Scientific Visualization
Part 2: Volume Visualizatino

John Keyser

Volume Visualization

- We have discussed how to handle geometrically given data
  - Finding points by nearest neighbor
  - Interpolation methods
- We have discussed constructing isosurfaces
  - Via marching cubes or other methods
- Now is another challenge: visualizing volumes of data

Occlusion

- The fundamental problem is occlusion – the stuff in front blocks the stuff in back
- We can't just render data directly, or we would only see the data on the boundary
- So, we need ways to get around the occlusion problem.

3D Visualizations

- For 3D data, we need to actually use 3D rendering approaches
- This falls under computer graphics
  - We can (and do) have a whole course on it…
- Some basic concepts:
  - Set up virtual world (specify environment)
    - Geometry, materials, lights
  - Set up camera
  - Rendering method
    - How do we determine what each pixel should be
    - What is seen, how it's colored, etc.

Camera View

- Generally need to set up a camera
  - **Position (in 3D world)**
  - **View direction (or look-at point)**
  - **“Up” direction**
  - Field of view
  - Type of Projection
    - Perspective or Parallel

Position, Look-At, View Up

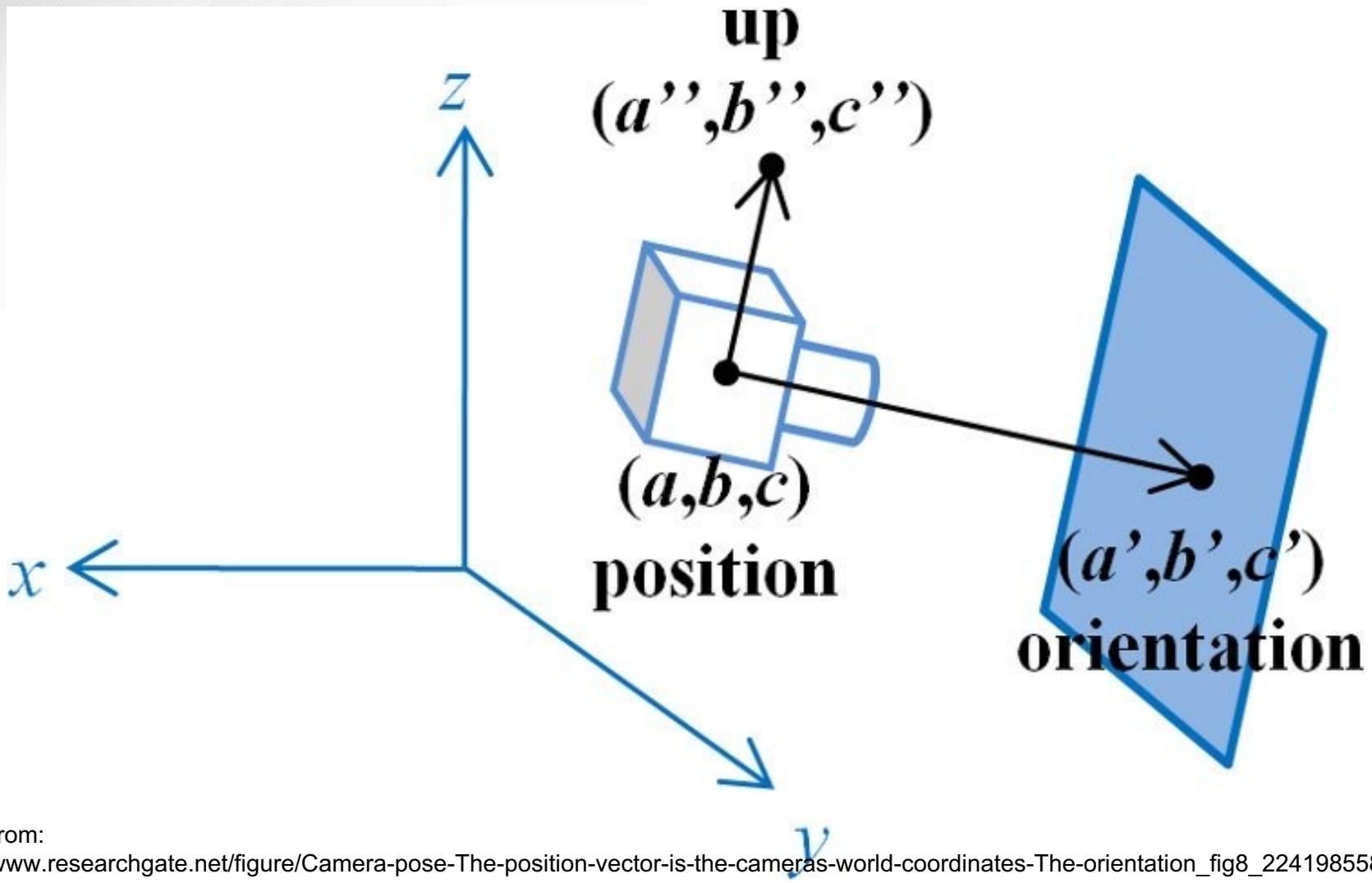

Image from:
https://www.researchgate.net/figure/Camera-pose-The-position-vector-is-the-cameras-world-coordinates-The-orientation_fig8_224198558

Camera View

- Generally need to set up a camera
  - Position (in 3D world)
  - View direction (or look-at point)
  - “Up” direction
  - **Field of view**
  - Type of Projection
    - Perspective or Parallel

Field of View

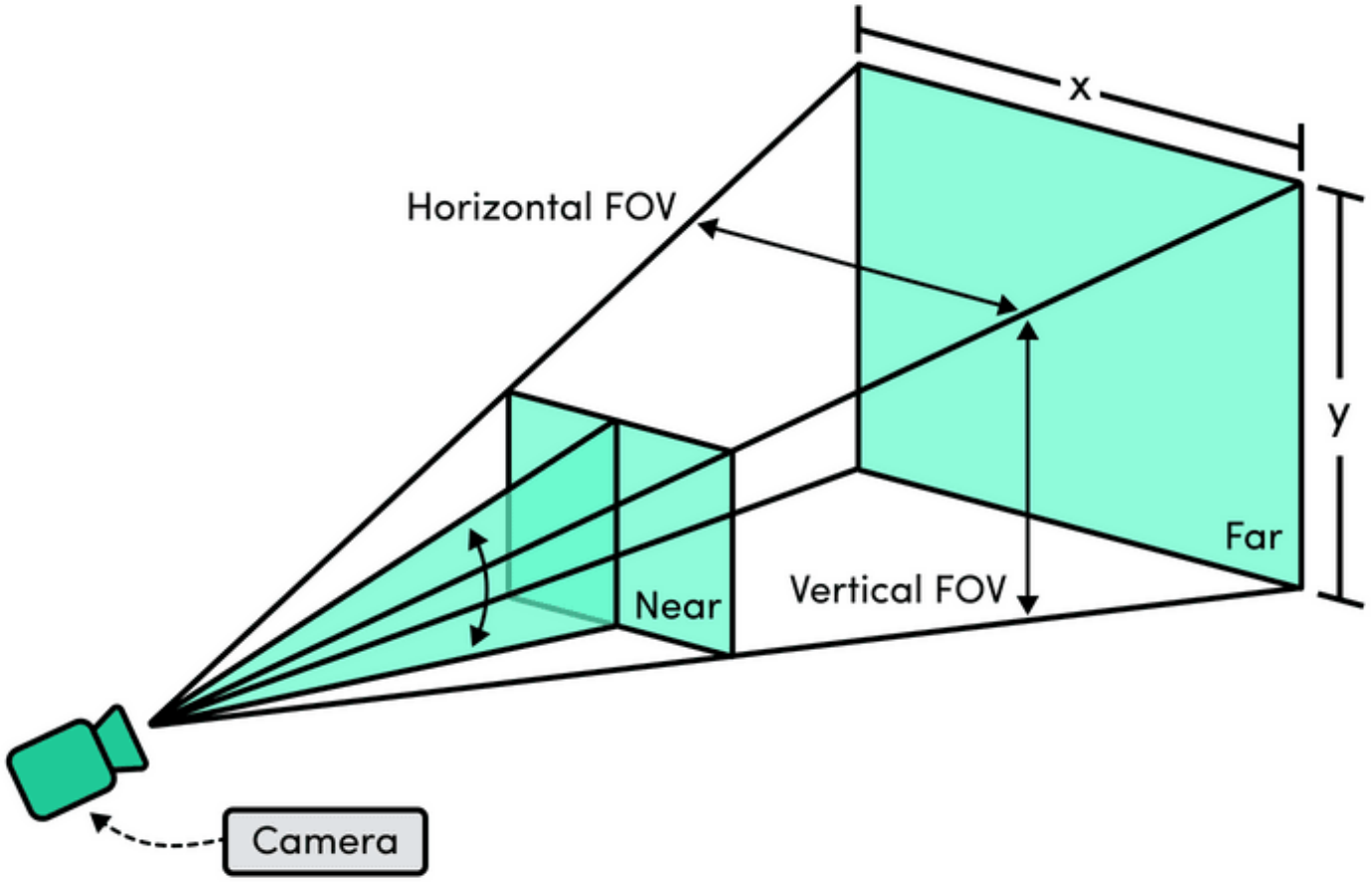

Image from:
https://www.gorrion.io/blog/three-js-series-basic-tutorial-of-three-js/

Camera View

- Generally need to set up a camera
  - Position (in 3D world)
  - View direction (or look-at point)
  - “Up” direction
  - Field of view
  - **Type of Projection**
    - **Perspective or Parallel**

Projection Type

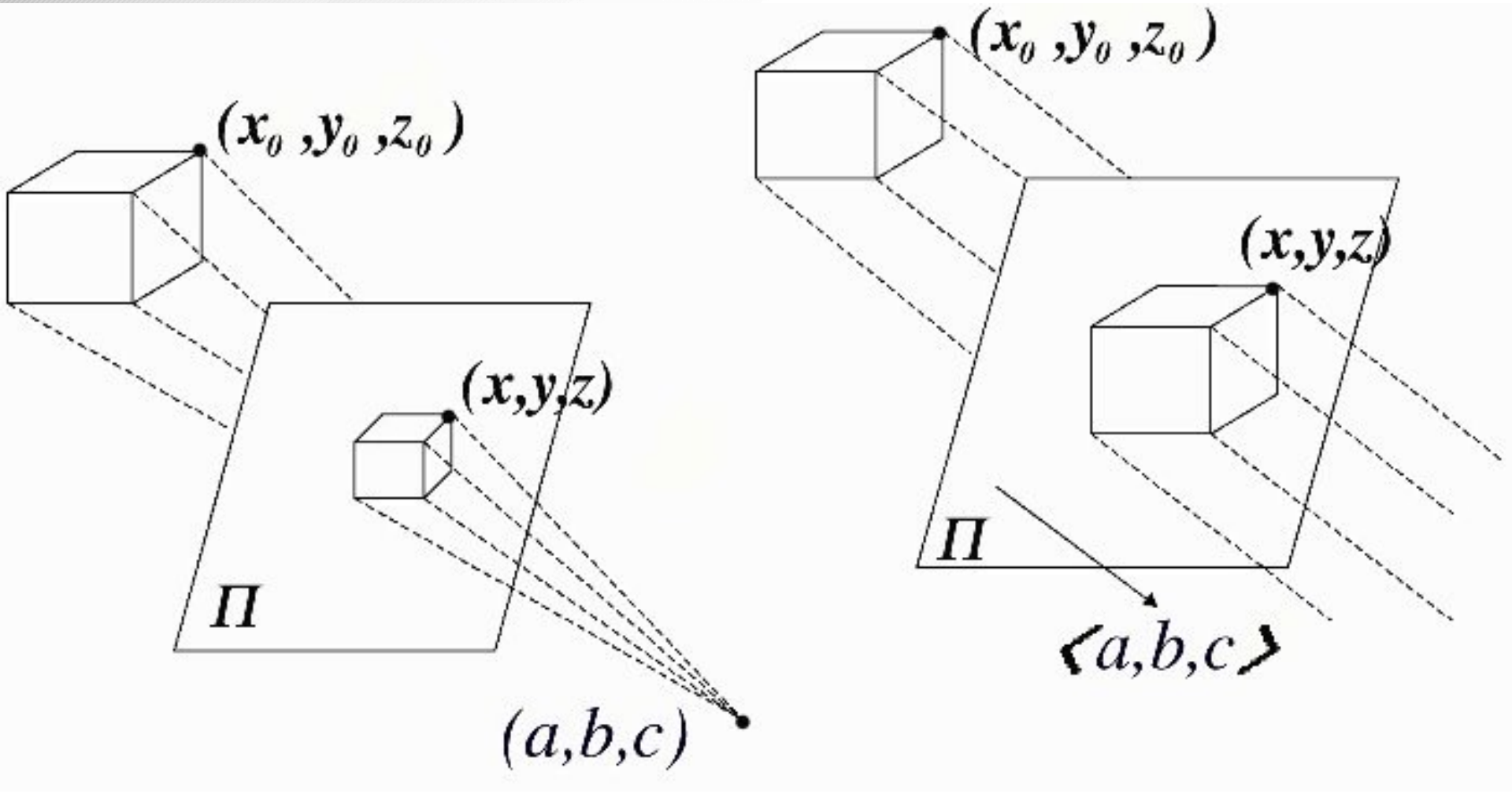

Image from:
https://glasnost.itcarlow.ie/~powerk/GeneralGraphicsNotes/projection/projection_viewing.html

Rendering

- With the world defined and a camera set up, the rendering process occurs

- For general graphics: find which surface is visible at each pixel, and calculate the color it should be based on light and material properties

Transparency

- Commonly used for Volume Visualization
- Usually there is some amount of transparency/opacity defined
- Multiple ways to implement:
  - Surfaces: Surface in front provides X % of color, everything behind provides (100-X)%
  - Grid Data: Provides a certain contribution to color that just adds to others seen

3D Display of One Variable

A 3D View that Isn't 3D: Height Fields

- A 2D Heat Map can be converted to a height field
  - Height is just the value that was encoded by color
  - Does not actually add any new information with 3rd dimension, but may be easier (or worse) to perceive

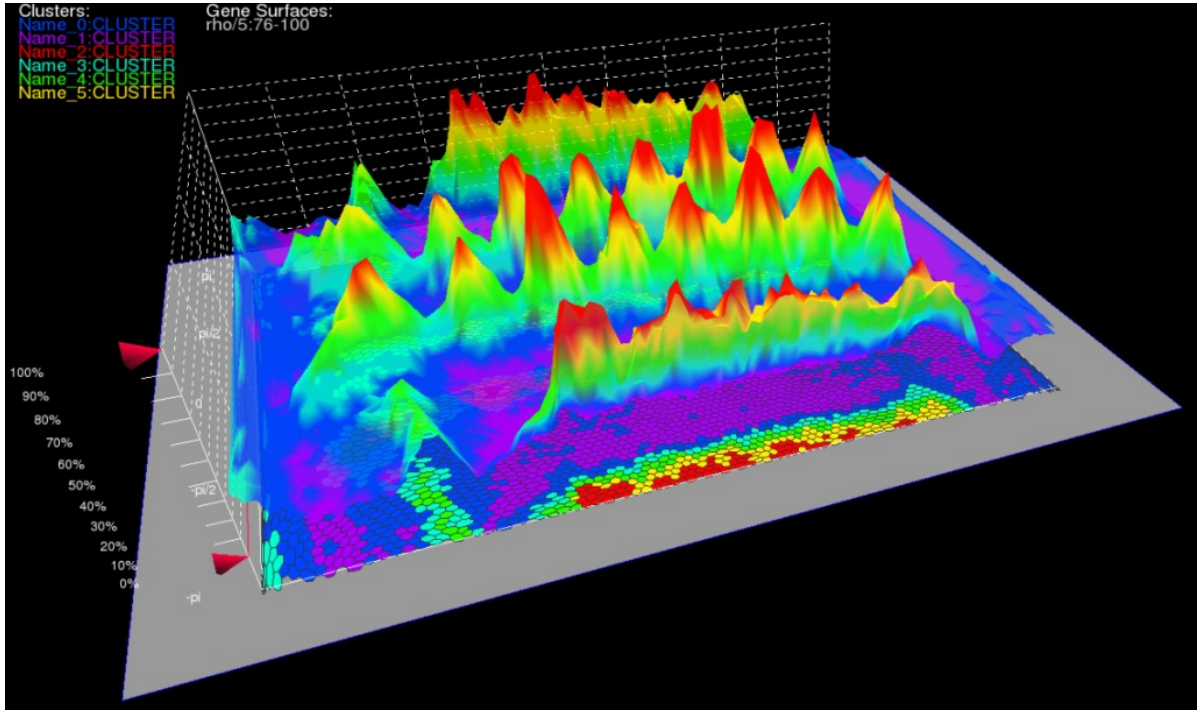

Image from:
https://www.omnisci.com/technical-glossary/scientific-visualization

Mapping Data onto a Height Field

- Height based on one variable (often real-world height), color on another
  - Perception can be difficult, though

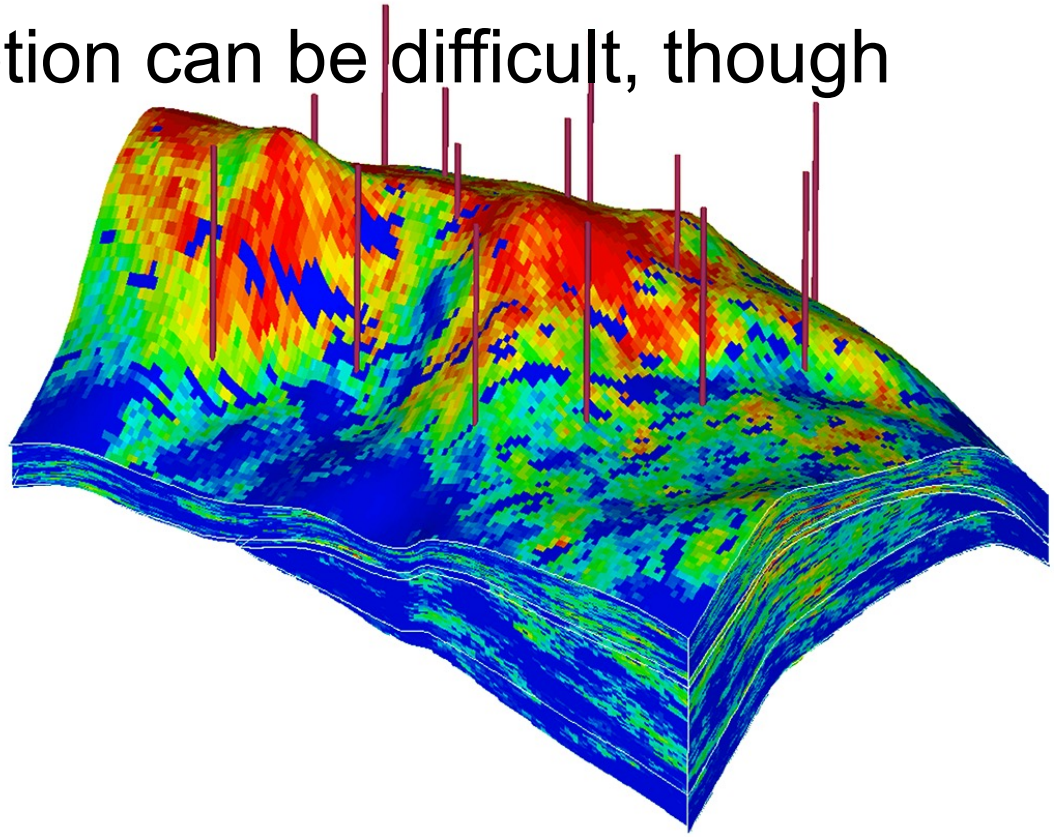

Image from:
https://www.specialoilfield.com/services/reservoir-simulation/

Direct Views of Grid Data

Cutaway 3D Views of grid data

- Remove part of the data (a geometric region
- Any one view shows only a portion/faces of region
- Requires interaction (moving plane) to understand full data set

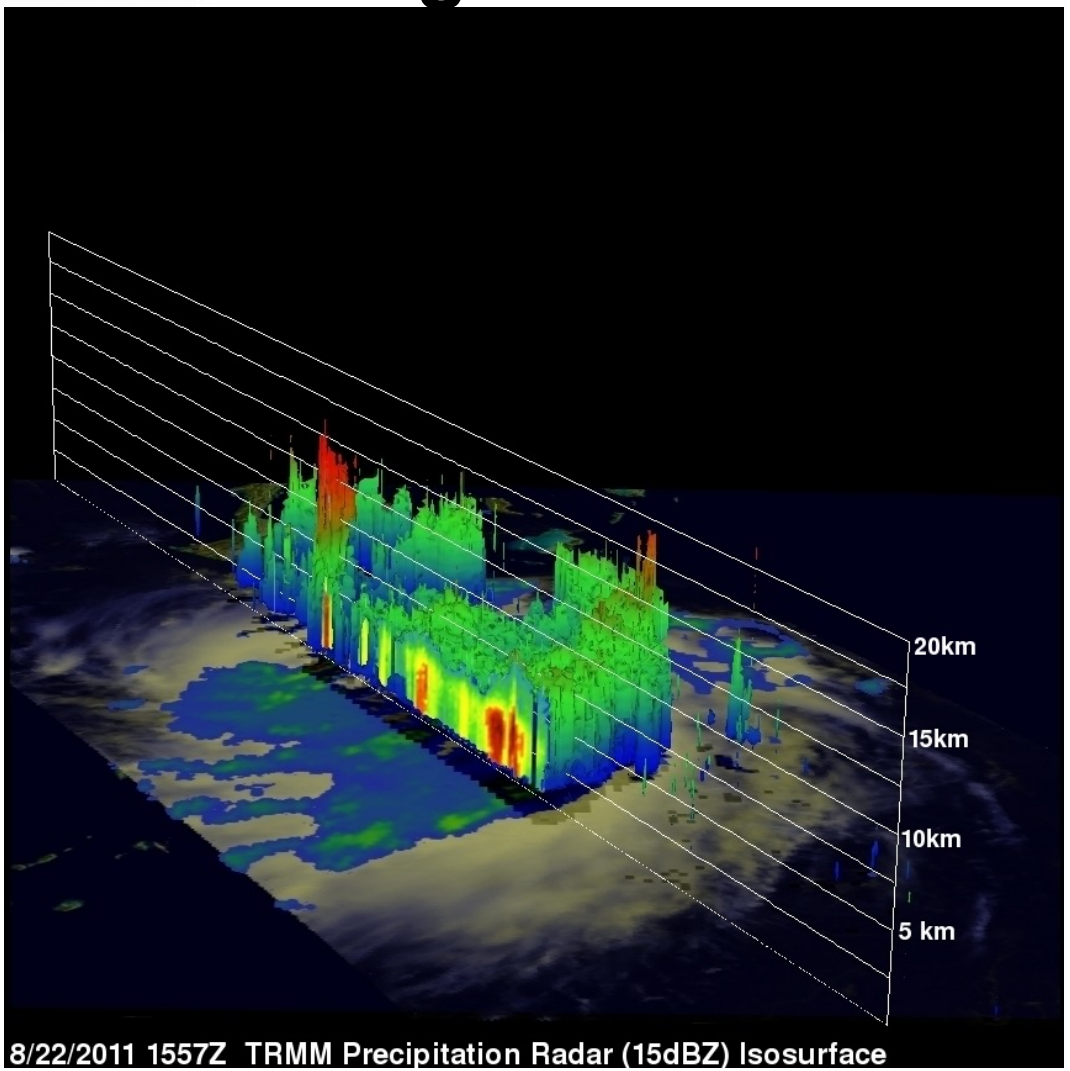

Image from:
http://3dviewpicture.blogspot.com/2013/08/3d-weather-radar.html

Slice View of Grid Data

- Can get slices through the 3D data, and render them on planes or other surfaces
- Interaction (moving planes) makes this more useful

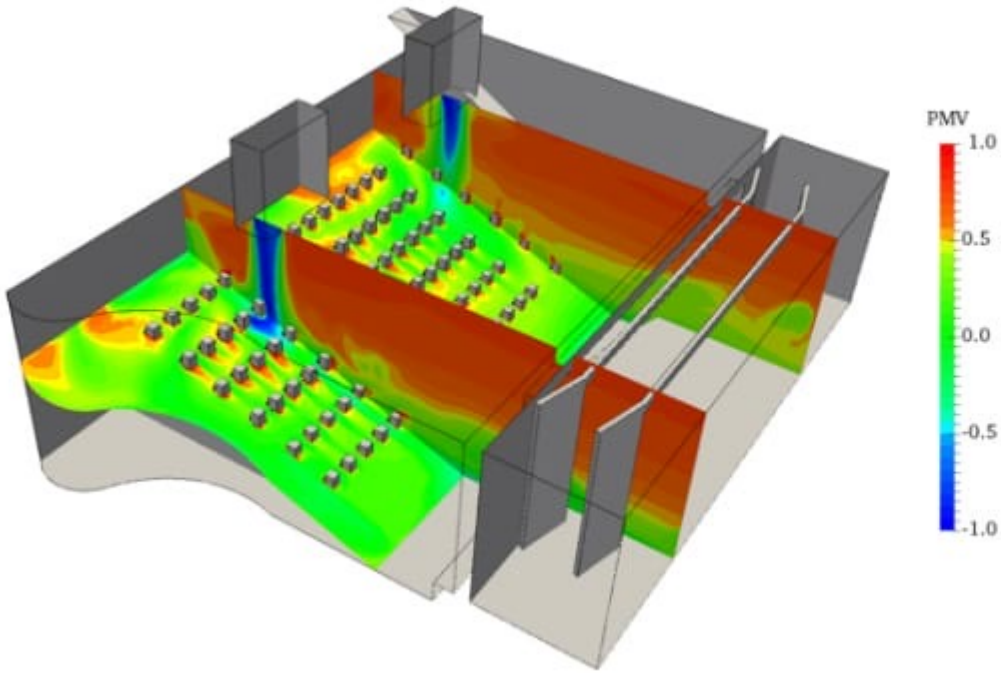

Image from:
https://www.simscale.com/blog/2018/04/en-15251-epbd-thermal-comfort/

Visualizing The Full Set of Data

Drawing All Samples

- Display all points with transparency or spacing
  - So you can see through to those behind, to a degree
- Can be difficult to determine depth, measure anything precisely
- But, can get general sense of data

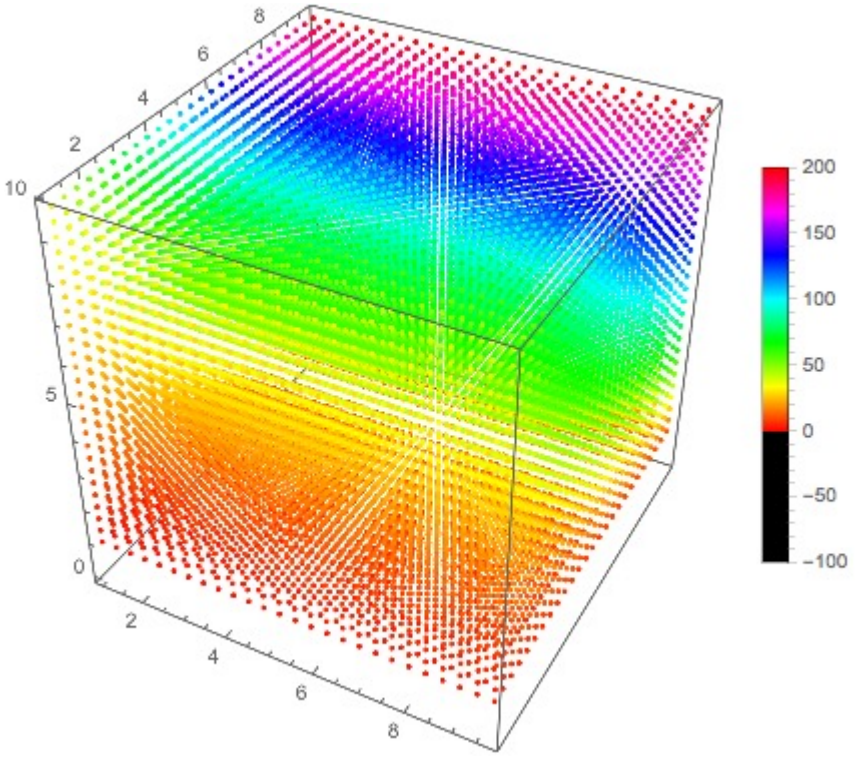

Image from:
https://mathematica.stackexchange.com/questions/152971/adding-color-bar-to-4d-graph

Isosurface Visualization

- Can render isosurfaces directly

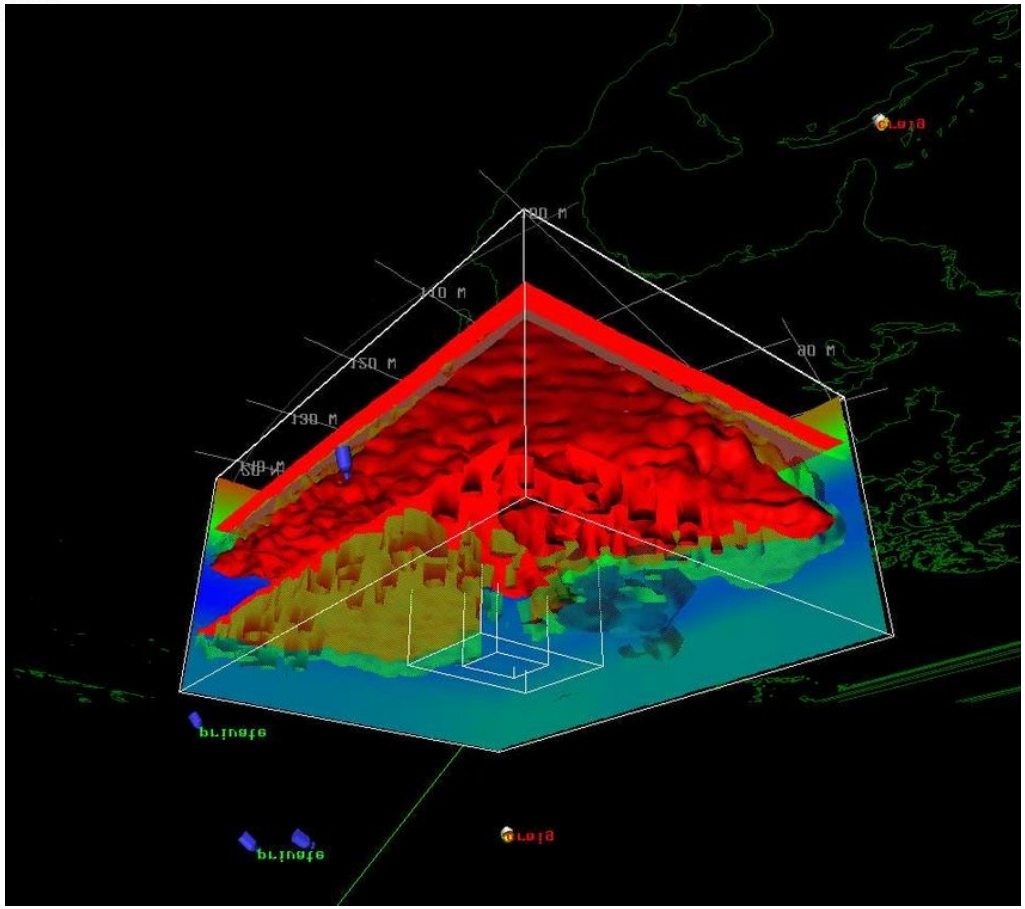

Image from:
https://www.researchgate.net/figure/Complex-isosurface-visualizations-of-humidity-and-temperature-elds_fig6_3208580/142779

Isosurfaces

- Rendering with transparency and controllable cutaways allows better views

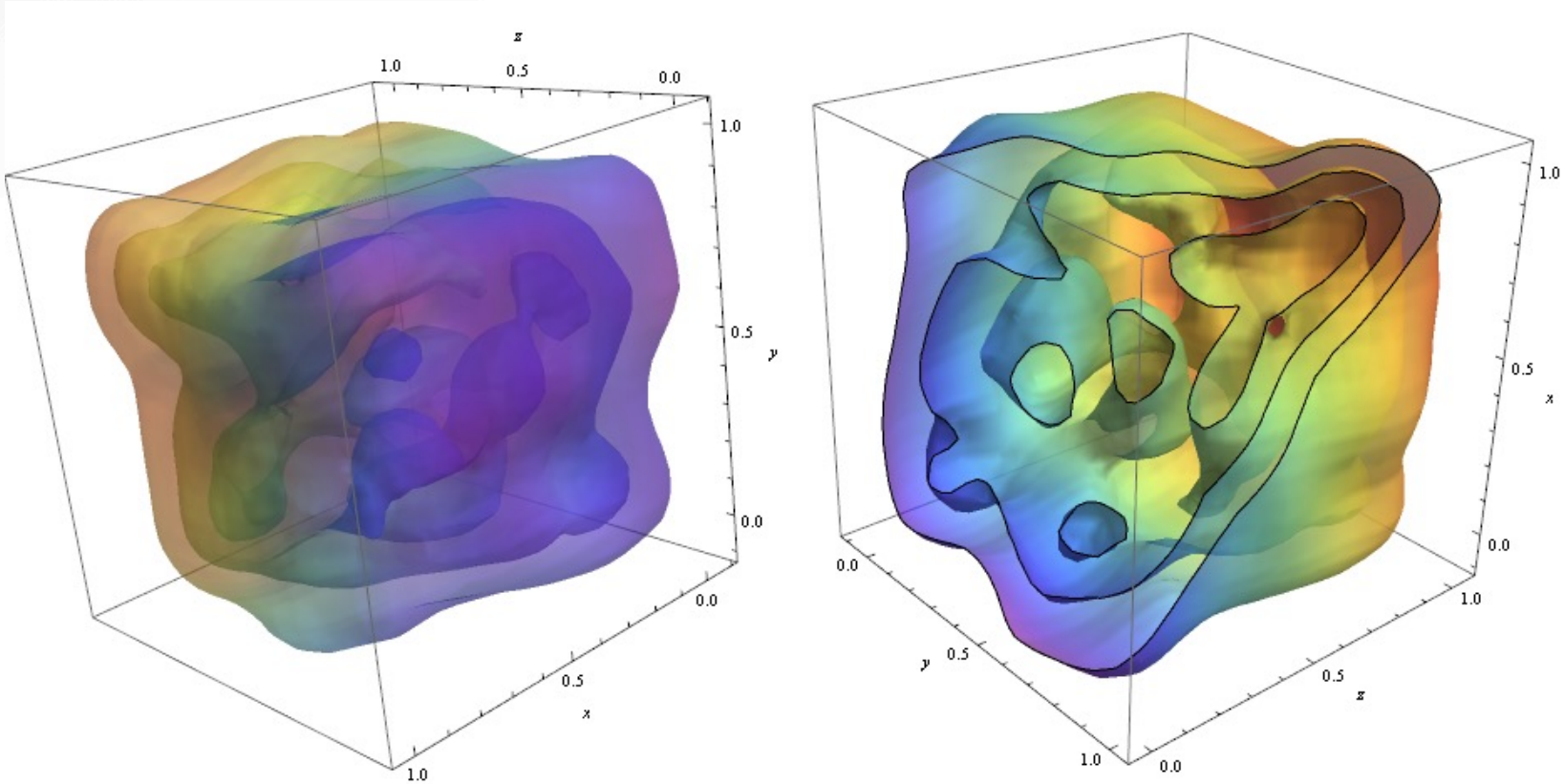

Image from:
https://tex.stackexchange.com/questions/142768/create-3d-heatmap-density-plot-using-tikz/142779

Transfer Functions

- Transfer functions describe how one parameter value should translate to what's rendered
- Sets opacity and color values based on the scalar value being rendered
- Users usually decide the transfer function
  - Based on what features they care to see
  - With some trial and error
- This is the fundamental tool for direct volume visualization

Transfer Function Interface

- Manual specification of TF requires an interface

Image from:
https://www.researchgate.net/figure/A-snapshot-of-the-transfer-function-interface-in-our-system-The-showing-curves-are_fig2_4188093

Transfer Functions

- Can highlight the key parts of the object
- Similar to isosurfaces, but without the additional extraction step

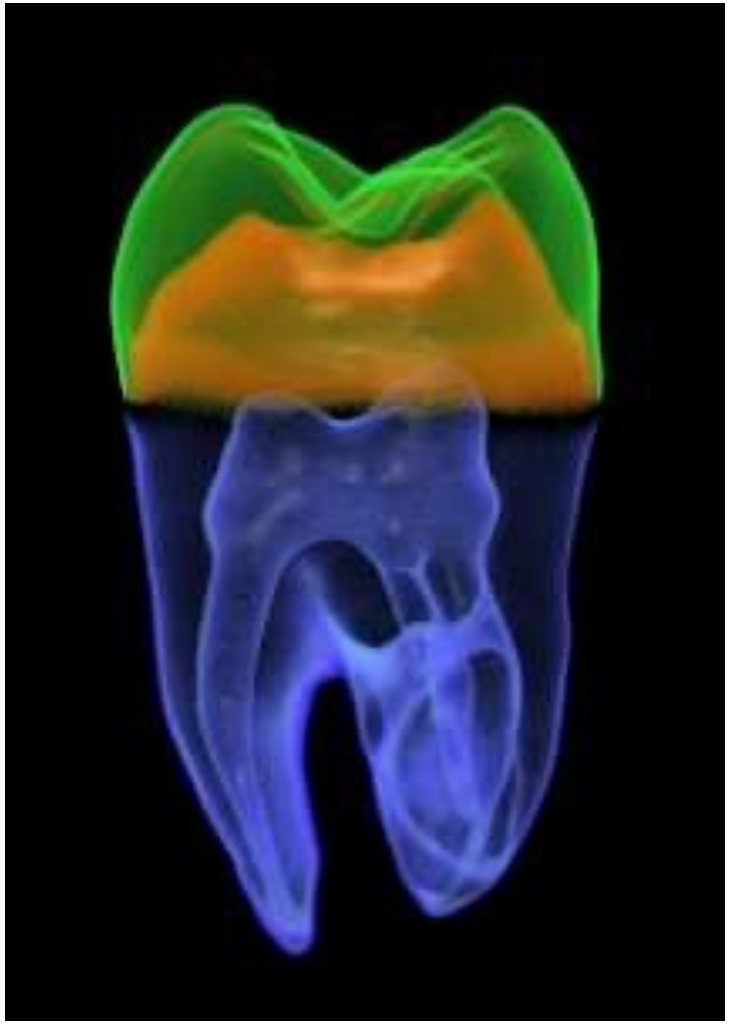

Image from:
https://www.researchgate.net/figure/This-figure-shows-the-effects-of-different-transfer-functions-on-a-tooth-dataset-25_fig4_265051079