Scientific Visualization

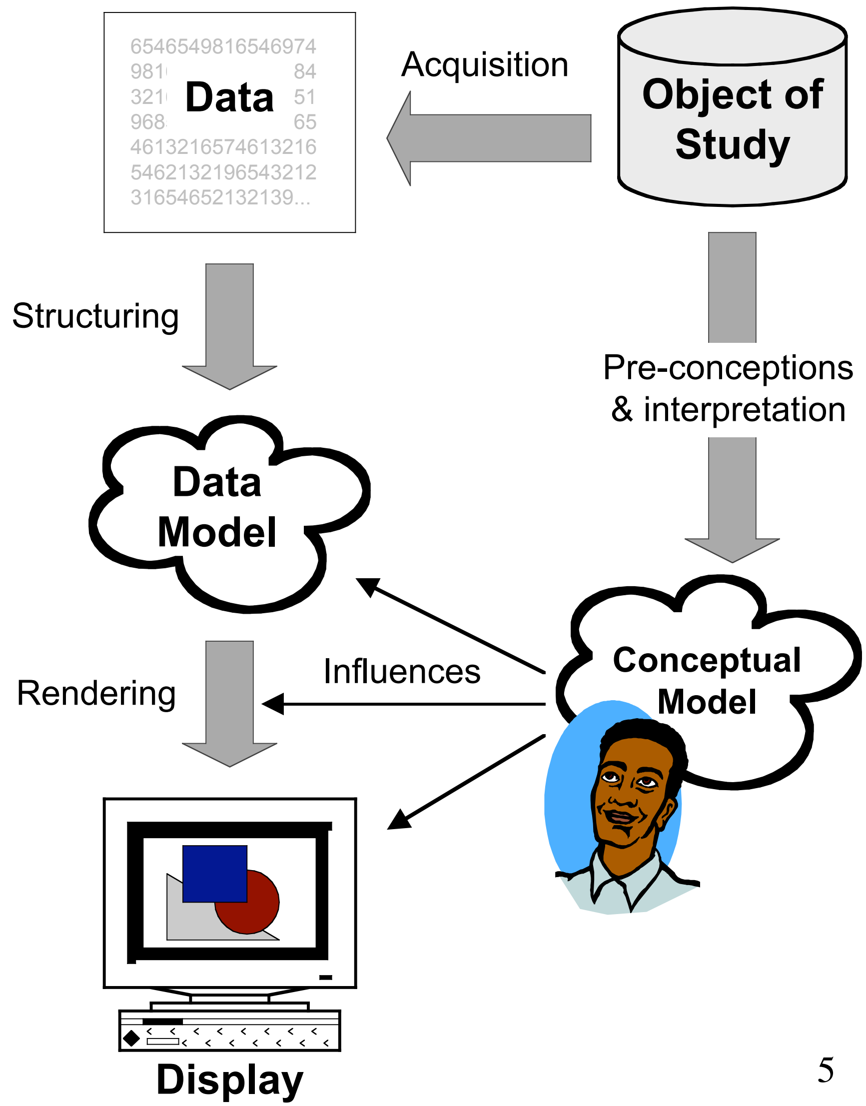
Torsten Möller

Acknowledgments: Prof. Raghu Machiraju, The Ohio State University as well as Prof. Klaus Mueller, SUNY Stony Brook, whose class notes I will re-use, re-interpret and manipulate.

Overview

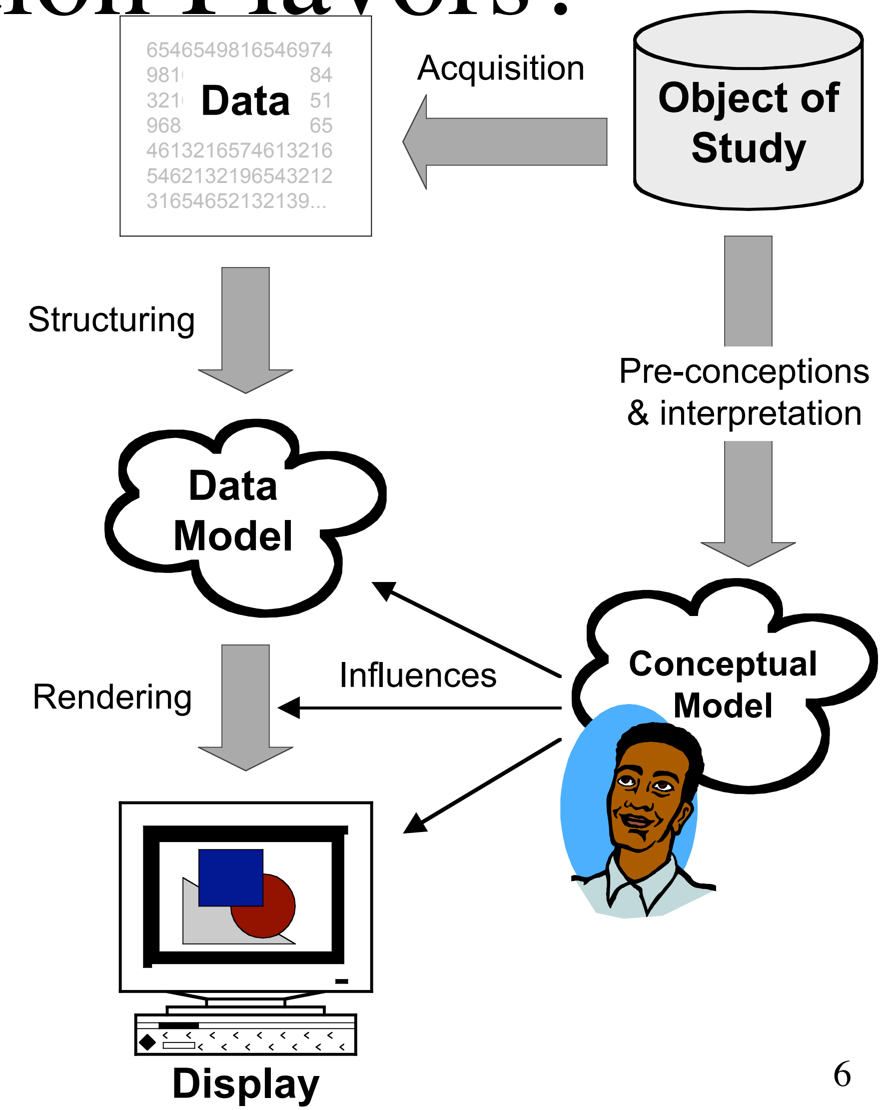
- What is SciVis?
- Acquisition Methods
- Iso-surfaces
- Direct-Rendering Pipeline
- Vector Visualization
- Challenges

What Is Visualization?



Visualization Flavors?

- Discrete or continuous data model
- Inherent spatial embedding or a chosen one?



Visualization Flavors?

Display Attributes

	Given	Constraint	Chosen
Continuous	✓Images (ie. Medical) ✓Molecular structures (distributions of mass, charge, etc.) ✓Globe (distribution data)	✓Distortions of given / continuous ideas (e.g., flattened medical structures, 2D ✓geographic maps, fish-eye lens views) ✓Arrangement of numeric variable values	✓Continuous mathematical functions ✓Continuous time-varying data, when time is mapped to a spatial dimension
Discrete	✓Segmented given / continuous data (e.g., segmented images) ✓Air traffic positions ✓Molecular structures (exact positions of components) ✓Globe (entity data)	✓Distortions of given / discrete ideas (e.g., 2D geographic maps, fish-eye lens views) ✓Arrangement of ordinal or numeric variable values	✓Discrete time-varying data, when time is mapped to a spatial dimension ✓Arbitrary entity-relationship data (e.g., file structures) ✓Arbitrary multi-dimensional data (e.g., employment statistics)

Visualization Flavors?

Display Attributes

	Given	Constraint	Chosen
Continuous	✓Images (ie. Medical) ✓Molecular structures (distributions of mass, charge, etc.) ✓Globe (distribution data)	✓Distortions of given / continuous ideas (e.g., flattened medical structures, 2D ✓geographic maps, fish-eye lens views) ✓Arrangement of numeric variable values	✓Continuous mathematical functions ✓Continuous time-varying data when time is mapped to a spatial dimension
Discrete	✓Segmented given / continuous data (e.g., segmented images) ✓Air traffic positions ✓Molecular structures (exact positions of components) ✓Globe (entity data)	✓Distortions of given / discrete ideas (e.g., 2D geographic maps, fish-eye lens views) ✓Arrangement of ordinal or nominal variable values	✓Discrete time-varying data, when time is mapped to a spatial dimension ✓Arbitrary entity-relationship data (e.g., file structures) ✓Arbitrary multi-dimensional data (e.g., employment statistics)

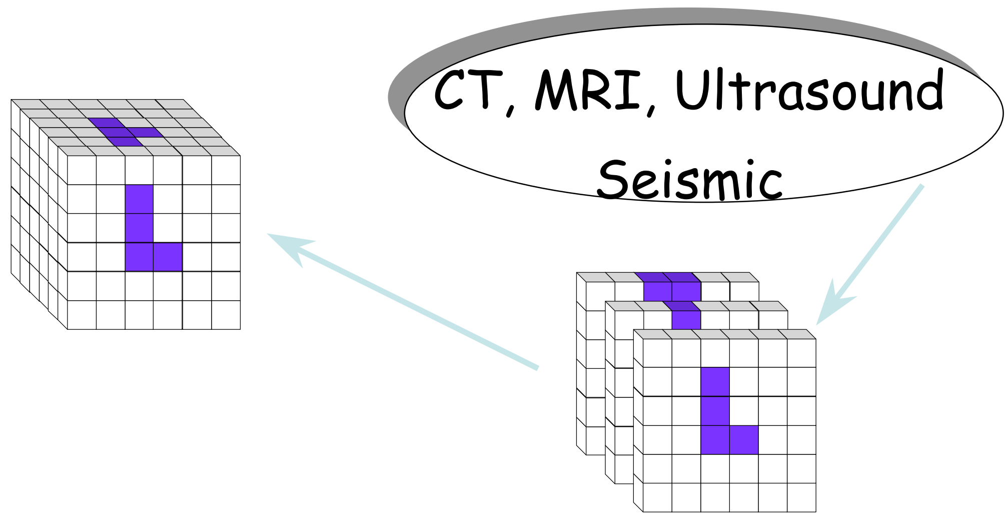
Scientific Visualization

Math Visualization

Bio Visualization

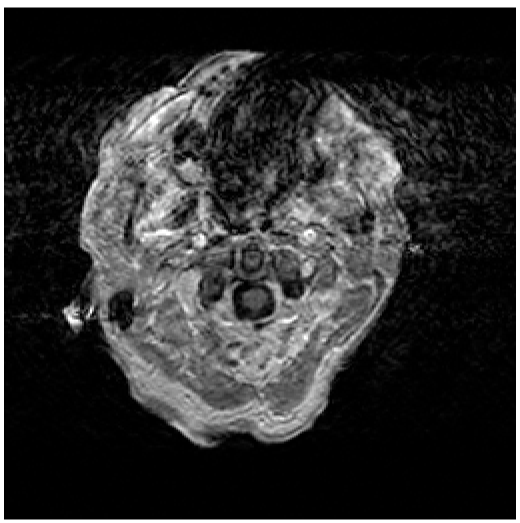
Information Visualization

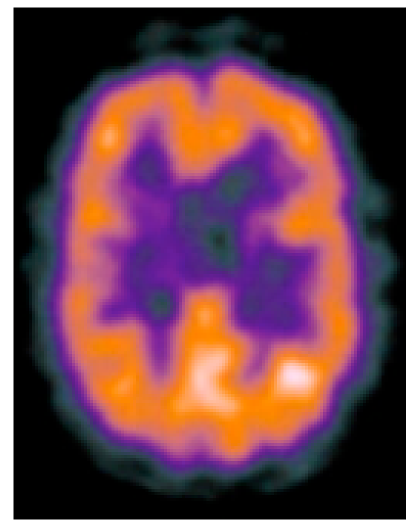
Visualization Pipeline



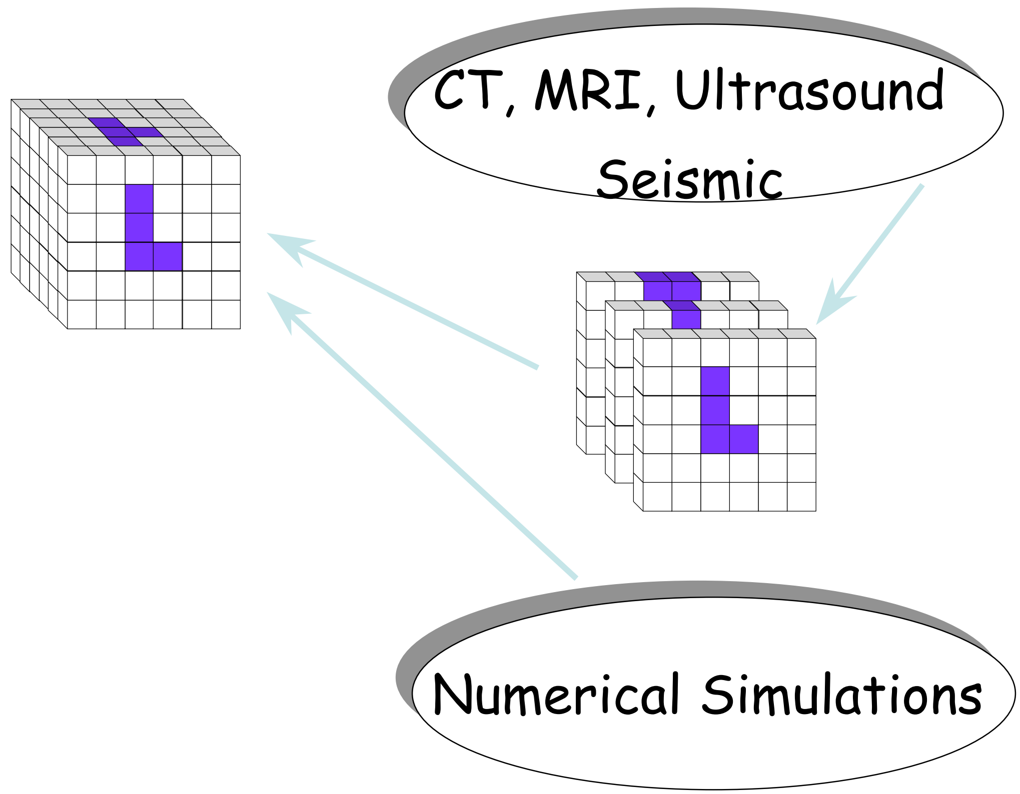
Scanning - Domains

- Medical scanners (MRI, CT, SPECT, PET, ultrasound)



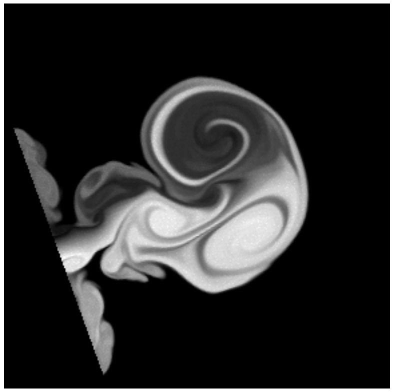


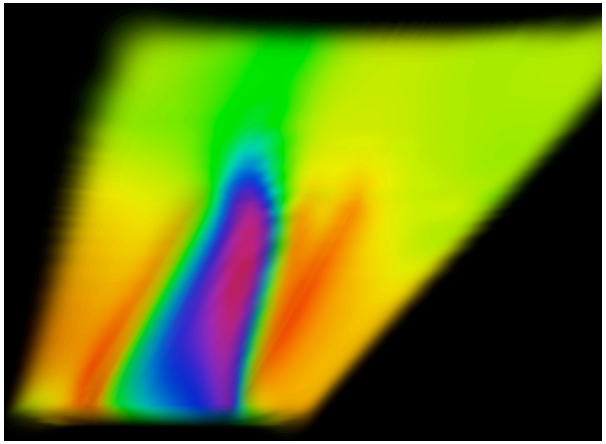
Visualization Pipeline



Scientific Computation - Domain

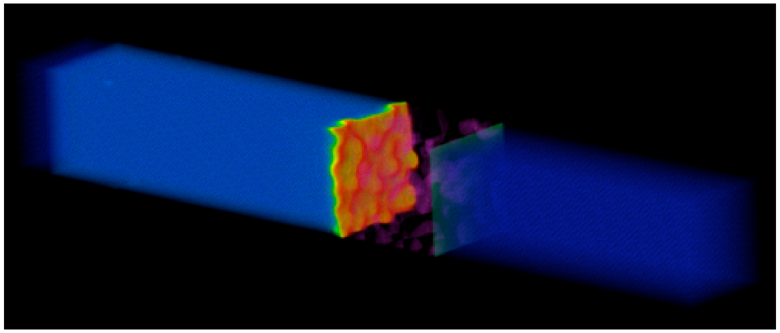
- Mathematical analysis
- ODE/PDE (ordinary and partial differential equations)
- Finite element analysis (FE),
- Supercomputer simulations,

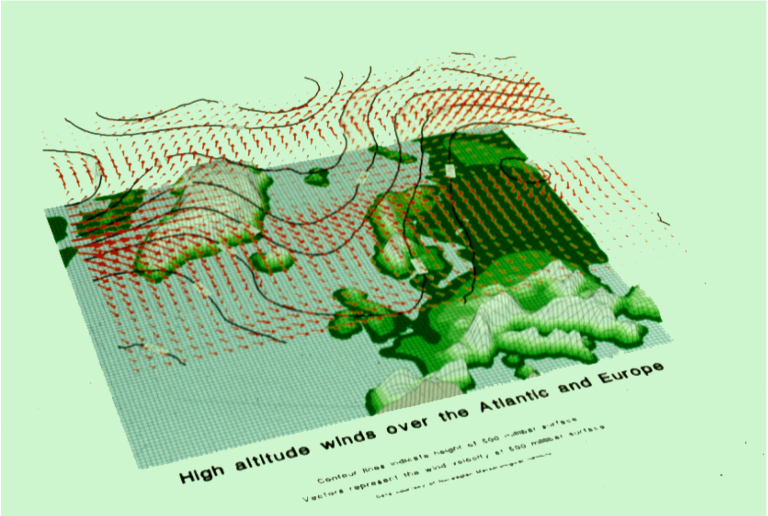




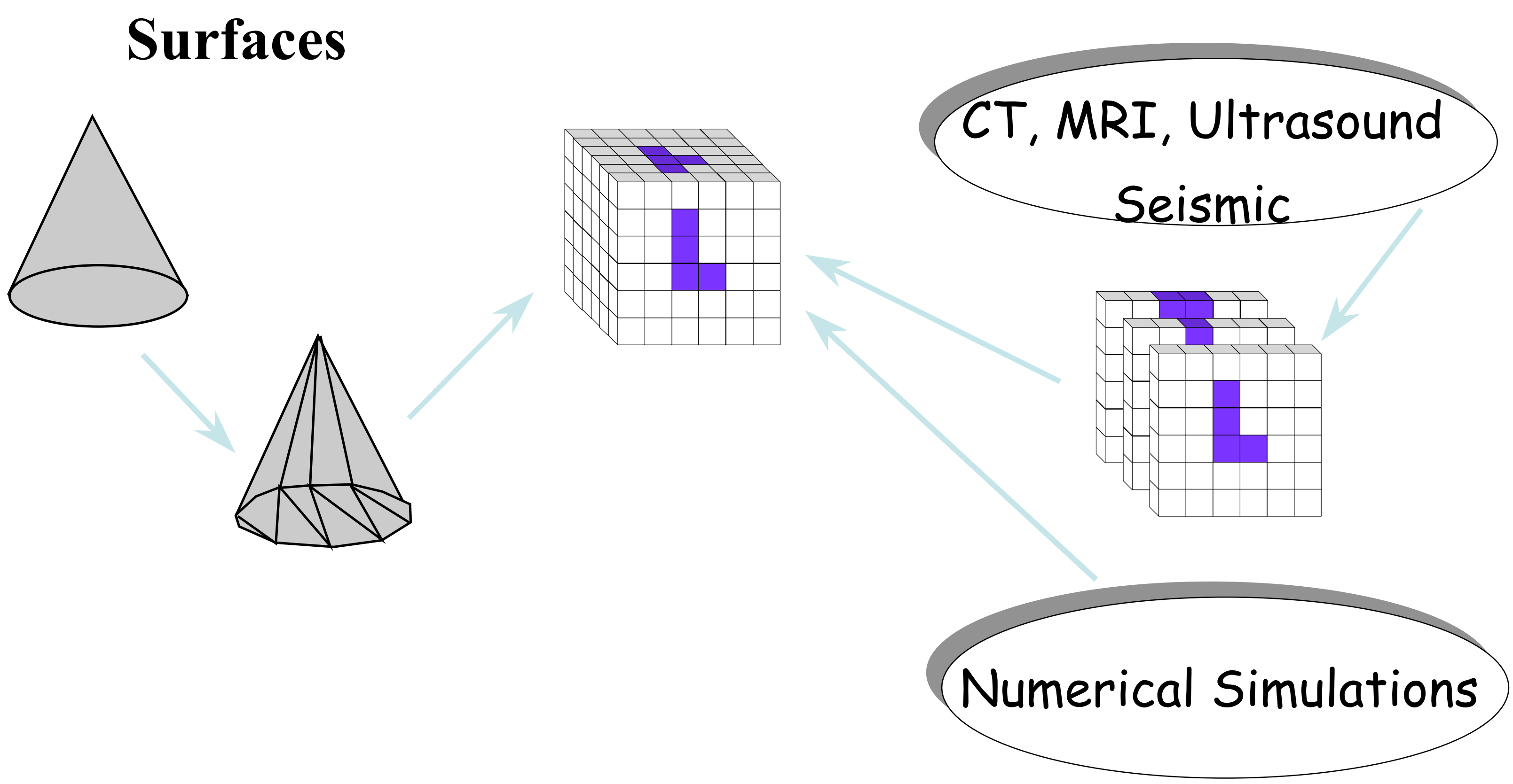
Scientific Computation - Apps

- Computational fluid dynamics (CFD),
- Computational field simulations (CFS),

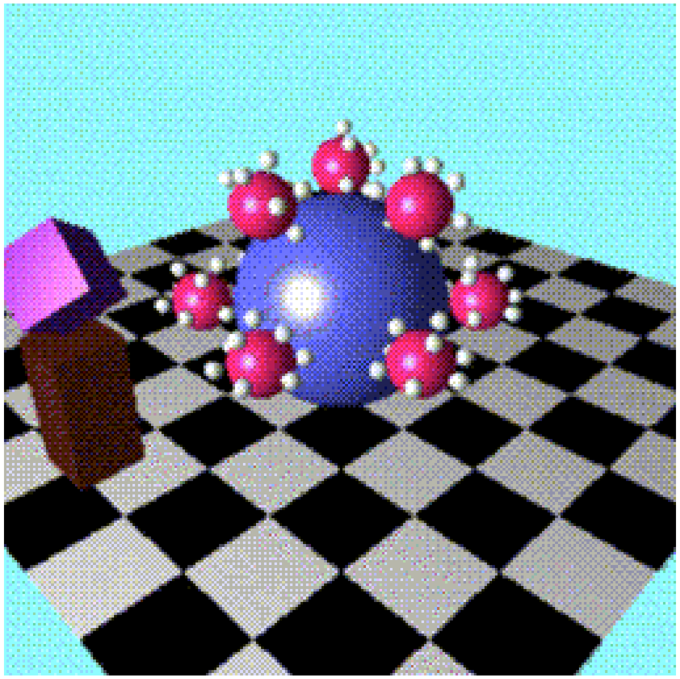


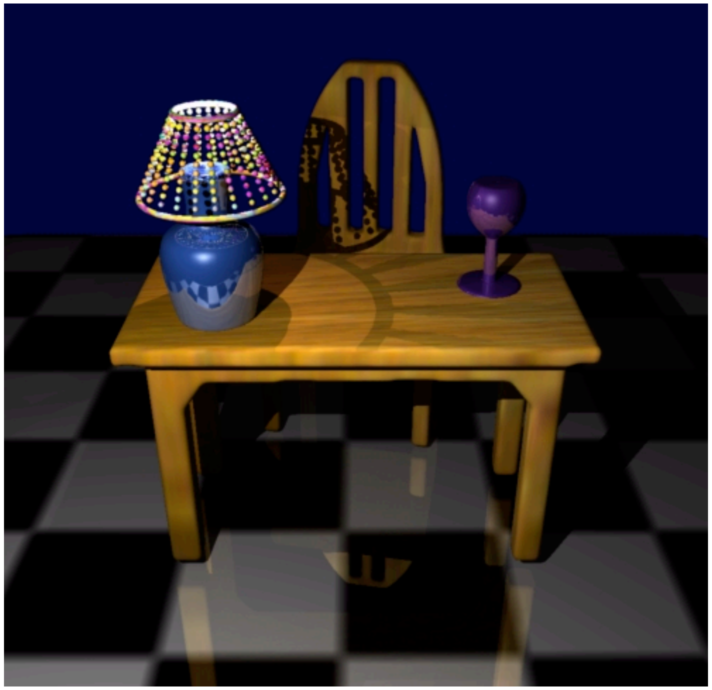


Visualization Pipeline

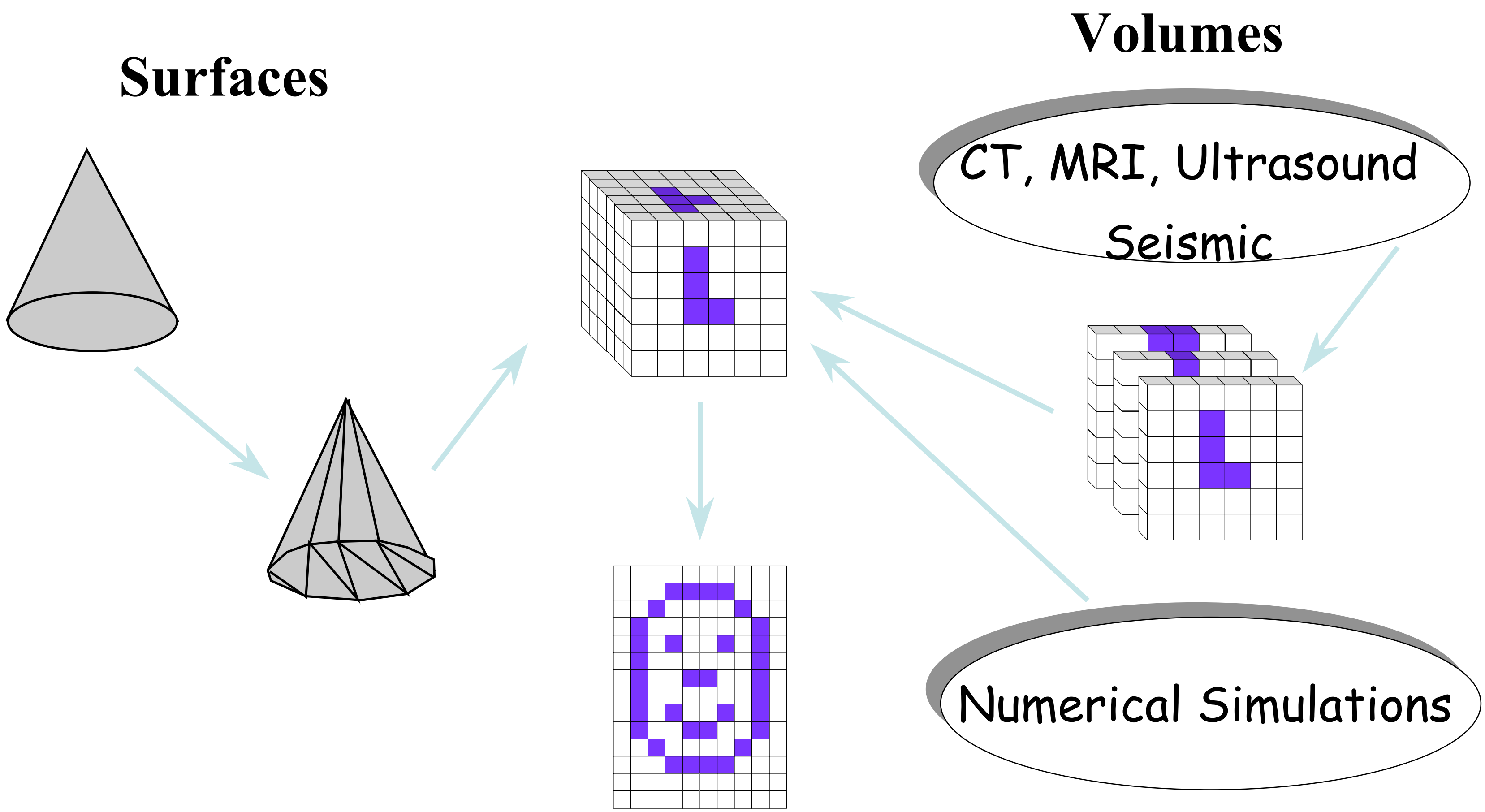


Surfaces

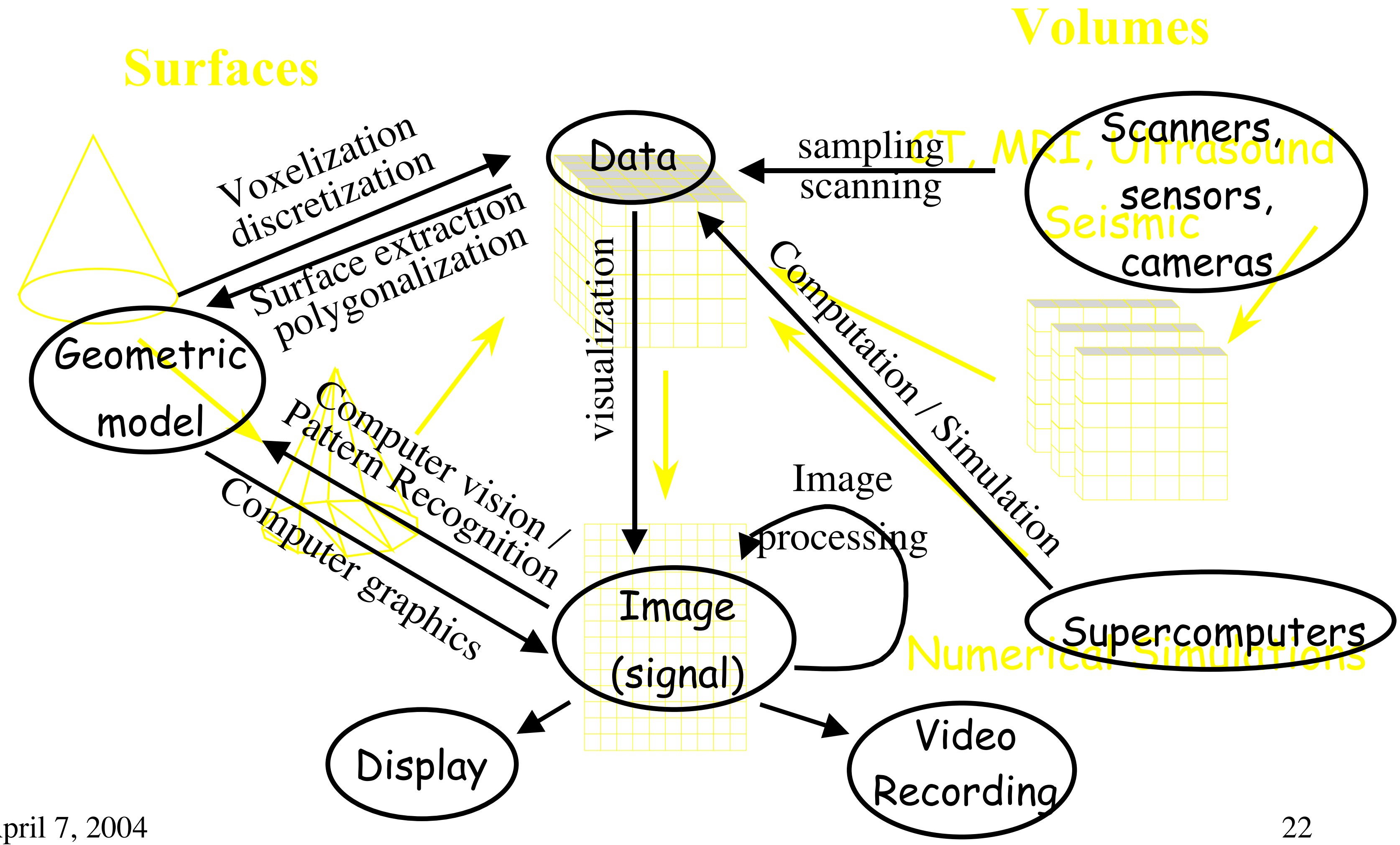




Visualization Pipeline



Taxonomy



Surfaces
Volumes
Voxelization
discretization
Surface extraction
polygonalization
Data
sampling
scanning
Scanners,
sensors,
cameras
CT, MRI, Ultrasound
Seismic
Geometric model
visualization
Computation / Simulation
Computer vision /
Pattern Recognition
Computer graphics
Image processing
Image (signal)
Supercomputers
Numerical simulations
Display
Video Recording

Overview

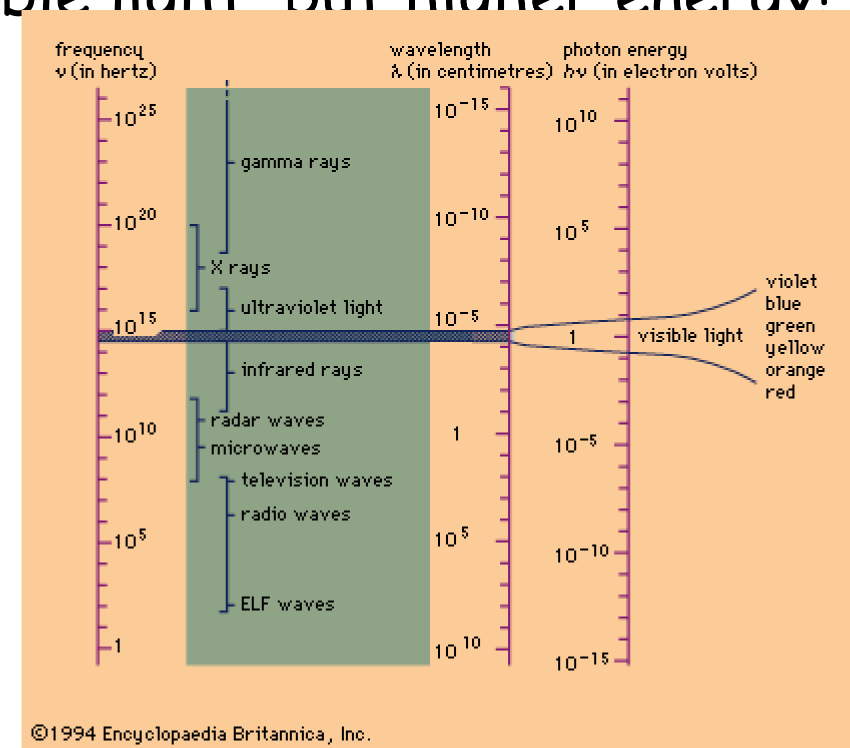
- What is SciVis?
- **Acquisition Methods**
- Iso-surfaces
- Direct-Rendering Pipeline
- Vector Visualization
- Challenges

Acquisition Methods

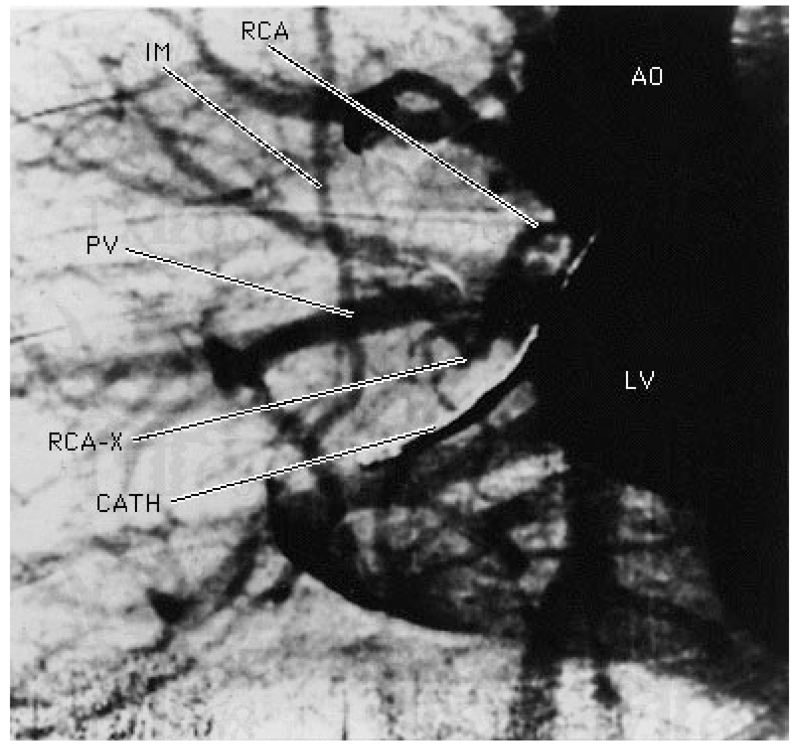
- X-Rays
- Computer Tomography (CT or CAT)
- MRI (or NMR)
- PET / SPECT
- Ultrasound
- Computational
- Synthetic

X-Rays

- photons produced by an electron beam
- similar to visible light, but higher energy!

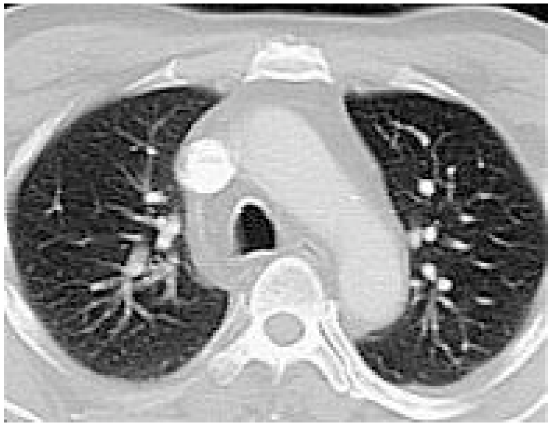


X-Rays - Images



CT or CAT - Methods

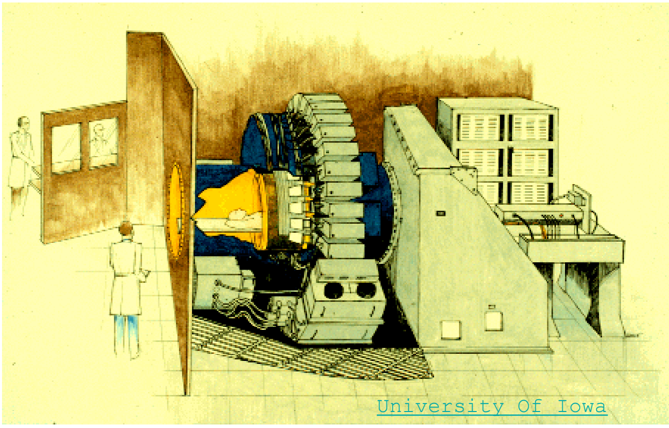
- measures the attenuation of X-rays from many different angles
- a computer reconstructs the organ under study in a series of cross sections or planes
- combine X-ray pictures from various angles to reconstruct 3D structures



video

CT - Beating Heart?

- Noise if body parts move!
- Heart - synchronize imaging with heart beat
  - can't capture beating well
  - need faster techniques
- Dynamic Spatial Reconstructor
  - has 14 X-ray/camera pairs
  - but turns slower
  - 2D projections seem more plausible
  - and cheaper



heart

beating

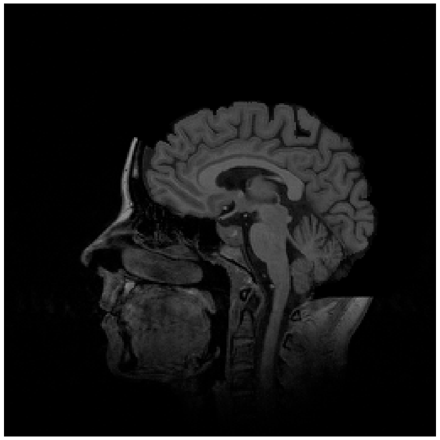
MRI

- Nuclear Magnetic Resonance (NMR) (or Magnetic Resonance Imaging - MRI)
- most detailed anatomical information
- high-energy radiation *is not* used, i.e. “save”
- based on the principle of nuclear resonance
- (medicine) uses resonance properties of protons

MRI - Signal to Noise Ratio

- proton density pictures - measures H
  MRI is good for tissues, but not for bone
- signal recorded in Frequency domain!!
- Noise - the more protons per volume unit, the more accurate the measurements - better SNR through decreased resolution

video

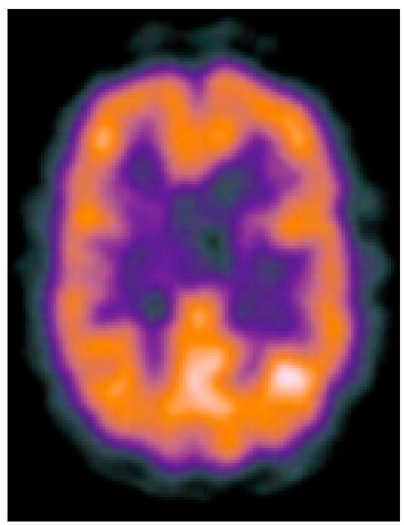


PET/SPECT

- Positron Emission Tomography
  Single Photon Emission Computerized Tomography
- recent technique
- involves the emission of particles of antimatter by compounds injected into the body being scanned
- follow the movements of the injected compound and its metabolism
- reconstruction techniques similar to CT - Filter Back Projection & iterative schemes

SPECT

- Emit (any) gamma rays
- collected with gamma camera
- cheap

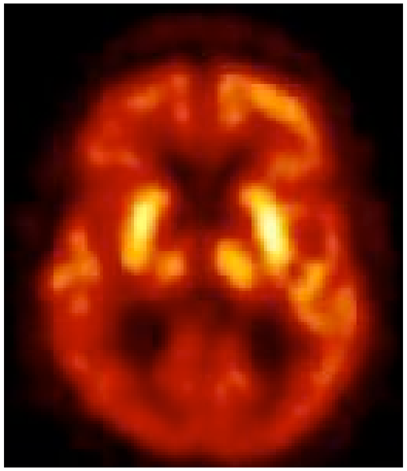


video

PET

- positrons collides with electron to emit photons in 180^0 angle
- both annihilation photons detected in coincidence
- higher sensitivity
- more expensive
- tracer has shorter half-live

video



Comparison

"CT and MRI show that
you have a brain;
PET and SPECT show
that you use it!"

Ultrasound

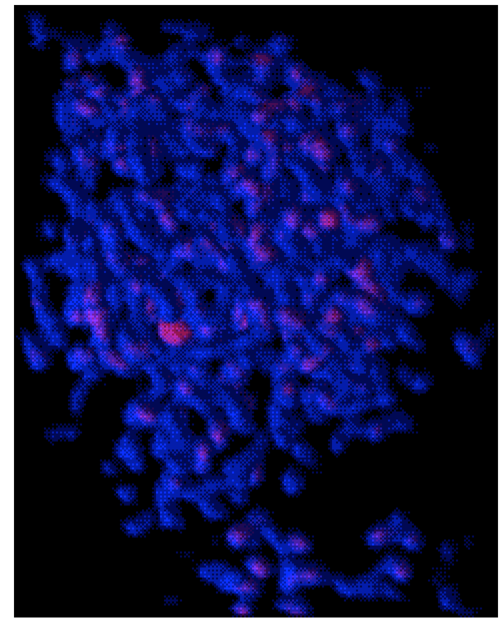
- by far least expensive
- very safe
- very noisy
- 1D, 2D, 3D scanners
- irregular sampling - reconstruction problems

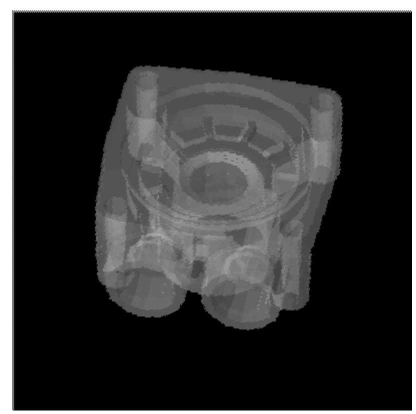
Comparison

	safety	tissue	time	quality
CT	--	bone	30-60s	high
MRI	+	soft tissue	30-60min	medium
PET/SPECT	++	functional	15-30min	low
Ultrasound	+++	borders	immediate	bad

Computational Methods (CM)

- Computational Field Simulations
- Computational Fluid Dynamics - Flow simulations
- Computational Chemistry - Electron-electron interactions, Molecular surfaces
- Computational Mechanics - Fracture
- Computational Manufacturing - Die-casting



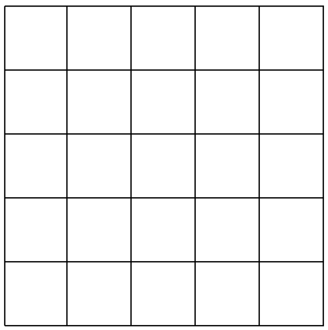


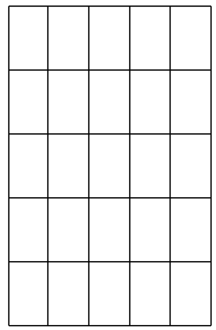
CM - Approach

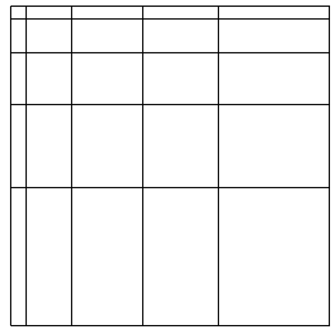
- (Continuous) physical model
  - Partial/Ordinary Differential Equation (ODE/PDE)
  - e.g. Navier-Stokes equation for fluid flow
  - e.g. Hosted Equations: $f_{xx} = g(x) : f(a) = A, f(b) = B, a < x < b$
  - e.g. Schrödinger Equation - for waves/quantum
- Continuous solution doesn't exist (for most part)
- Numerical Approximation/Solution
  1. Discretize solution space - Grid generation explicit
  2. Replace continuous operators with discrete ones
  3. Solve for physical quantities

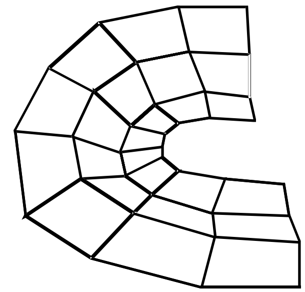
CM - Grid Types

Structured Grids:

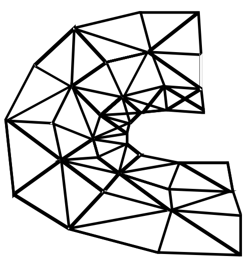

regular

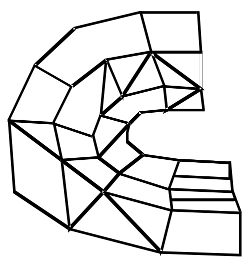

uniform

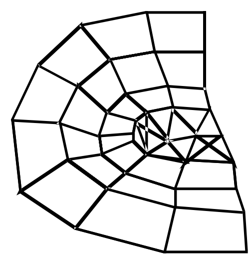

rectilinear

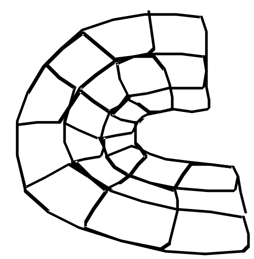

curvilinear

Unstructured Grids:

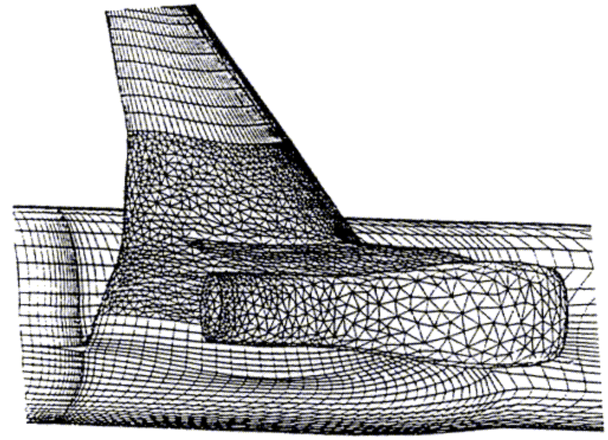

regular

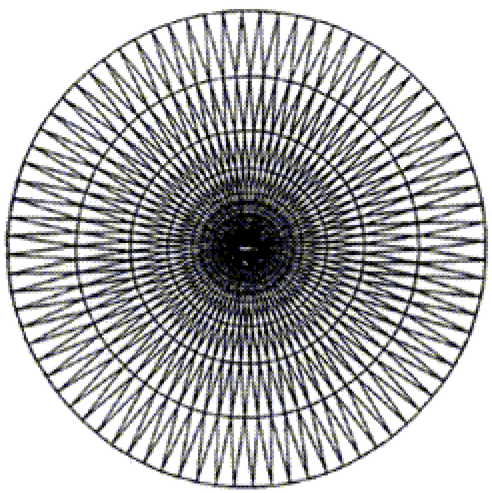

irregular

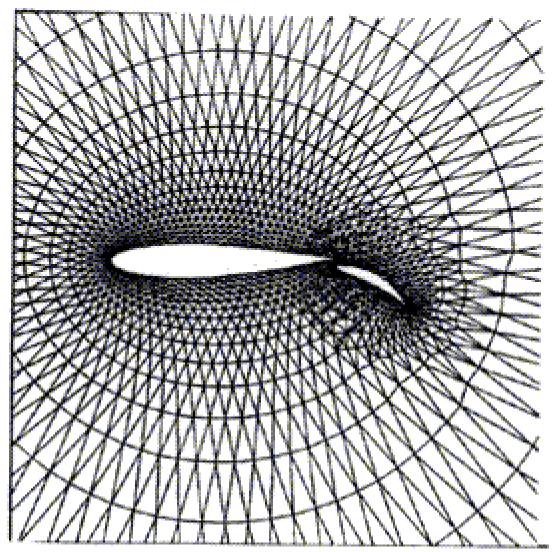

hybrid

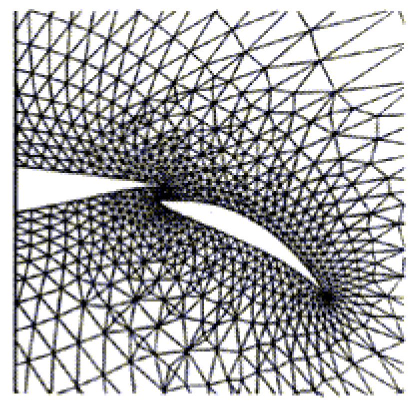

curved

CM - Grid Exc







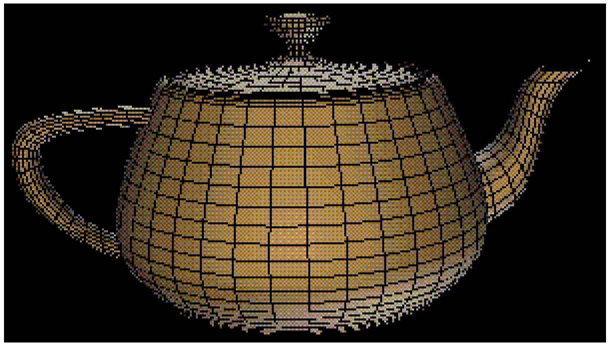


CM - Solution Spaces

- desirable Grids:
  - smooth grids
  - non-folding grids
- time-varying (4D)
- vector data (as opposed to scalar data)

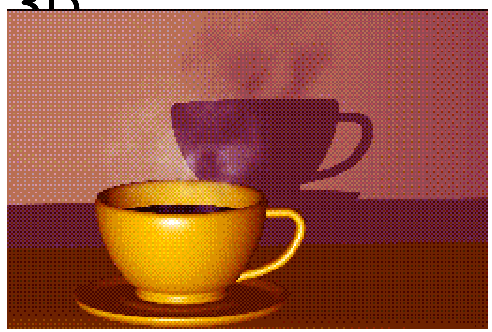
Synthetic Methods

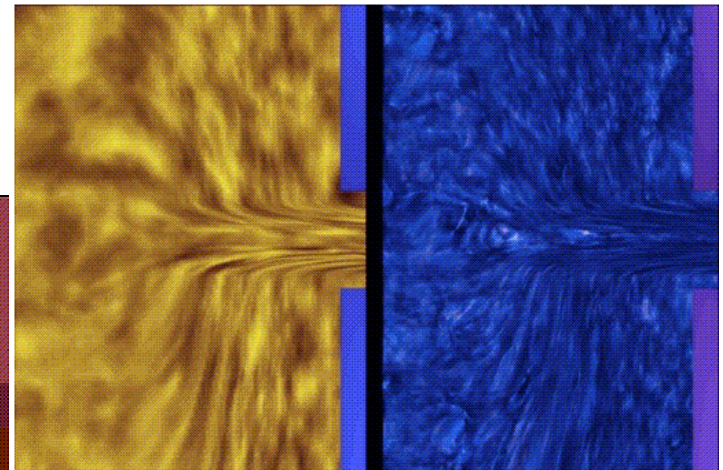
- 3D Discretization Techniques ☛ *Voxelization*
- Scan Conversion of Geometric Objects
  - Planes / Triangles
  - Cylinders
  - Sphere
  - Cone
  - NURBS, Bezier patches

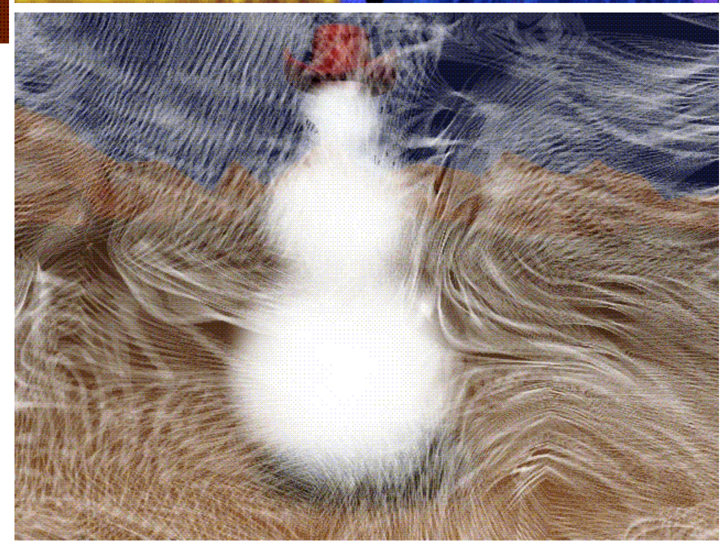


Synthetic Methods

- Solid Textures
- Hyper Texture - 3D Textures
  - Fur
  - Marble
  - Hair
  - Turbulent flow
- 3D Regular grid has texture values







Volume Generation

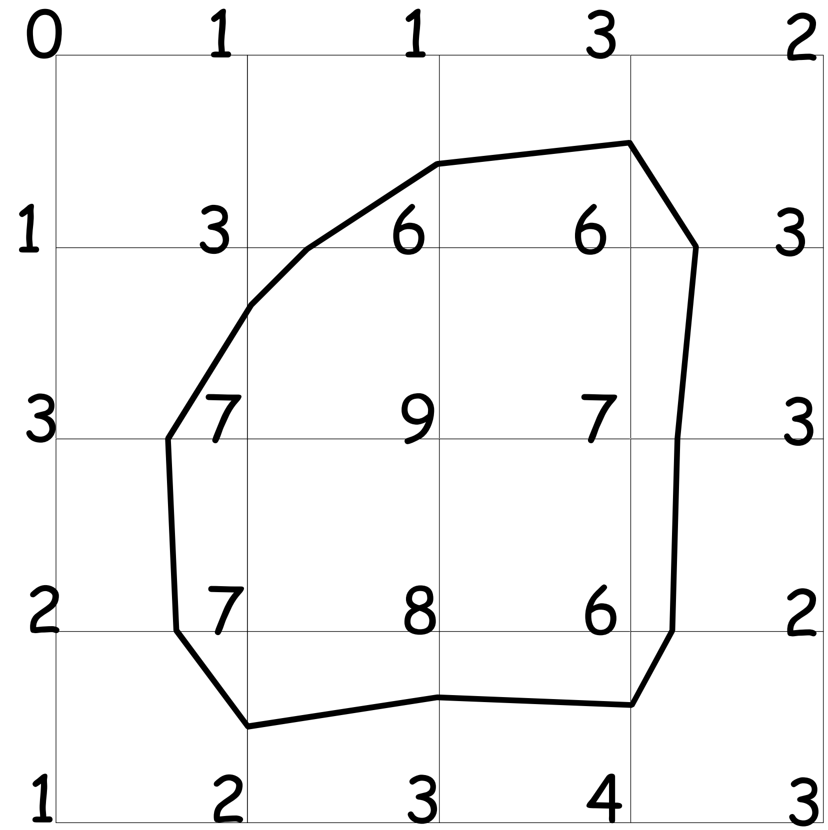
- Capture original function accurately
  - Sampling Theorems
  - Sufficient resolution
- Should not create
  - Noise - Medical
  - Small Triangles - CAGD
  - Flaws (Cracks) - CAGD
- For computational simulations
  - capture geometry
  - adapt to solution
  - time varying, vector fields

Overview

- What is SciVis?
- Acquisition Methods
- **Iso-surfaces**
- Direct-Rendering Pipeline
- Vector Visualization
- Challenges

Isosurface Extraction

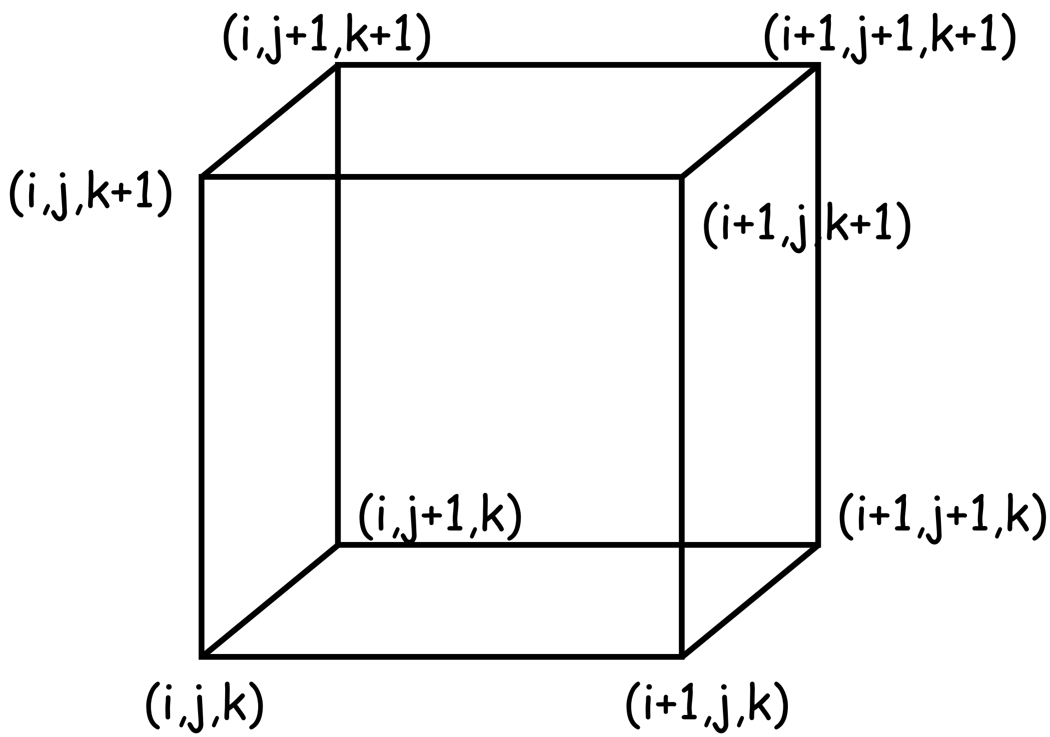
- by contouring
  - closed contours
  - continuous
  - determined by iso-value
- several methods
  - marching cubes
  - dividing cubes
  - surface tracking
  - span space



Iso-value=5

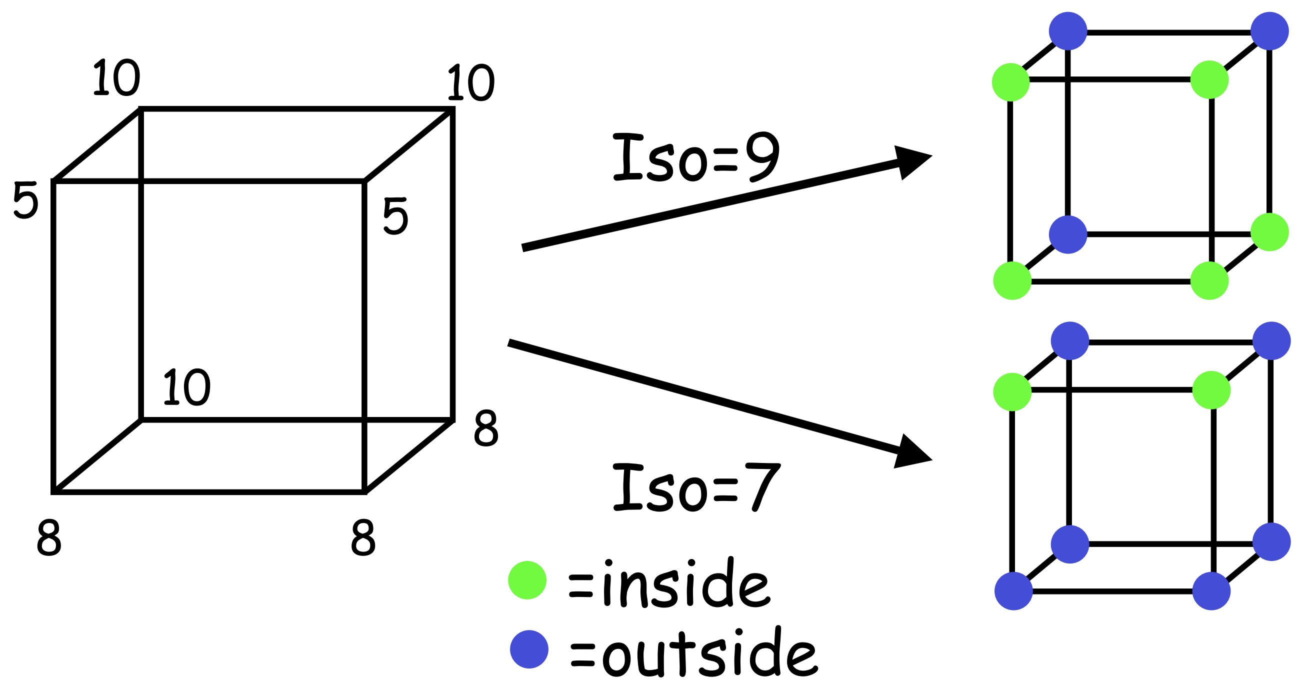
MC 1: Create a Cube

- Consider a Cube defined by eight data values:



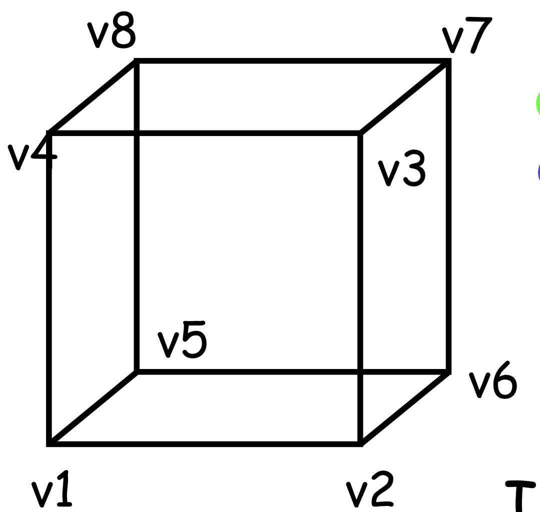
MC 2: Classify Each Voxel

- Classify each voxel according to whether it lies outside the surface (value > iso-surface value) inside the surface (value <= iso-surface value)

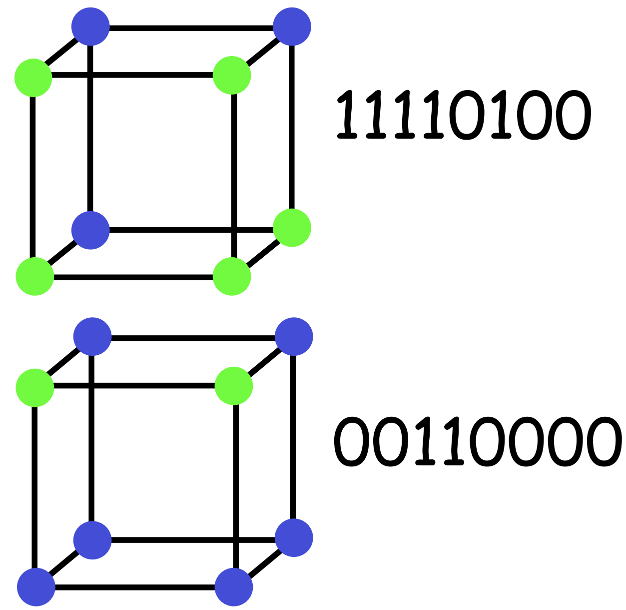


MC 3: Build An Index

- Use the binary labeling of each voxel to create an index



inside =1
outside=0

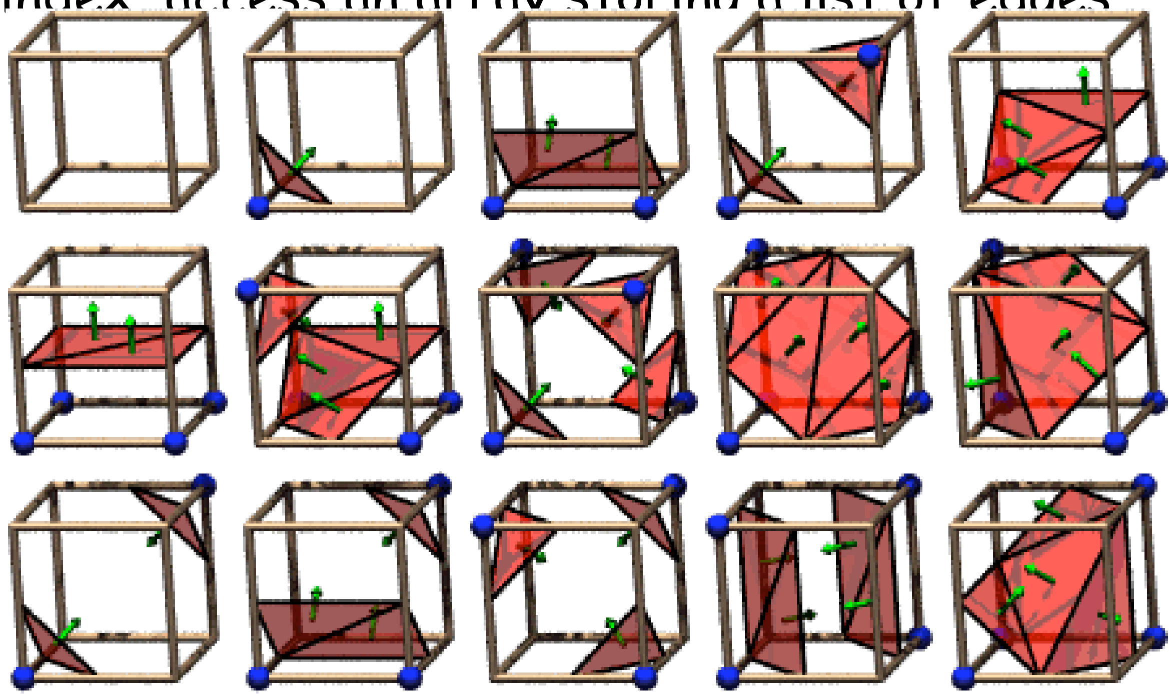


Index:

v1	v2	v3	v4	v5	v6	v7	v8

MC 4: Lookup Edge List

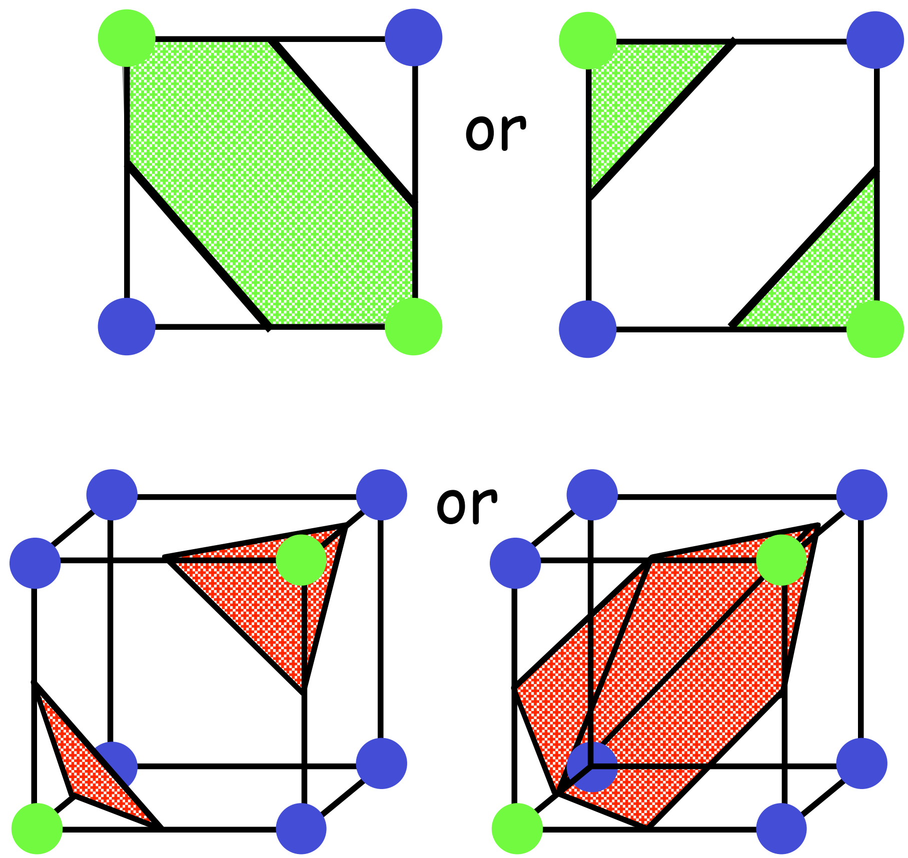
- For a given index, access an array storing a list of edges


The 15 Cube Combinations

✓ all 256 cases can be derived from 15 base cases

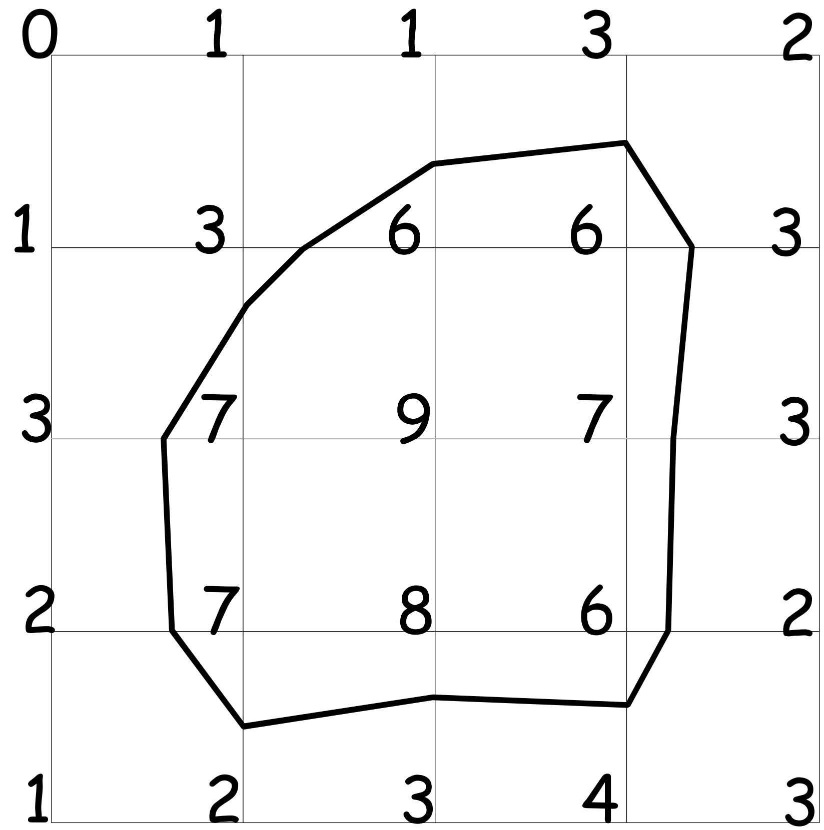
Ambiguous Cases

- Ambiguous cases:
  3, 6, 7, 10, 12, 13
- Adjacent vertices:
  different states
- Diagonal vertices:
  same state
- Resolution:
  decide for one case



Isosurface Extraction

- MC
  - Most popular one
  - There are faster ones
  - Not as simple to program
- Isosurface rendering doesn't really show "thickness" of features etc.



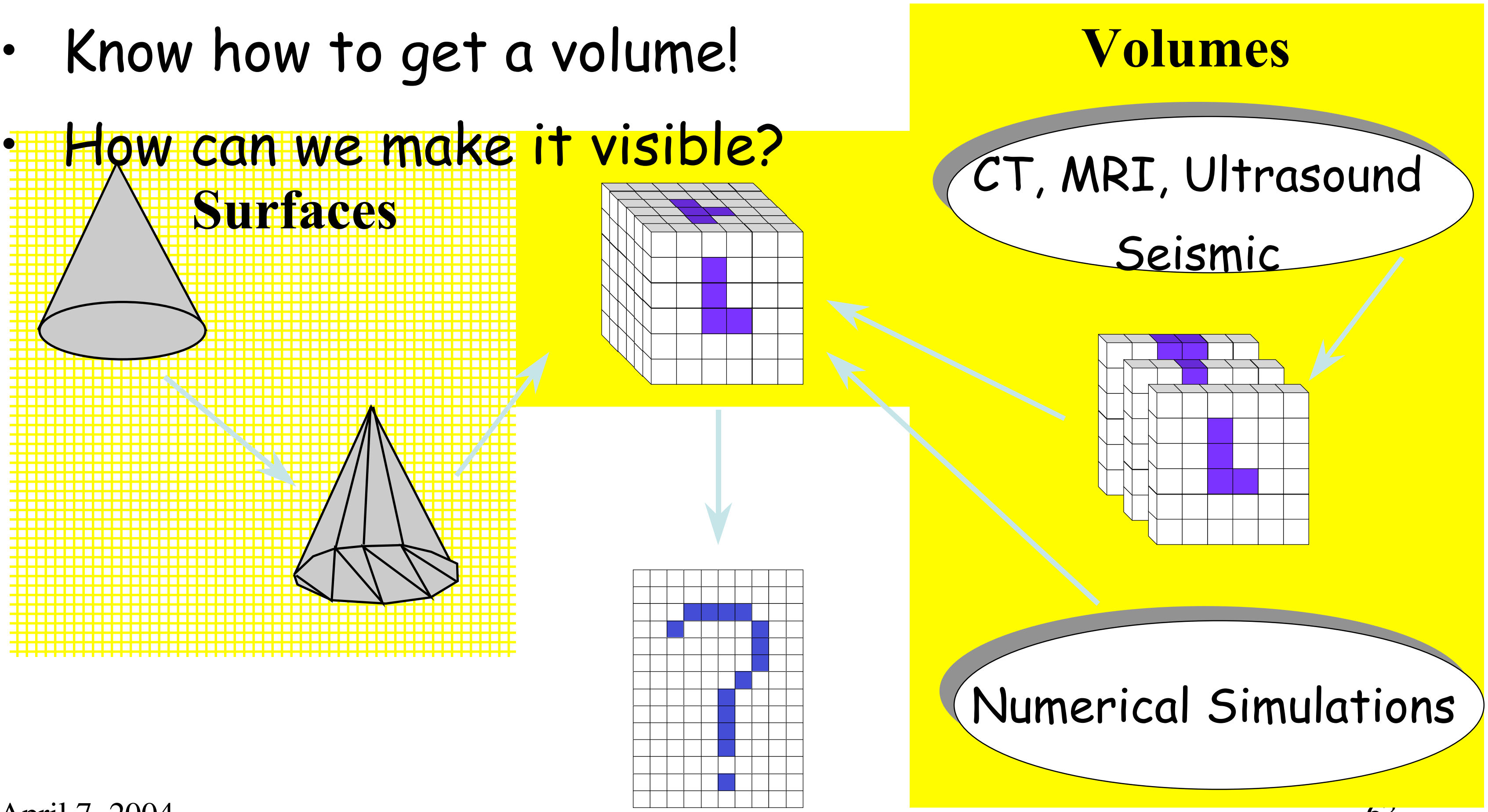
Iso-value=5

Overview

- What is SciVis?
- Acquisition Methods
- Iso-surfaces
- **Direct-Rendering Pipeline**
- Vector Visualization
- Challanges

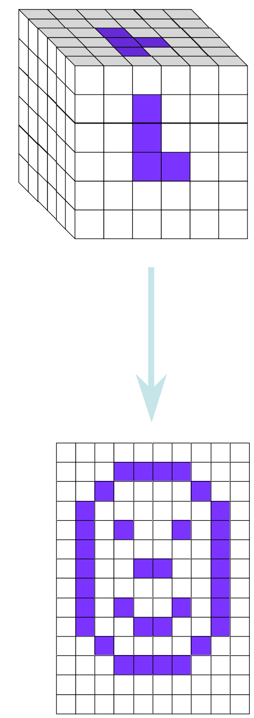
What Now?

- Know how to get a volume!
- How can we make it visible?

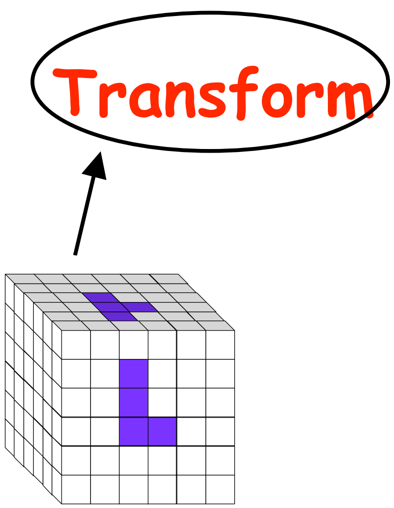


Rendering Pipeline (RP)

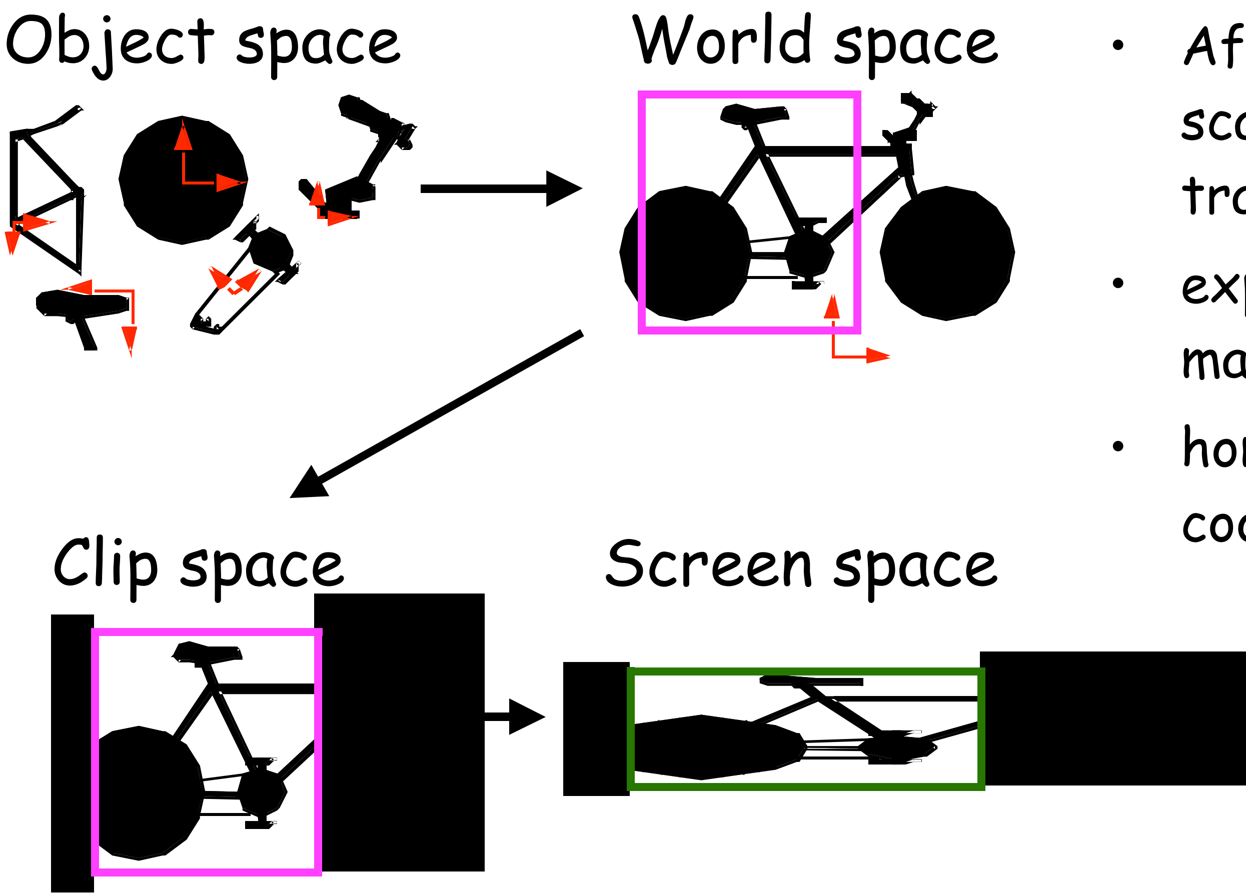
- Transform
- Classify
- Shading
- Interpolation
- Composite



Rendering Pipeline (RP)

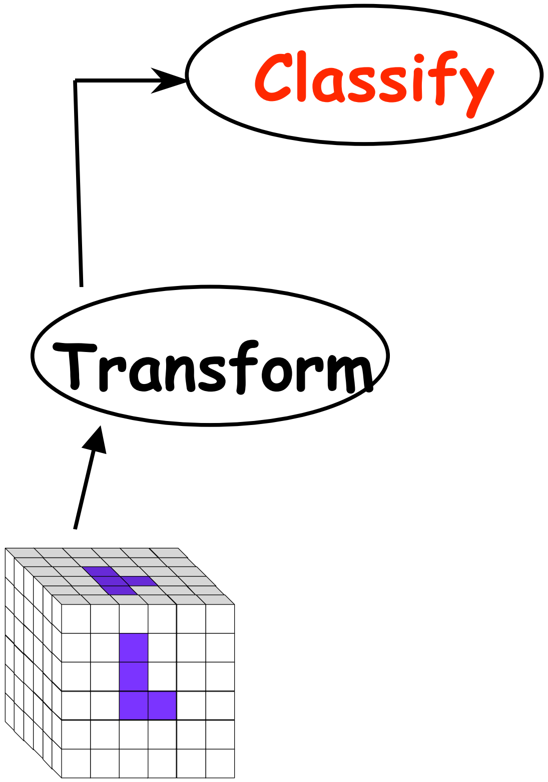


Transformation



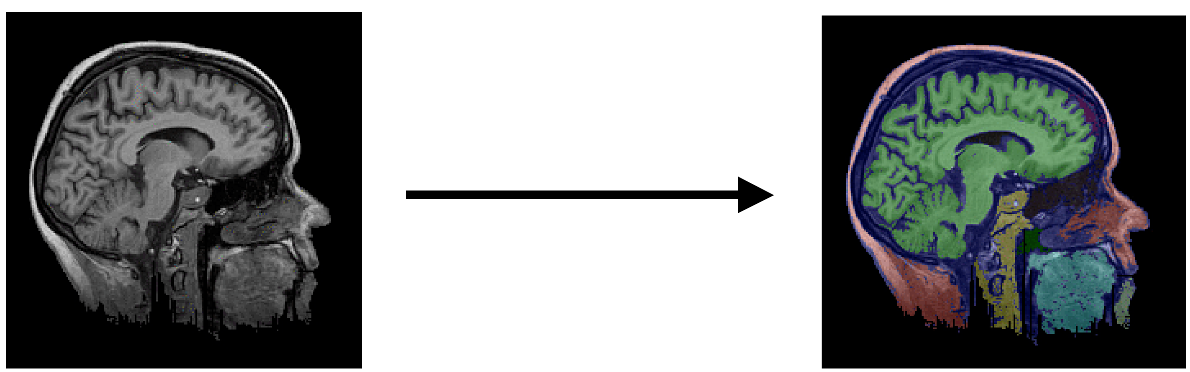
- Affine: rotate + scale + translate
- expressed in matrix form
- homogenous coordinates

Rendering Pipeline (RP)



Classification

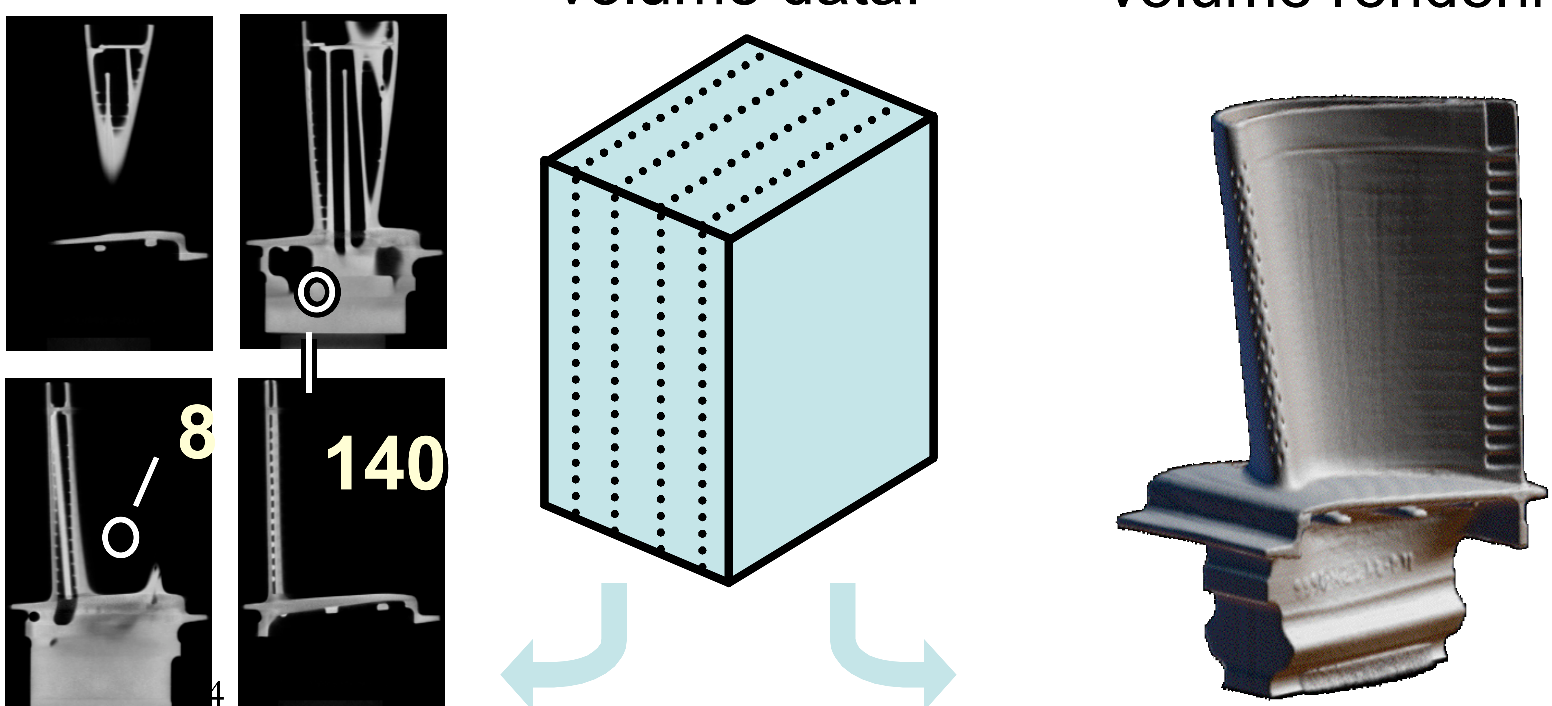
- original data set acquires application specific scalars/vectors (temperature, velocity, proton density etc.)
- make sense of / explore given data set
- assign material properties (surface graphics)
- assign color/opacity value, that guide visualization
- achieved through transfer functions



Transfer Functions (TF's)

- Transfer functions make volume data visible
- by mapping data values to optical properties

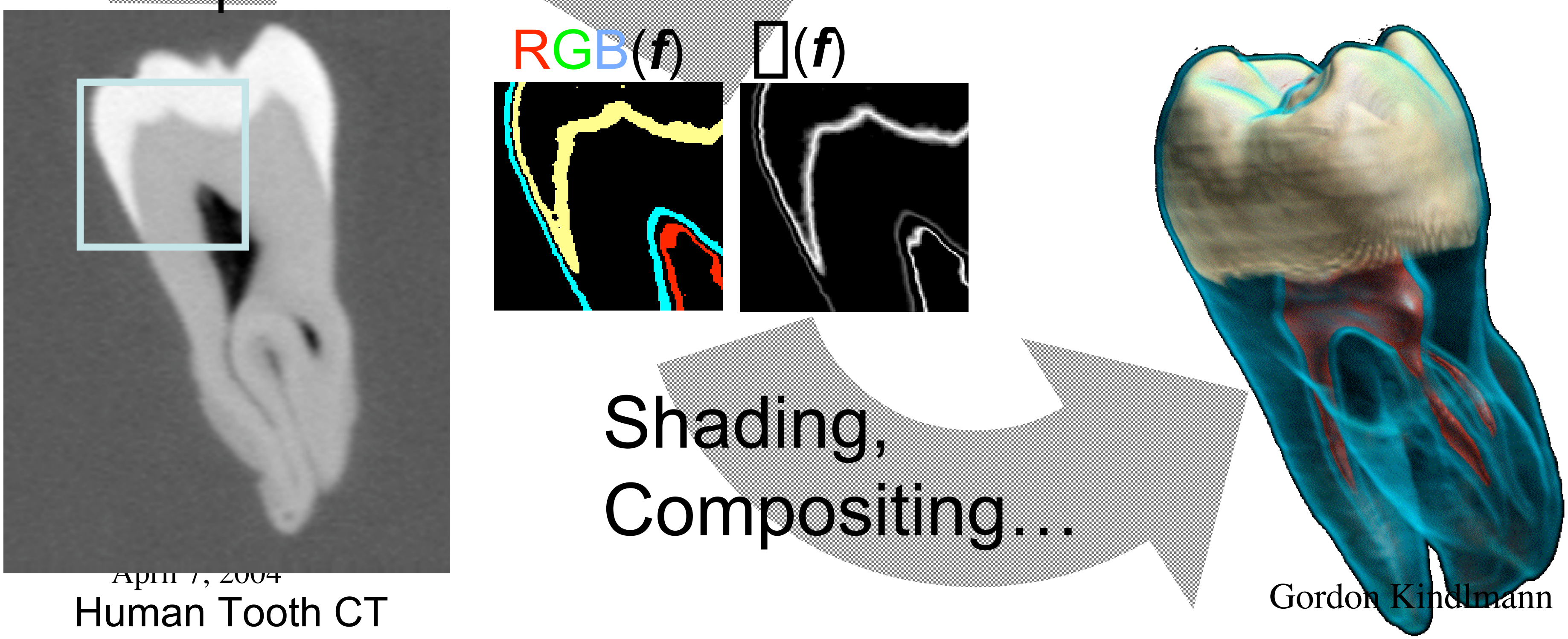
slices: volume data: volume rendering:



Gordon Kindlmann

Transfer Functions (TF's)

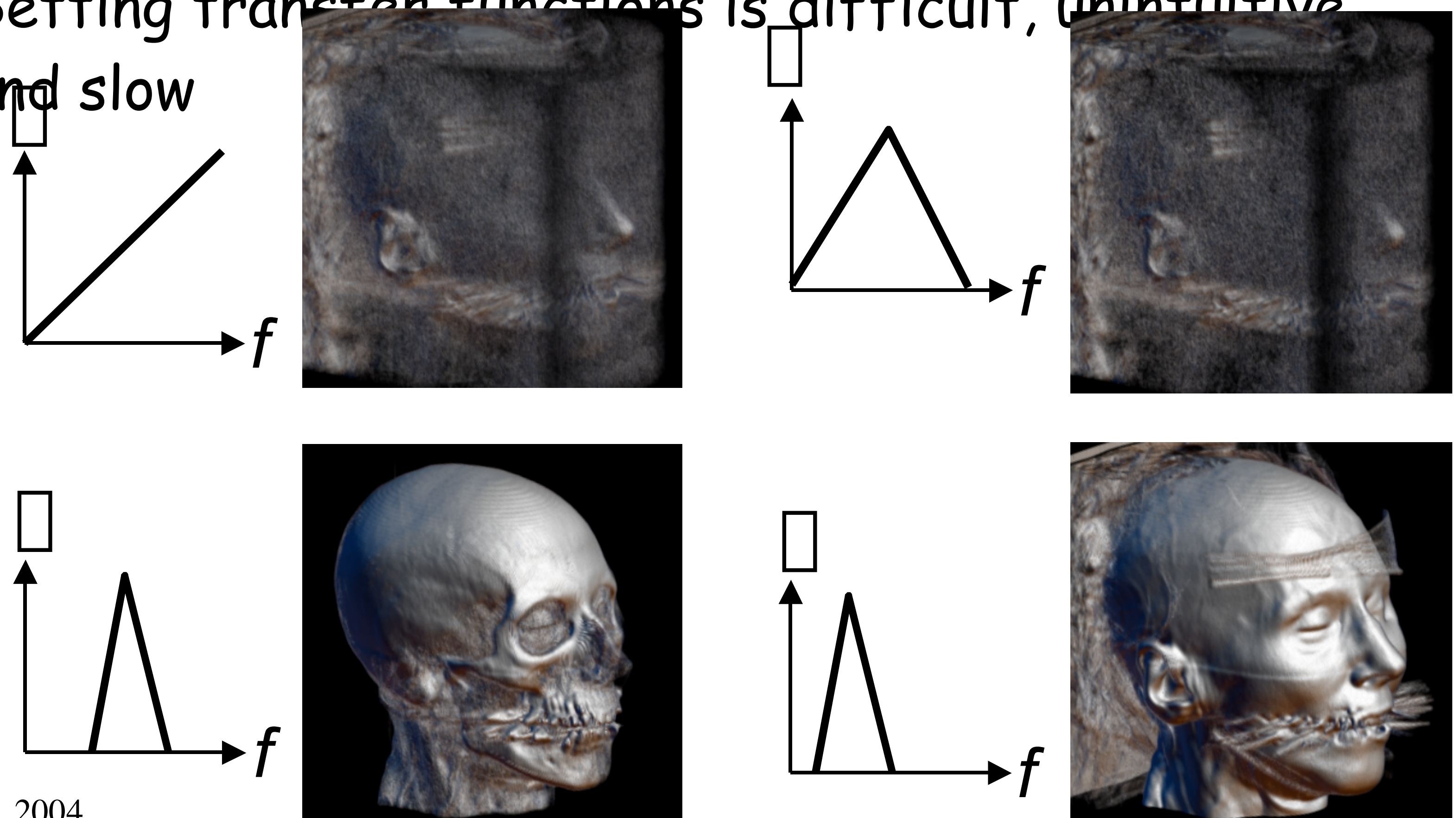
- Simple (usual) case: Map data value f to color and opacity



Human Tooth CT

TF's

- Setting transfer functions is difficult, unintuitive and slow



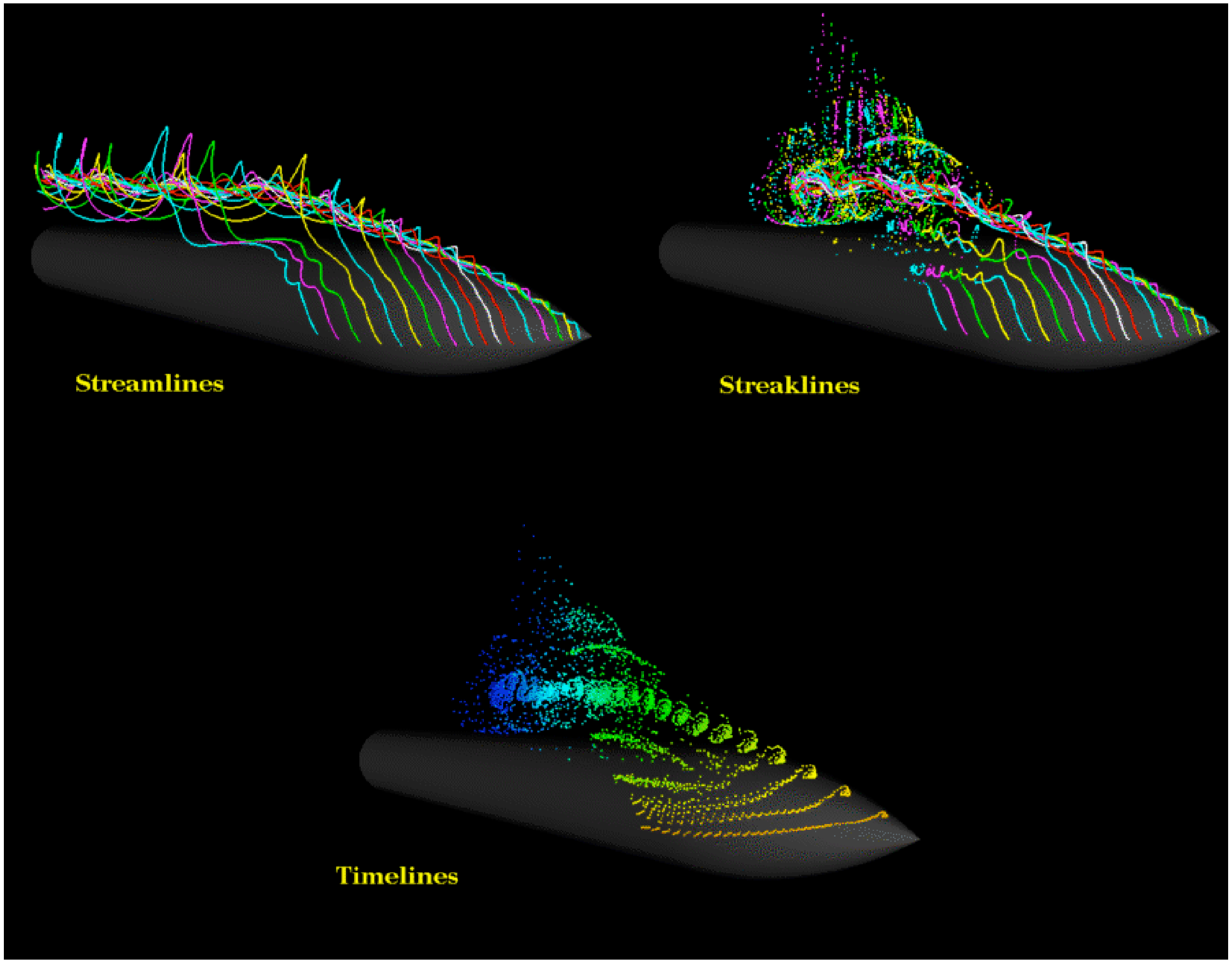
Goals

- Make good renderings easier to come by
- Make space of TFs less confusing
- Remove excess “flexibility”
- Provide one or more of:
  - Information
  - Guidance
  - Semi-automation
  - Automation

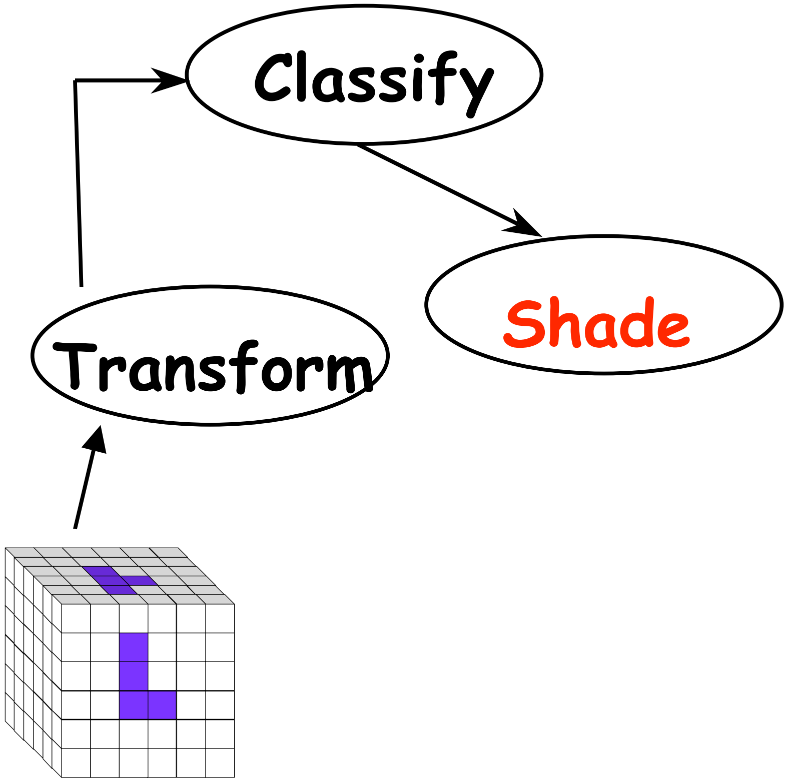
Classification - Vector

- Scalar data sets: typically color lookup
- Vector data sets: be creative
  - glyphs
  - streamlines / streaklines / particle methods
  - line bundles
  - spot noise
  - line integral convolution (LIC)

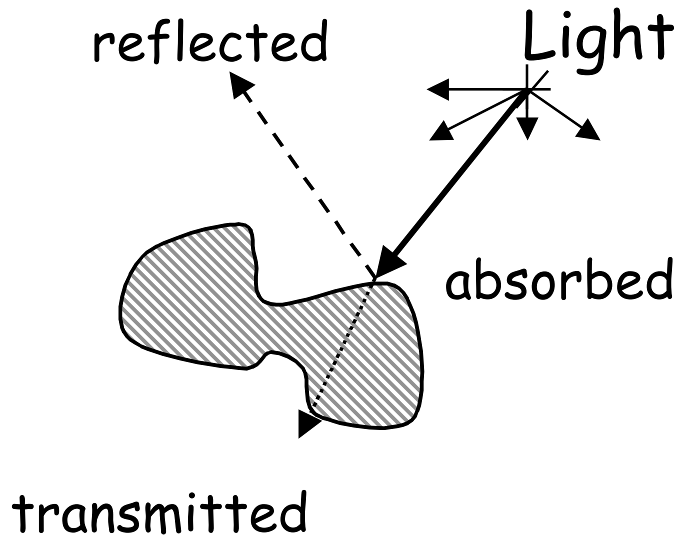
Classification - Vector (2)



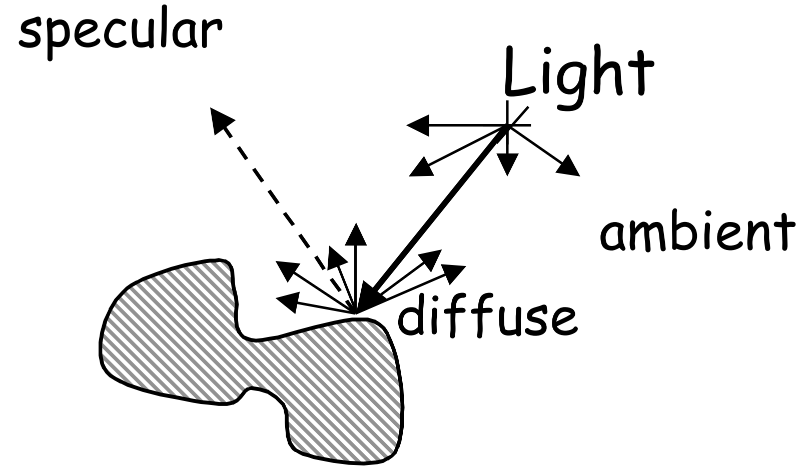
Rendering Pipeline (RP)



Light Effects



- Usually only considering reflected part

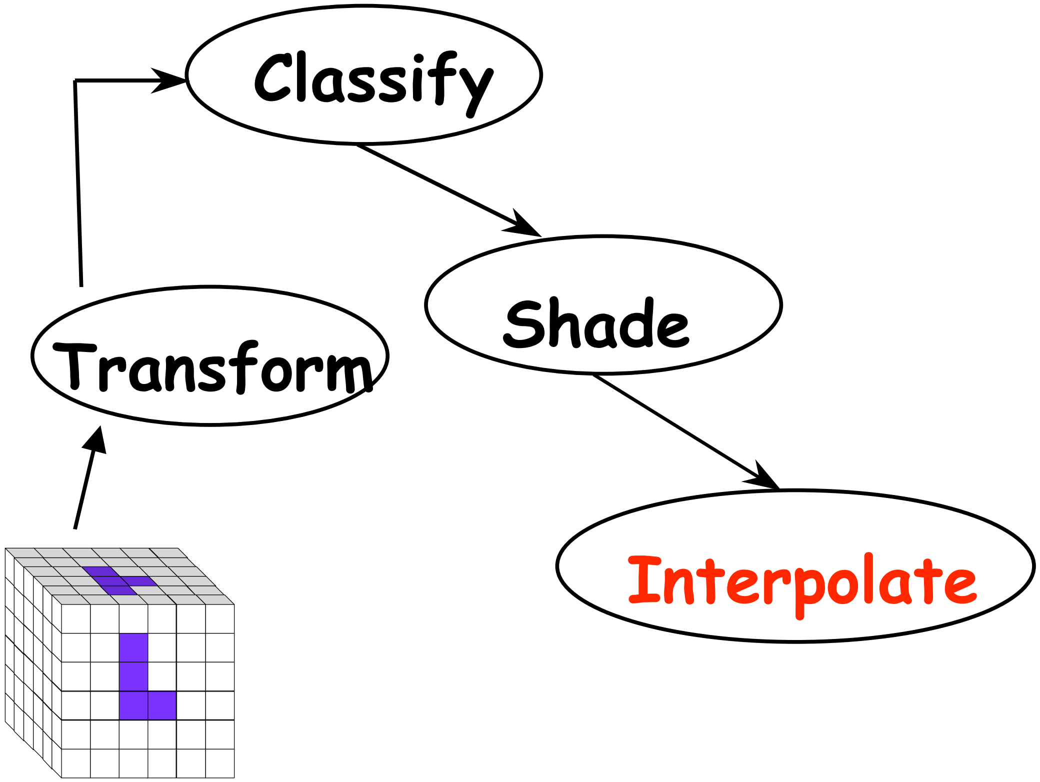


Light=refl.+absorbed+trans.

Light=ambient+diffuse+specular

$$I = k_a I_a + k_d I_d + k_s I_s$$

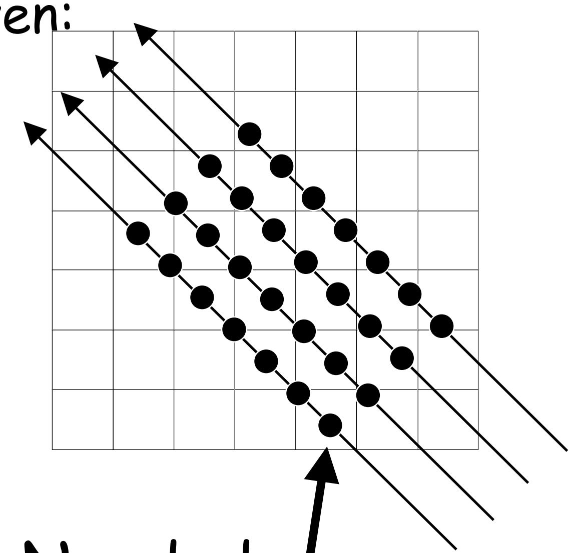
Rendering Pipeline (RP)



Interpolation

2D

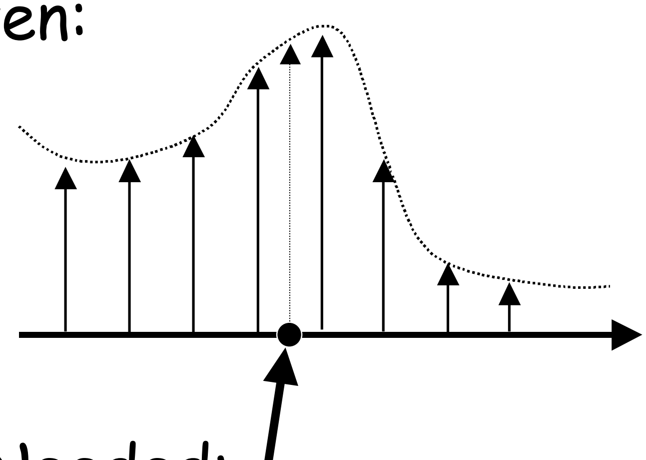
- Given:



- ✓ Needed:

1D

- ✓ Given:



- ✓ Needed:

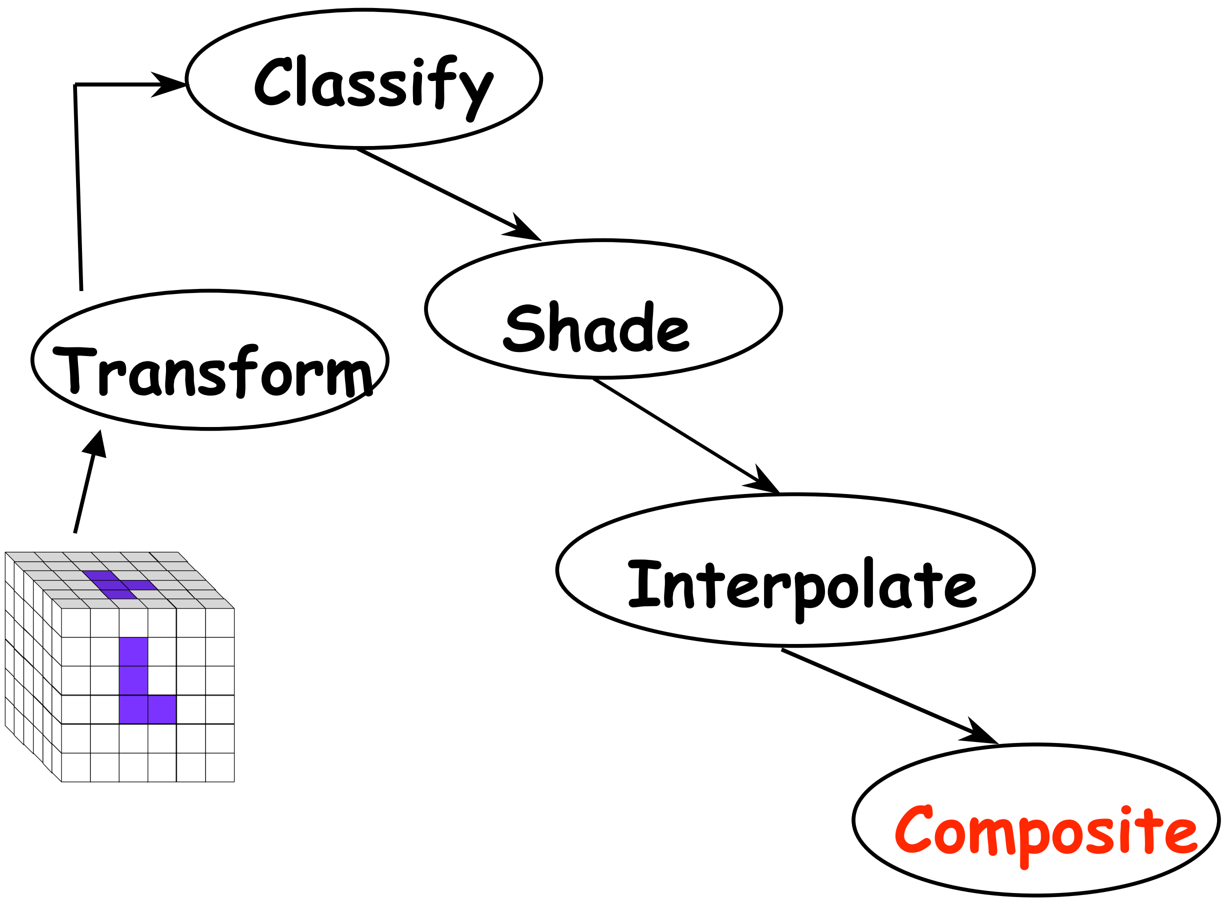
Interpolation (summary)

- Very important; regardless of algorithm
- expensive => done very often for one image
- Requirements for good reconstruction
  - performance
  - stability of the numerical algorithm
  - accuracy

Nearest neighbor

Linear

Rendering Pipeline (RP)



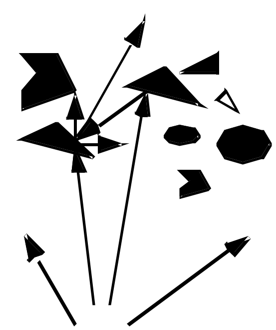
Semi -Transparent - How?

- Radiative transport theory
- model the interaction of light with the material

Transport of Light

observer

Emission(+)
Scattering(+)
Absorption(-)



light

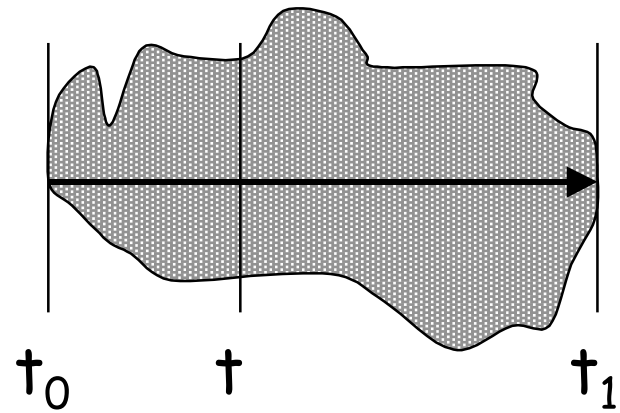
Semi -Transparent - How?

- Rendering Integral (Sabella, Max, …)

$$I(t) = \int_{t_0}^{t} c(s)\, e^{-\tau(s)}\, ds$$

$$\tau(s) = k \int_{t_0}^{s} \rho(u)\, du$$

✓ Discretize Integral!!

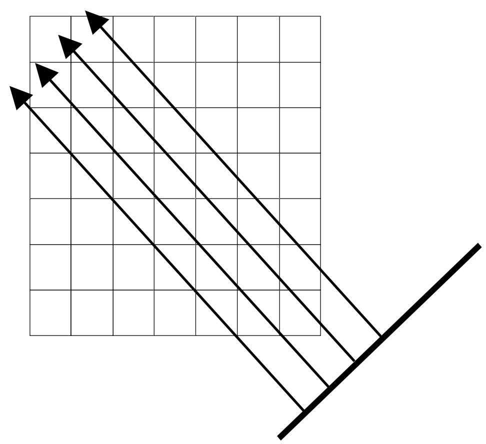
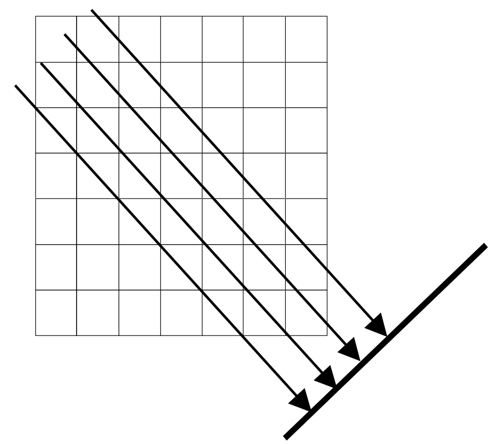
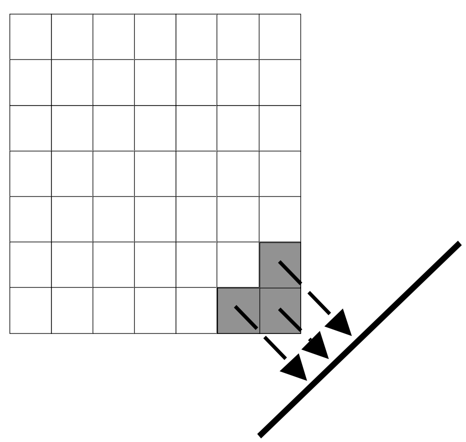
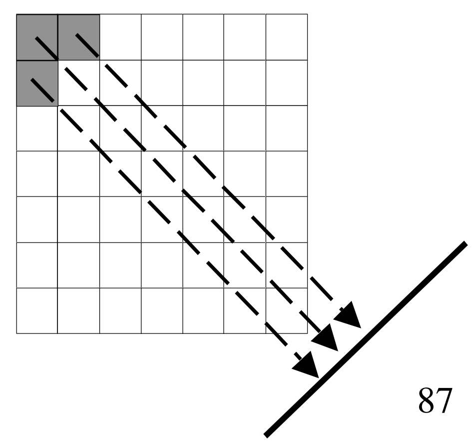


C(t): shade

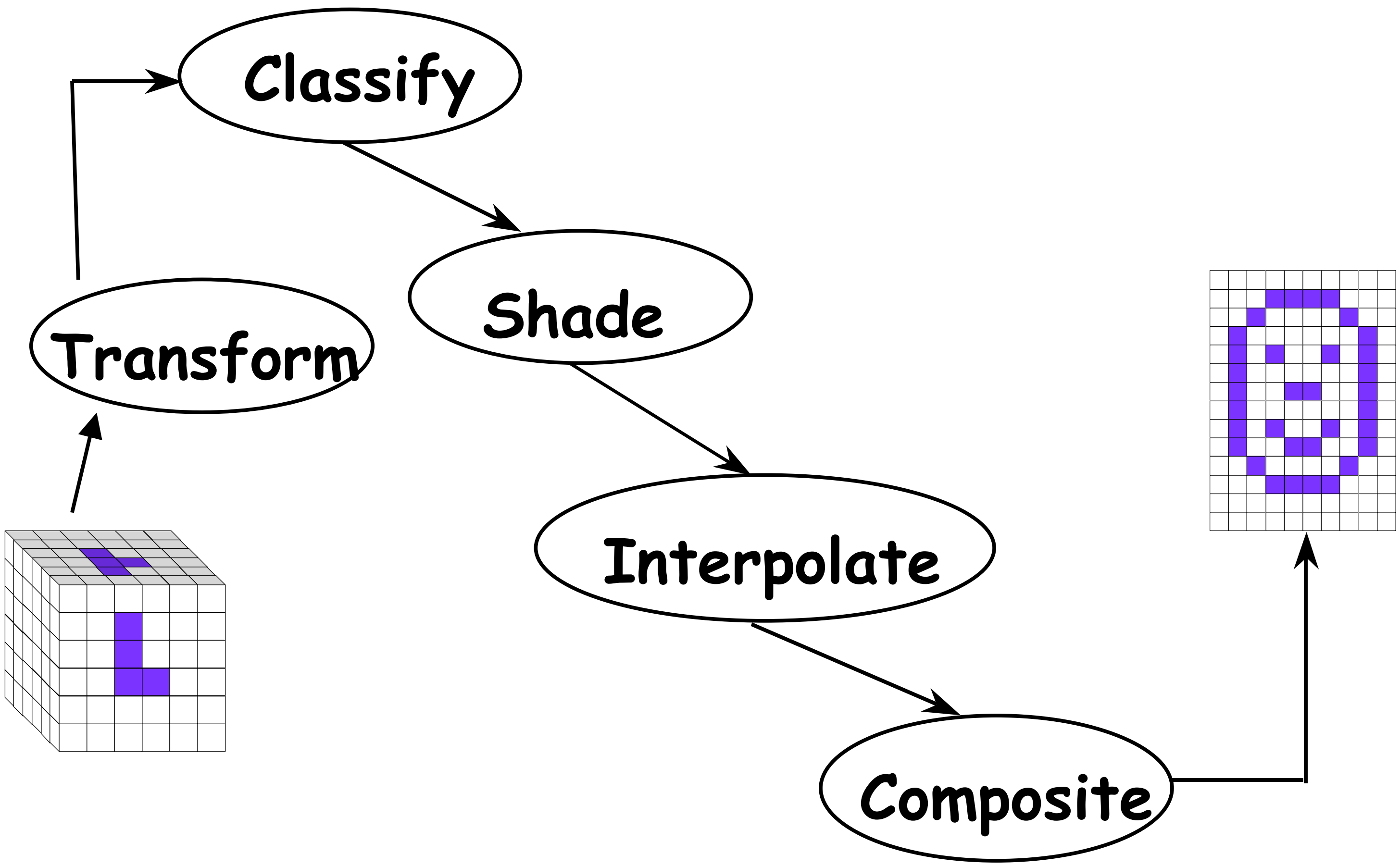
τ(t): opacity

ρ(t): “density”

Compositing (2)

	Front-To-Back	Back-To-Front
Image order		
Object order		

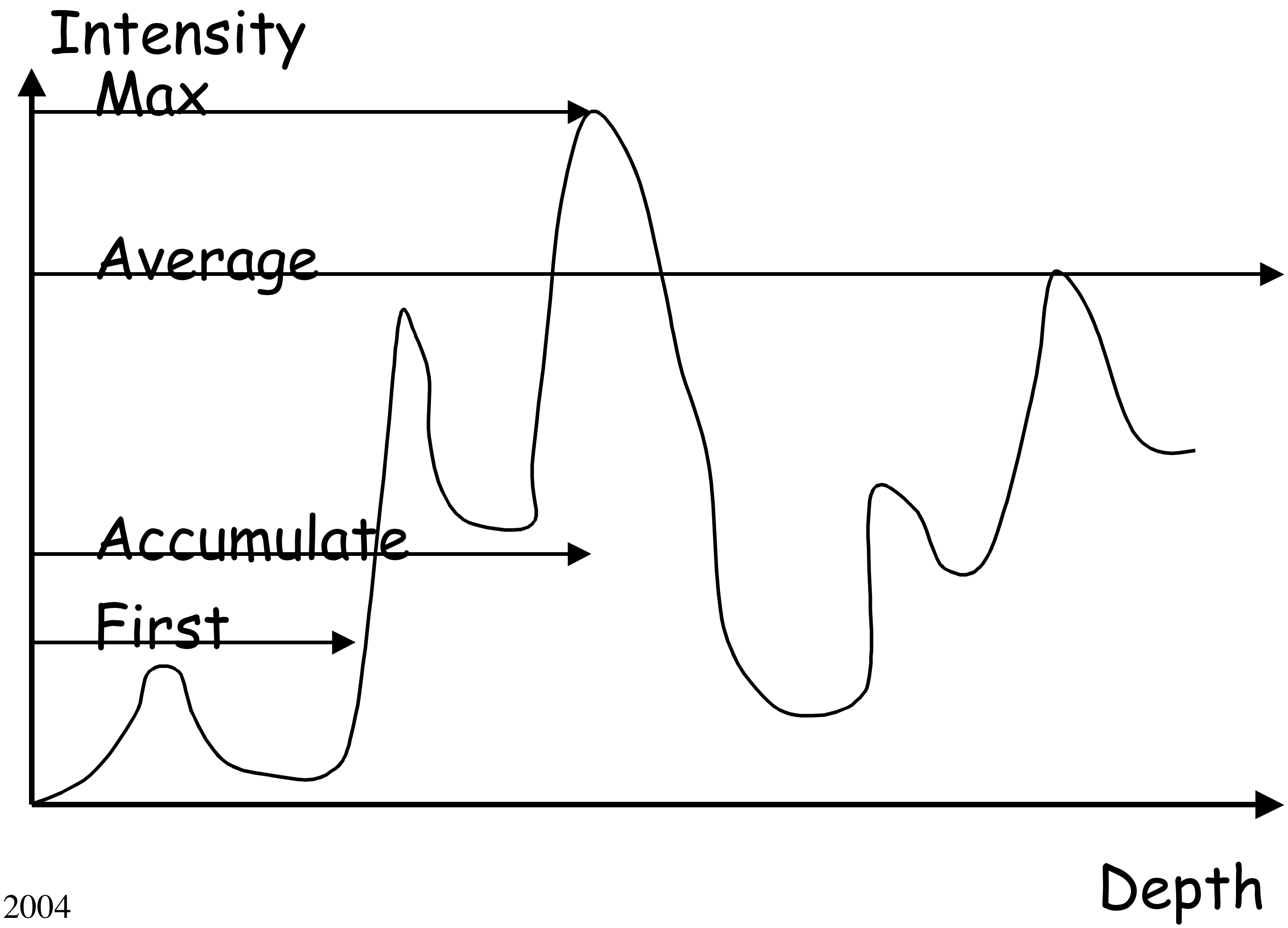
Rendering Pipeline (RP)



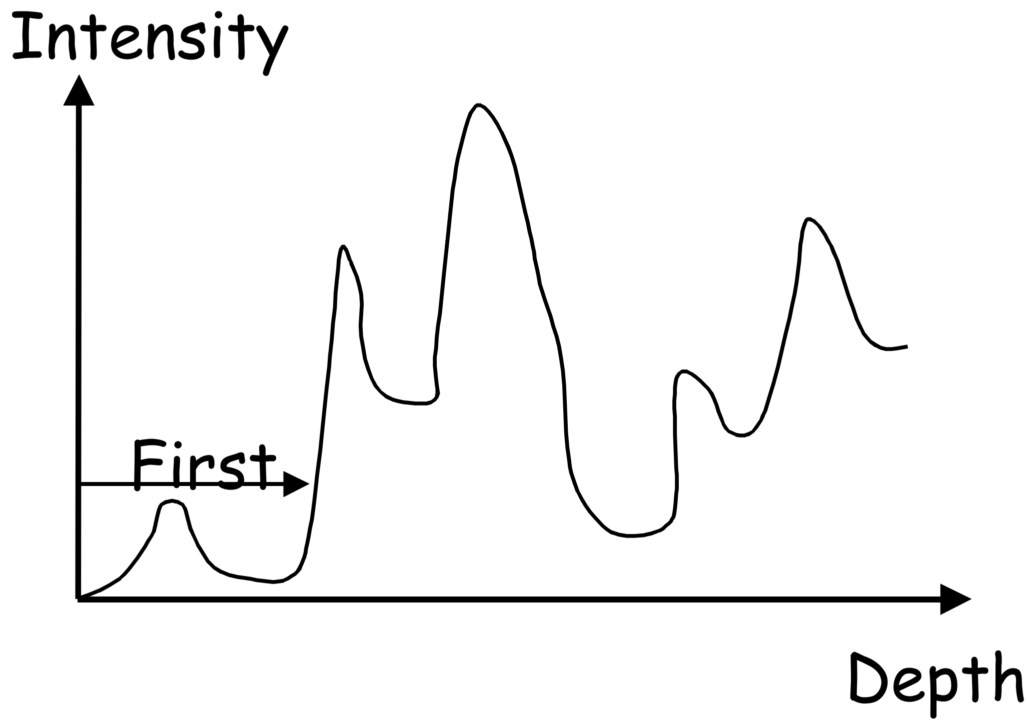
Ray Tracing

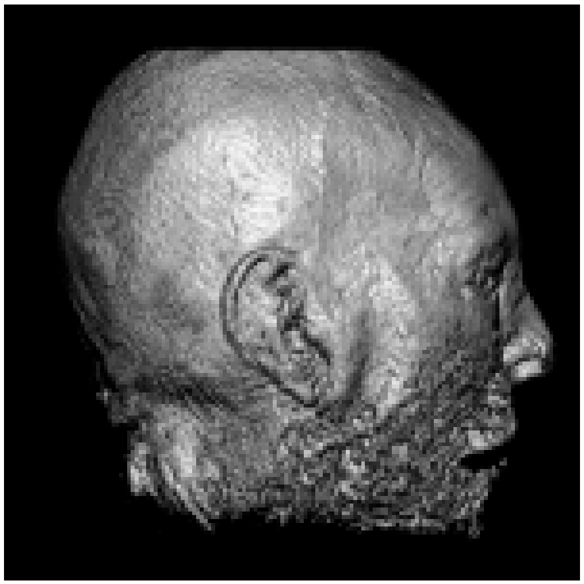
- "another" typical method from traditional graphics
- Typically we only deal with primary rays - hence: **ray-casting**
- a natural image-order technique
- as opposed to surface graphics - how do we calculate the ray/surface intersection???
- Since we have no surfaces - we need to carefully step through the volume

Ray Traversal Schemes



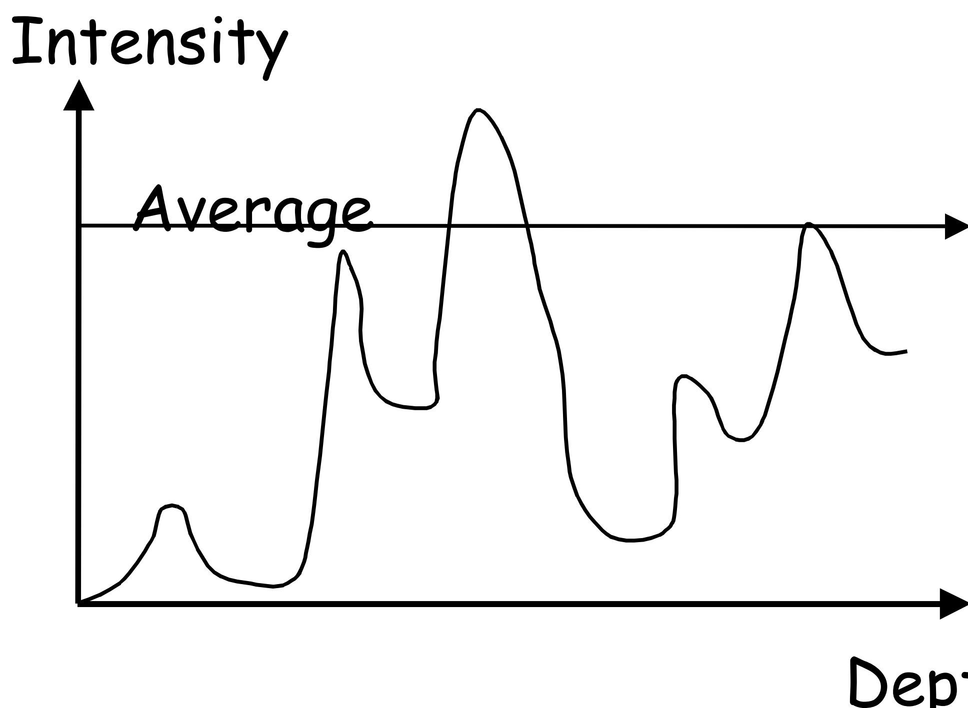
Ray Traversal - First

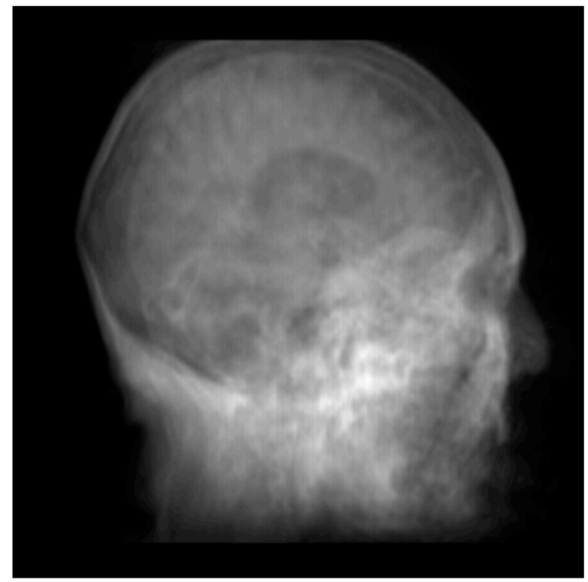




- **First**: extracts iso-surfaces (again!) done by Tuy&Tuy '84

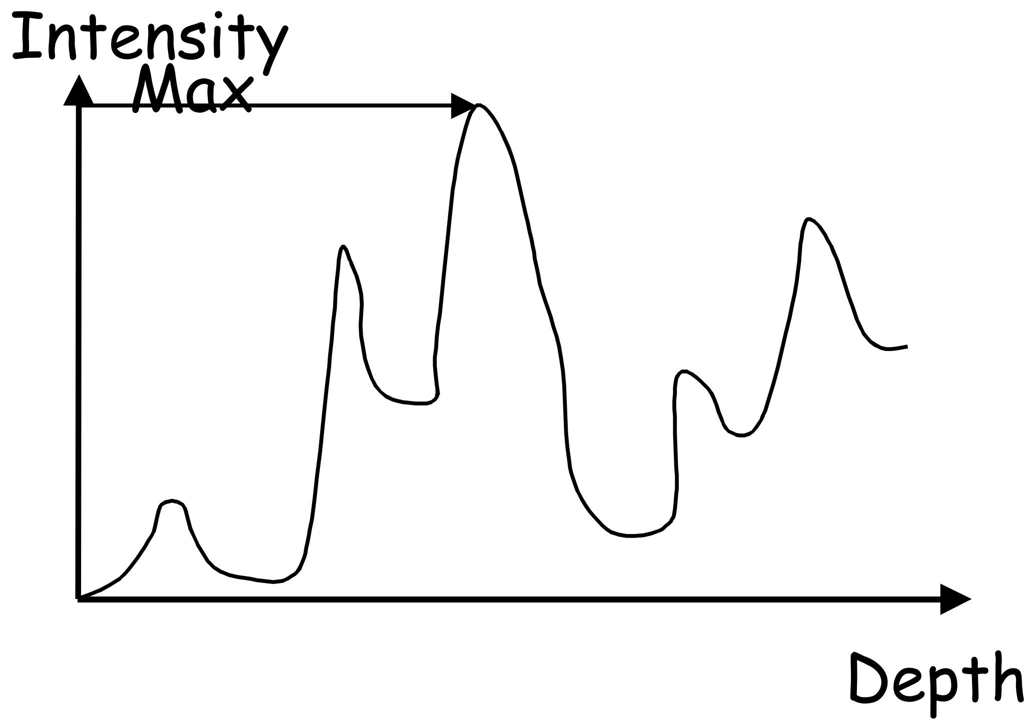
Ray Traversal - Average

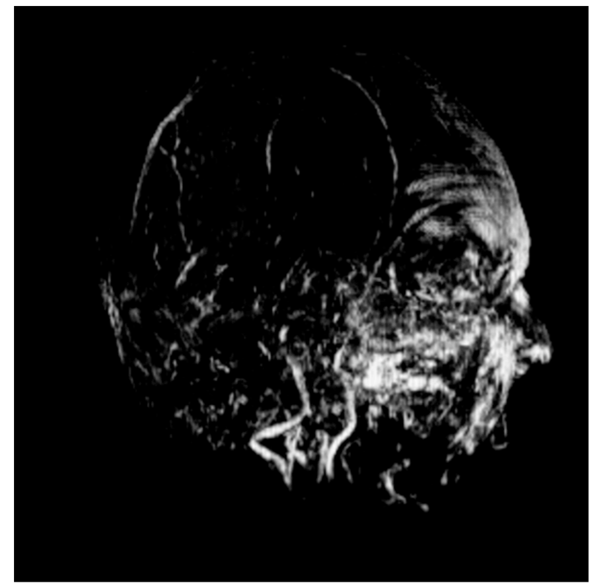




- **Average**: produces basically an X-ray picture

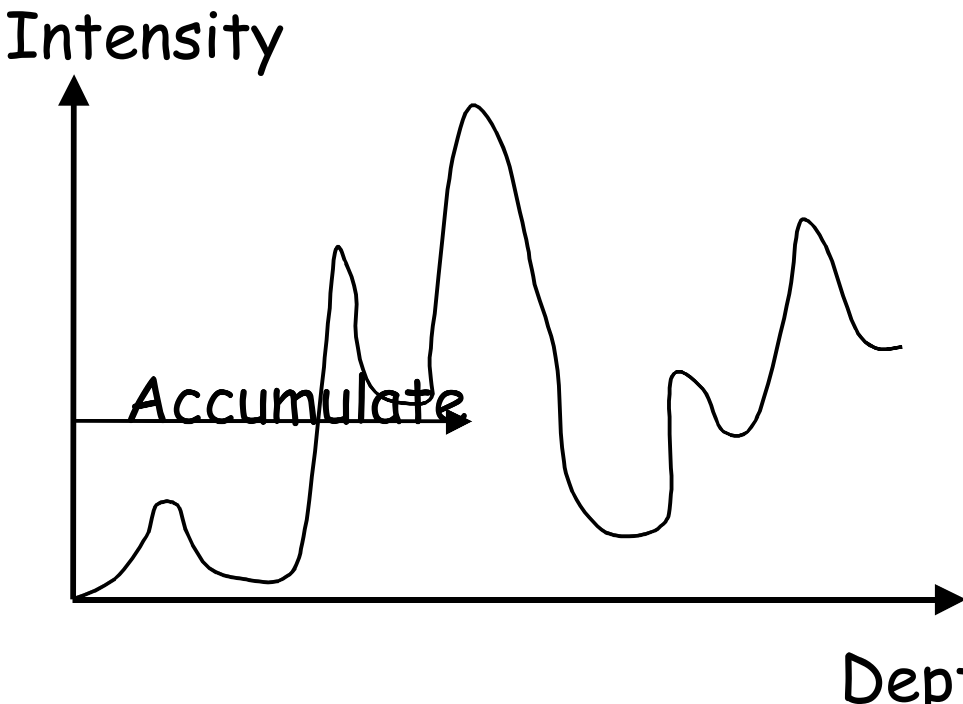
Ray Traversal - MIP

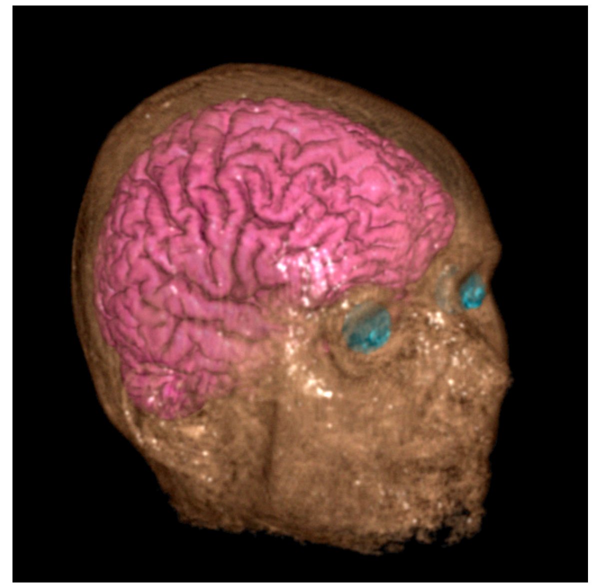




- **Max**: Maximum Intensity Projection used for Magnetic Resonance Angiogram

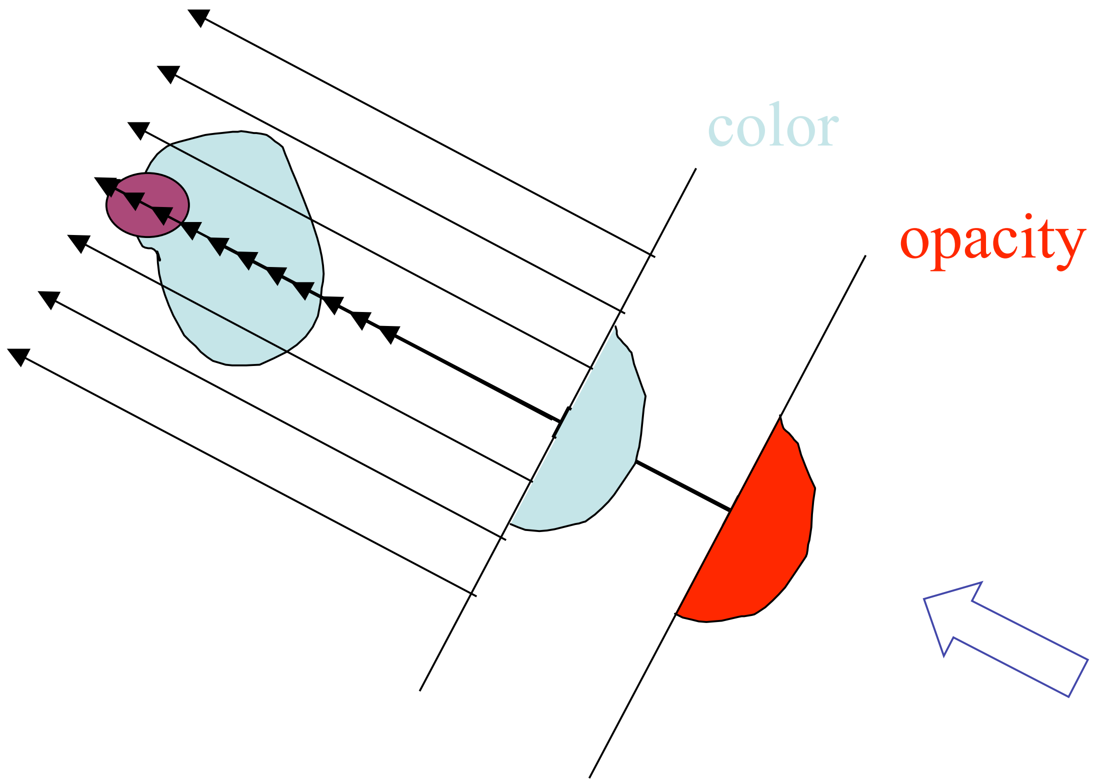
Ray Traversal - Accumulate





- **Accumulate**: make transparent layers visible!
  Levoy '88

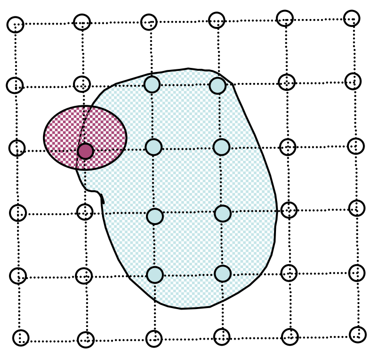
Volumetric Ray Integration

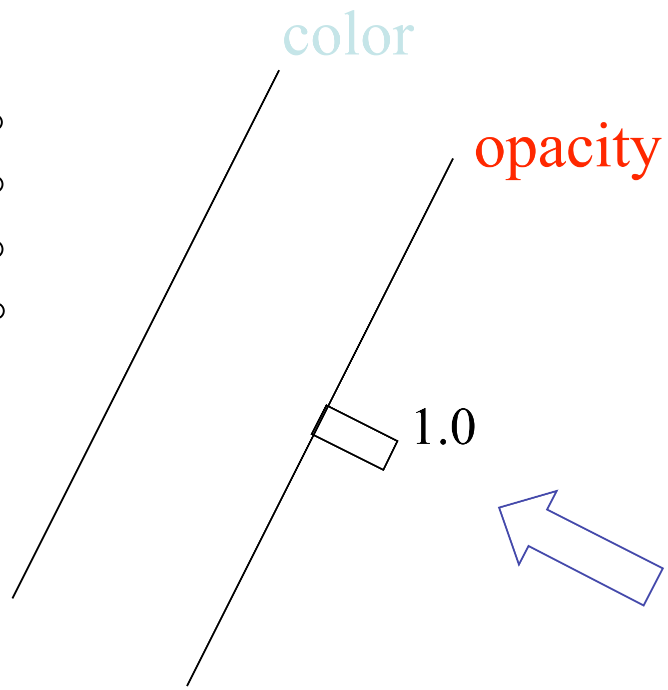


object (color, opacity)

Raycasting

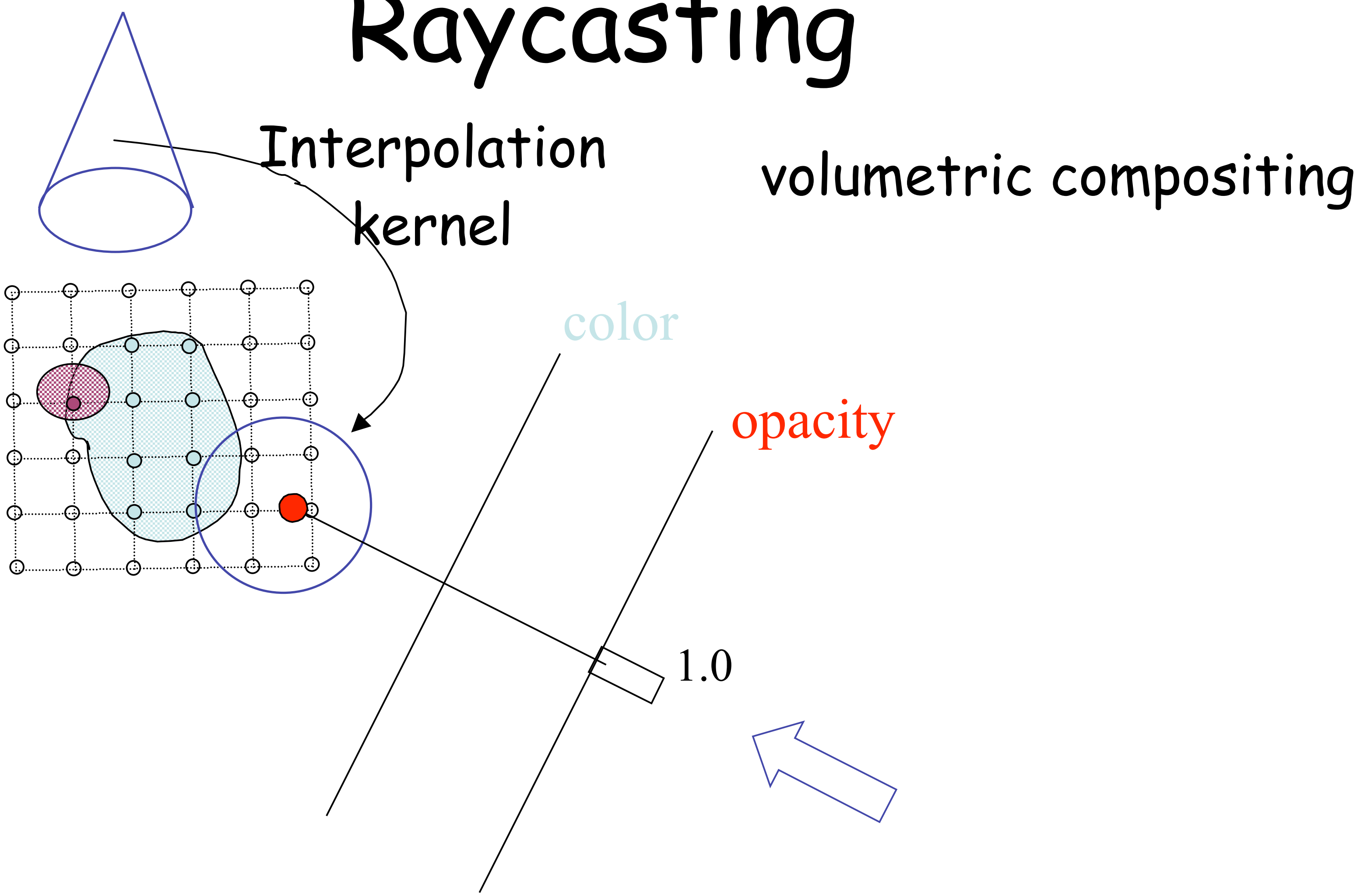
volumetric compositing





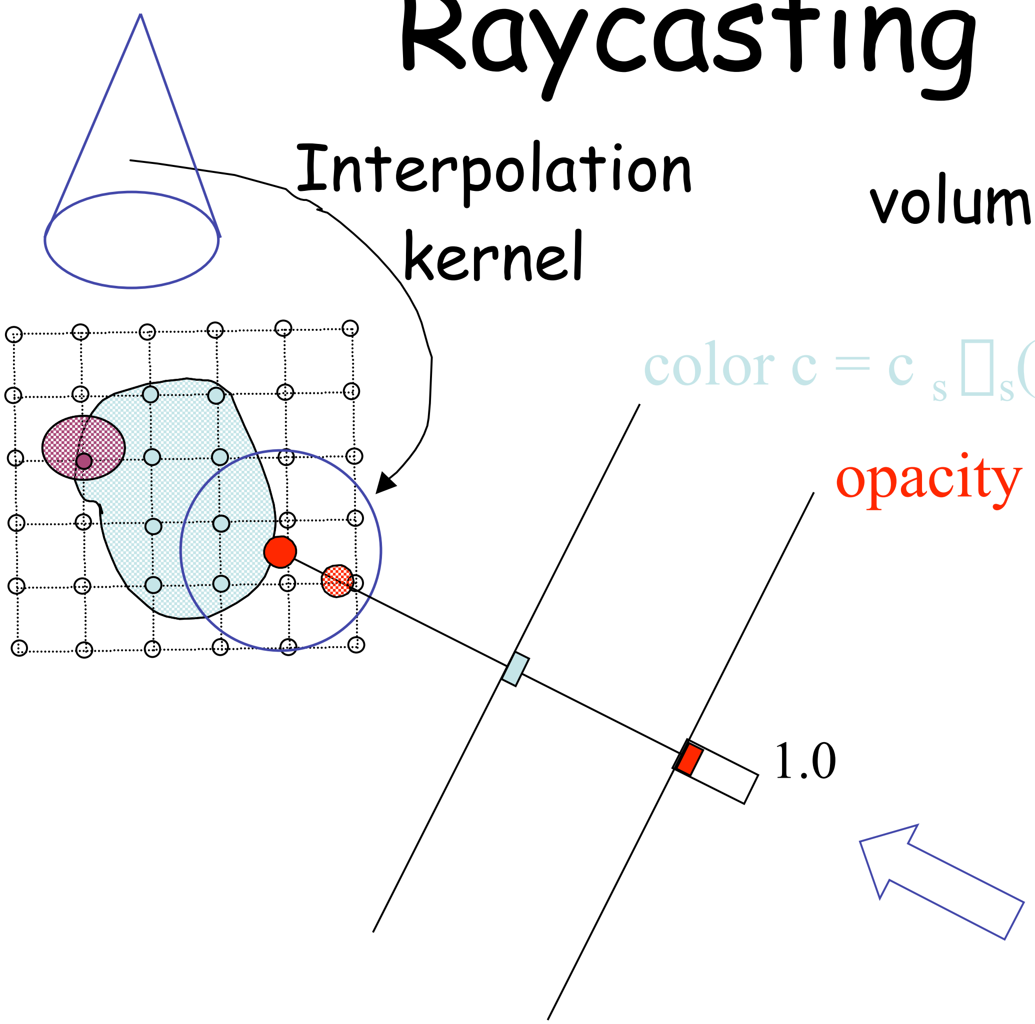
object (color, opacity)

Raycasting



object (color, opacity)

Raycasting



volumetric compositing

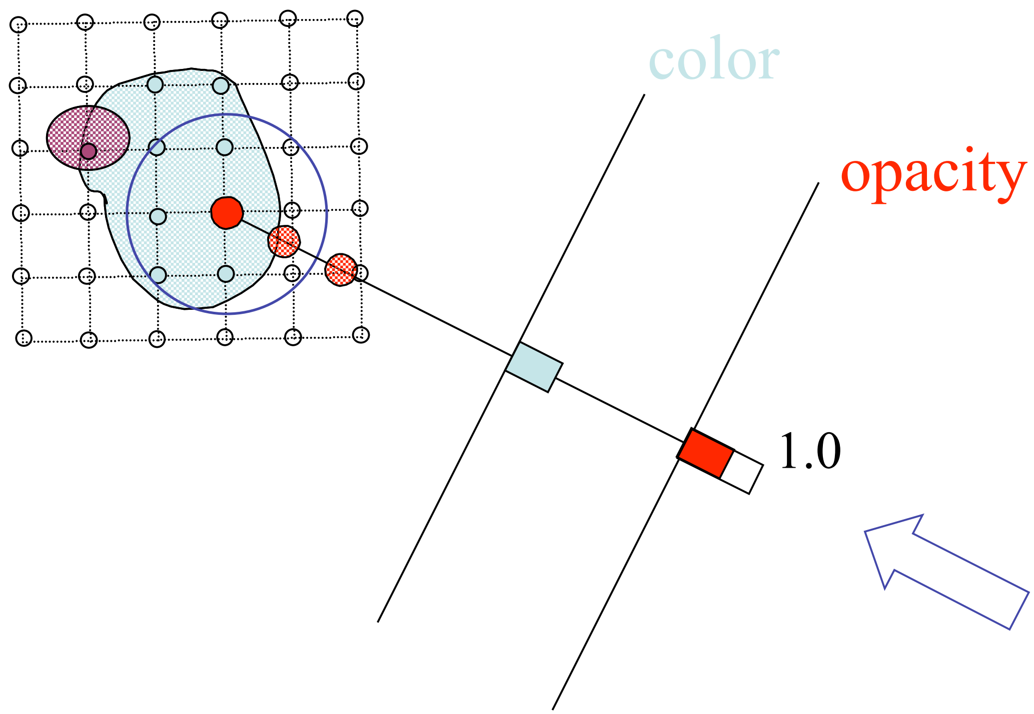
color $c = c_s \alpha_s (1 - \alpha) + c$

opacity $\alpha = \alpha_s (1 - \alpha) + \alpha$

object (color, opacity)

Raycasting

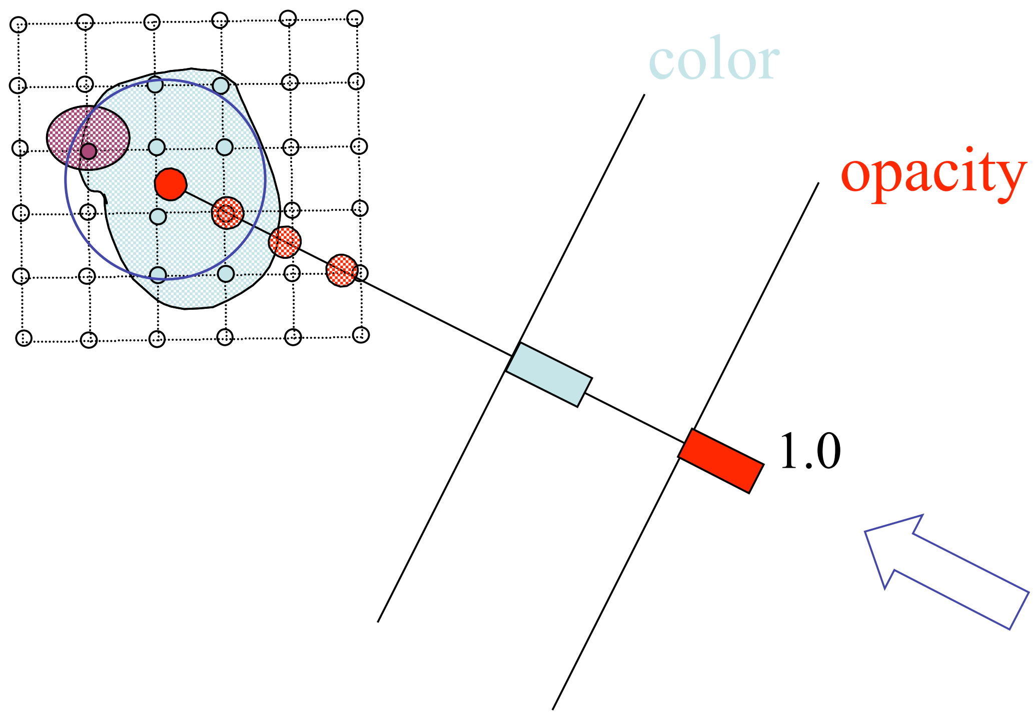
volumetric compositing



object (color, opacity)

Raycasting

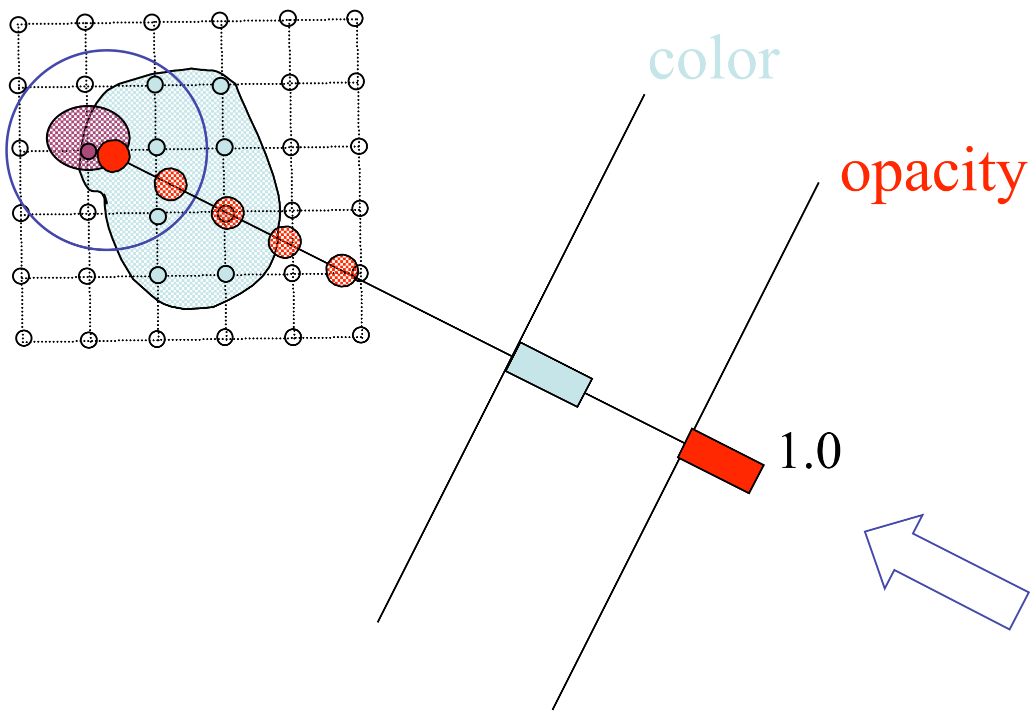
volumetric compositing



object (color, opacity)

Raycasting

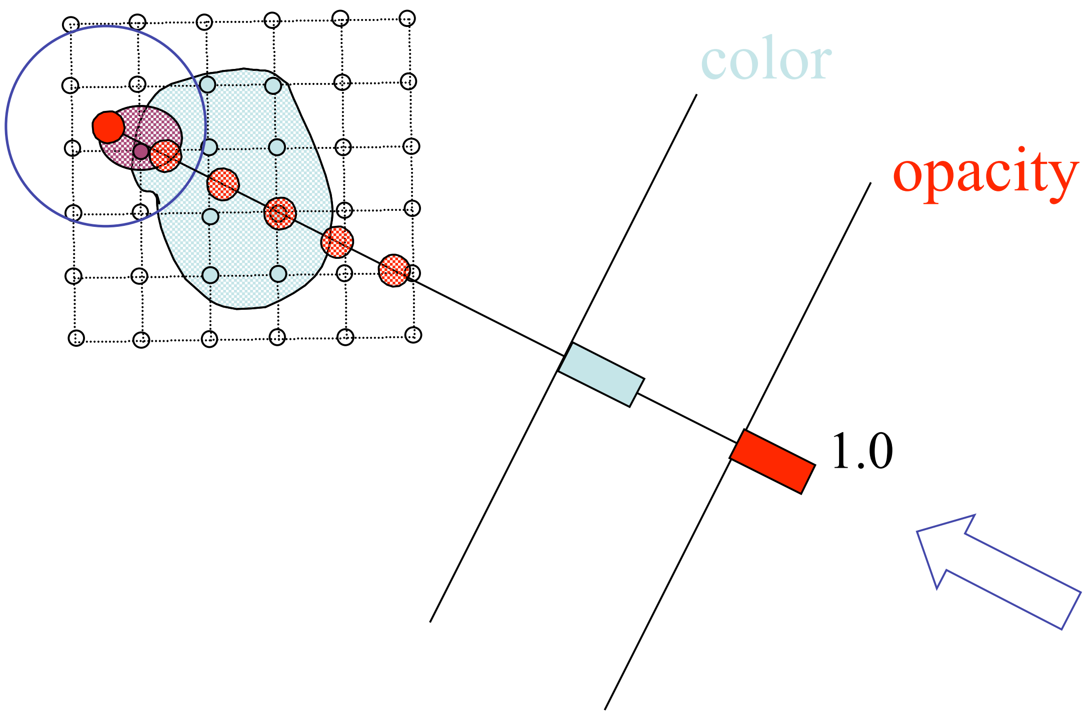
volumetric compositing



object (color, opacity)

Raycasting

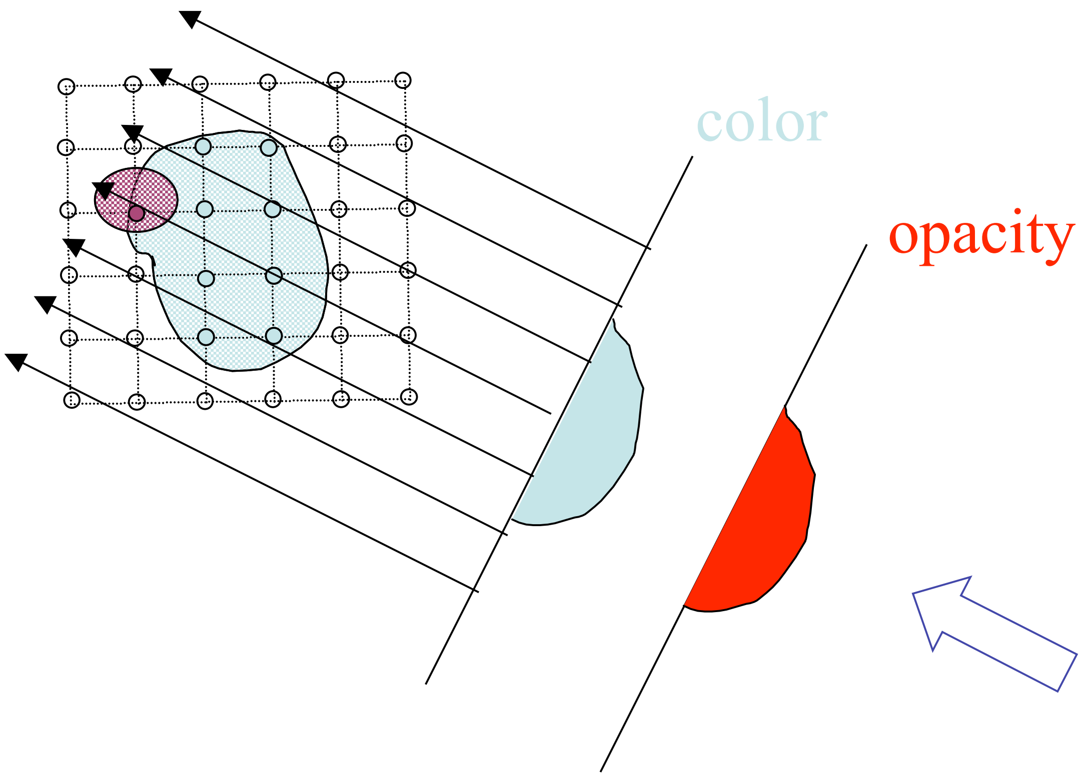
volumetric compositing



object (color, opacity)

Raycasting

volumetric compositing



object (color, opacity)

Overview

- What is SciVis?
- Acquisition Methods
- Iso-surfaces
- Direct-Rendering Pipeline
- **Vector Visualization**
- Challenges

Flow Visualization – traditionally

- Traditionally - Experimental Flow Vis
- purpose:
  - get an impression of flow around a scale model of a real object
  - as a source of inspiration for the development of new and better theories
  - to verify a new theory or model

video

Flow Visualization - How

- How is it done?
- three basic techniques:
  - adding foreign material
  - optical techniques
  - adding heat and energy

Flow Visualization - Examples

Flow Visualization - add material

- Time Lines:
- lines, that once released in the fluid, are moved and transformed by the flow. The motion and formation of the line, which is often released perpendicular to the flow, shows the flow.
- Practice - often consist of row of small particles, such as hydrogen bubbles.

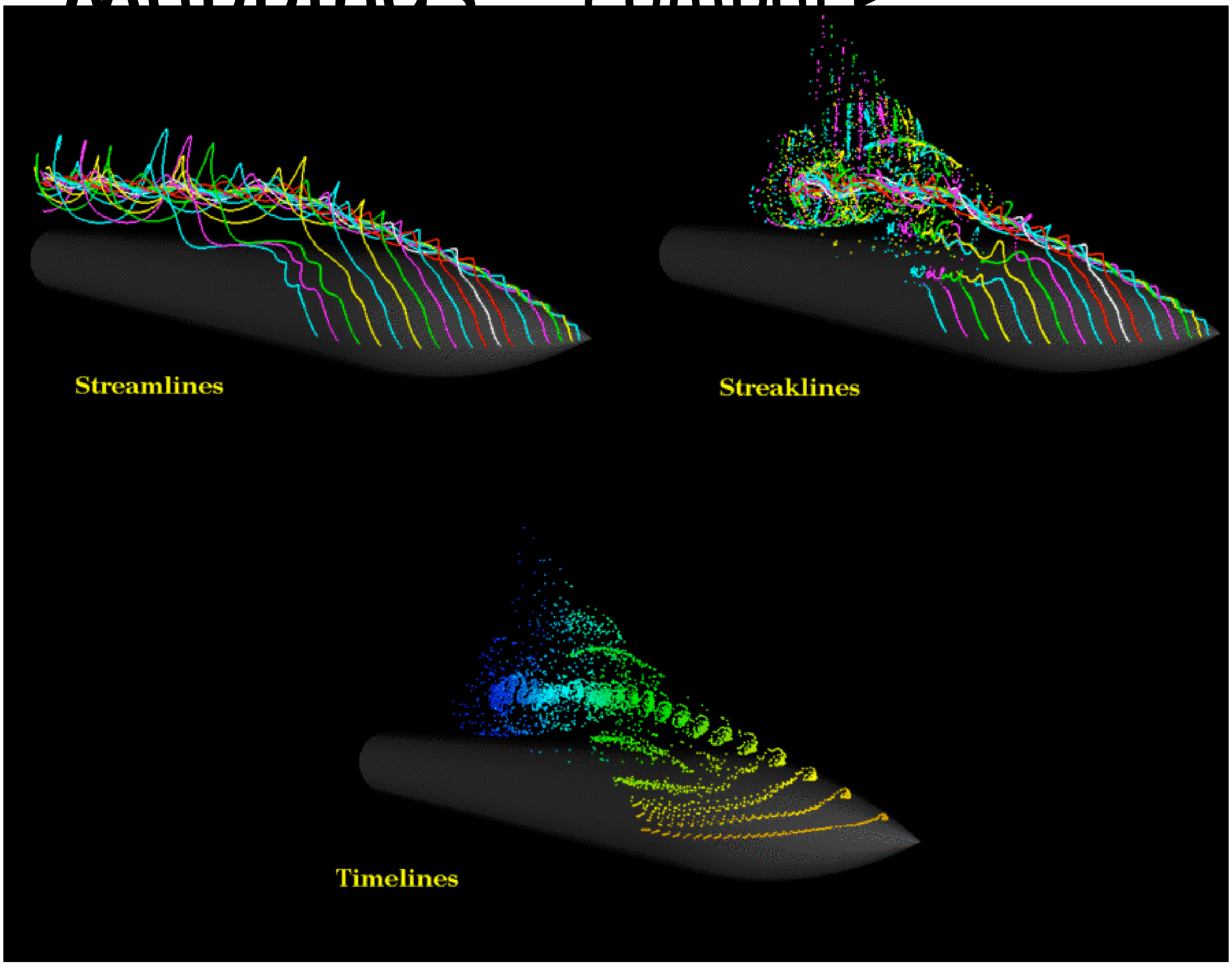
Flow Visualization – add material

- Streak Lines:
- arises when dye is injected in the flow from a fixed position.
- Practice -Injecting the dye for a period of time gives a line of dye in the fluid, from which the fluid flow can be seen.

Flow Visualization - add material

- Path Lines or Streamlines:
- is the path of a (massless) particle in the fluid. Imagine a light emitting particle in the flow. *A* path line is obtained when a photographic plate is exposed for several seconds.
- Steady flows - path and streak lines are identical to stream lines - lines that are everywhere tangent to the velocity field.

Mappings - compare

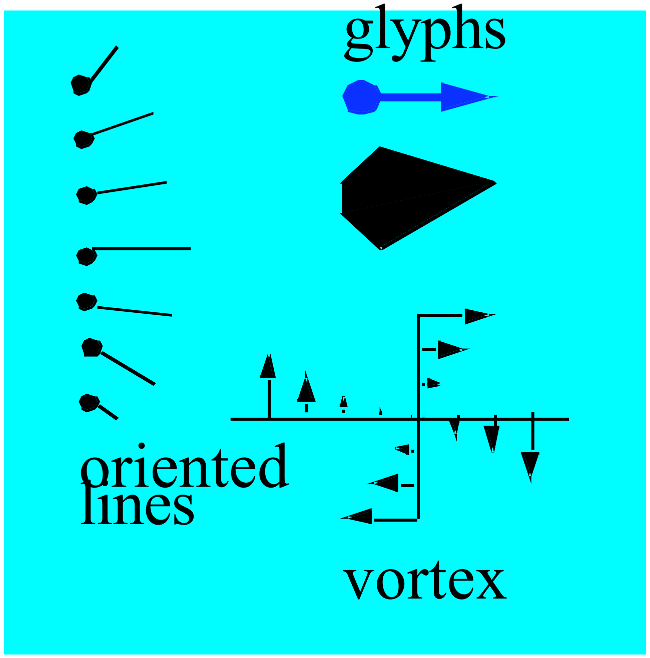


Flow Visualization - add material

- Flow on a surface:
- fix tufts (small threads) at several points on the surface or
- coat the surface with viscous material (oil)

Mappings - Hedgehogs, Glyphs

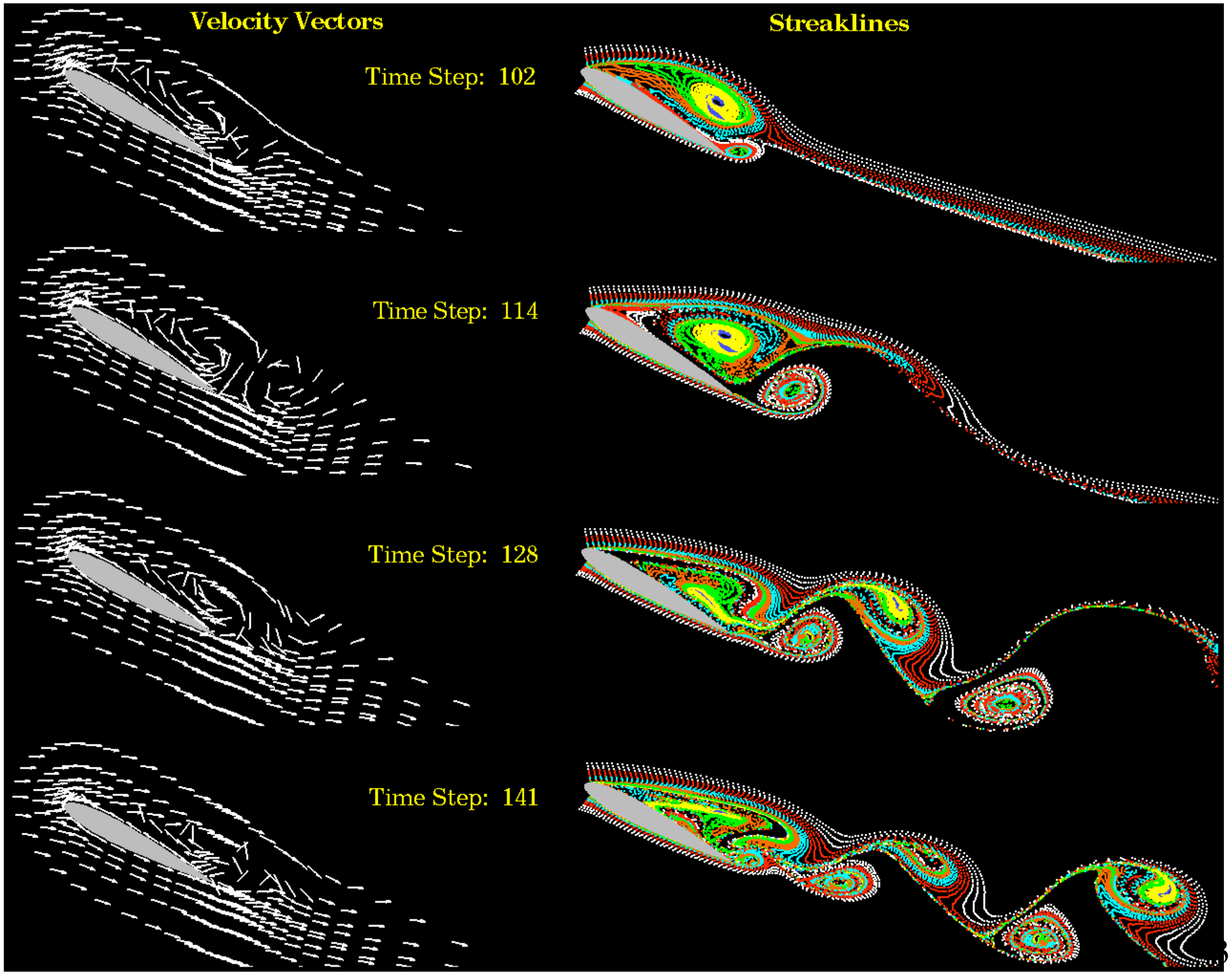
- Put "icons" at certain places in the flow
- e.g. arrows - represent direction & magnitude
- other primitives are possible



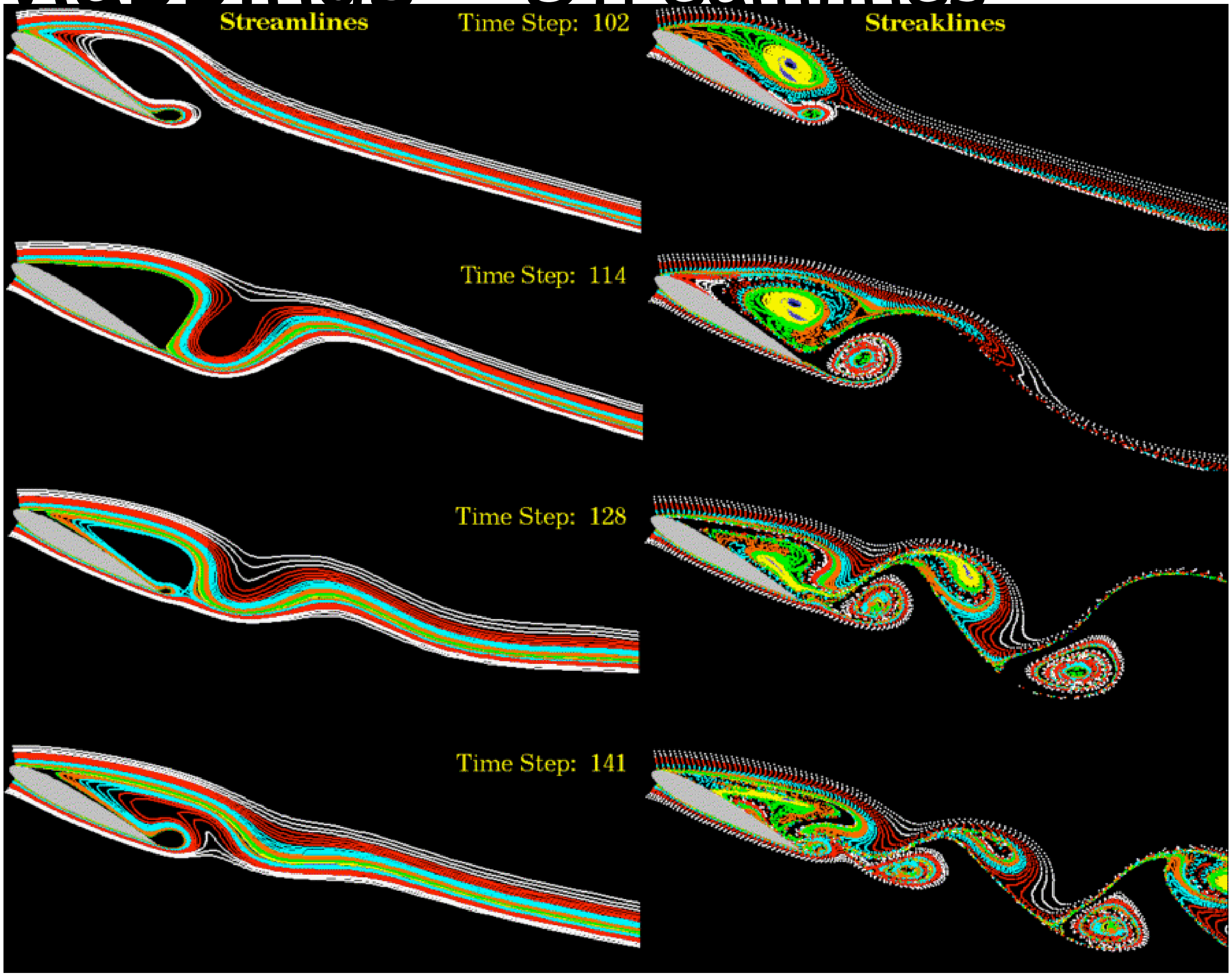
Mappings - Hedgehogs, Glyphs

- analogous to tufts or vanes from experimental flow visualization
- clutter the image real quick
- maybe ok for 2D
- not very informative

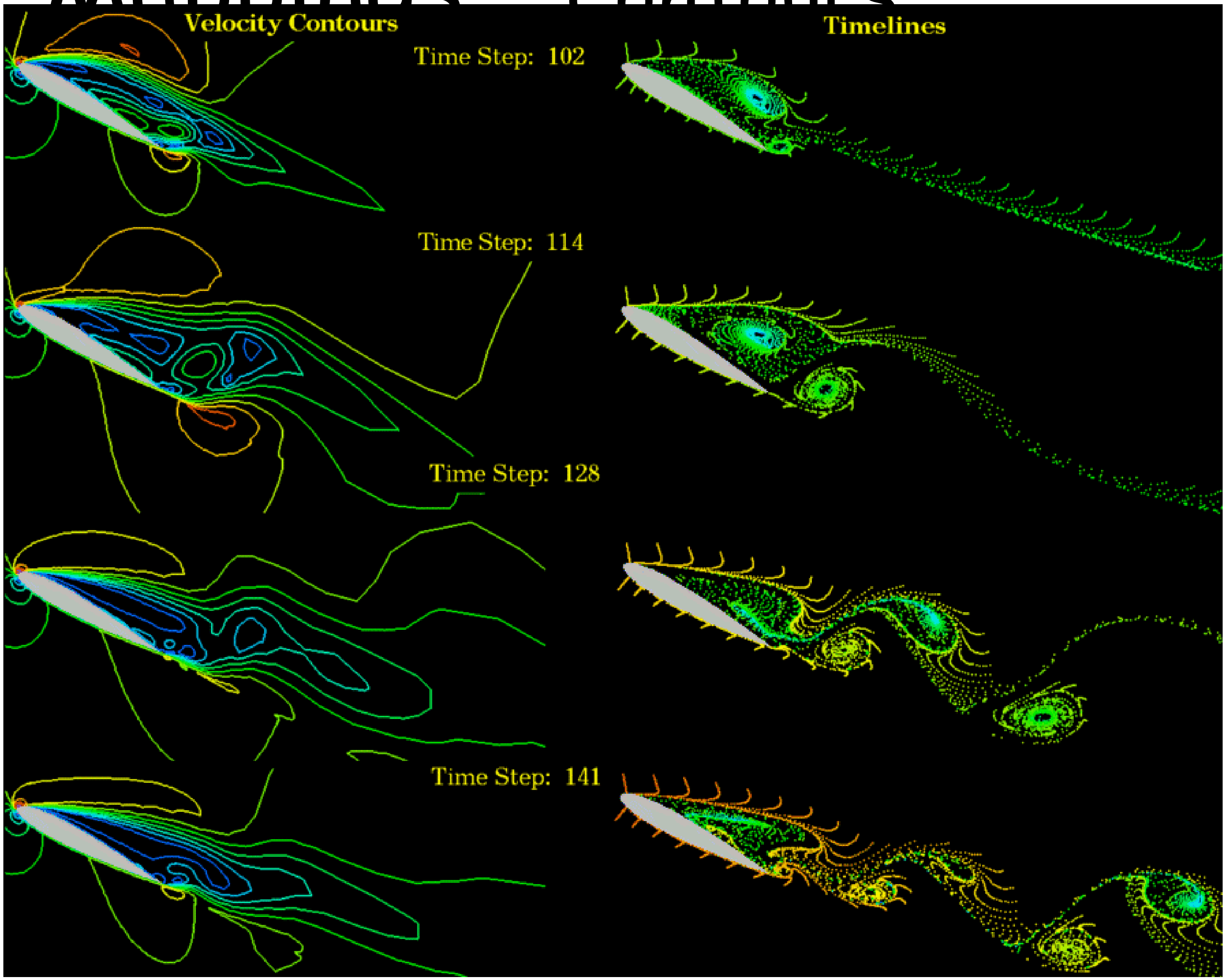
Mappings - Streak-lines



Mappings - Streamlines



Mappings - Contours

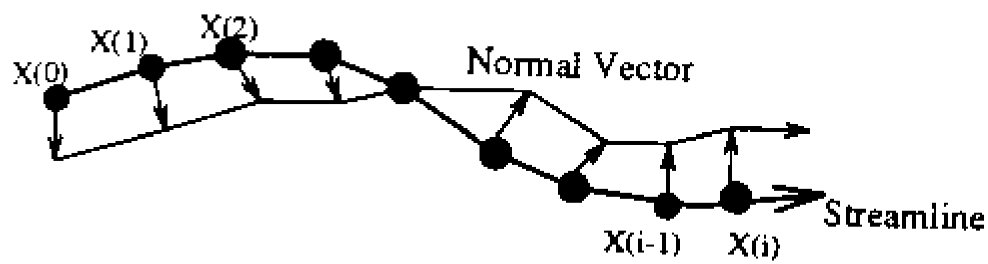


Mappings - Stream-ribbon

- We really would like to see vorticities, I.e. places were the flow twists.
- A point primitive or an icon can hardly convey this
- idea: trace neighboring particles and connect them with polygons
- shade those polygons appropriately and one will detect twists

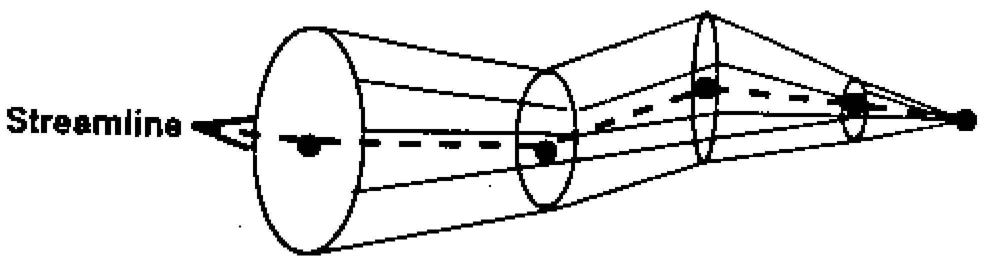
Mappings - Stream-ribbon

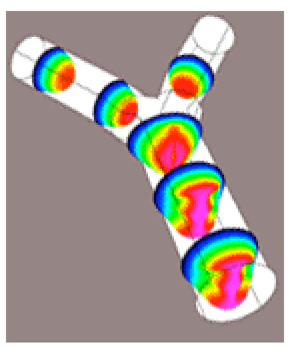
- Problem - when flow diverges
- Solution: Just trace one streamline and a constant size vector with it:

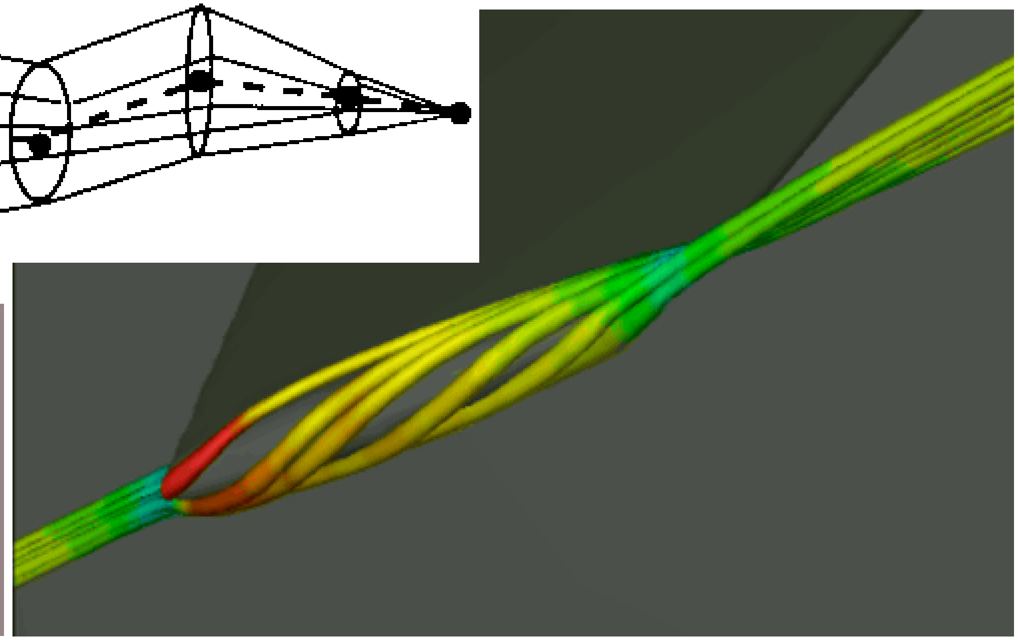


Mappings - Stream-tube

- Generate a stream-line and connect circular crossflow sections along the stream-line







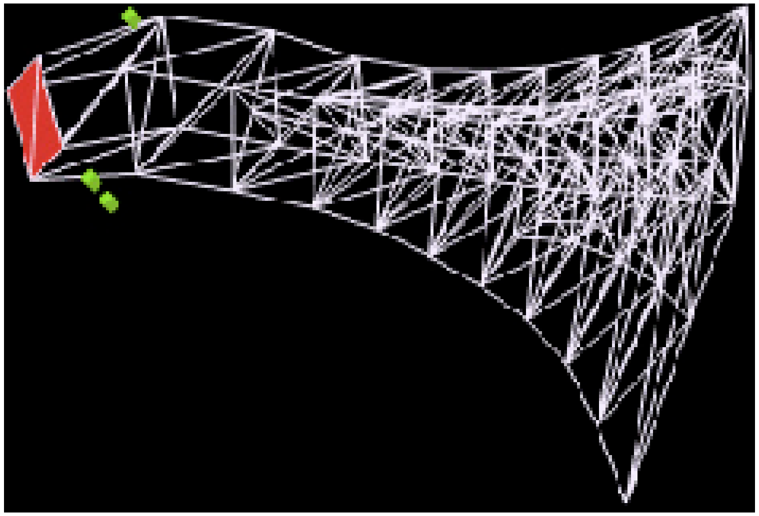
Mappings - Stream-balls

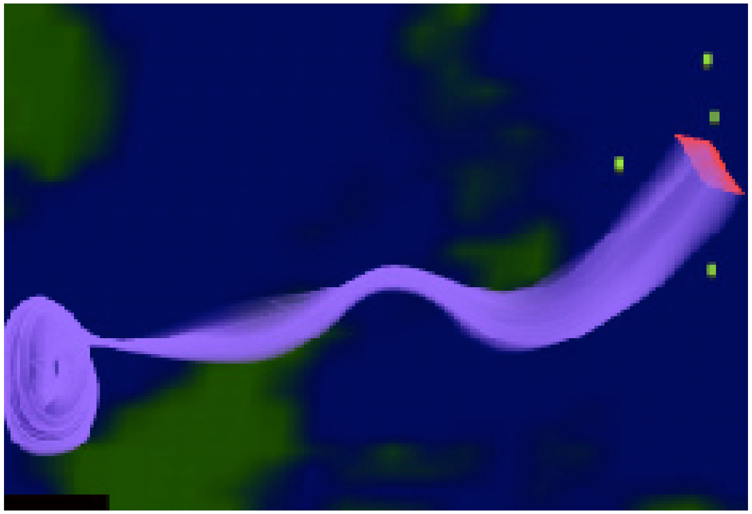
- Another way to get around diverging stream-lines
- simply put implicit surface primitives at particle traces - at places where they are close they'll merge elegantly ...

video

Mappings - Flow Volumes

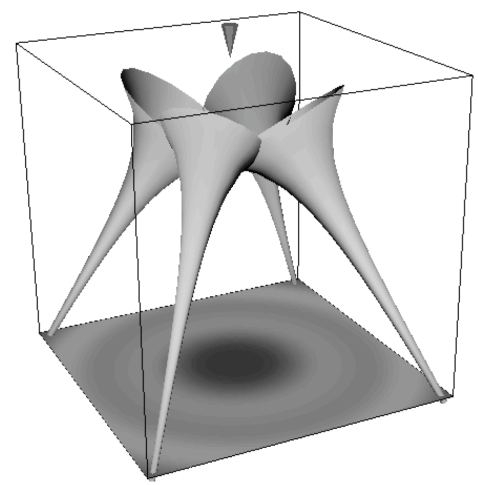
- Instead of tracing a line - trace a small polyhedra





Data Preparation - Tensors

- Hyper-streamlines:
  - look at eigen-values and eigen-vectors of tensor
  - visualize streamlines for one of the eigenvectors
  - use a geometric primitive that sweeps along that streamline
  - major, medium, and minor hyper-streamlines
  - depending on the magnitude of the eigenvector
- collection:
  - "critical" points (global) when one of the eigenvalues is zero

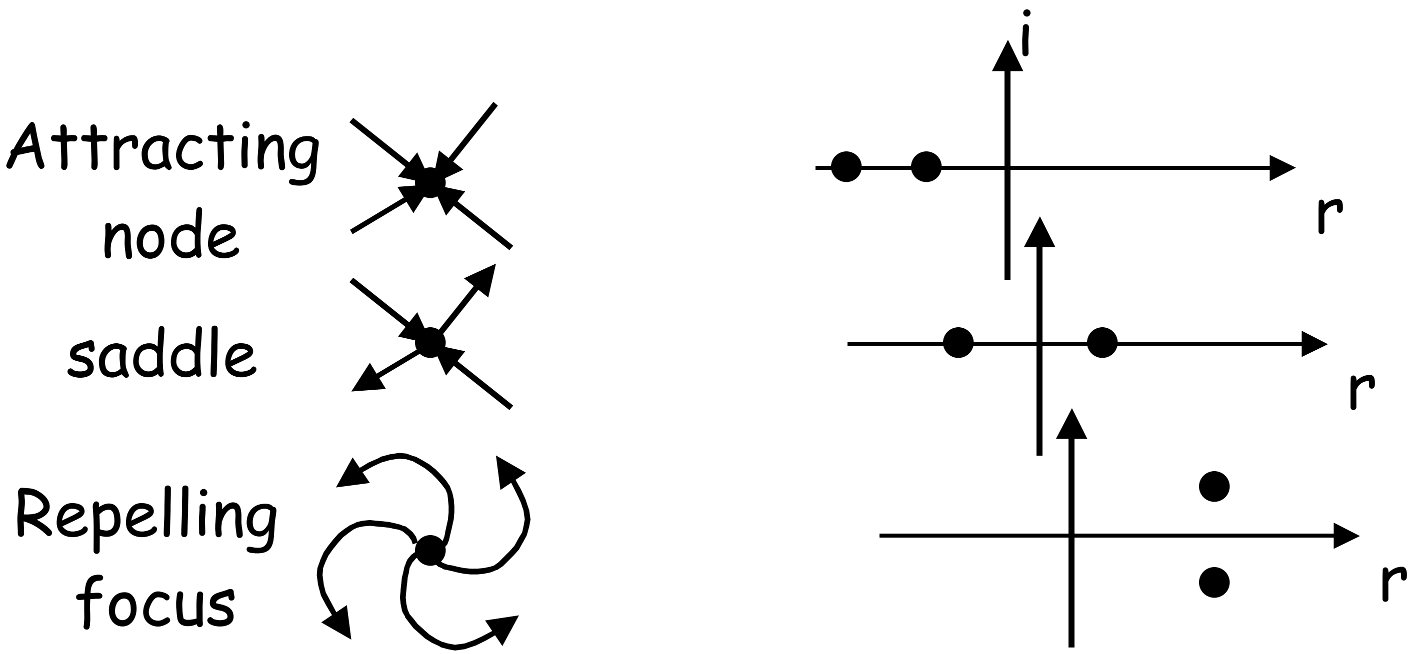


Data Preparation - Topology

- Finding “critical” points
- what is critical in a flow?
- Well - when it doesn’t flow anymore!
- I.e - critical points are places without change: $v = 0$!
- Try to
  - find these places
  - classify them

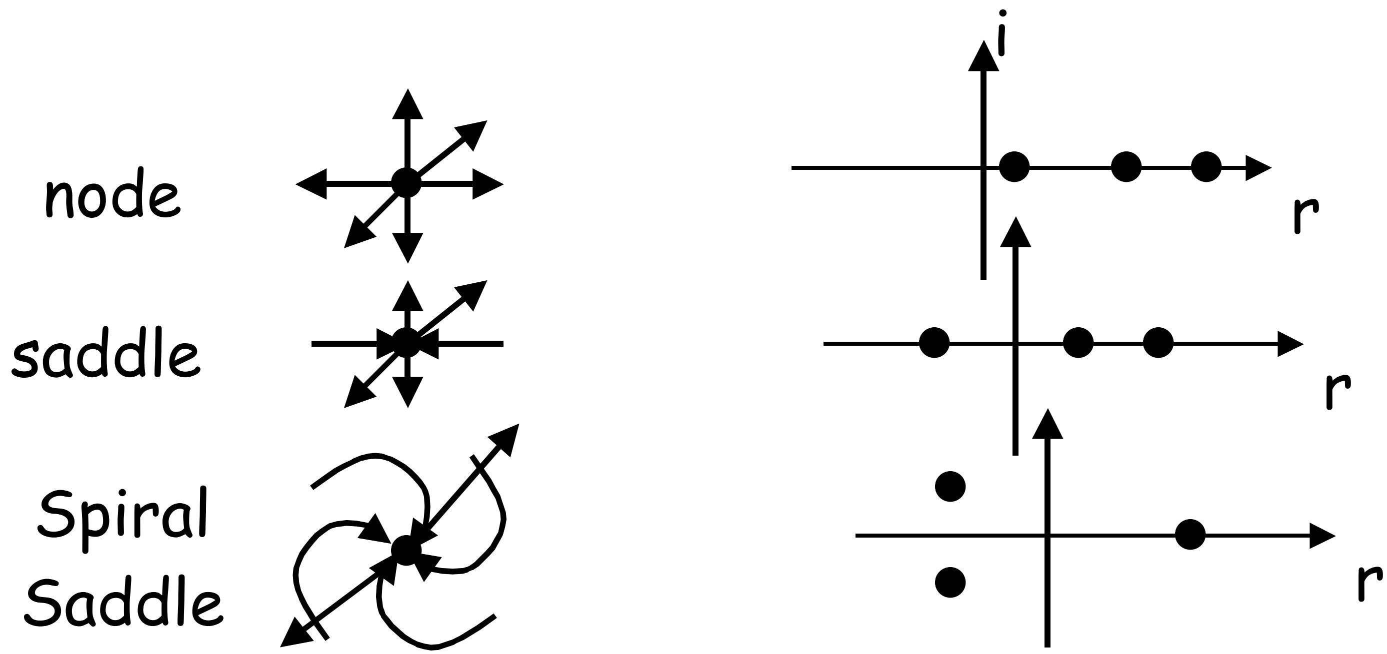
Data Preparation - Topology

- 2D classification (and higher D):
- according to eigen-values of derivative matrix



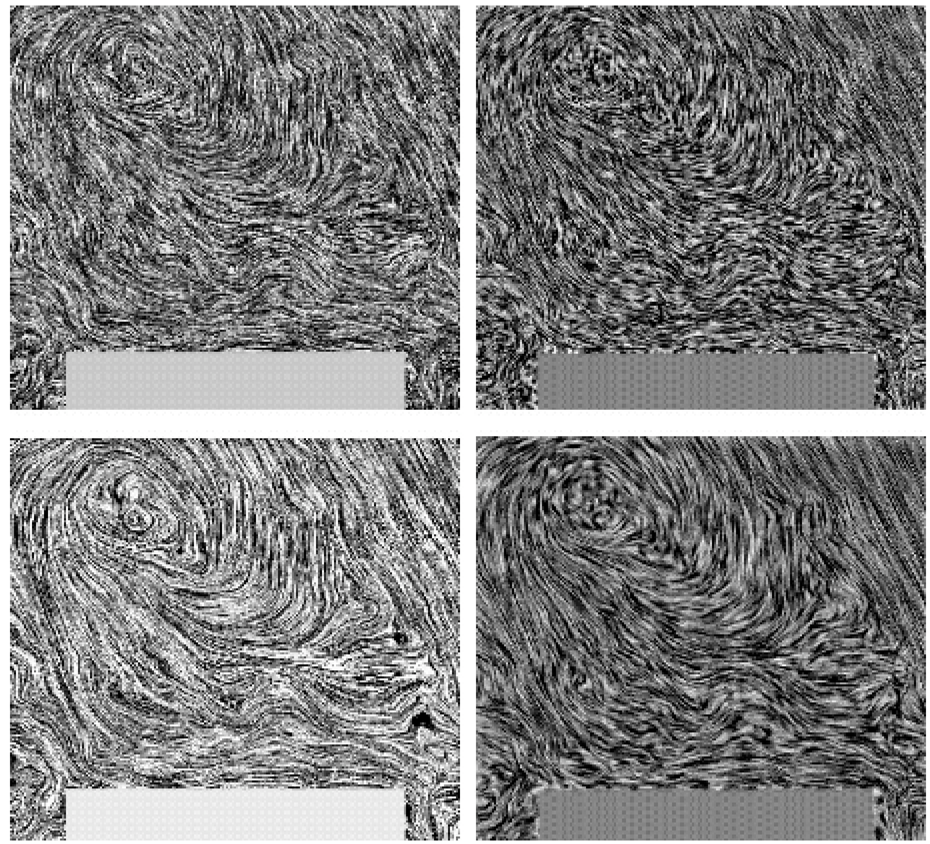
Data Preparation - Topology

- 3D classification
- more complicated



Rendering - LIC

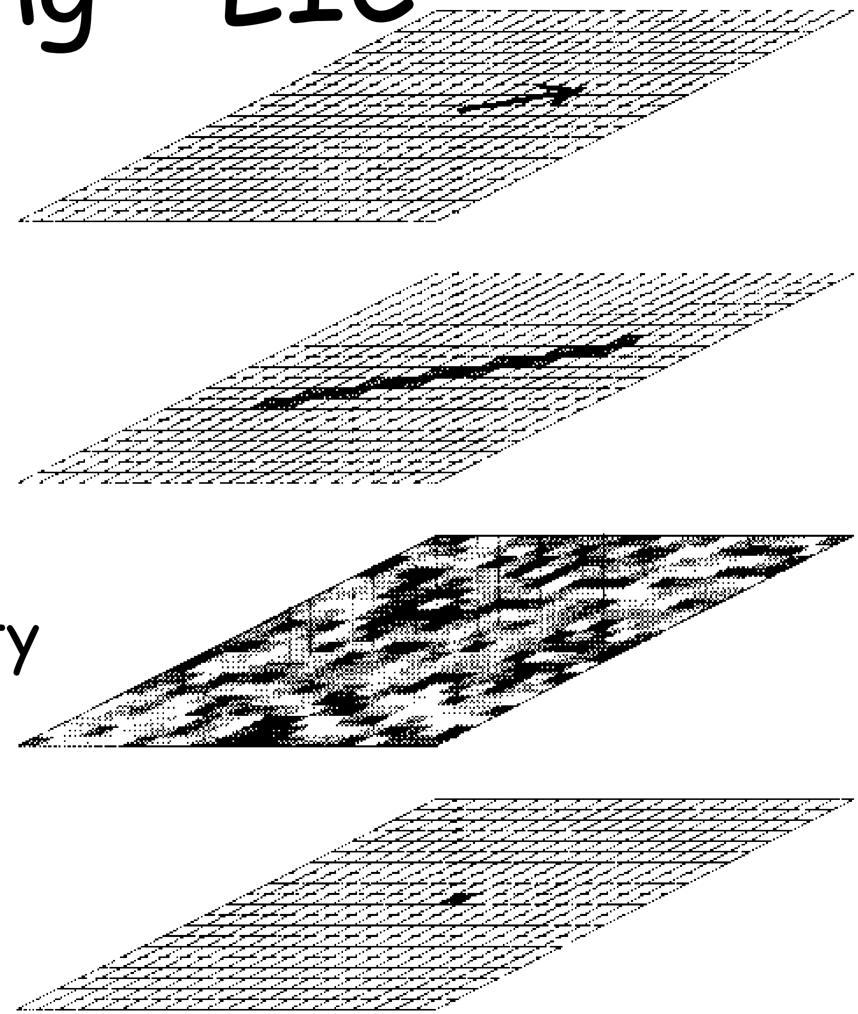
- Similar to spot noise
- underlying a noise texture under the vector field
- difference - integrates along a streamline



LIC Spot Noise

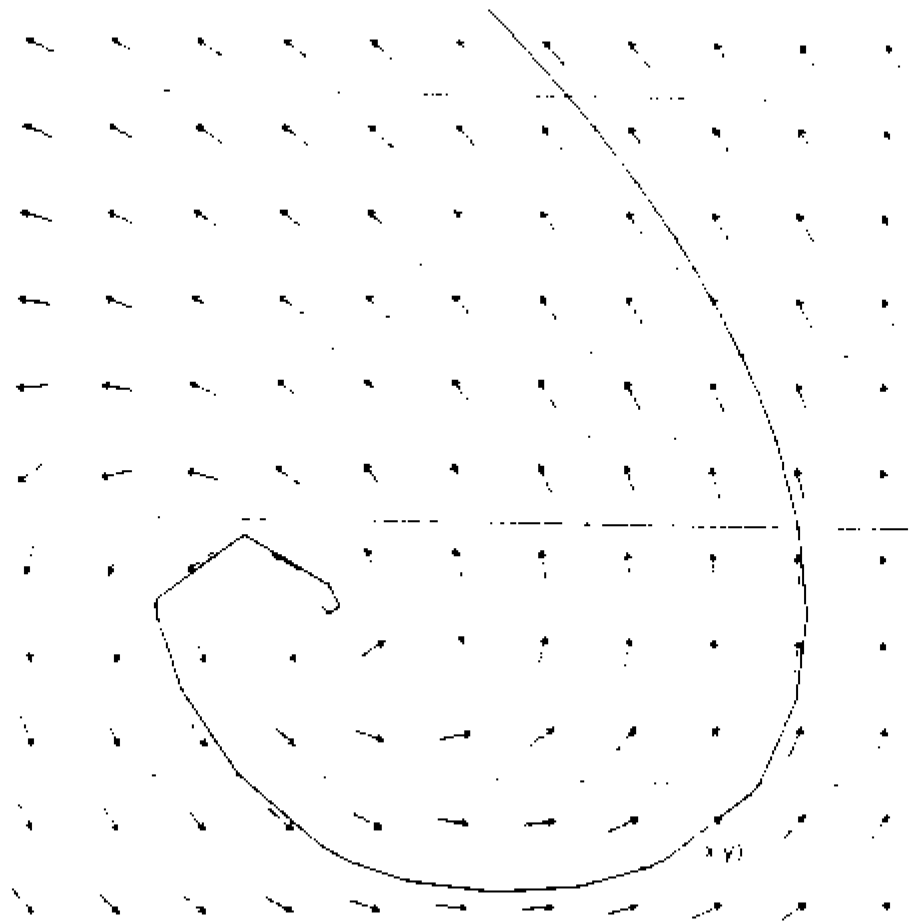
Rendering - LIC

- DDA convolution:
- translates each vector into a straight line (DDA line drawing)
- multiplies each pixel with a texture intensity to come up with a new value for the pixel

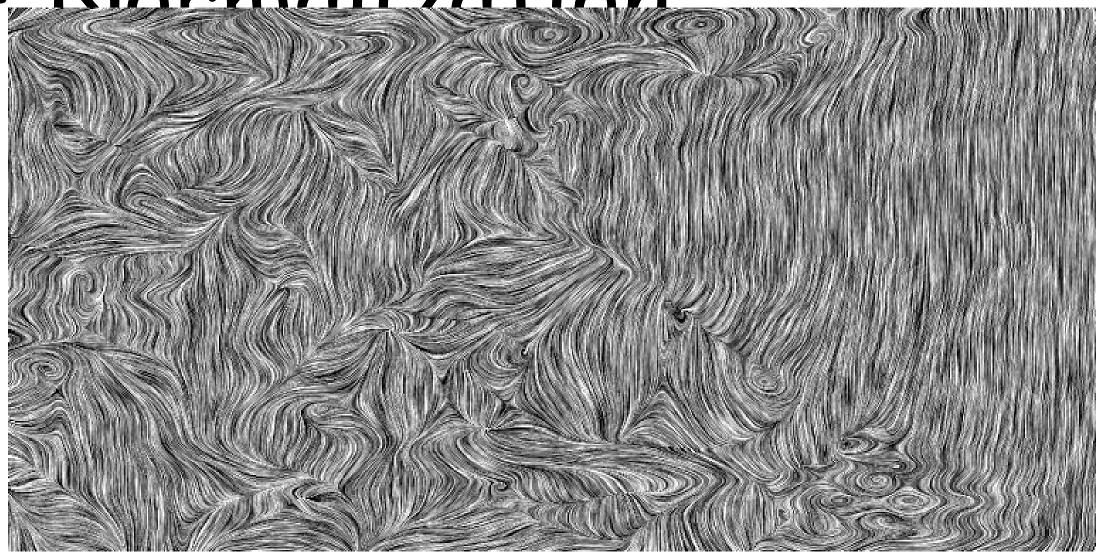


Rendering - LIC

- Compute a local stream line of a pre-determined size
- integrate the noise texture along that streamline

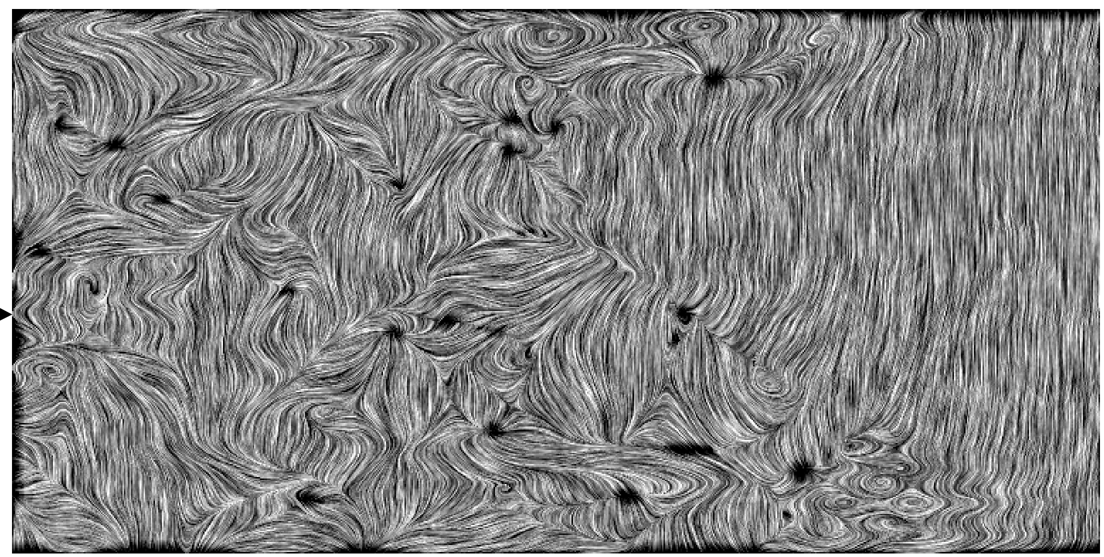


LIC - Normalization



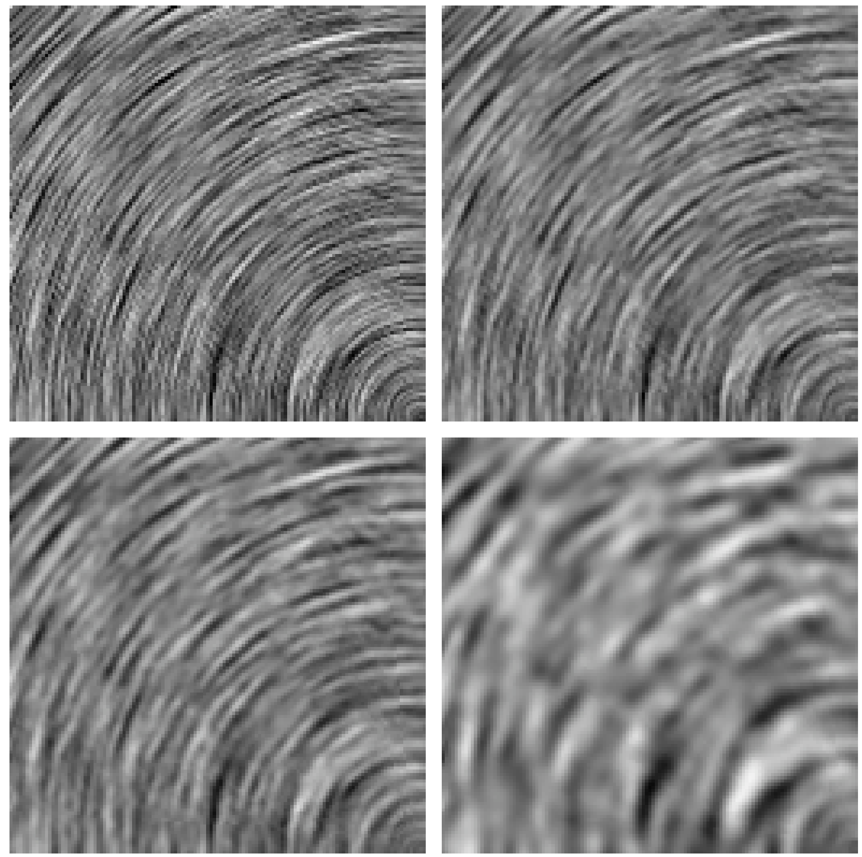
- We need to normalize by the sum of the filter weights

No normalization



LIC

- Aliasing can be a problem
- hence low-pass noise!



LIC - Image Processing

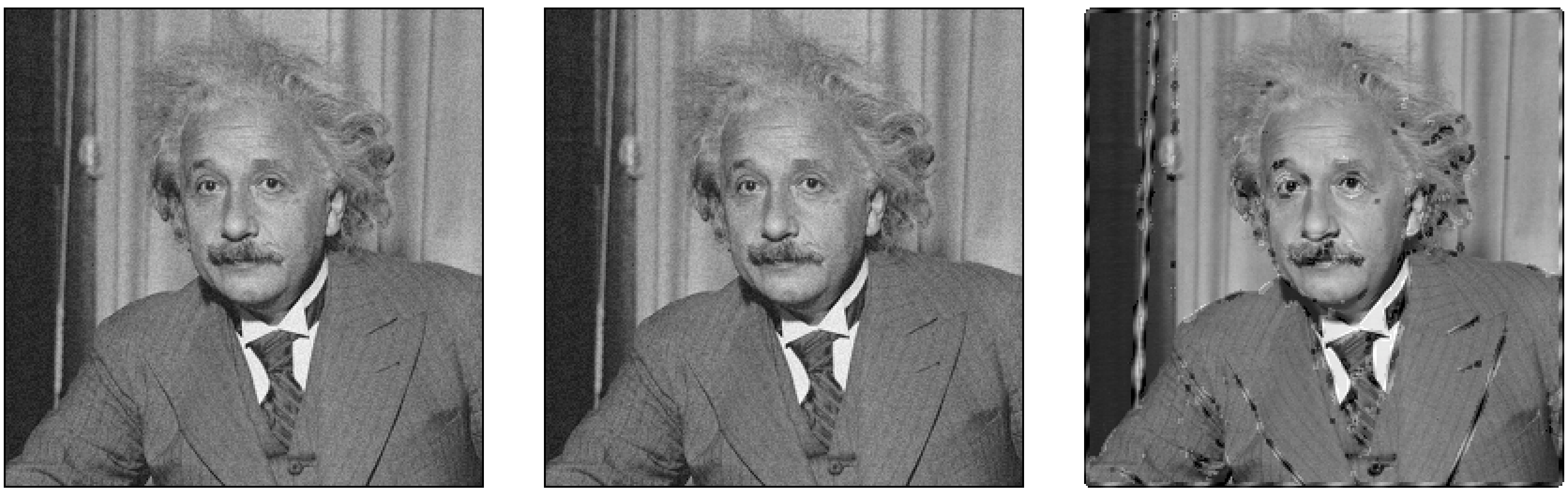
- We can apply a vector field to an image to change the image

Overview

- What is SciVis?
- Acquisition Methods
- Iso-surfaces
- Direct-Rendering Pipeline
- Vector Visualization
- **Challenges**

Challenges - Accuracy

- Analysis of rendering pipeline
- Need metrics -> perceptual metric



(a) Original (b) Bias-Added (c) Edge-Distorted

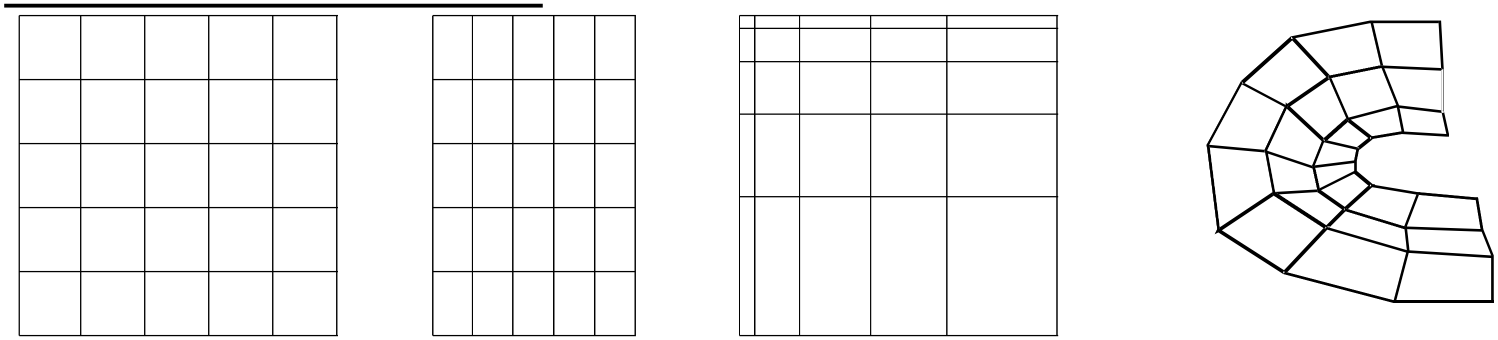
Challenges - Accuracy

- Deal with unreliable data (noise, Ultrasound)

Challenges - Accuracy

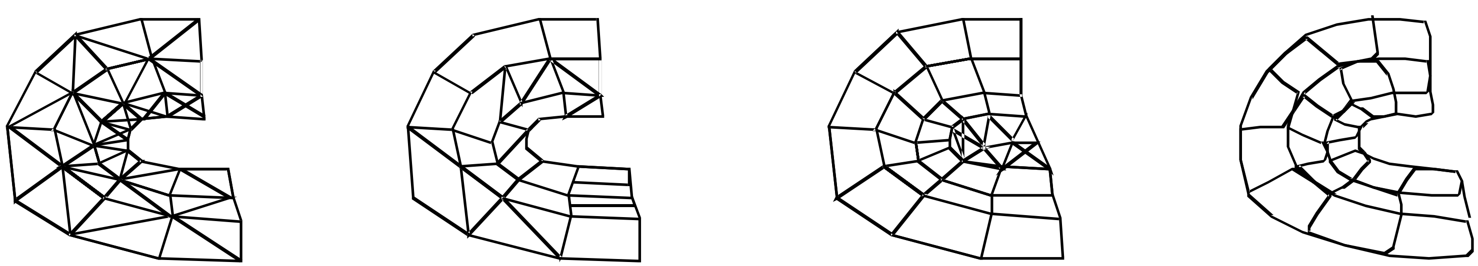
- Irregular data sets

Structured Grids:



regular uniform rectilinear curvilinear

Unstructured Grids:



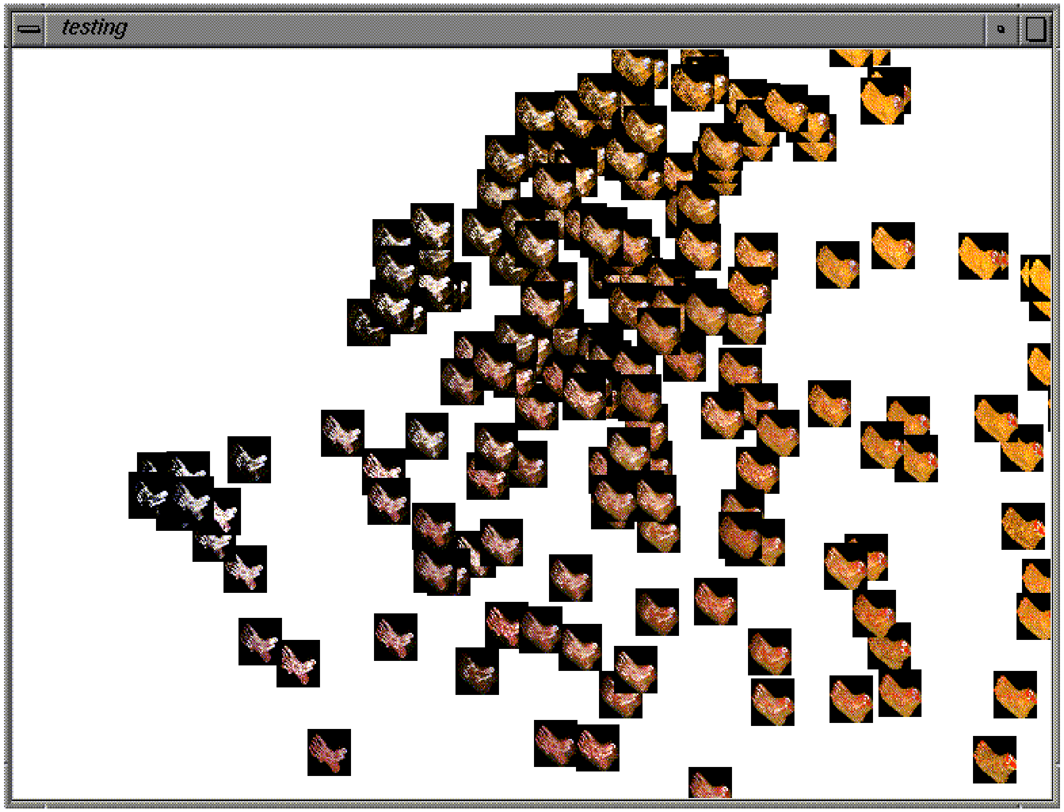
regular irregular hybrid curved

Challenges - Speed/Size

- Efficient algorithms
- Hardware developments (VolumePro)
- Utilize current hardware (nvidia, ATI)
- Compression schemes
- Tera-byte data sets

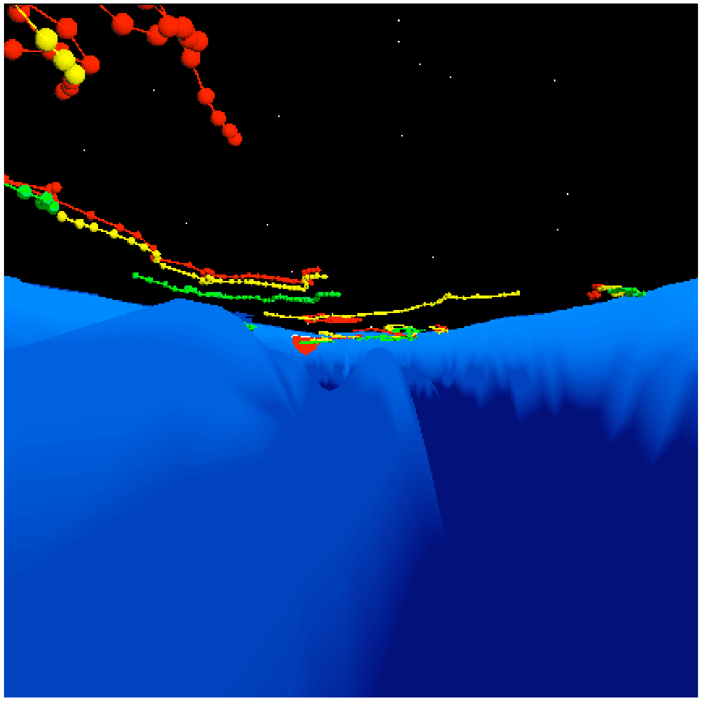
Challenges - HCI

- How to explore data set?
- Identify regions of interest quickly



Challenges - HCI

- "Augmented" reality
- Explore novel I/O devices



Tera-Scale Visualization

- Time-varying multi-modal data sets
- Common in engineering problems
- Accuracy - irregular data sets
- Speed - compression/supercomputers
- HCI - Regions of interests

Still Questions?

torsten@cs.sfu.ca

http://gruvi.cs.sfu.ca/