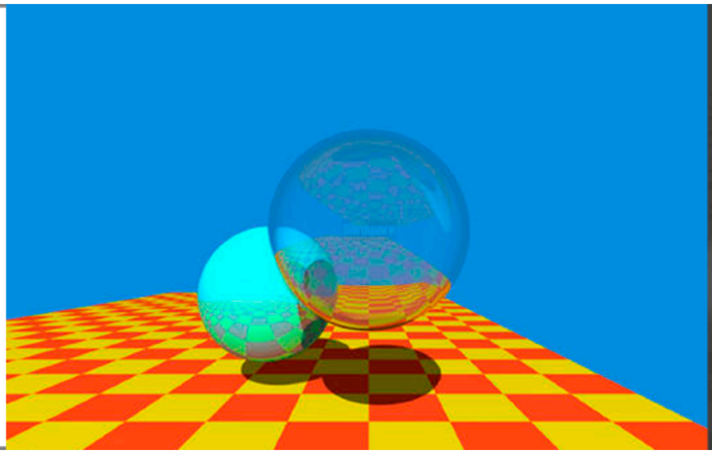
Introduction to Computer Graphics

GAMES101, Lingqi Yan, UC Santa Barbara

Lecture 10:
Geometry 1 (Introduction)






http://www.cs.ucsb.edu/~lingqi/teaching/games101.html

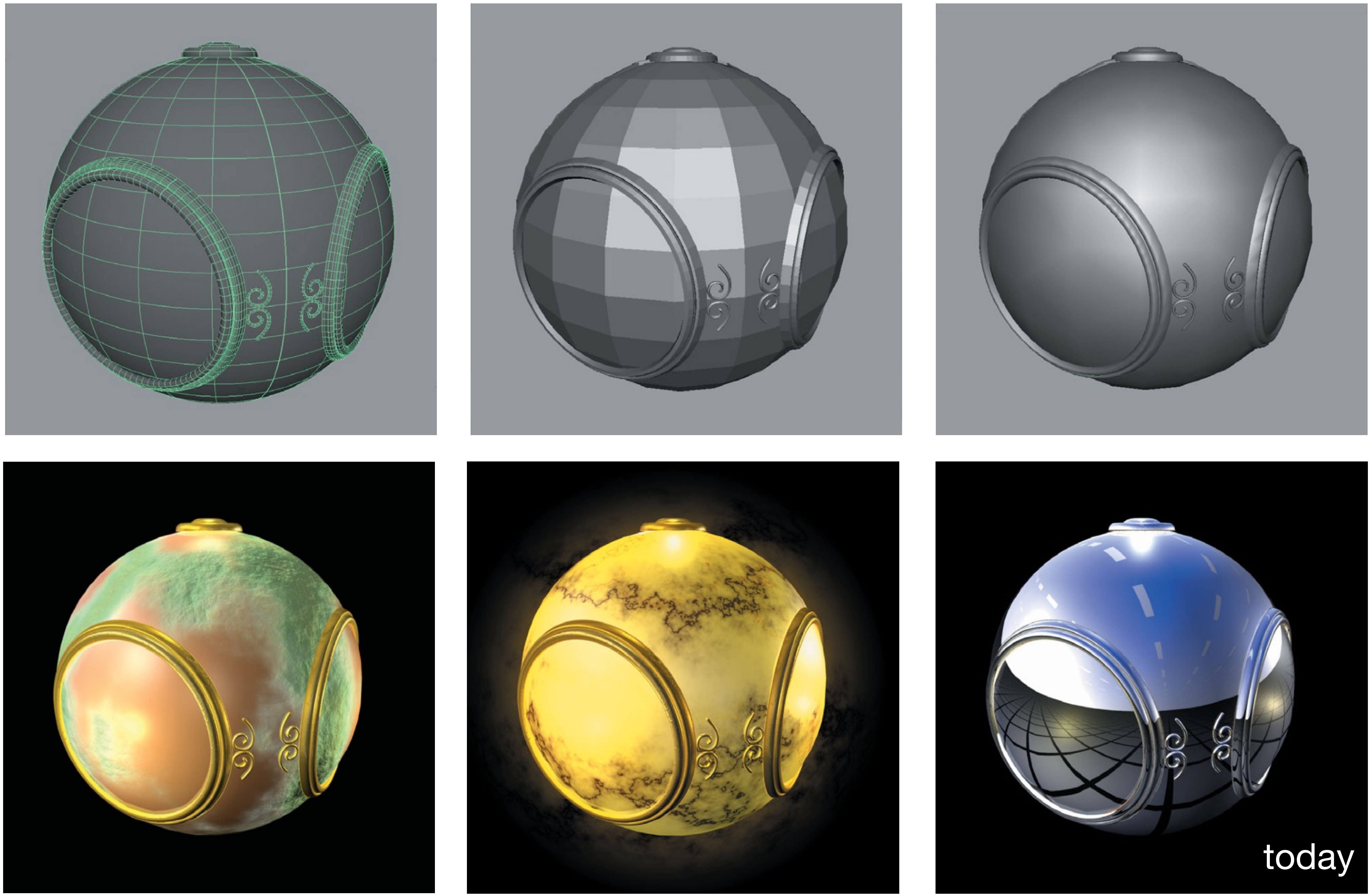
Announcements

- Homework 3
  - The framework has been updated
  - Together with an “FAQ” section in the BBS
- New TAs
  - Peng Yu (禹鹏)，BUAA，y2505418927@gmail.com
  - Wenxian Guo (郭文鲜)，ZJU，wxguojlu@hotmail.com

Last Lectures

- Shading 1 & 2
  - Blinn-Phong reflectance model
  - Shading models / frequencies
  - Graphics Pipeline
  - Texture mapping
- Shading 3
  - Barycentric coordinates
  - Texture antialiasing (MIPMAP)
  - Applications of textures

Last Lectures



Today

- Applications of textures
- Introduction to geometry (2nd part of this course!)
    - Examples of geometry
    - Various representations of geometry

Applications of Textures

Many, Many Uses for Texturing

In modern GPUs, texture = memory + range query (filtering)

- General method to bring data to fragment calculations

Many applications

- Environment lighting
- Store microgeometry
- Procedural textures
- Solid modeling
- Volume rendering
- …

Environment Map

假设光源无限远，只记录光照的方向信息，这种贴图被称作环境光贴图

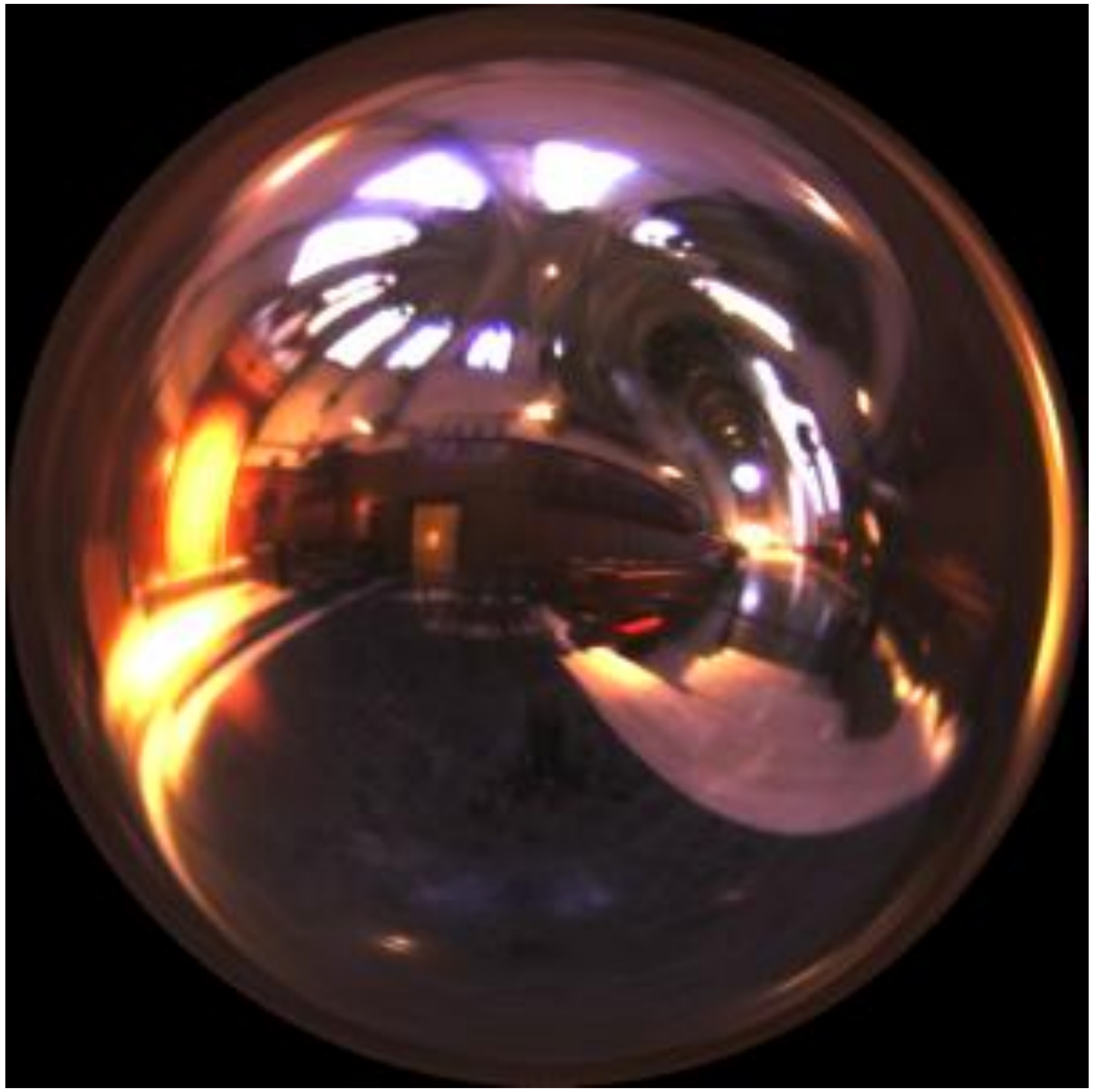
Light from the environment

Rendering with the environment

[Blinn & Newell 1976]

Utah Teaport 犹他茶壶

Environmental Lighting



Environment map (left) used to render realistic lighting

Spherical Environment Map

球心为世界中心。类比地球仪展开铺平，存在纹理的拉升扭曲问题，解决方法:Cube Map

Hand with Reflecting Sphere. M. C. Escher, 1935. lithograph

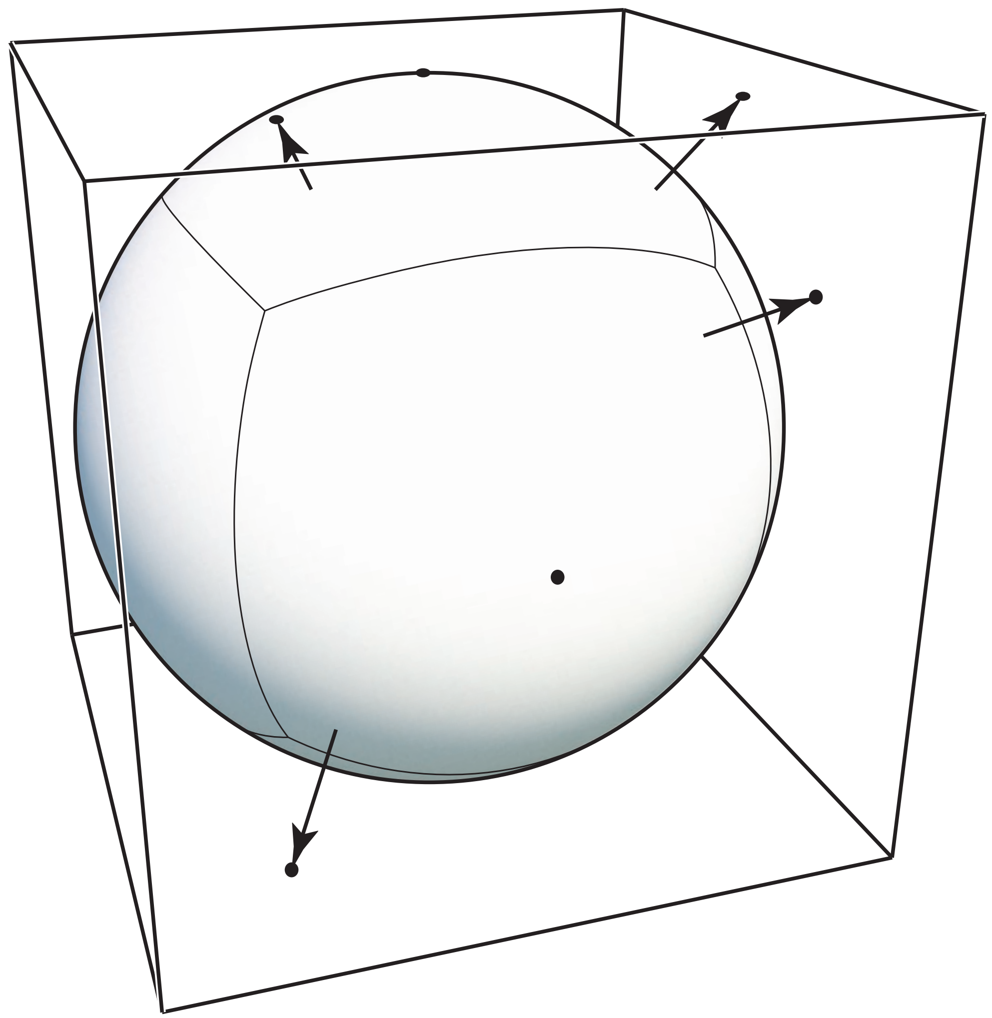
Light Probes, Paul Debevec

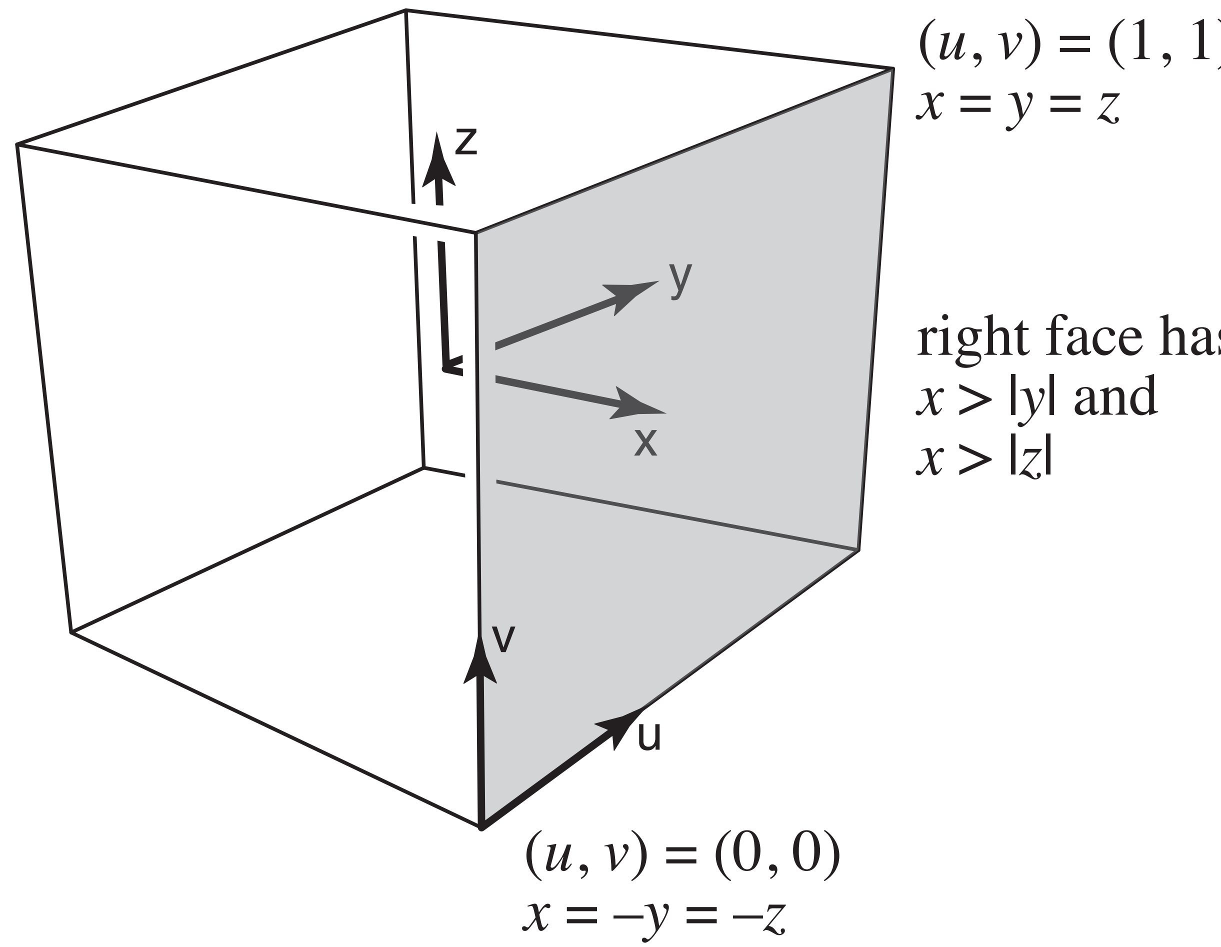
Spherical Map — Problem

Prone to distortion (top and bottom parts)!

Cube Map

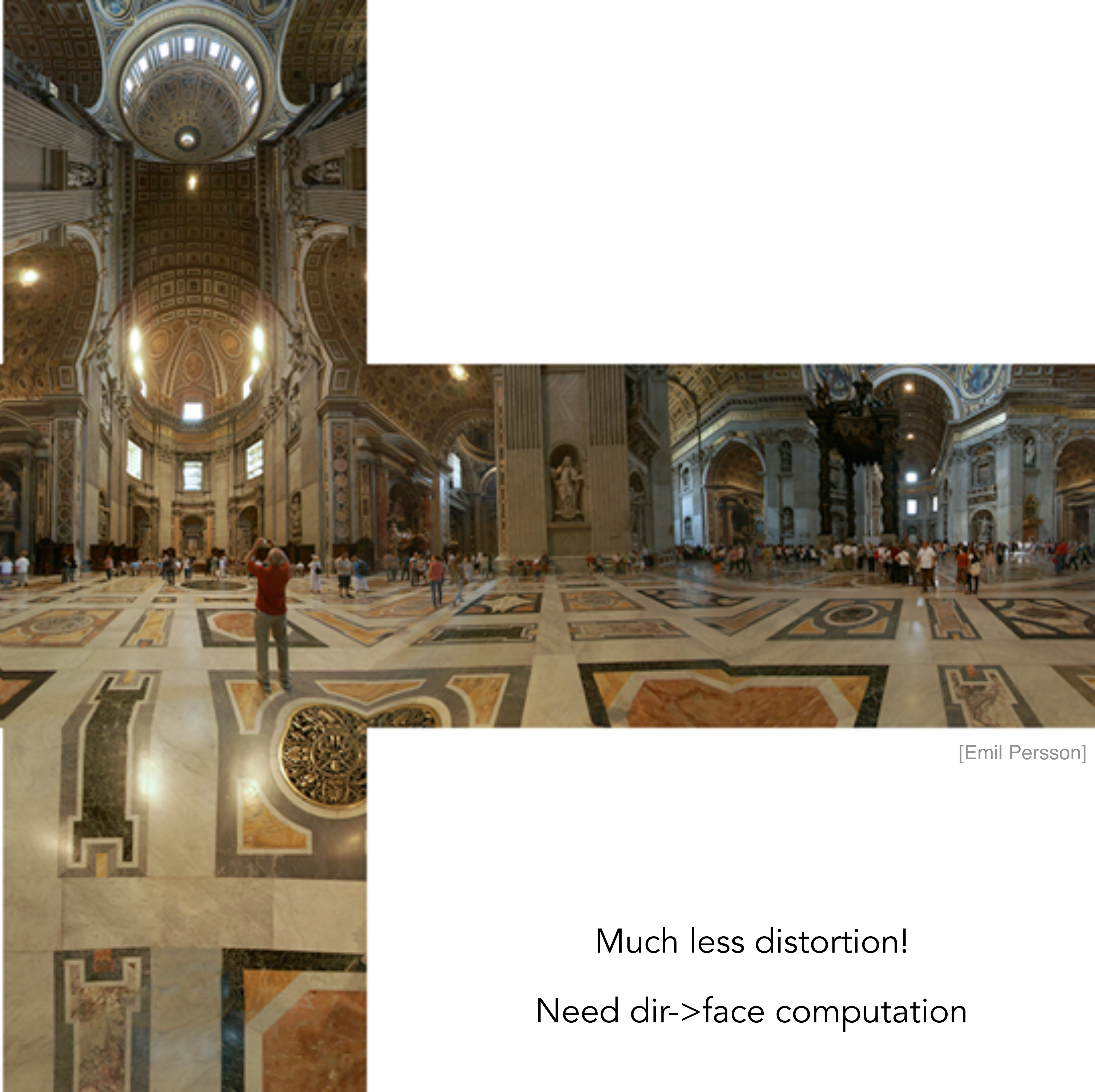
将环境光照信息记录在一个立方体表面上，但会需要额外判断某一方向上的光照应该记录在立方体的哪个面上，计算量更大





A vector maps to cube point along that direction.

The cube is textured with 6 square texture maps.



[Emil Persson]

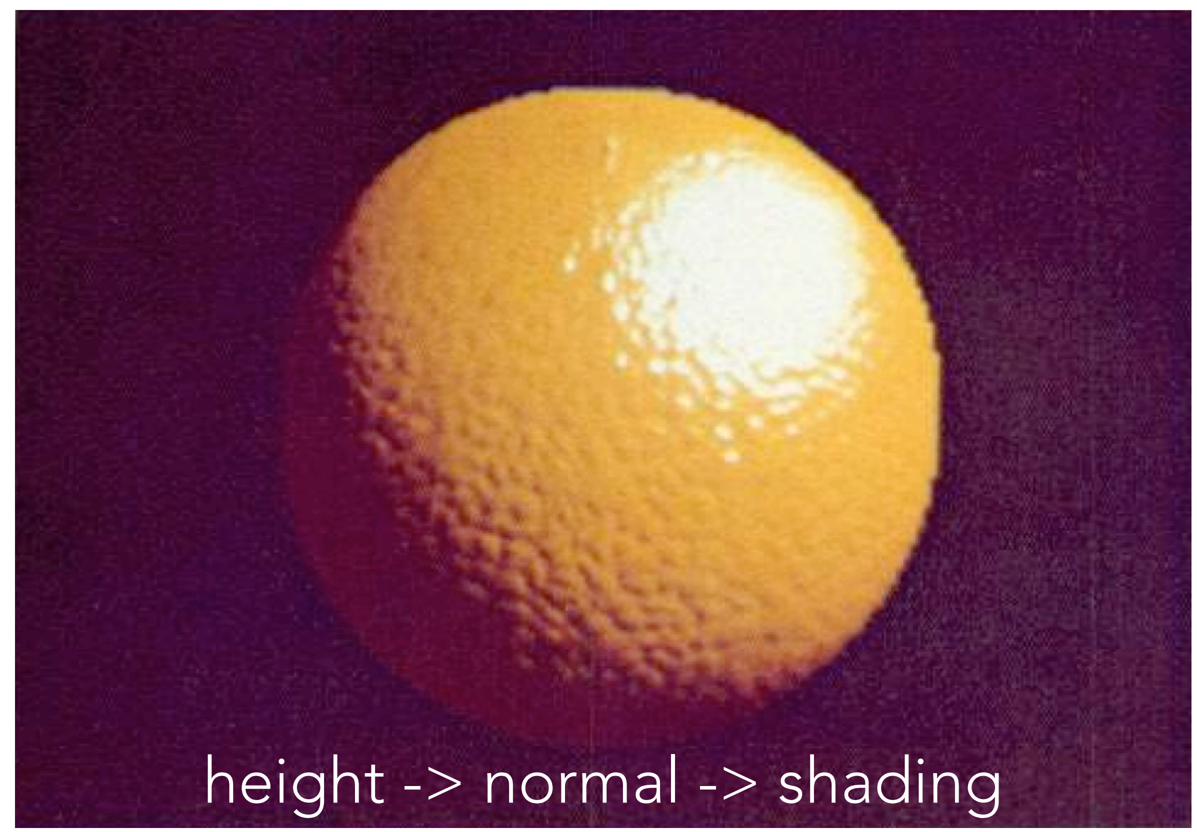
Much less distortion!

Need dir->face computation

Textures can affect shading!

- Textures doesn't have to only represent colors
  - What if it stores the height / normal?
  - Bump / normal mapping
  - **Fake** the detailed geometry

Relative height to the underlying surface



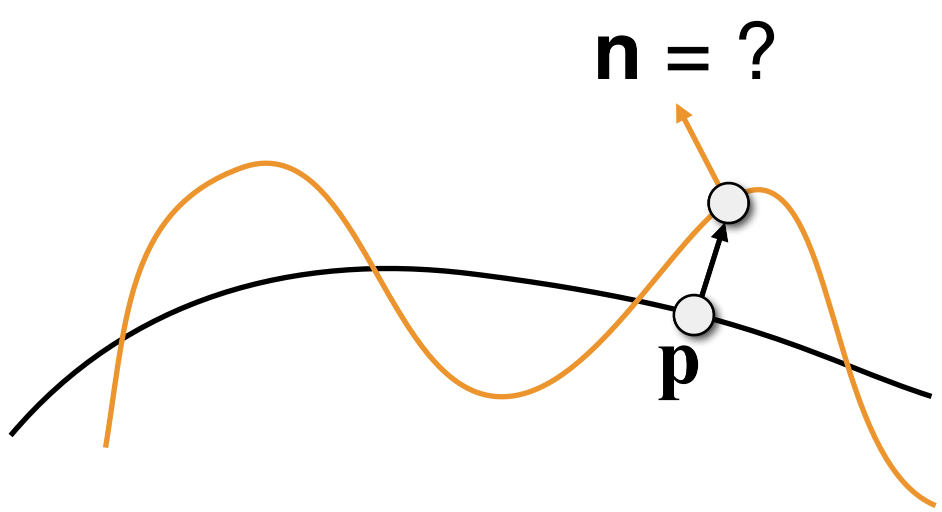
Bump Mapping

记录了纹理的高度移动，并不改变原来模型的几何信息，通过法线扰动，得到模拟出来的着色效果，以假乱真

Adding surface detail without adding more triangles

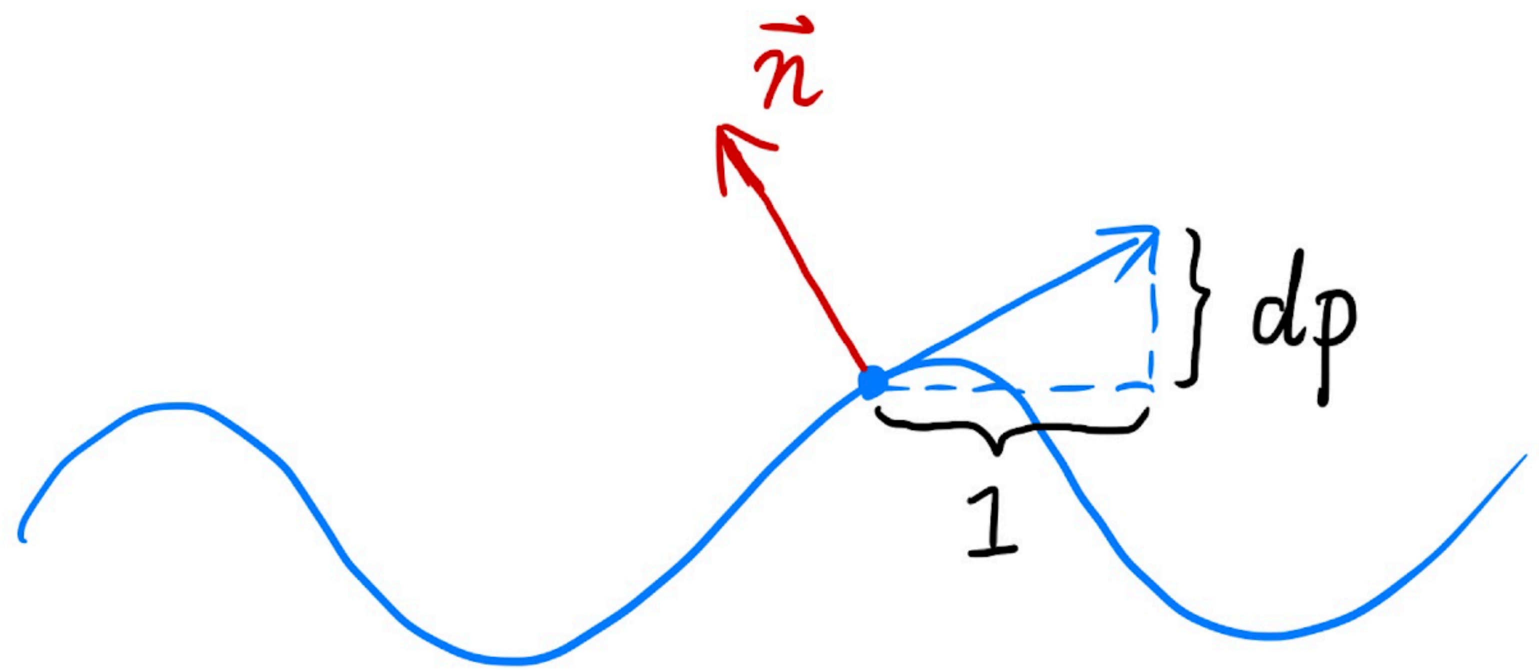
- Perturb surface normal per pixel
  (for shading computations only)
- “Height shift” per texel defined by a texture
- How to modify normal vector?

Bump texture
b(u,v)



How to perturb the normal (in flatland)

- Original surface normal n(p) = (0, 1)
- Derivative at p is dp = c * [h(p+1) - h(p)]
- Perturbed normal is then n(p) = **(-dp, 1)**.normalized()



二维上计算法线的方式是差分求导再旋转

How to perturb the normal (in 3D)

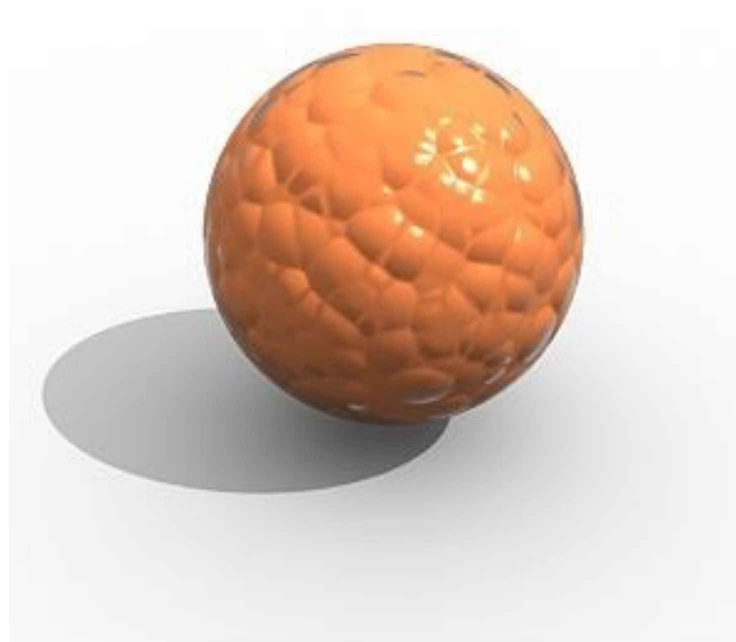
- Original surface normal n(p) = (0, 0, 1)
- Derivatives at p are
  - dp/du = c1 * [h(**u**+1) - h(**u**)]
  - dp/dv = c2 * [h(**v**+1) - h(**v**)]
- Perturbed normal is n = **(-dp/du, -dp/dv, 1)**.normalized()
- Note that this is in **local coordinate**!
  More will be elaborated in FAQ of HW3

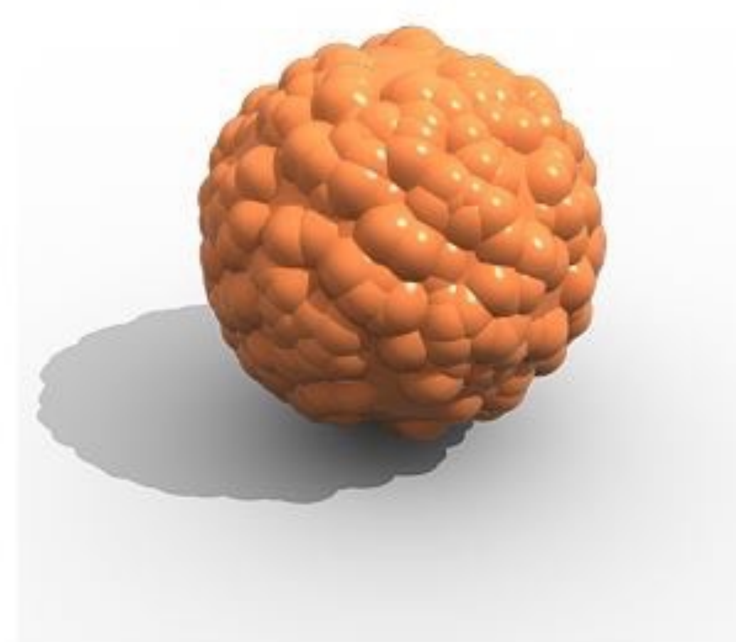
凹凸贴图是有局限性的，因为其本质只是改变了这个面上面呈现出的颜色，这个凹凸的面本质上还是光滑的

Textures can affect shading!

- **Displacement mapping** — a more advanced approach
  - Uses the same texture as in bumping mapping
  - Actually **moves the vertices**

与凹凸贴图类似，但位移贴图是真的改变了几何信息，去对模型的顶点做位移，会比凹凸贴图更加逼真，但对模型的精度(三角面数量)要求更高，并且运算量也会随之上升

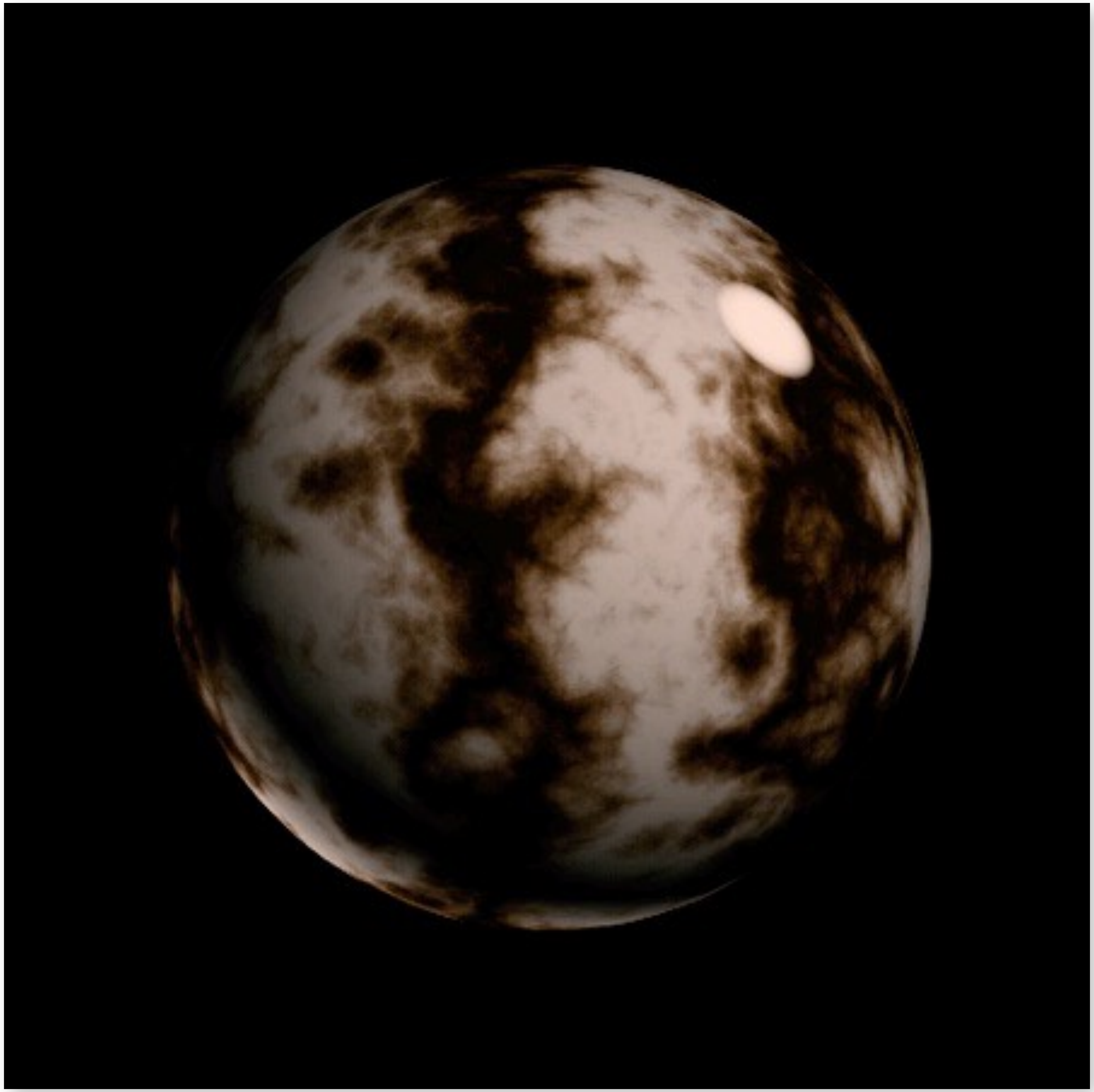


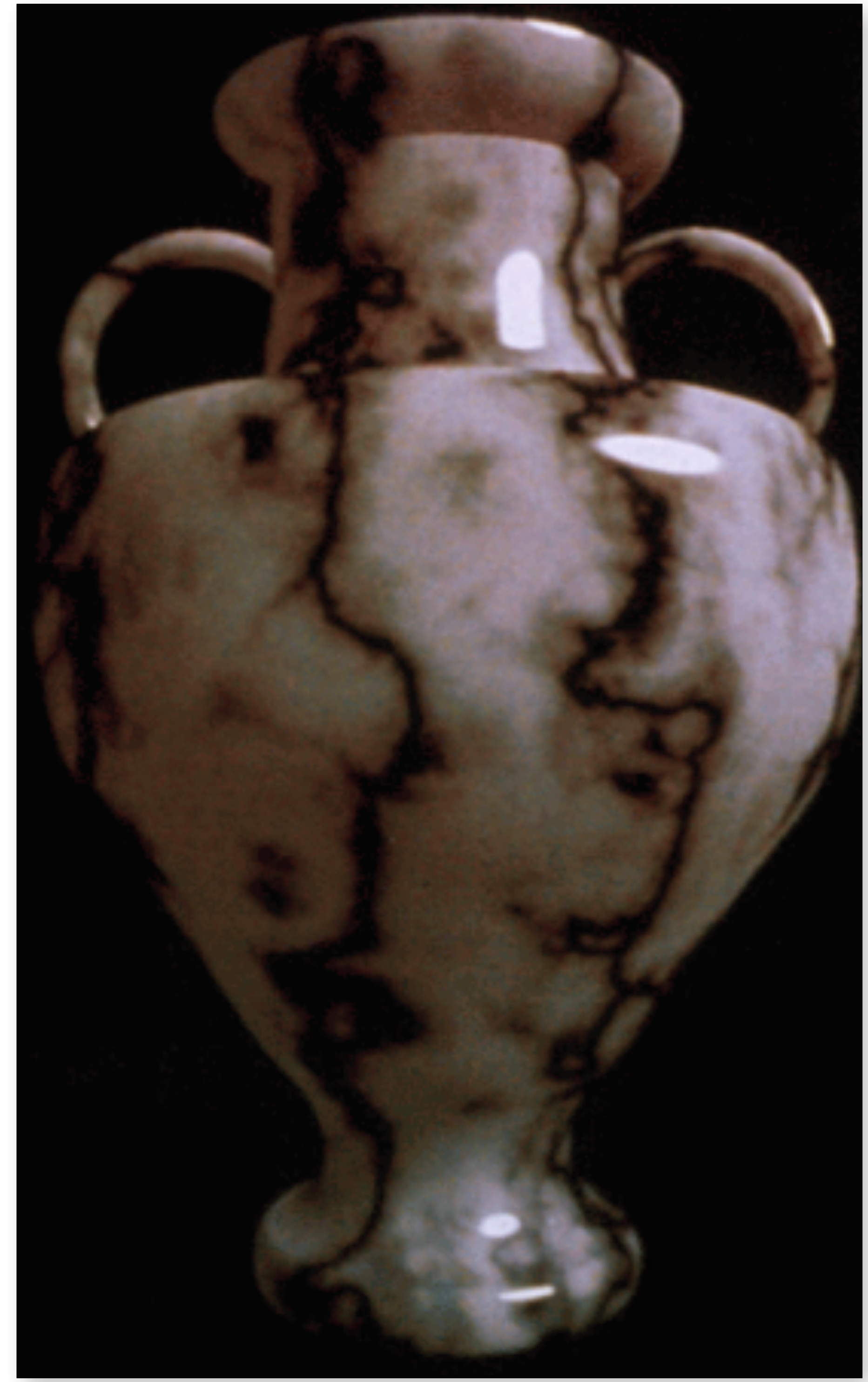


Bump / **Normal** mapping   Displacement mapping

3D Procedural Noise + Solid Modeling

三维的纹理，并非真正生成了纹理的图，而是定义空间中任意点的颜色 定义三维空间中的噪声函数，再通过映射，得到预想的效果





Perlin noise, Ken Perlin

Provide Precomputed Shading

将环境光进行预计算处理，再附在原先纹理上做一层遮蔽，再将纹理贴到模型上

Simple shading

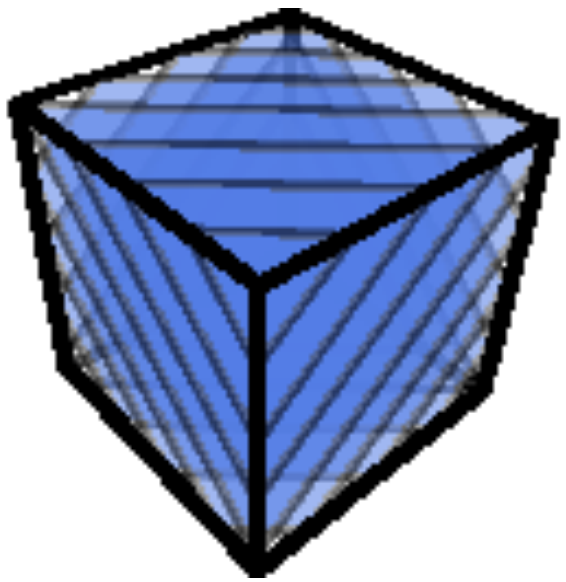
Ambient occlusion texture map

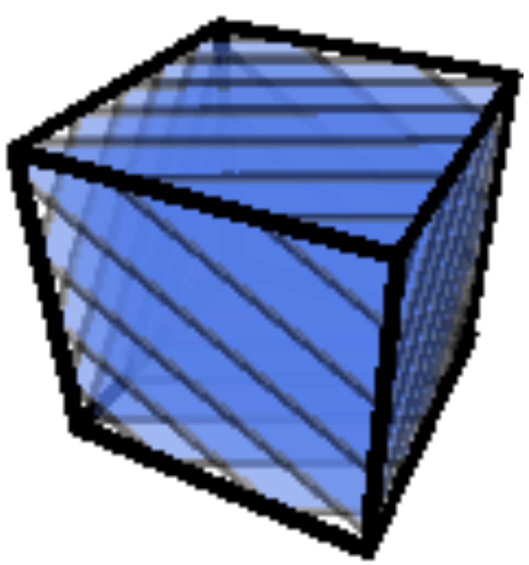
With ambient occlusion

Autodesk

3D Textures and Volume Rendering

广泛应用于物体渲染，如核磁共振等扫描后得到的体积信息，通过这些信息进行渲染，得到结果





Marc Levoy

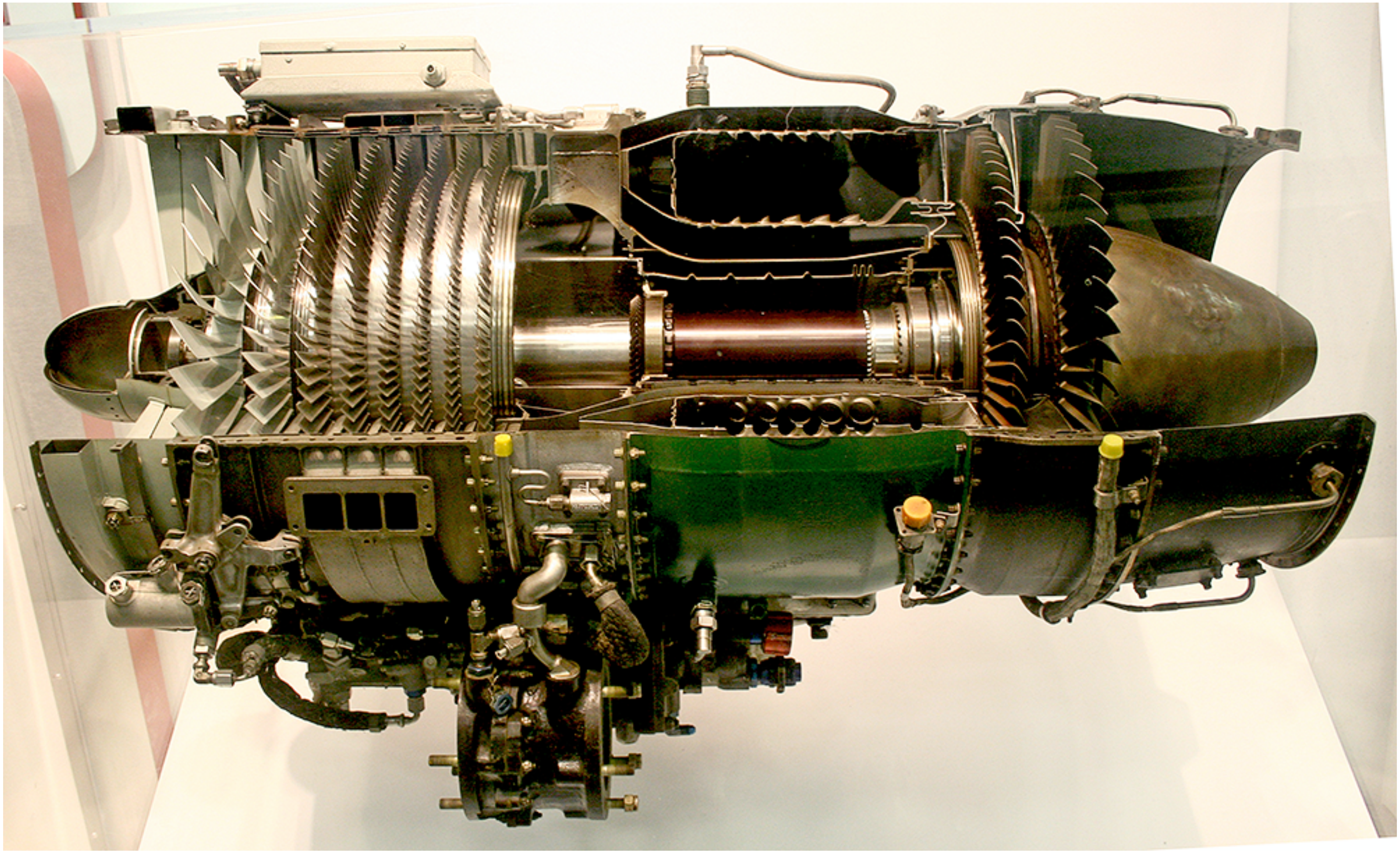
Today

- Shading 3
  - Applications of textures
- Introduction to geometry
  - Examples of geometry
  - Various representations of geometry

Examples of Geometry

Examples of Geometry

Examples of Geometry



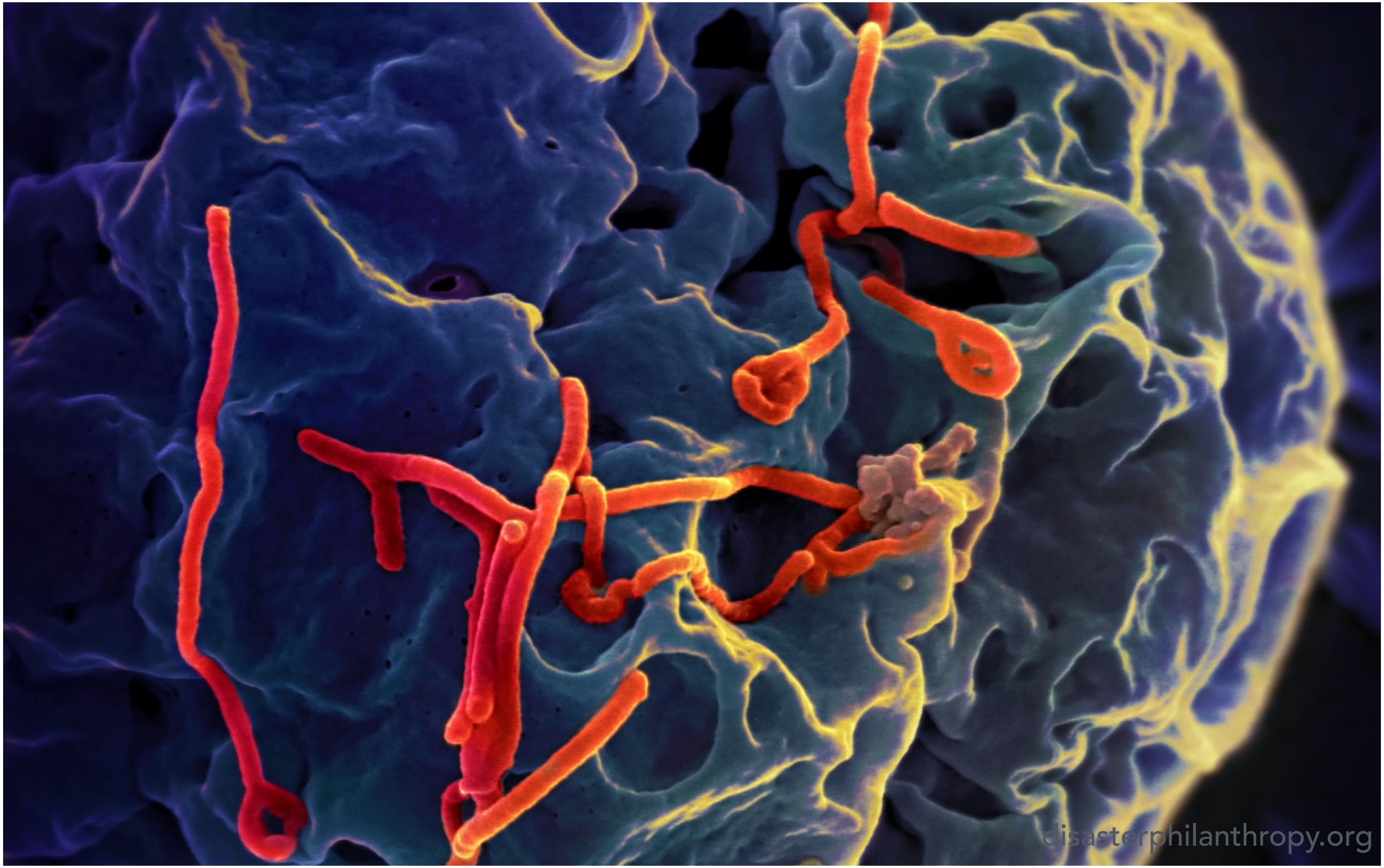
Examples of Geometry

Examples of Geometry

Examples of Geometry

Examples of Geometry

Examples of Geometry



Examples of Geometry

Adriana Franco, National Geographic

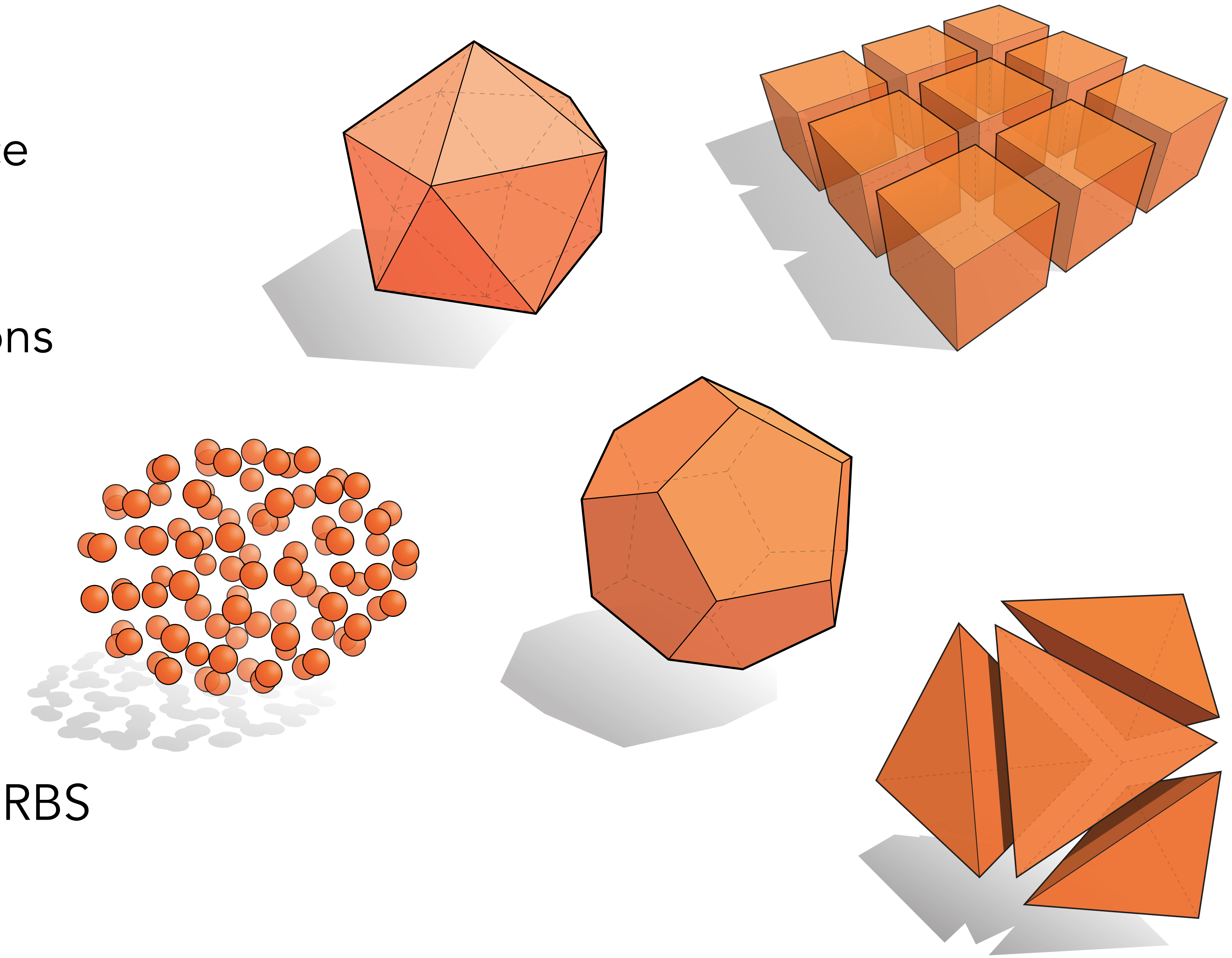
Many Ways to Represent Geometry

Implicit

- algebraic surface
- level sets
- distance functions
- …

Explicit

- point cloud
- polygon mesh
- subdivision, NURBS
- …



Each choice best suited to a different task/type of geometry

"Implicit" Representations of Geometry

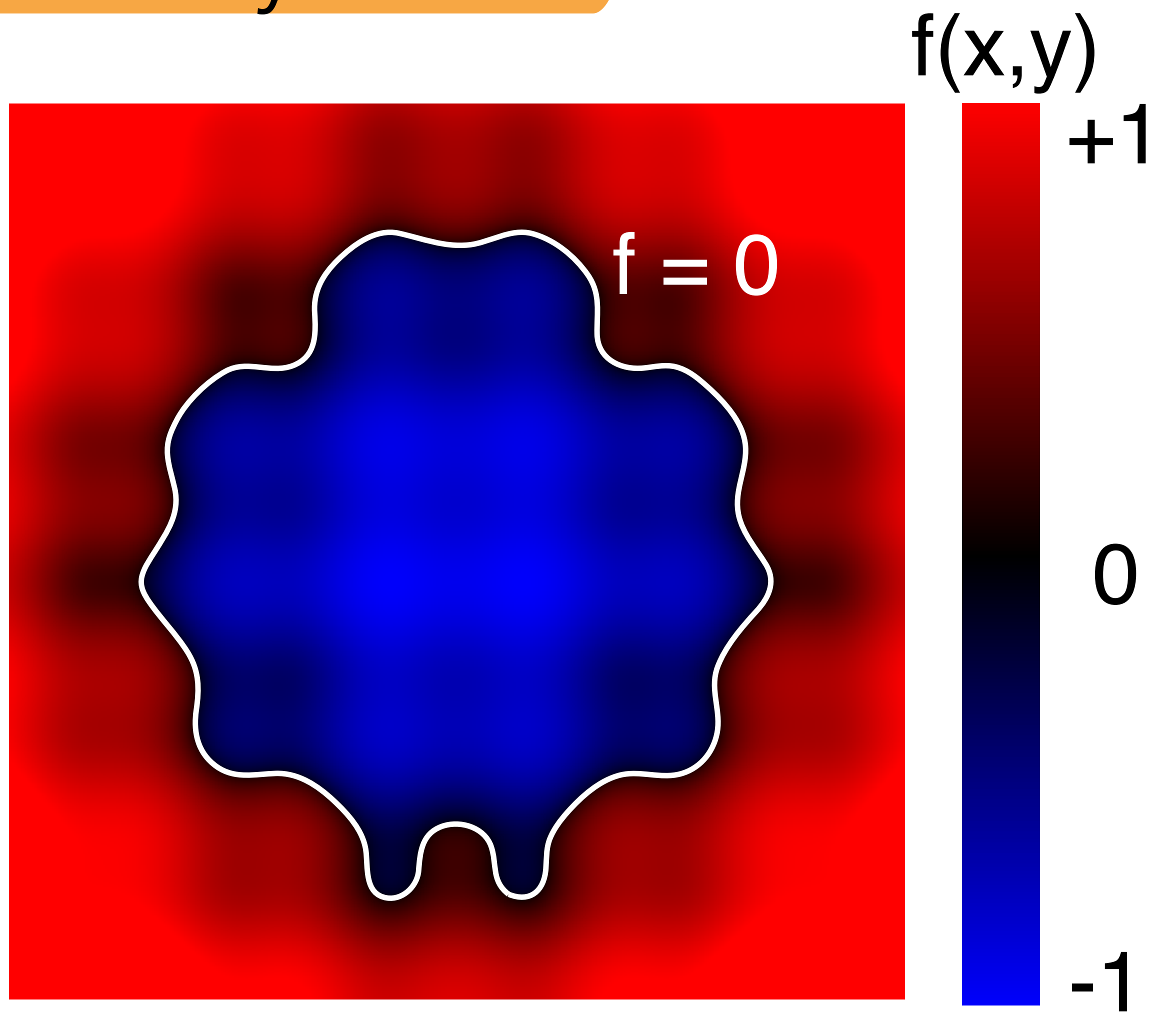
用空间中的满足一定条件的点的集合来表示面，隐式几何不会表示点的具体位置信息，而是告诉我们这些点满足的函数关系

Based on classifying points

- Points satisfy some specified relationship

E.g. sphere: all points in 3D, where $x^2+y^2+z^2 = 1$

More generally, $f(x,y,z) = 0$

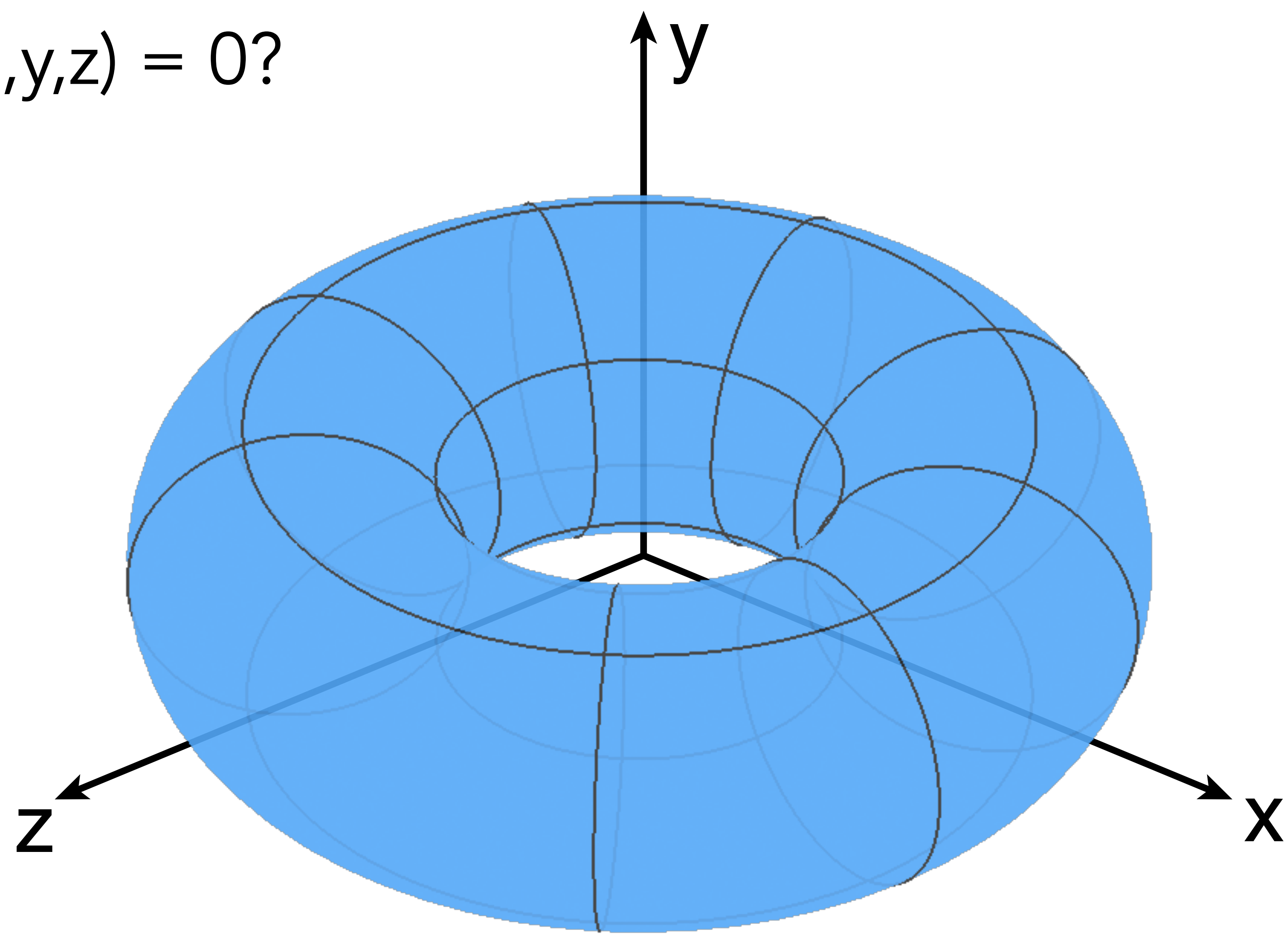


Implicit Surface – Sampling Can Be Hard

我们很难看出隐式想表达的形状是什么，也无法判定所有在表面上的点，但对于判断点的位置关系(在内，在外还是在表面)会很方便

$$f(x, y, z) = (2 - \sqrt{x^2 + y^2})^2 + z^2 - 1$$

What points lie on f(x,y,z) = 0?



Some tasks are hard with implicit representations

Implicit Surface – Inside/Outside Tests Easy

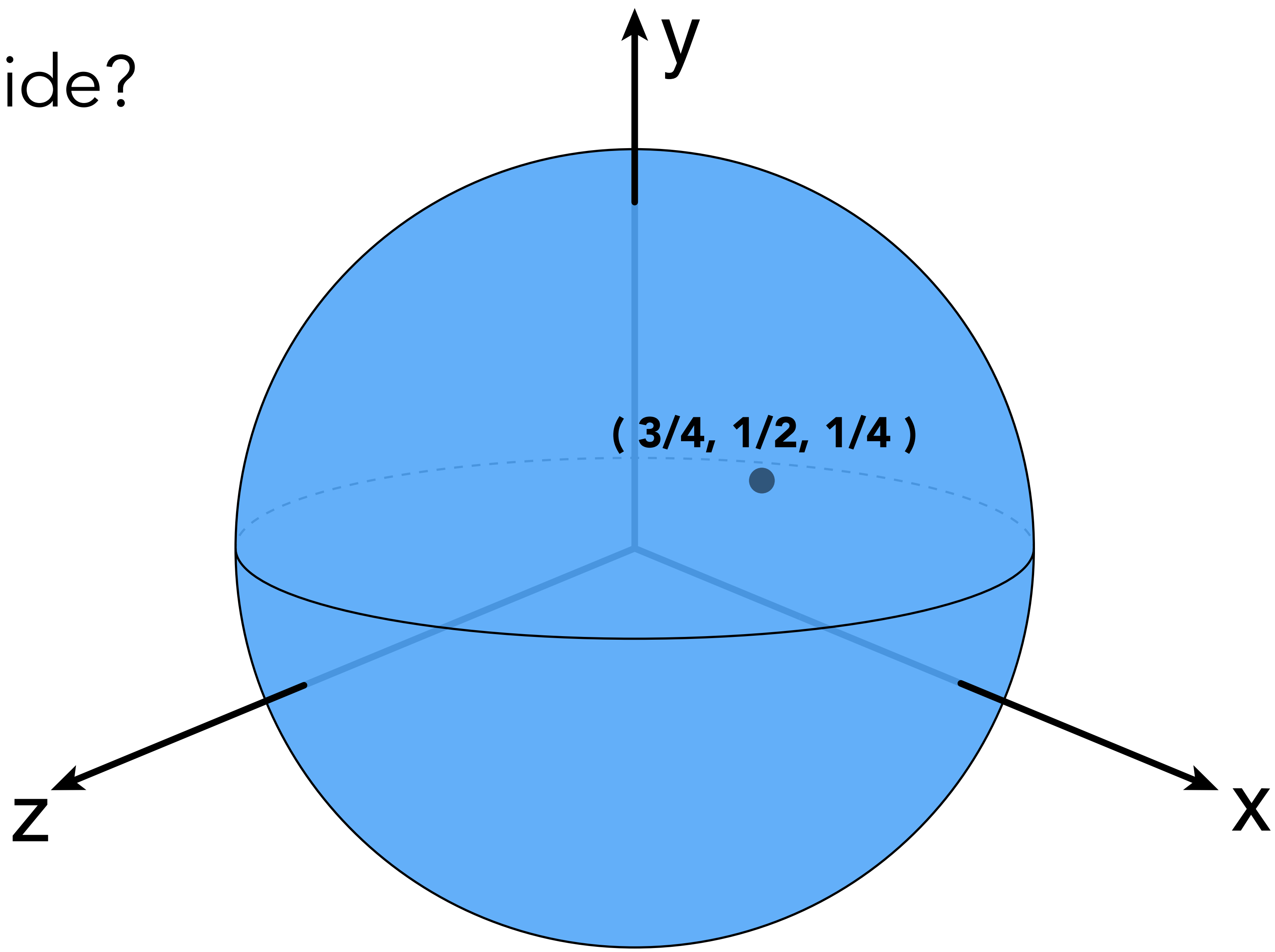
$$f(x, y, z) = x^2 + y^2 + z^2 - 1$$

Is (3/4, 1/2, 1/4) inside?

Just plug it in:

f(x,y,z) = –1/8 < 0

Yes, inside.



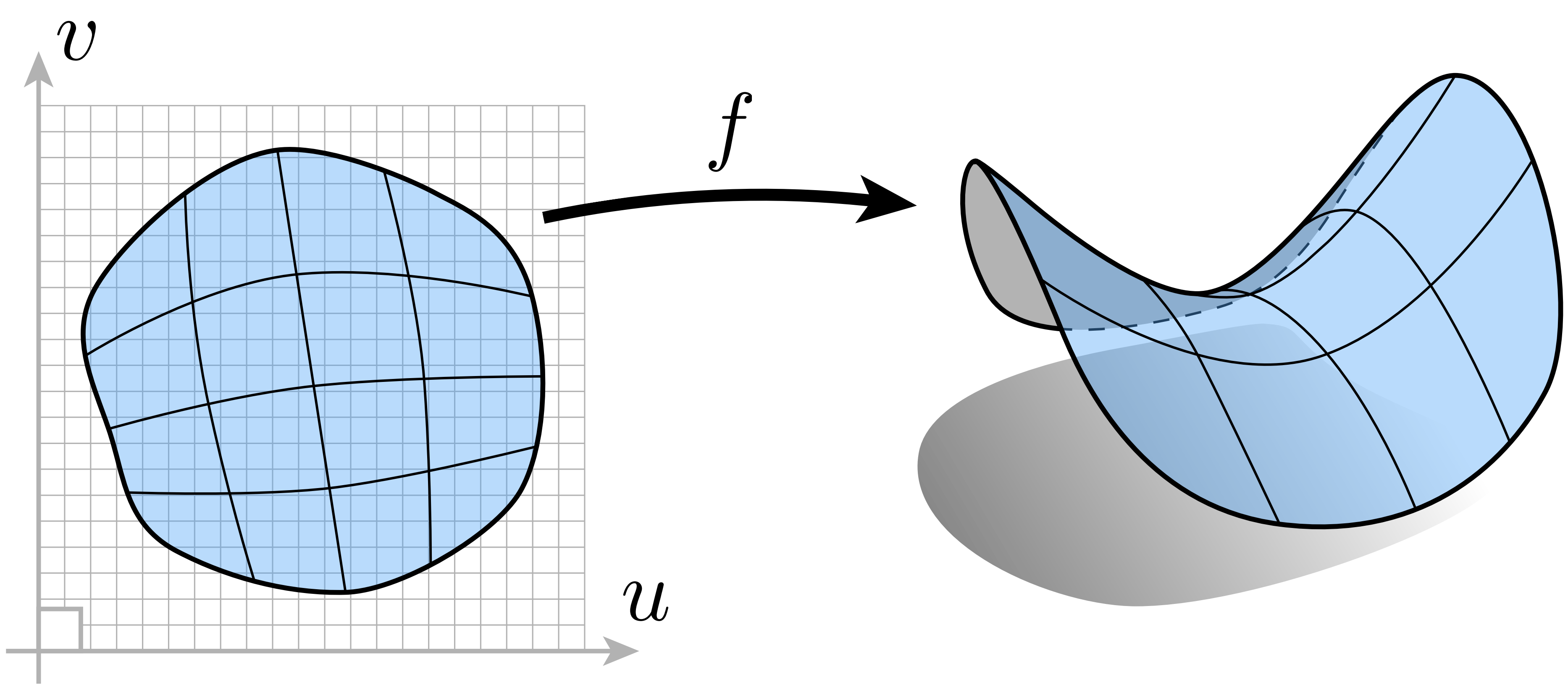
Implicit representations make some tasks easy

"Explicit" Representations of Geometry

几何的显式表示：显式建模就是类似纹理查表一样，直接用一个表储存这个表面上的所有点等直接得到曲面的方式

All points are **given directly** or **via parameter mapping**

Generally: $f : \mathbb{R}^2 \to \mathbb{R}^3; (u, v) \mapsto (x, y, z)$

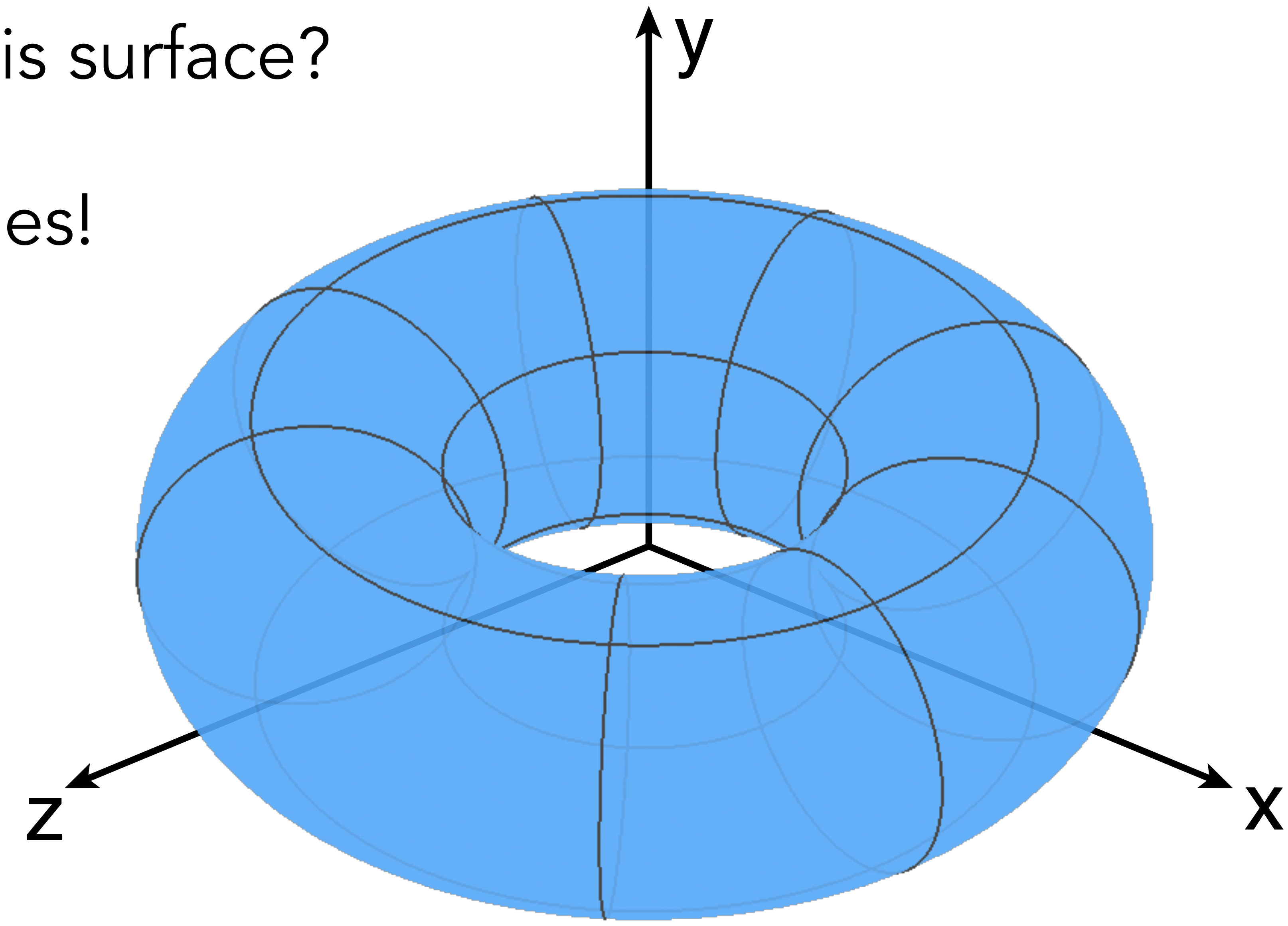


Explicit Surface – Sampling Is Easy

$$f(u,v) = ((2+\cos u)\cos v, (2+\cos u)\sin v, \sin u)$$

What points lie on this surface?

Just plug in (u,v) values!

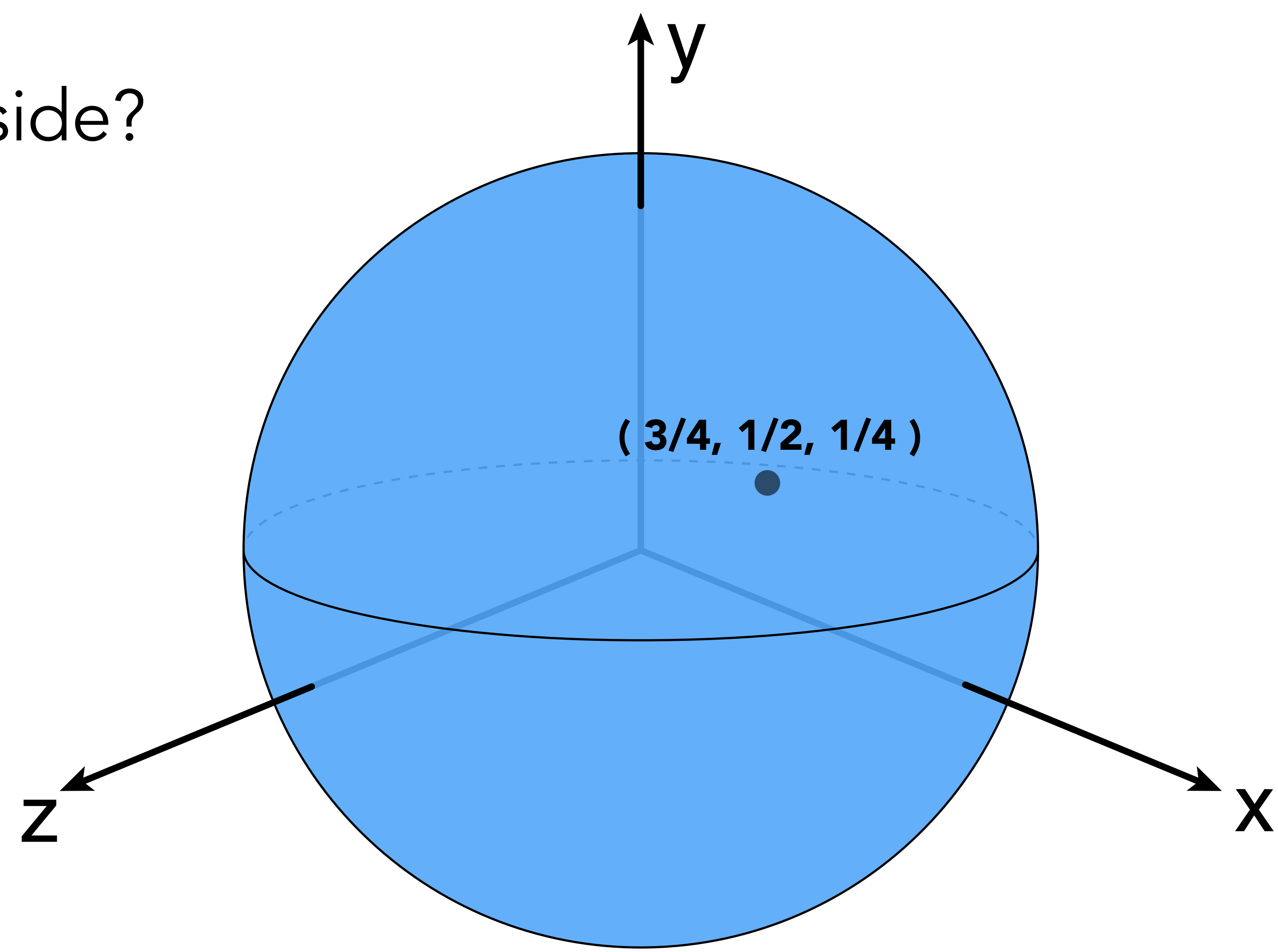


Explicit representations make some tasks easy

Explicit Surface – Inside/Outside Test Hard

$$f(u,v) = (\cos u \,\sin v, \sin u \,\sin v, \cos v)$$

Is (3/4, 1/2, 1/4) inside?



Some tasks are hard with explicit representations

No “Best” Representation – Geometry is Hard!

“I hate meshes.
I cannot believe how hard this is.
Geometry is hard.”

— David Baraff
Senior Research Scientist
Pixar Animation Studios

Slide cribbed from Keenan Crane, cribbed from Jeff Erickson.

Best Representation
Depends on the Task!

More Implicit Representations in Computer Graphics

Many Implicit Representations in Graphics

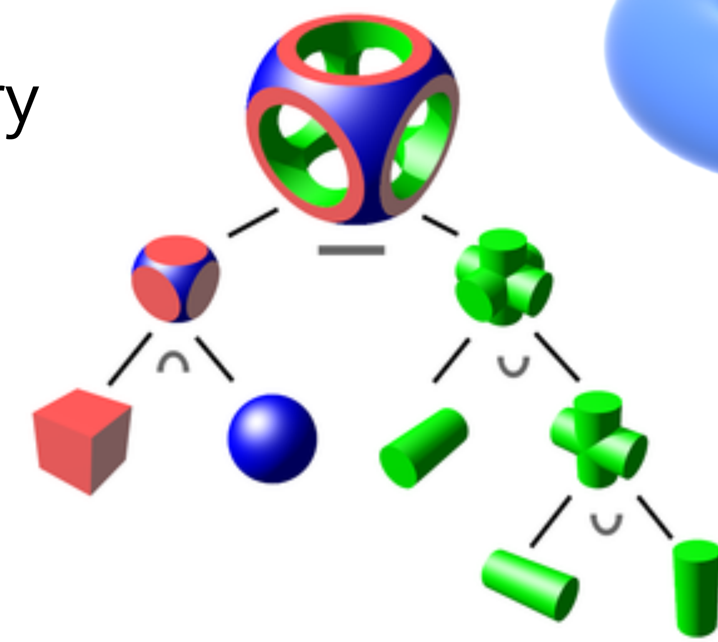
Algebraic surfaces

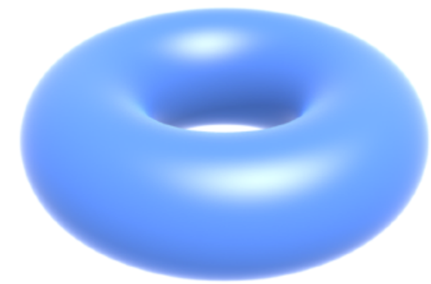
Constructive solid geometry

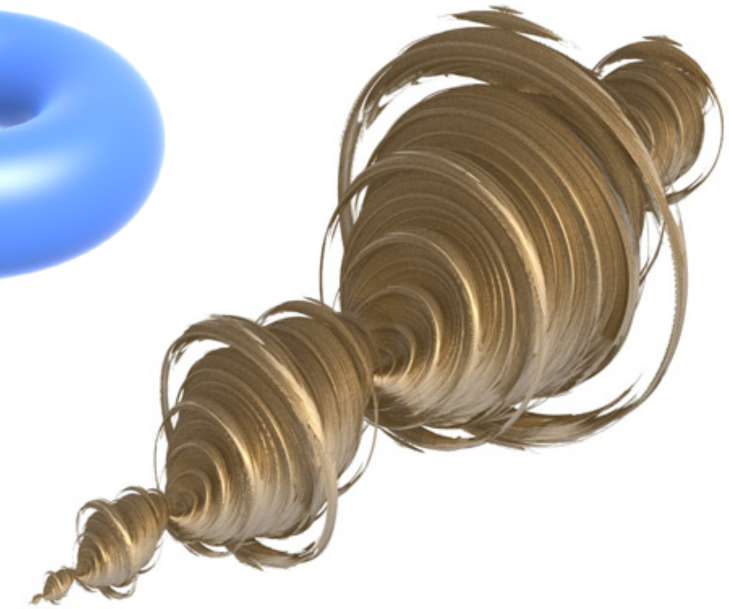
Level set methods

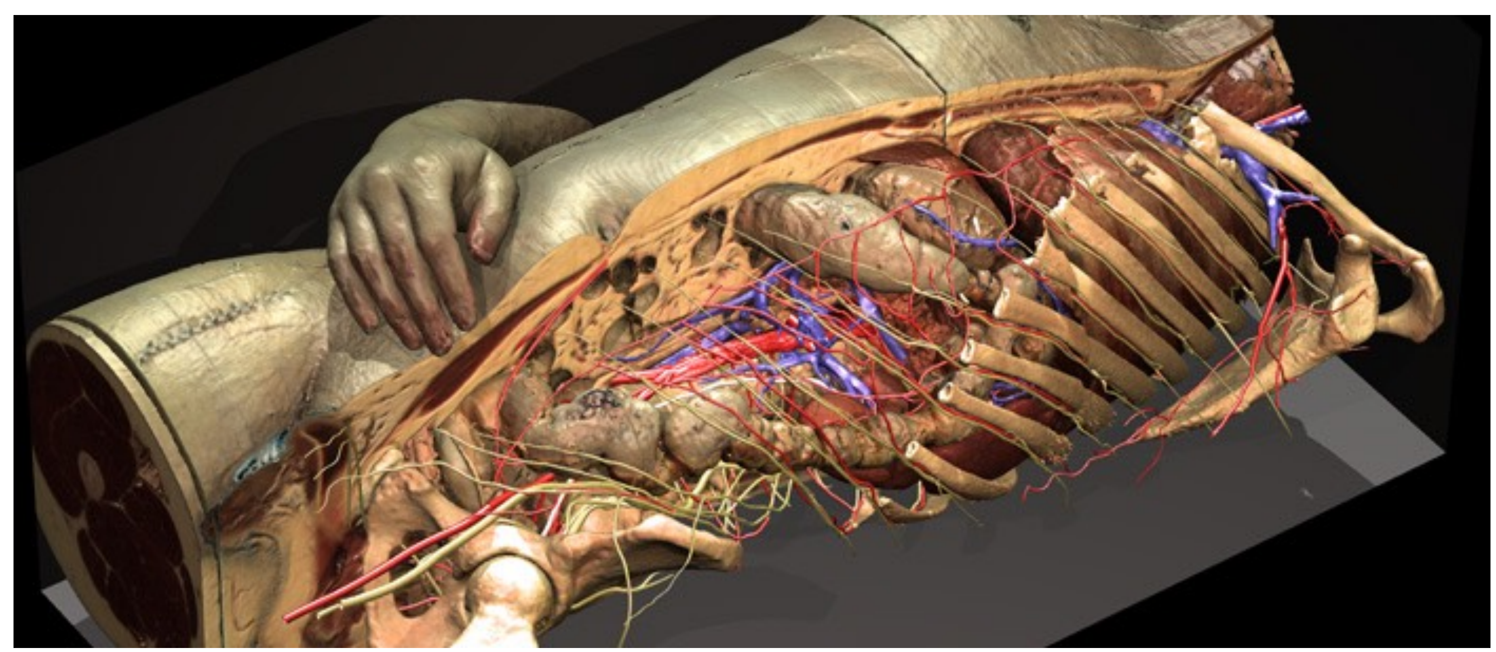
Fractals

…



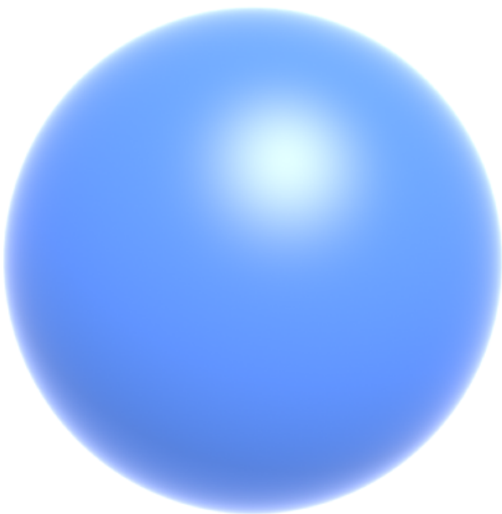




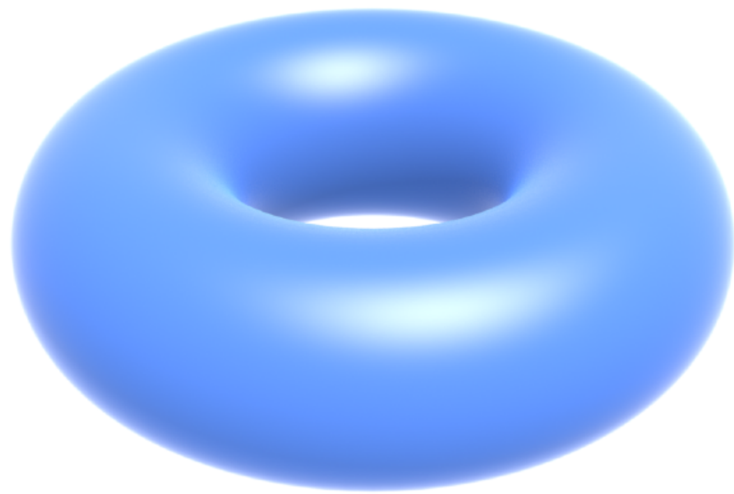


Algebraic Surfaces (Implicit)

Surface is zero set of a polynomial in x, y, z



$$x^2+y^2+z^2=1$$



$$(R-\sqrt{x^2+y^2})^2+z^2=r^2$$

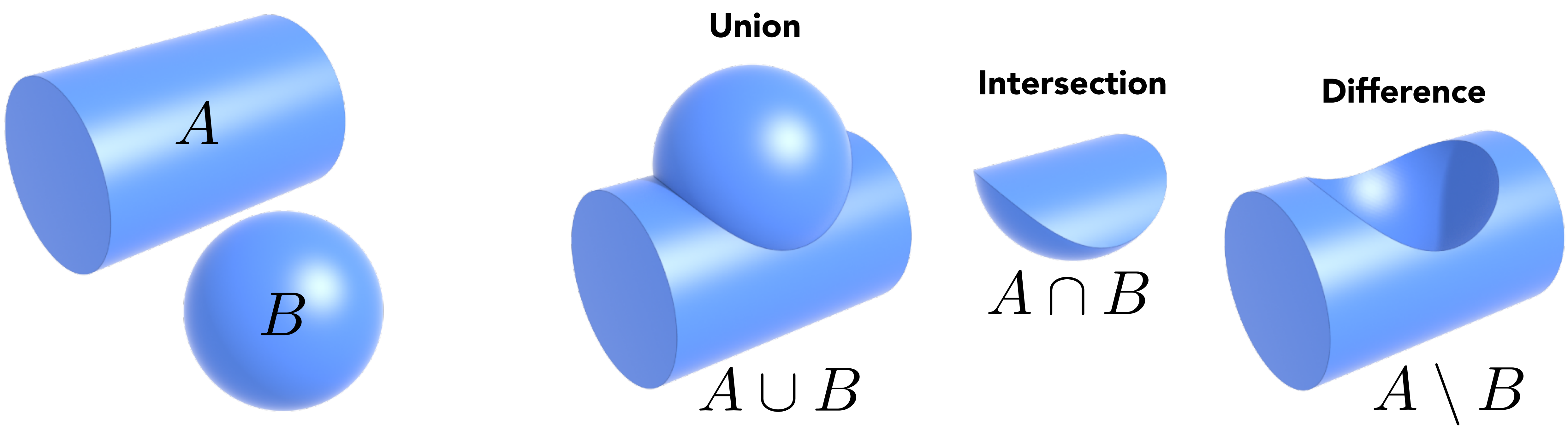
$$(x^2+\frac{9y^2}{4}+z^2-1)^3 = x^2z^3+\frac{9y^2z^3}{80}$$

More complex shapes?

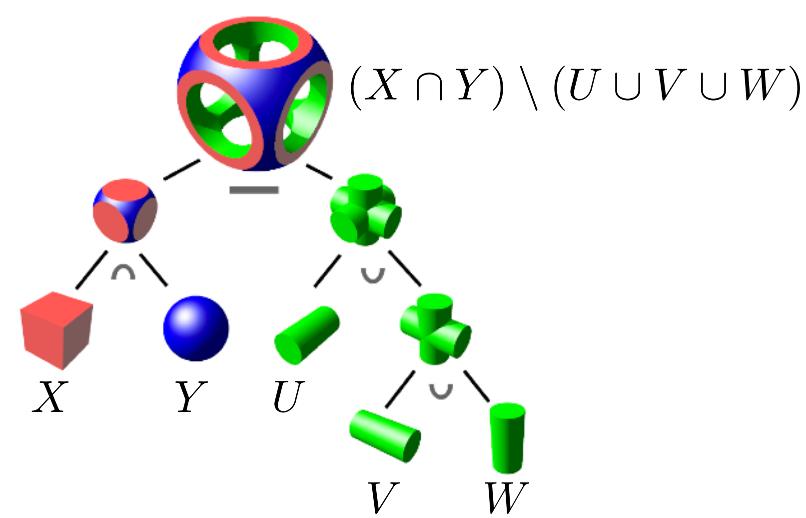
Constructive Solid Geometry (Implicit)

CSG就是使用基础几何的简单布尔运算来得到复杂几何

Combine implicit geometry via Boolean operations



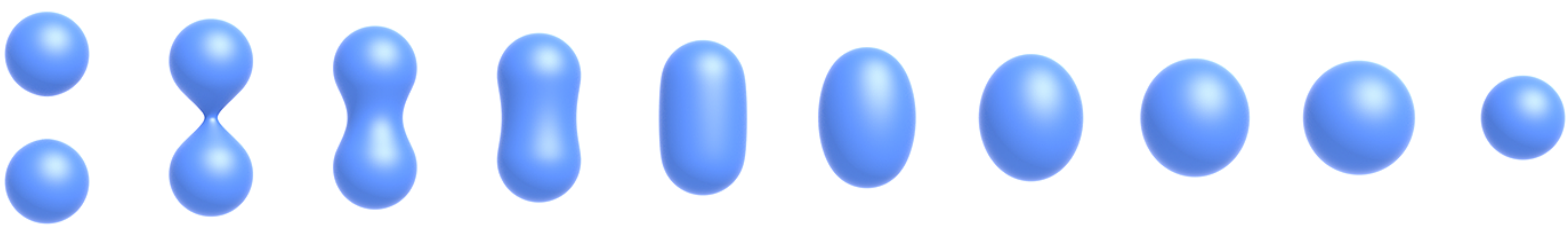
Boolean expressions:



Distance Functions (Implicit)

Instead of Booleans, gradually blend surfaces together using

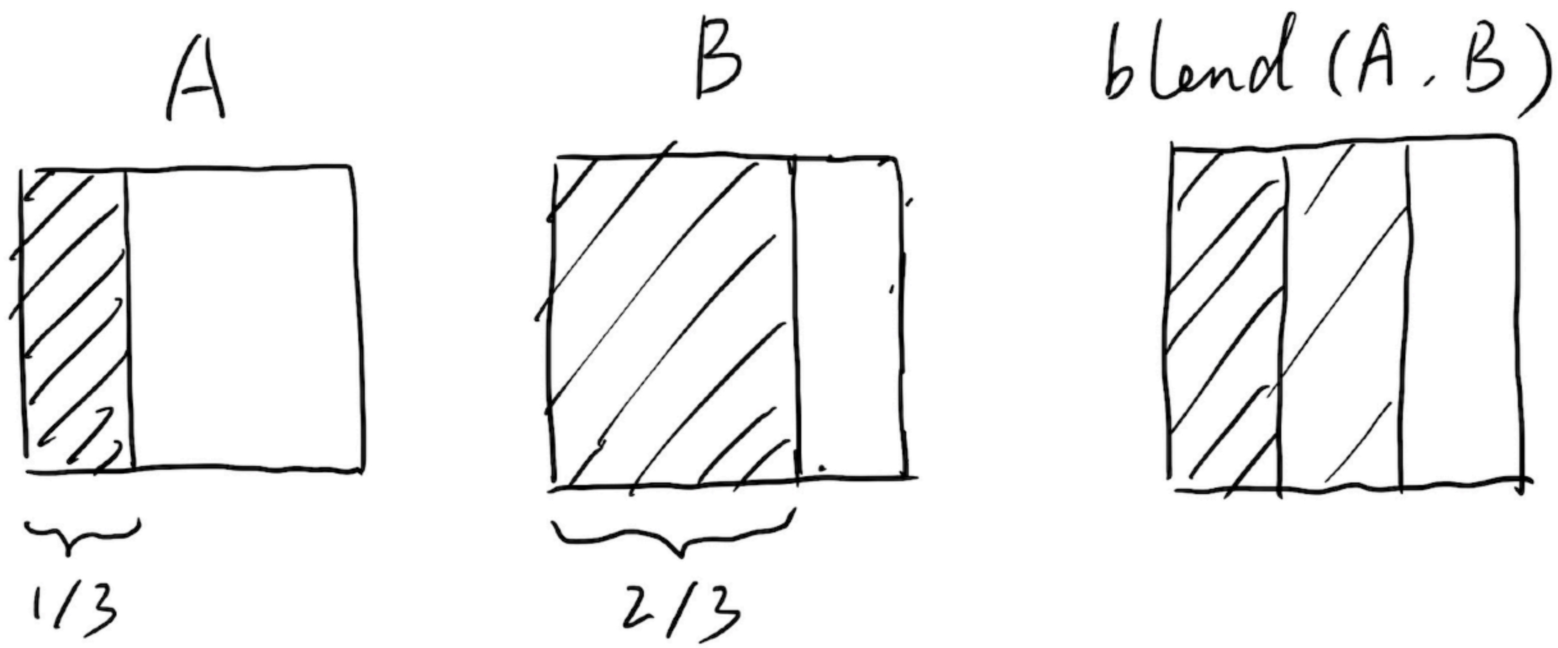
Distance functions:
giving minimum distance (could be **signed** distance)
from anywhere to object



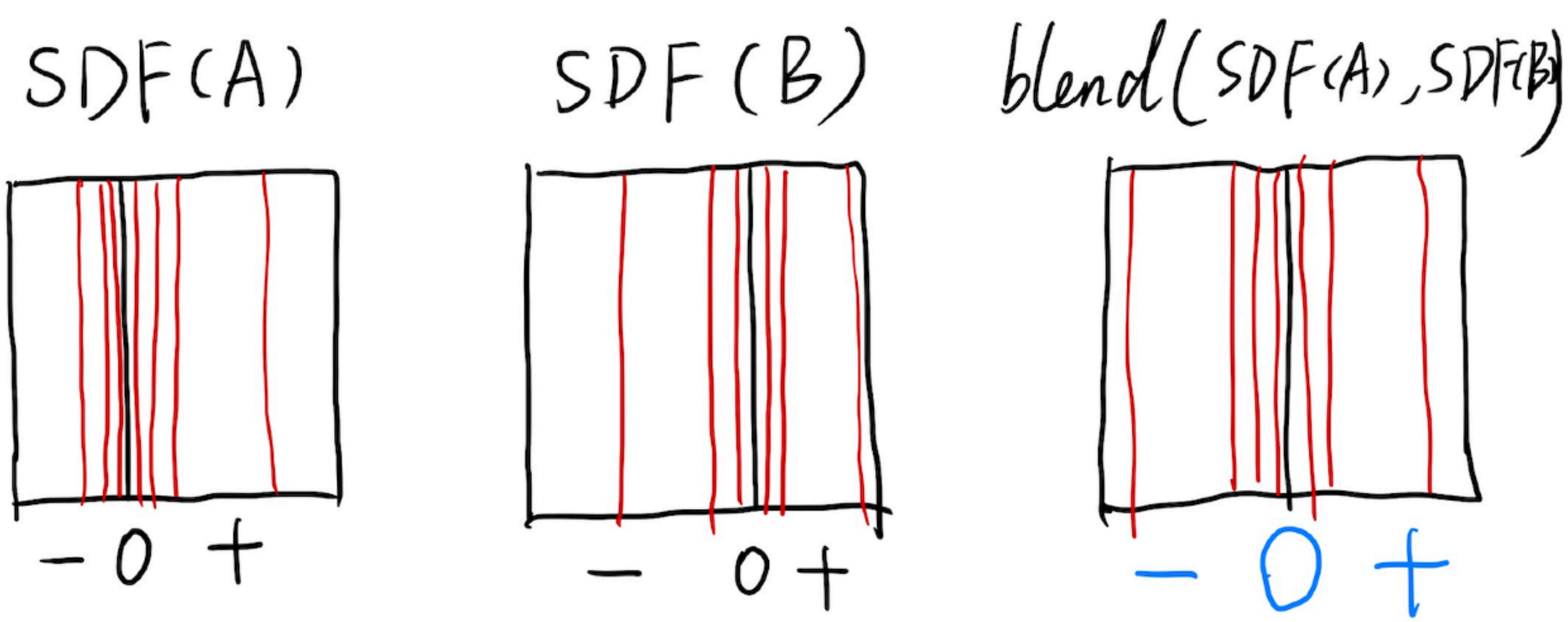
首先距离函数是表达空间中任意一个点到几何表面的最小距离的函数。这种隐式表达方式就是用一个点到几何表面的距离来表示这个表面（不仅仅是外部点还有内部的点）

Distance Functions (Implicit)

An Example: Blending a moving boundary

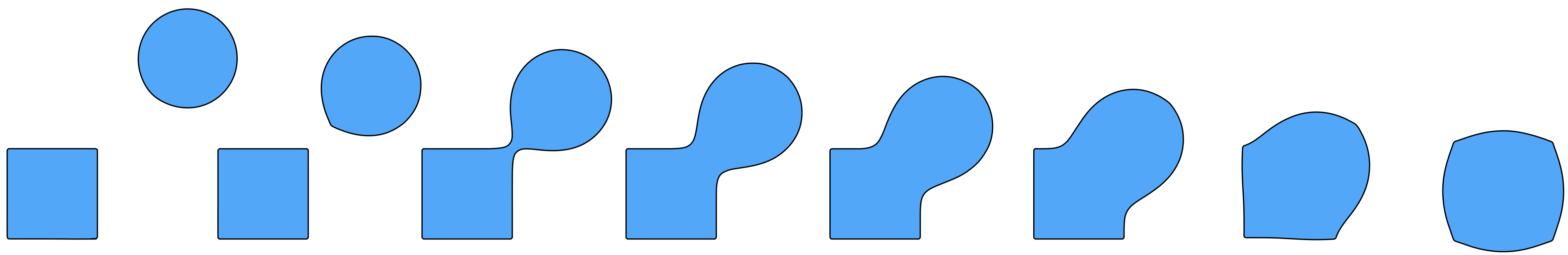


使用距离函数，融合（SDF）AB的距离函数值（在A边界右边正值B边界左边是负值）就可以直接得到正确的结果。



Blending Distance Functions (Implicit)

Can blend any two distance functions d1, d2:



Scene of Pure Distance Functions

See https://iquilezles.org/www/articles/raymarchingdf/raymarchingdf.htm

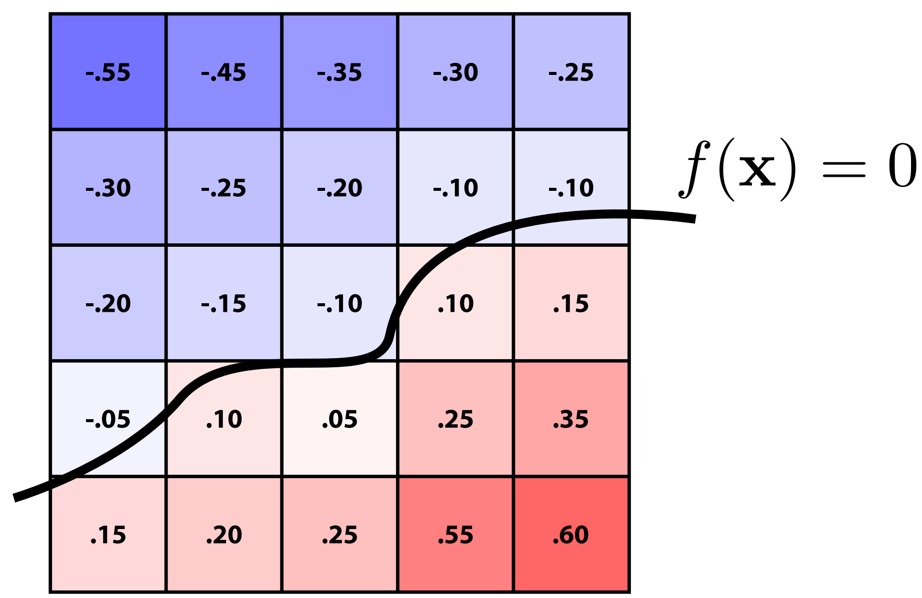
Level Set Methods (Also implicit)

（水平集）

水平集就相当于通过离散近似连续的距离函数的形式，它将距离函数表示为离散的数值网格，这种离散化的优势就在于可以统一表示所有有距离函数或无距离函数的模型。

Closed-form equations are hard to describe complex shapes

Alternative: store a grid of values approximating function

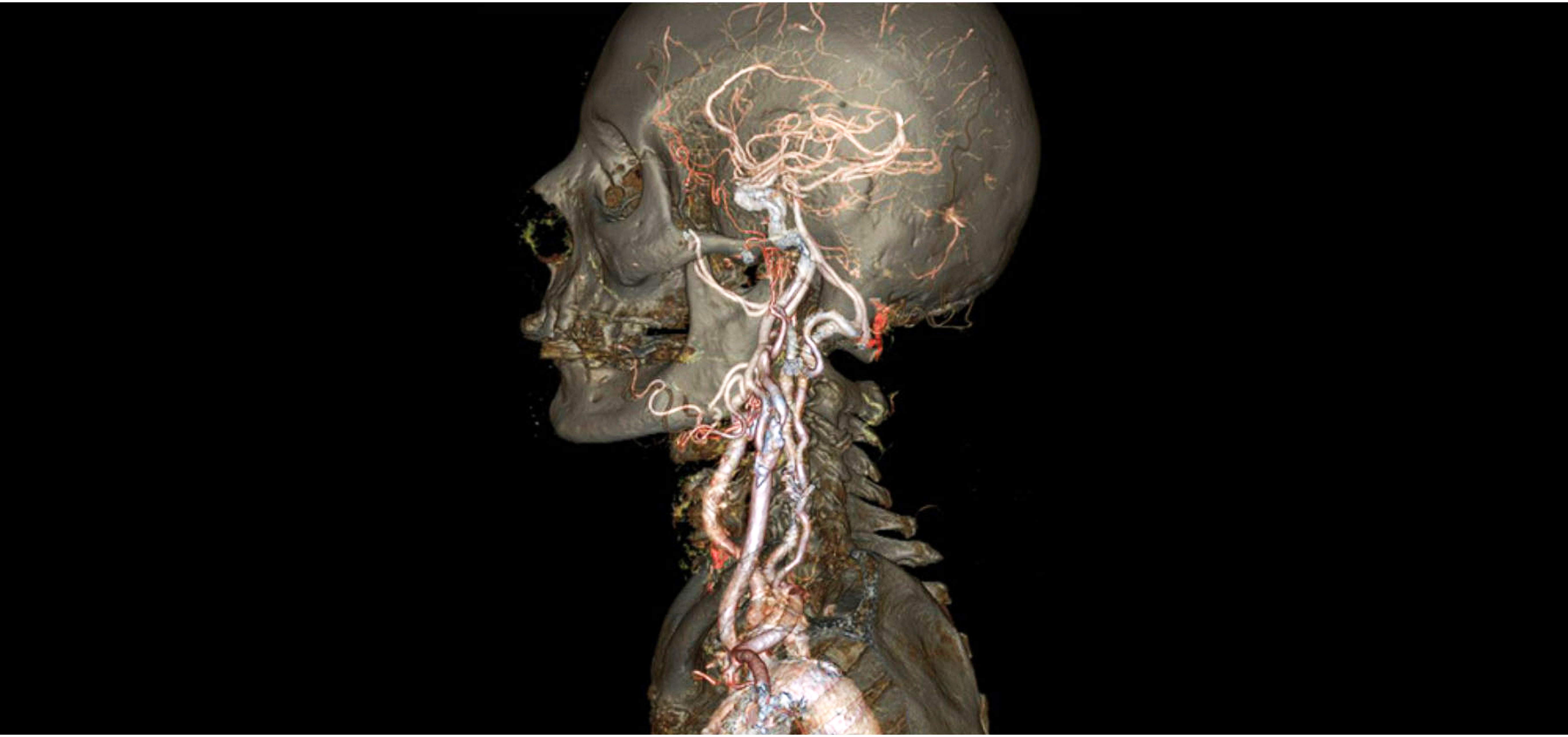


Surface is found where interpolated values equal zero

Provides much more explicit control over shape (like a texture)

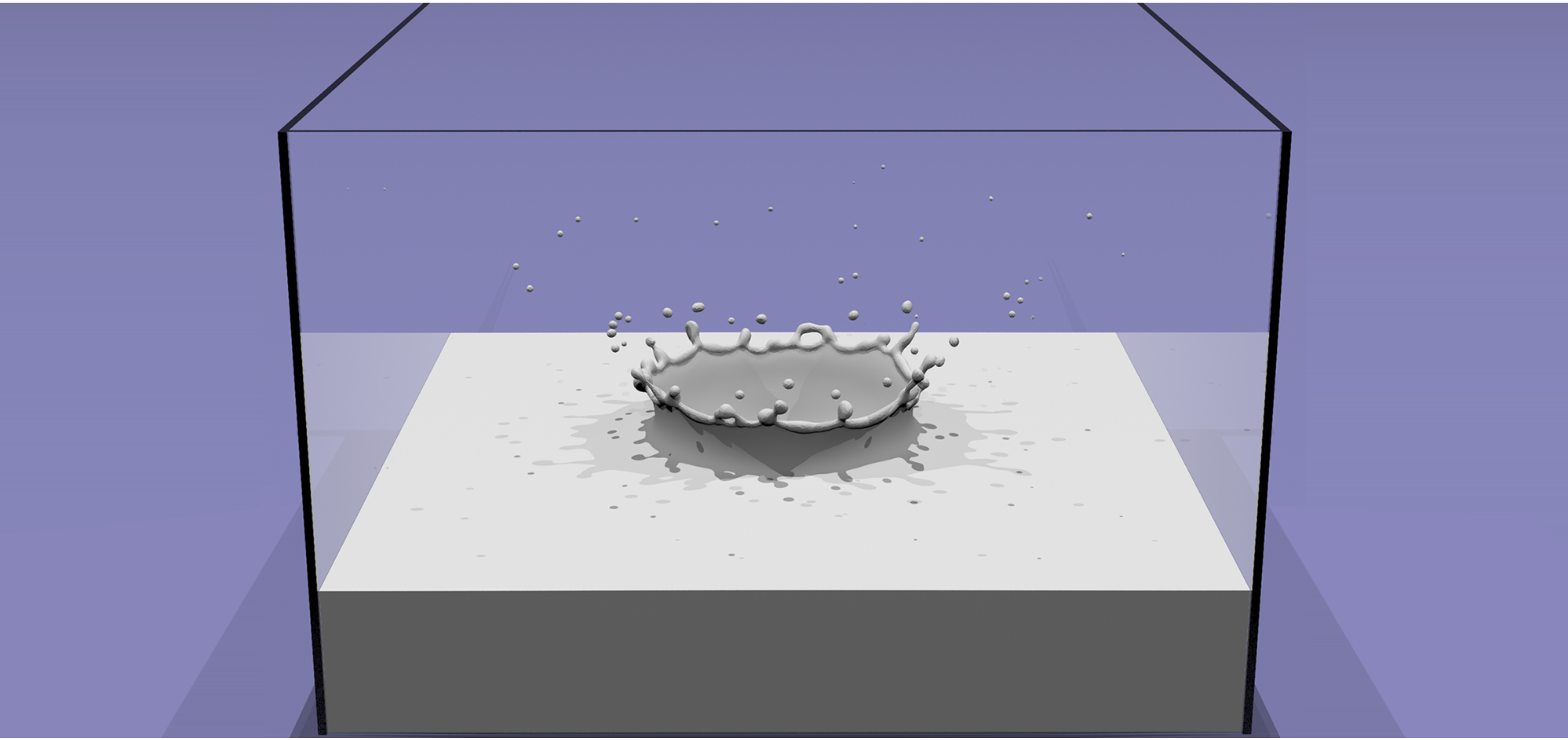
Level Sets from Medical Data (CT, MRI, etc.)

Level sets encode, e.g., constant tissue density



Level Sets in Physical Simulation

Level set encodes distance to air-liquid boundary



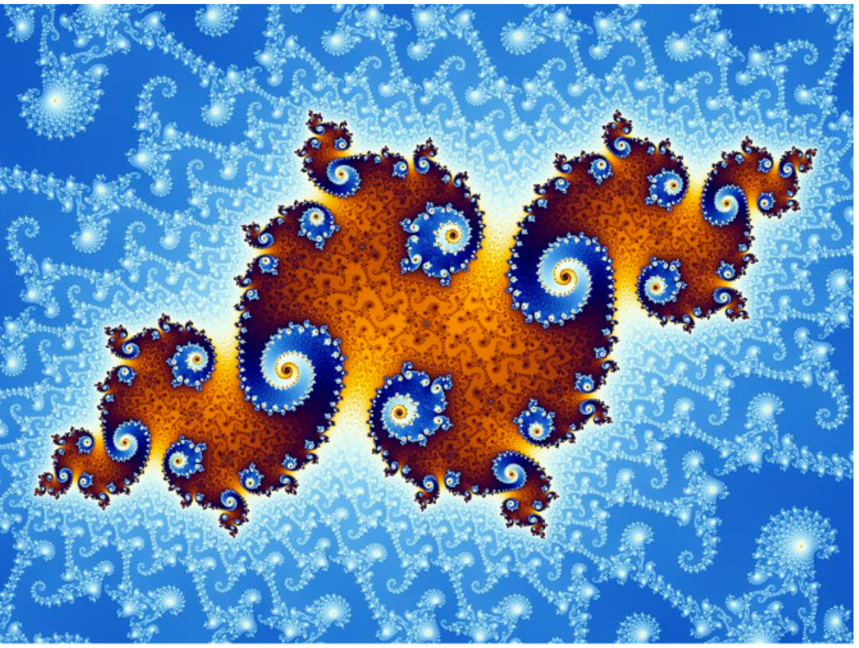
See http://physbam.stanford.edu

Fractals (Implicit)

Exhibit self-similarity, detail at all scales

“Language” for describing natural phenomena

Hard to control shape!



Implicit Representations - Pros & Cons

Pros:

- compact description (e.g., a function)
- certain queries easy (inside object, distance to surface)
- good for ray-to-surface intersection (more later)
- for simple shapes, exact description / no sampling error
- easy to handle changes in topology (e.g., fluid)

Cons:

- difficult to model complex shapes

Thank you!

(And thank Prof. Ravi Ramamoorthi and Prof. Ren Ng for many of the slides!)