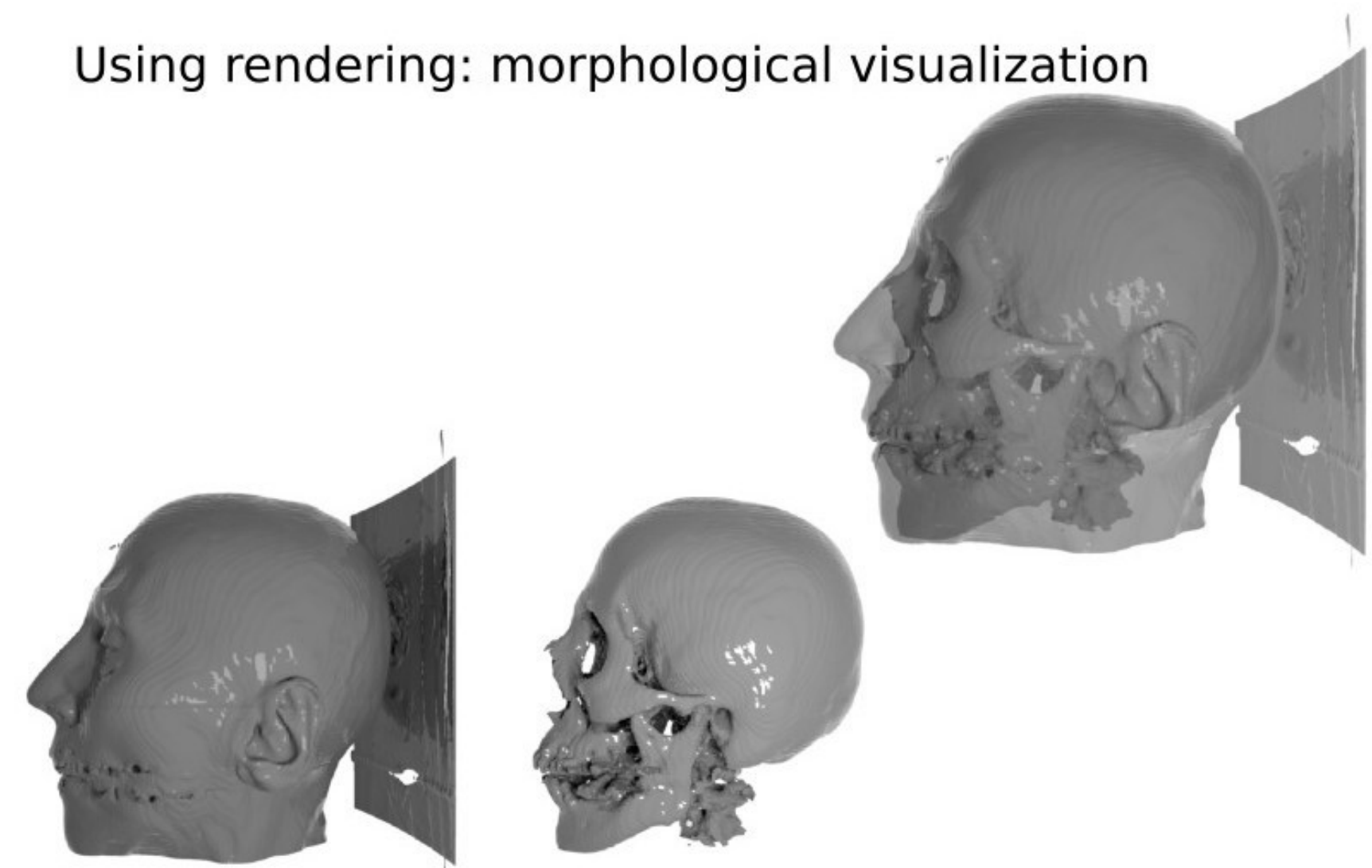
Accumulation of surface with transparency



046

Isosurface example: CT

Using rendering: morphological visualization



047

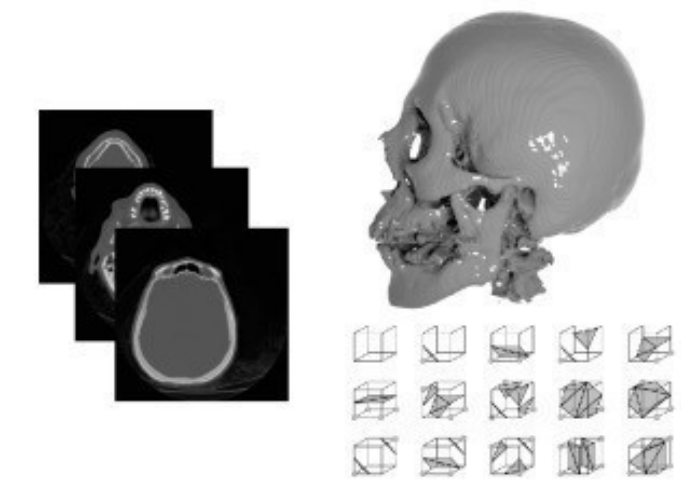
Ray casting

048

Ray casting

What we have seen:

- Slice in a volume
- Isosurface extraction (marching cubes/tetrahedron)



What we are going to see

- Transparency rendering
= **volume rendering**



049

Ray casting

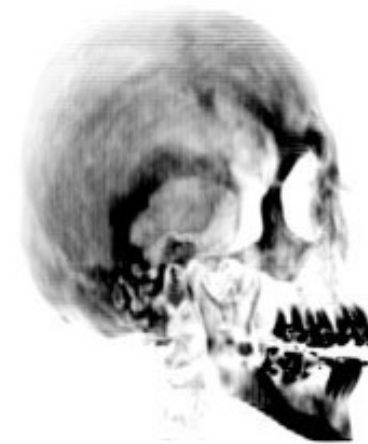
Surface rendering

- + Accurate
- + Data reduction
- Local information
- A-priori knowledge



Volume rendering

- + Global information, direct visualization
- Not accurate, transparency can be tricky



Pipe-line: Volume rendering to guide a surface extraction

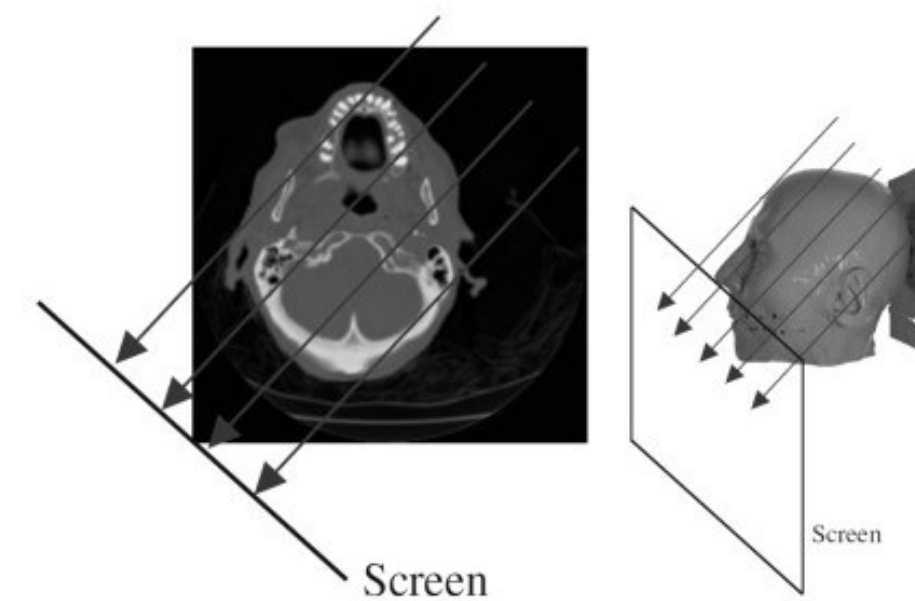
050

Ray casting

Goal: Modeling a data acquisition using transparency

Problem: Human are not used to see transparency

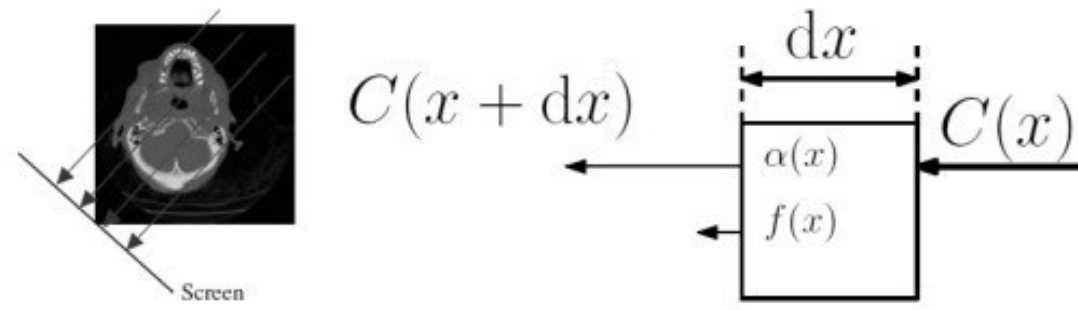
General approach: Ray-tracing/casting = Throw rays and set the color as a function of path and obstacles.



051

Ray casting: equations

For attenuated emission



$$C(x + dx) = [1 - \alpha(x) dx] C(x) + f(x)$$

$$\Rightarrow C'(x) = -\alpha(x)C(x) + f(x)$$

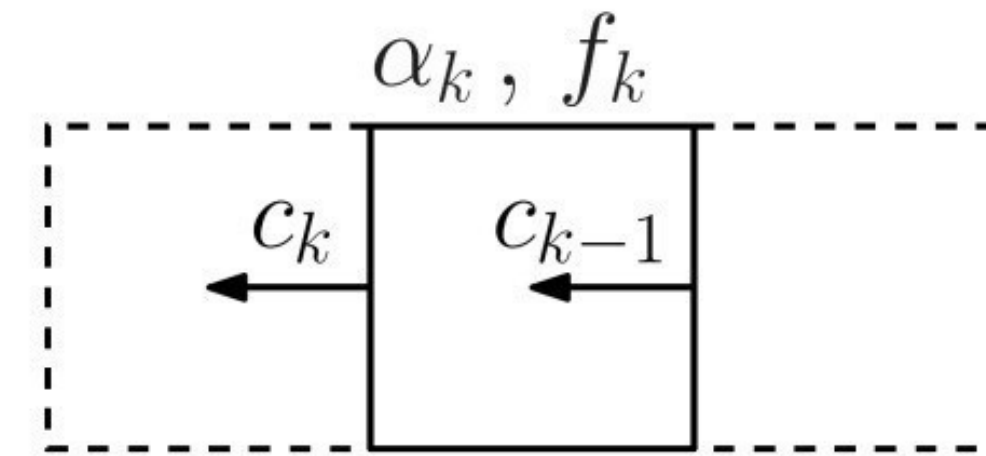
$$\Rightarrow C(x) = \left(\int_{x_0}^x f(u) e^{\int_{x_0}^u \alpha(t) dt} + C(x_0) \right) e^{-\int_{x_0}^x \alpha(t) dt}$$

For a given α, f , find C = Volume rendering

For a given C , find α, f = Tomography

052

Ray casting: discretization



In discrete form $c_k = (1 - \alpha_k) c_{k-1} + f_k$

α_k, f_k are functions of the intensity I of the current voxel
Can also depends on the derivatives

ex. $\alpha_k = A \Delta x I_k, f_k = B \Delta x I_k$

More generally, transfert functions are used

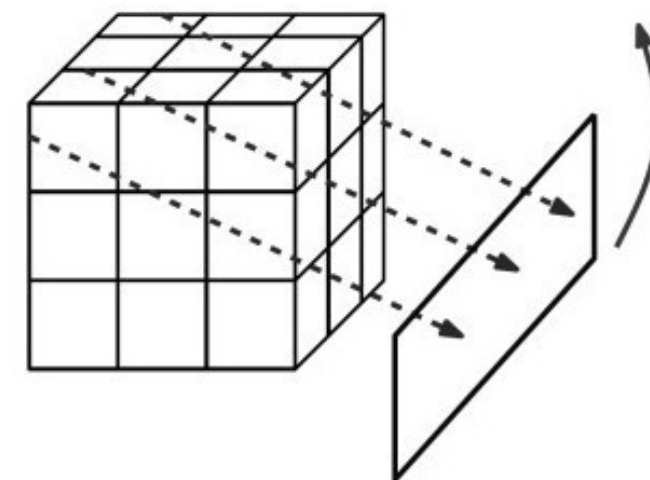
$$\alpha_k = \mathcal{F}(I_k) \quad f_k = \mathcal{G}(I_k)$$

053

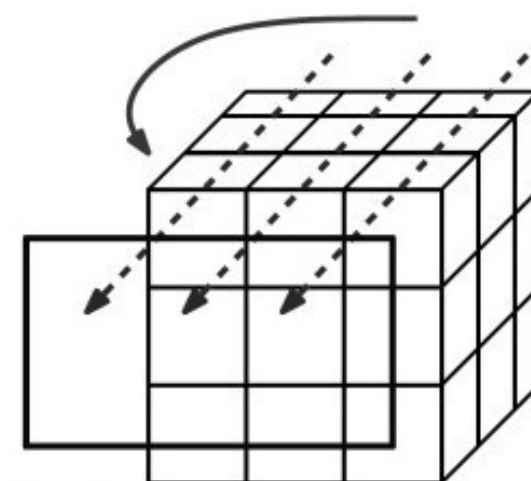
Ray casting: Implementation

2 Approches

Throw rotated lines in fixed grid



Rotate grid and integrate along fixed axis
(3D texture)



Trivial parallelisation

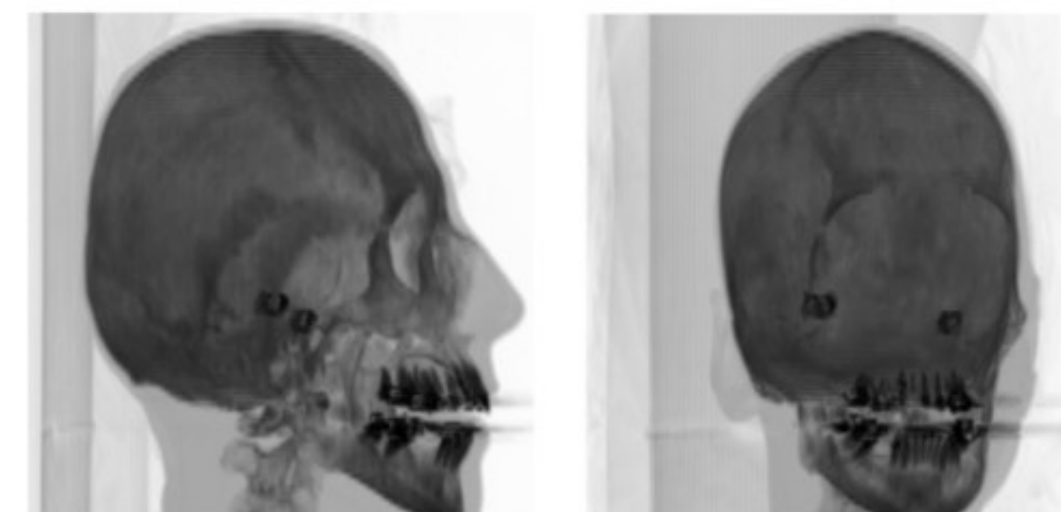
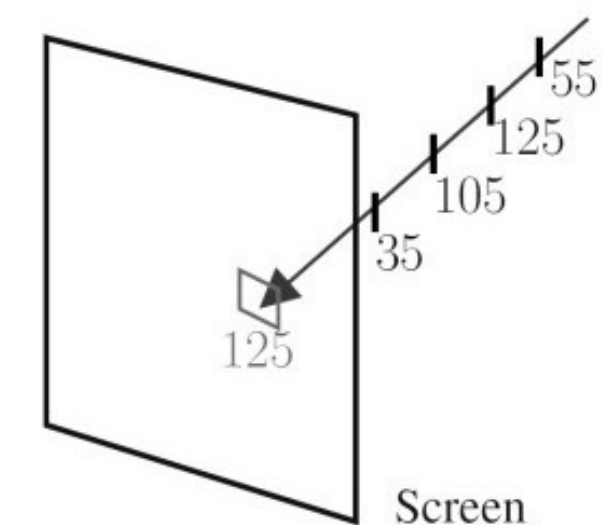
054

MIP

MIP = Maximum Intensity Projection $c = \max_k(I_k)$

+ Fast, simple
+ Standard in medical domain

- No information about depth
(without motion)



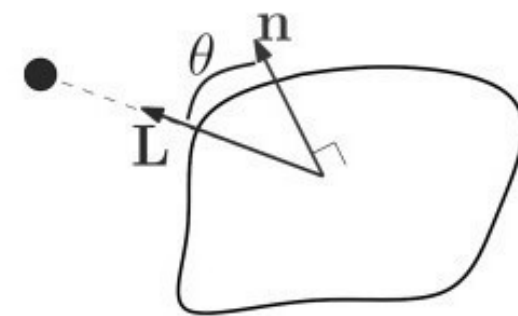
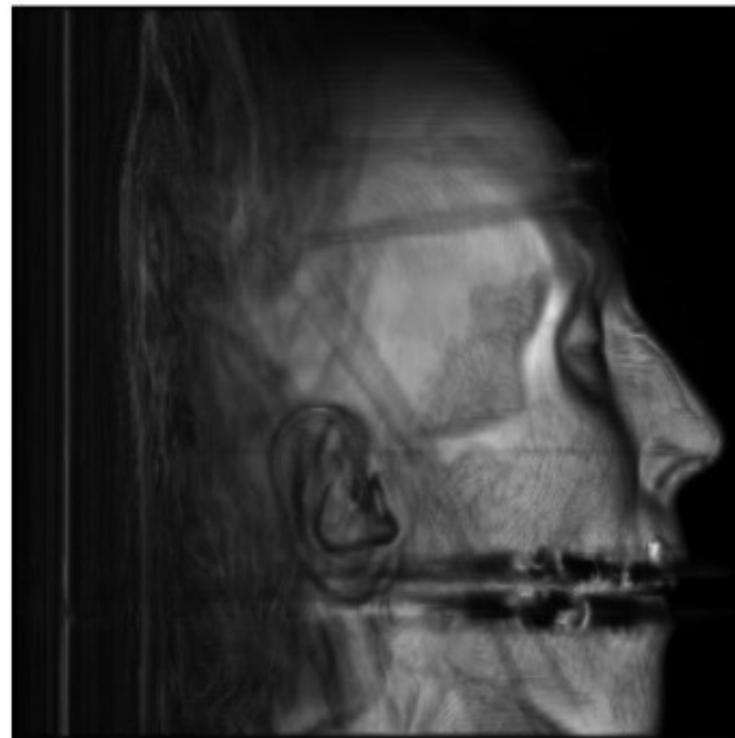
055

Shading

Diffuse shading $\cos(\theta) = \langle \mathbf{L}, \mathbf{n} \rangle$

In a given voxel, approximates the surface normal

$$\mathbf{n} = \frac{\nabla I}{\|\nabla I\|}$$

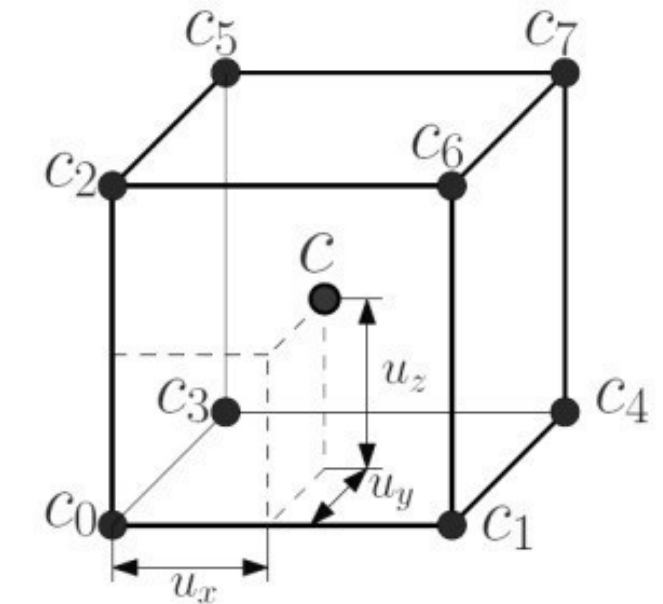


$$\nabla I = \begin{pmatrix} I(k_x + 1, k_y, k_z) - I(k_x - 1, k_y, k_z) \\ I(k_x, k_y + 1, k_z) - I(k_x, k_y - 1, k_z) \\ I(k_x, k_y, k_z + 1) - I(k_x, k_y, k_z - 1) \end{pmatrix}$$

056

Trilinear interpolation

$$c = \begin{pmatrix} (1 - u_x)(1 - u_y)(1 - u_z) & c0+ \\ u_x(1 - u_y)(1 - u_z) & c1+ \\ (1 - u_x)(1 - u_y)u_z & c2+ \\ (1 - u_x)u_y(1 - u_z) & c3+ \\ u_xu_y(1 - u_z) & c4+ \\ (1 - u_x)u_yu_z & c5+ \\ u_x(1 - u_y)u_z & c6+ \\ u_xu_yu_z & c7 \end{pmatrix}$$



057

Libraries

VTK: The Visualization ToolKit

Heavy and complete set of tools.

<http://www.vtk.org>

Volume rendering library (Stanford).

Standard, old.

<http://www-graphics.stanford.edu/software/volpack/>

ImageVis3D (Utah).

<http://www.sci.utah.edu/cibc/software/41-imagevis3d.html>

V3.

Fast on the GPU

<http://www.stereofx.org/volume.html>

058