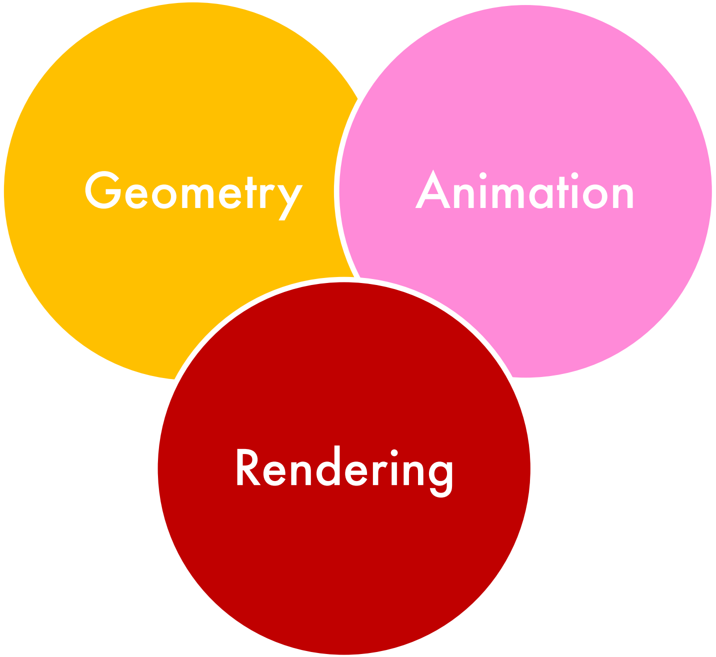
Further Graphics 2022

UNIVERSITY OF CAMBRIDGE

Cengiz Öztireli

Computer Graphics



Goal:

understand the fundamentals of **representing and rendering scenes** for computer aided image generation

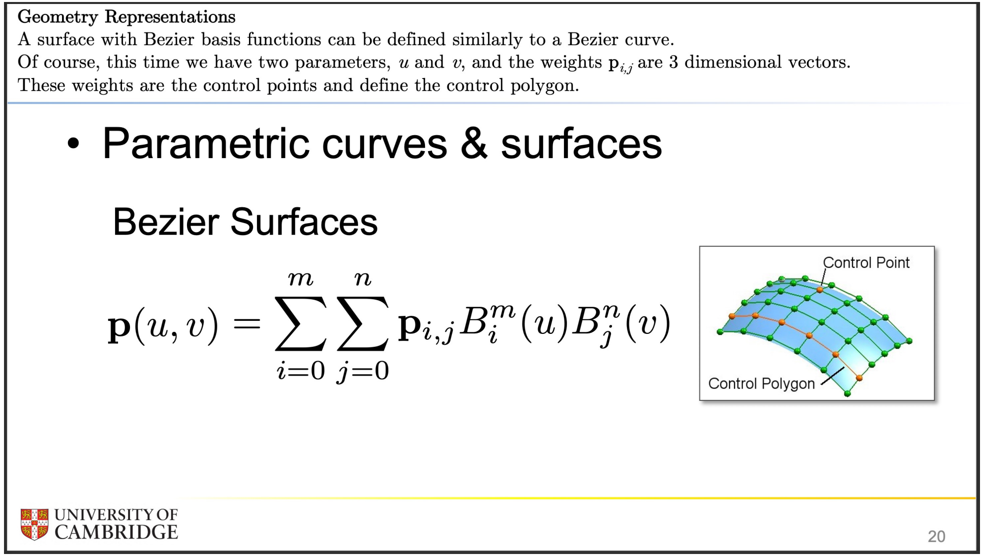
Lectures & Tick

Geometry Representations
Discrete Differential Geometry
Geometry Processing
Animation I
Animation II
The Rendering Equation
Distributed Ray Tracing
Inverse Rendering

Tick:
Approximately 2 weeks
Two coding exercises
Deadline:
Thursday, October 27, 12:00 PM

Lectures & Tick

Lecture notes with detailed notes for each slide shown in the lectures will be available via the **course webpage** and **Moodle**.



Geometry Representations

A surface with Bezier basis functions can be defined similarly to a Bezier curve.
Of course, this time we have two parameters, u and v, and the weights $\mathbf{p}_{i,j}$ are 3 dimensional vectors.
These weights are the control points and define the control polygon.

- Parametric curves & surfaces

Bezier Surfaces

$$\mathbf{p}(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} \mathbf{p}_{i,j} B_i^m(u) B_j^n(v)$$

UNIVERSITY OF CAMBRIDGE

20

Geometry Representations

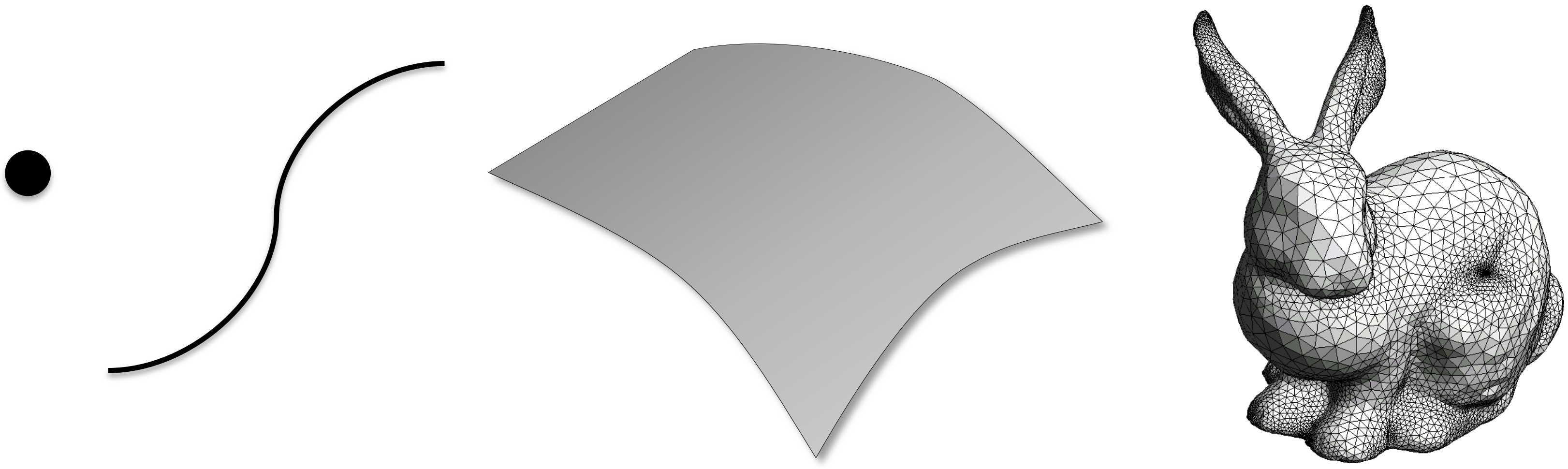
Dr Cengiz Öztireli

UNIVERSITY OF CAMBRIDGE

Geometry in Graphics

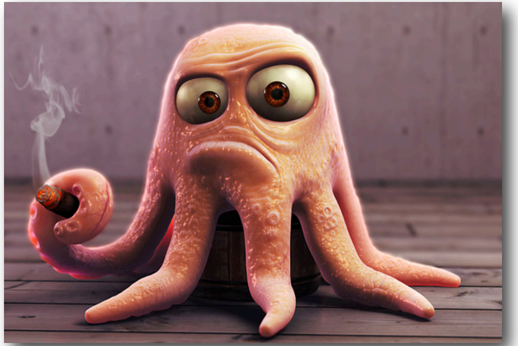
Geometry comes in many forms.

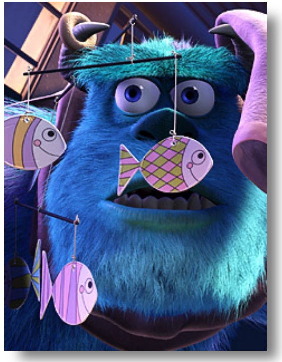
Examples, left to right: points, lines, continuous surfaces, surface meshes.



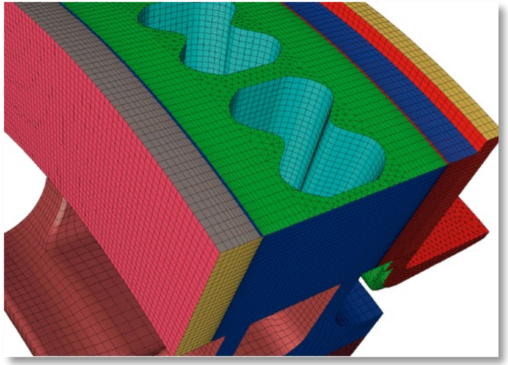
UNIVERSITY OF CAMBRIDGE

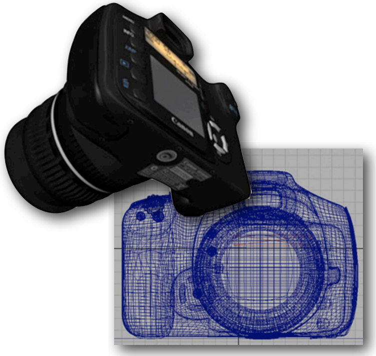
Applications





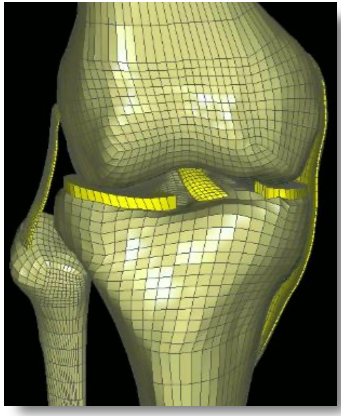
Games/Movies



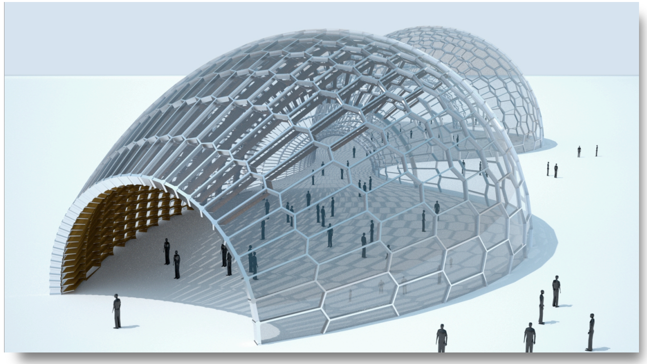
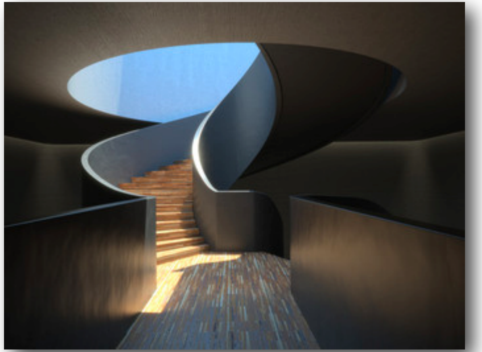


Engineering/Product design

Applications




Medicine/Biology

Architecture

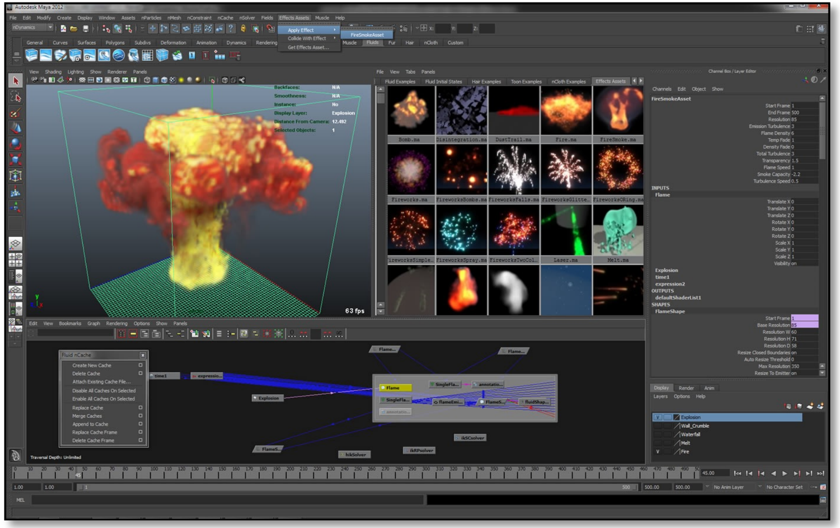
Sources of Geometry

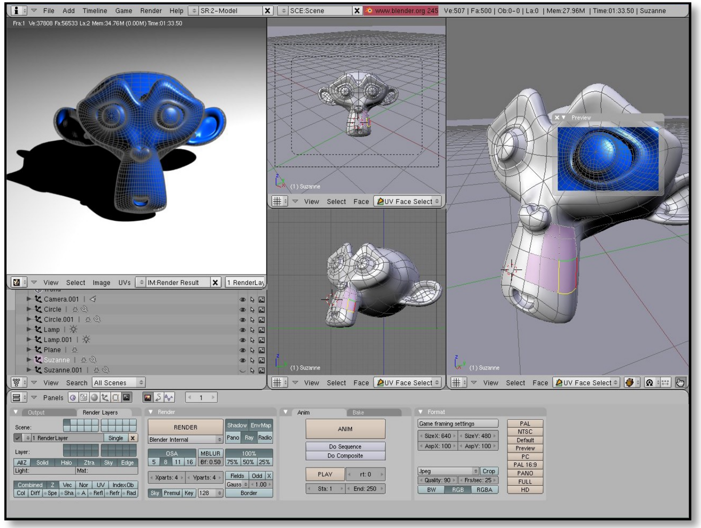
- Acquired real-world objects

3D Scanning

Sources of Geometry

- Digital 3D modeling



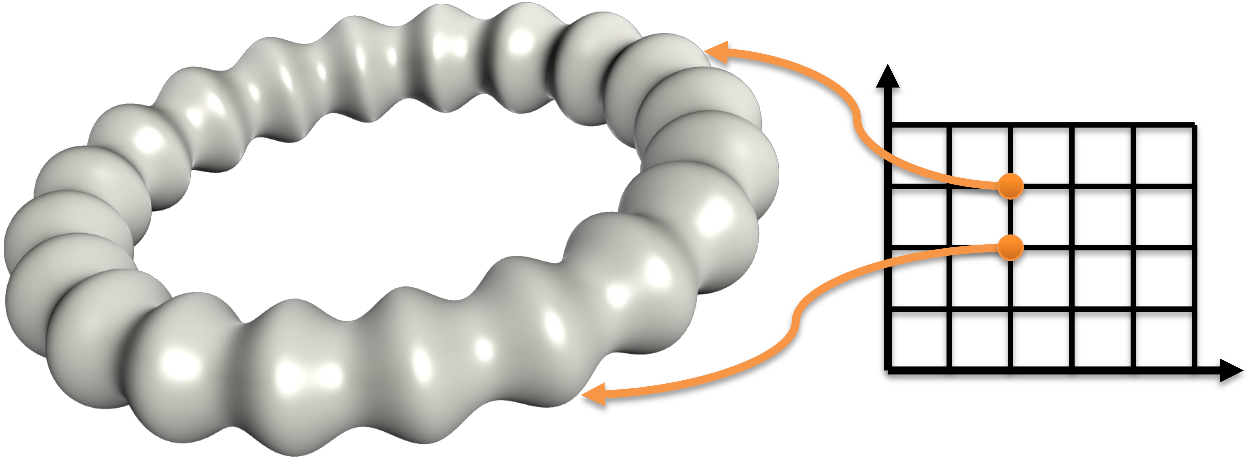


Geometry Representations

- Considerations
  - Storage
  - Acquisition of shapes
  - Creation of shapes
  - Editing shapes
  - Rendering shapes

The surface is represented as a map from (in this case) a 2D plane to a surface in 3D.
We thus have m degrees of freedom, although the object is embedded in an n - dimensional space.

- Parametric curves & surfaces



$$f : X \rightarrow Y, X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n$$

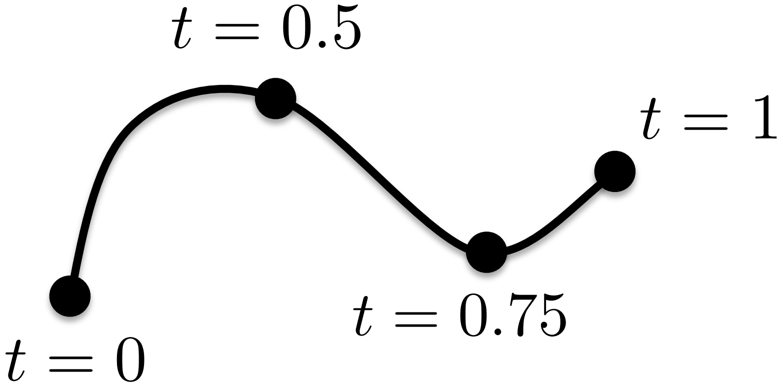
As another simple example, we have $m = 1$ and $n = 2$ for curves on a plane.
As we change t from 0 to 1, we trace a curve in 2D.

- Parametric curves & surfaces

Planar Curves

$$f : X \to Y, X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n \quad m = 1, n = 2$$

$$\mathbf{p}(t) = (x(t), y(t))$$

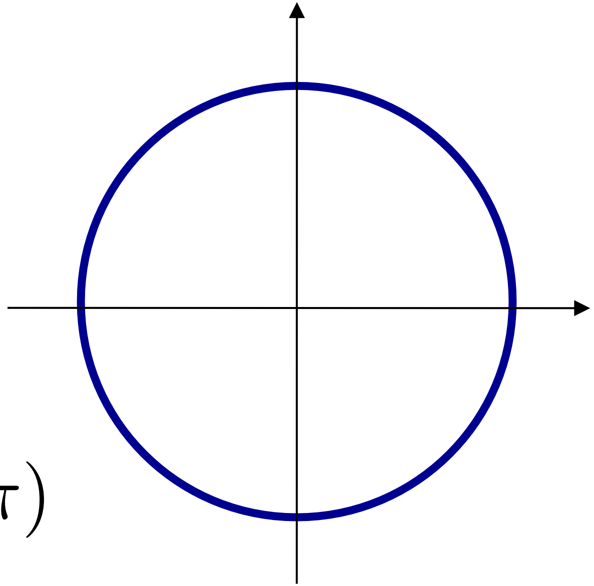


A special case of a planar curve is a circle.
As we change t from 0 to 2 pi, we trace a circle on a plane.

- Parametric curves & surfaces

Circle

$$\mathbf{p} : \mathbb{R} \to \mathbb{R}^2$$
$$t \mapsto \mathbf{p}(t) = (x(t), y(t))$$



$$\mathbf{p}(t) = r\,(\cos(t), \sin(t)) \quad t \in [0, 2\pi)$$

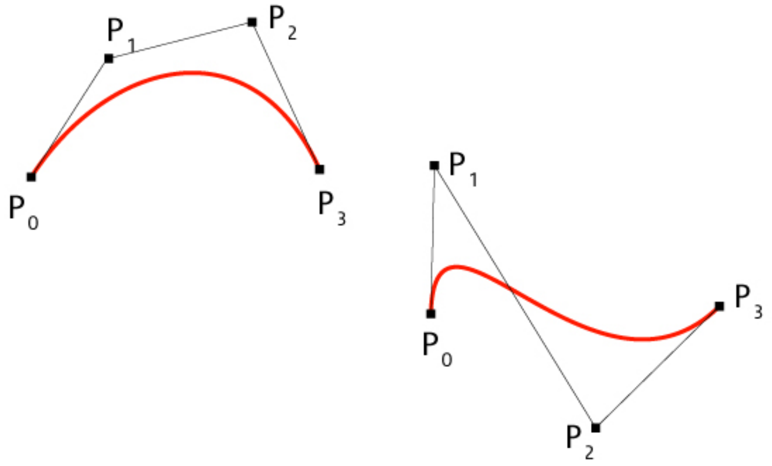
Geometry Representations
A more complex case is a Bezier curve, where we have a weighted combination of basis functions.
The weights $\mathbf{p}_i$, also called control points, are vectors in 2D.
Note: We always denote vectors with boldface lowercase letters.

- Parametric curves & surfaces

Bezier Curves

$$\mathbf{p}(t) = \sum_{i=0}^{n} \mathbf{p}_i B_i^n(t)$$

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$



We can similarly define a curve in 3D by having three functions, one for each coordinate.
For each t, we have a different point in 3D on the curve.

- Parametric curves & surfaces

Space Curves in 3D

$$f : X \to Y, X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n$$

$$m = 1, n = 3$$

$$\mathbf{p}(t) = (x(t), y(t), z(t))$$

Geometry Representations

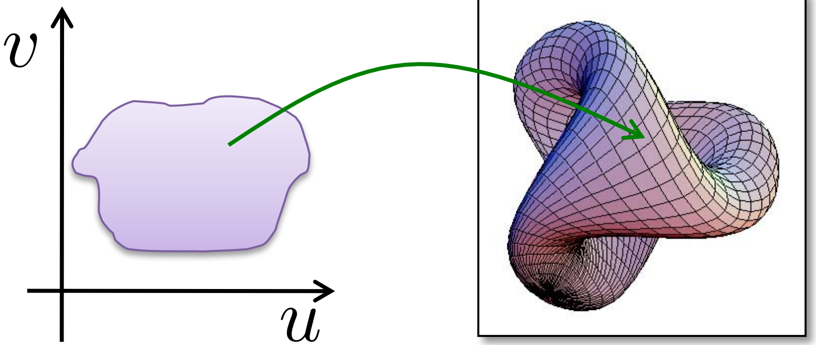
If we set $m = 2$, we get two degrees of freedom, which defines a surface.
Denoting the parameter space with u and v, each point on the surface in 3D is given by three functions x, y, z.

- Parametric curves & surfaces

Surfaces

$$f : X \to Y, X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n \qquad m = 2, n = 3$$

$$\mathbf{p}(u, v) = (x(u, v), y(u, v), z(u, v))$$



A special case of a 2-surface embedded in 3D is a sphere.
Note the ranges of the $(u,\ v)$ pair.

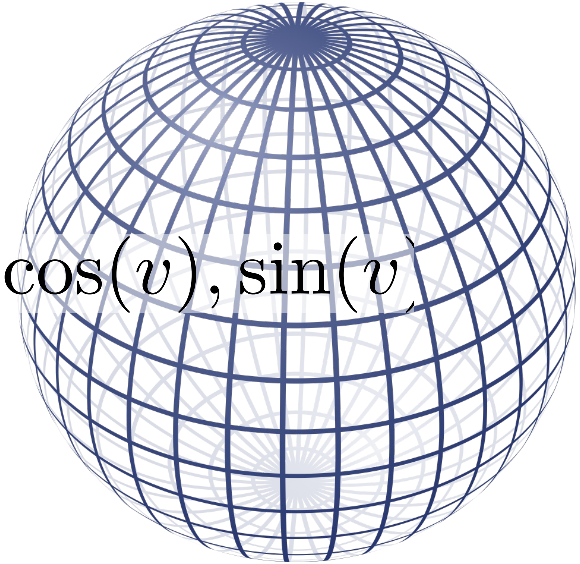
• Parametric curves & surfaces

Sphere

$$\mathbf{p} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$

$$\mathbf{p}(u,v) = r\left(\cos(u)\cos(v), \sin(u)\cos(v), \sin(v)\right)$$

$$(u,v) \in [0, 2\pi) \times [-\pi/2, \pi/2]$$

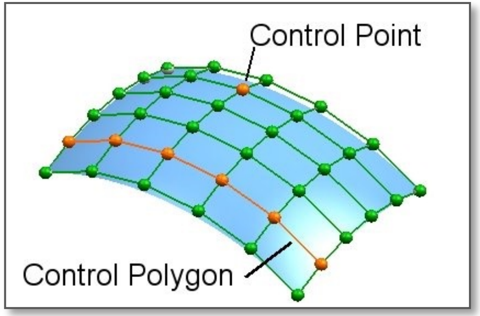


A surface with Bezier basis functions can be defined similarly to a Bezier curve.
Of course, this time we have two parameters, u and v, and the weights $\mathbf{p}_{i,j}$ are 3 dimensional vectors.
These weights are the control points and define the control polygon.

- Parametric curves & surfaces

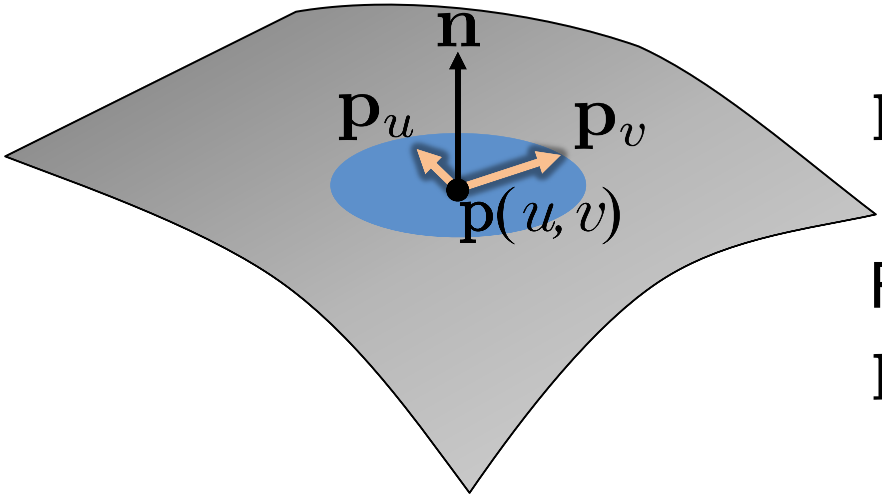
Bezier Surfaces

$$\mathbf{p}(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n}\mathbf{p}_{i,j}B_i^m(u)B_j^n(v)$$



Perhaps the most important local feature of a surface is a tangent plane.
It is defined as the plane spanned by the vectors computed by taking the derivatives w.r.t. u and v.
The surface normal $\mathbf{n}$ is defined as the vector orthogonal to the tangent plane.

- Parametric curves & surfaces



$$\mathbf{p}_u = \frac{\partial \mathbf{p}(u,v)}{\partial u}, \quad \mathbf{p}_v = \frac{\partial \mathbf{p}(u,v)}{\partial v}$$

Regular parametrization:

$$\mathbf{p}_u \times \mathbf{p}_v \neq 0$$

$$\mathbf{n}(u,v) = \frac{\mathbf{p}_u \times \mathbf{p}_v}{\|\mathbf{p}_u \times \mathbf{p}_v\|}$$

Volumetric representations of geometry can be thought of as fields in 3D, where at each point we have a value that determines e.g. the density of a tissue at that point. We can similarly define color fields, etc.

- Parametric curves & surfaces

Volumetric Representations

$$f : X \to Y, X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n \quad m = 3, n = 1$$

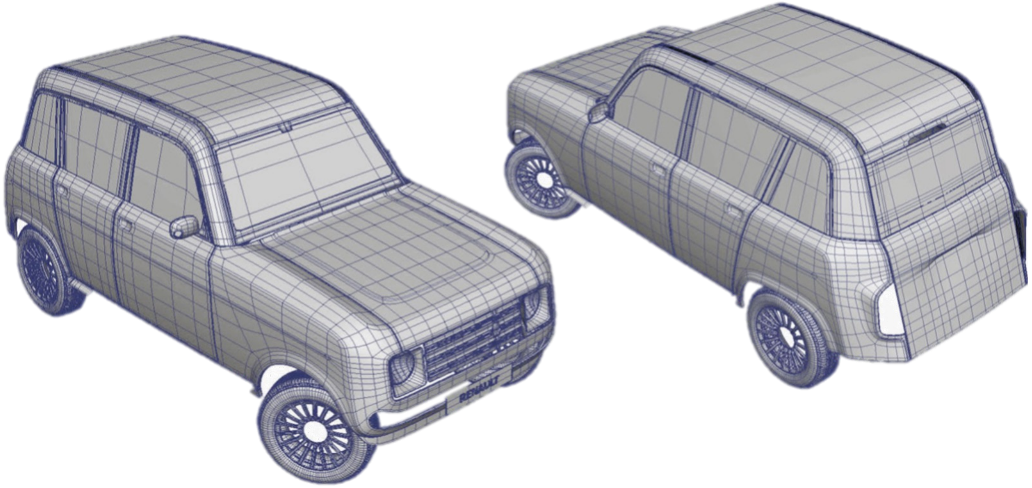
Geometry Representations

- Parametric curves & surfaces
  - \+ Easy to generate points on a curve/surface
  - \+ Easy point-wise differential properties
  - \+ Easy to control by hand
  - − Hard to determine inside/outside
  - − Hard to determine if a point is on a curve/surface
  - − Hard to generate by reverse engineering

UNIVERSITY OF CAMBRIDGE

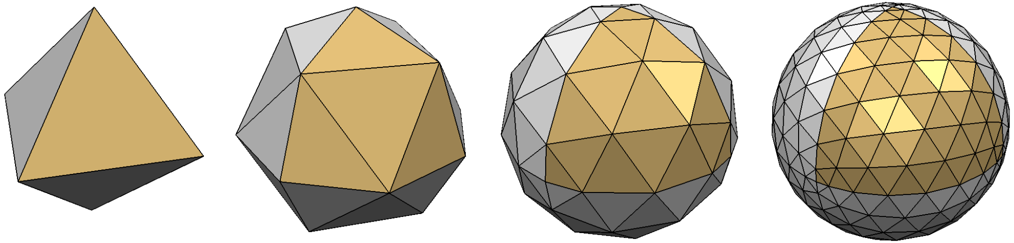
Meshes in computer graphics are points in 2D or 3D connected by edges forming polygons, e.g. triangles.
They represent the boundary of an object as a discrete surface.
They are the standard for modeling geometry e.g. for games.

- Polygonal Meshes



A polygonal mesh is a piecewise linear approximation of an underlying continuous surface.
As we increase the number of polygons, we get better approximations.

- Polygonal Meshes



Piecewise linear approximation

A surface mesh is a graph embedded in 3D. Each vertex in V is associated with a point $\mathbf{p}$ in 3D. The vertices are connected by edges in E. The edges form faces stored as a set F. Hence, each edge stores two vertex indices and each face stores three vertex indices.

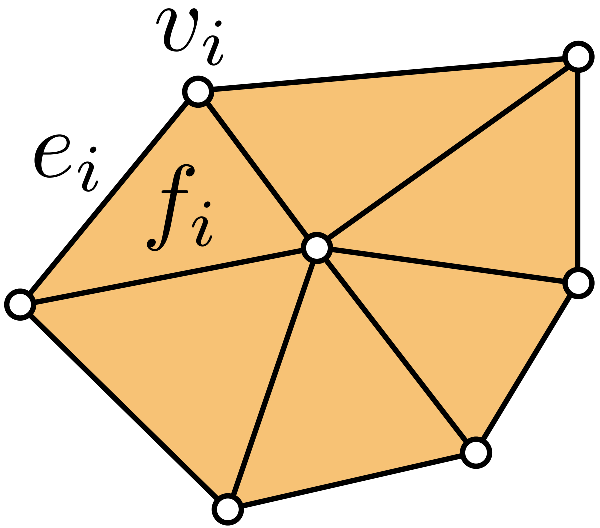
- Triangle Meshes

$$V = \{v_1, \ldots, v_n\}$$

$$E = \{e_1, \ldots, e_k\}, \quad e_i \in V \times V$$

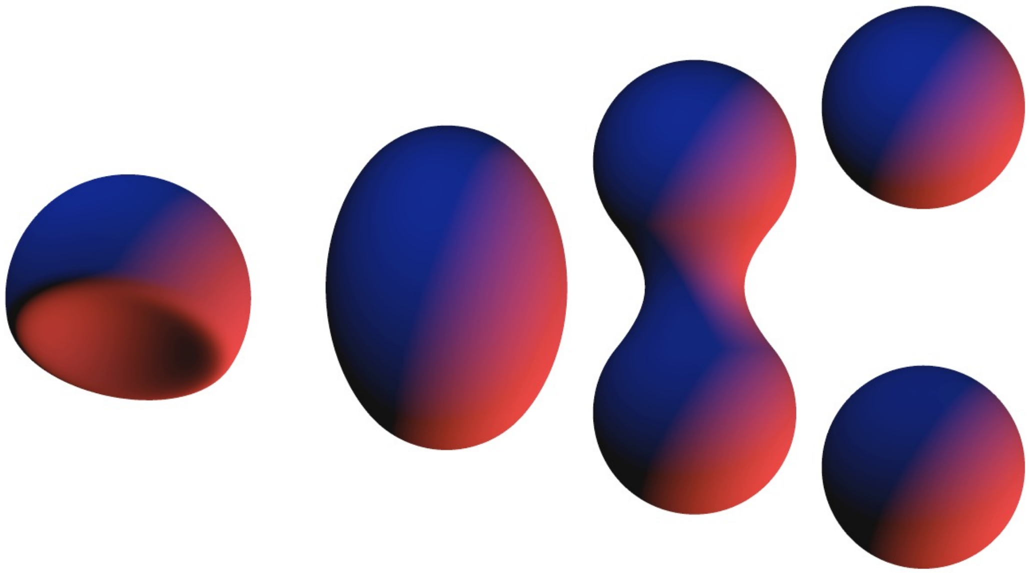
$$F = \{f_1, \ldots, f_m\}, \quad f_i \in V \times V \times V$$

$$P = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}, \quad \mathbf{p}_i \in \mathbb{R}^3$$



Geometry Representations

- Implicit surfaces

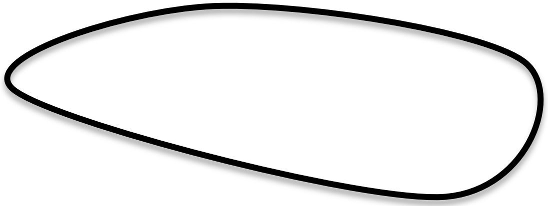


- Implicit curves & surfaces

$$f : \mathbb{R}^m \rightarrow \mathbb{R}$$

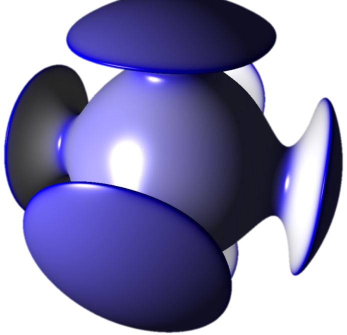
Planar Curves

$$S = \{x \in \mathbb{R}^2 | f(x) = 0\}$$



Surfaces in 3D

$$S = \{x \in \mathbb{R}^3 | f(x) = 0\}$$



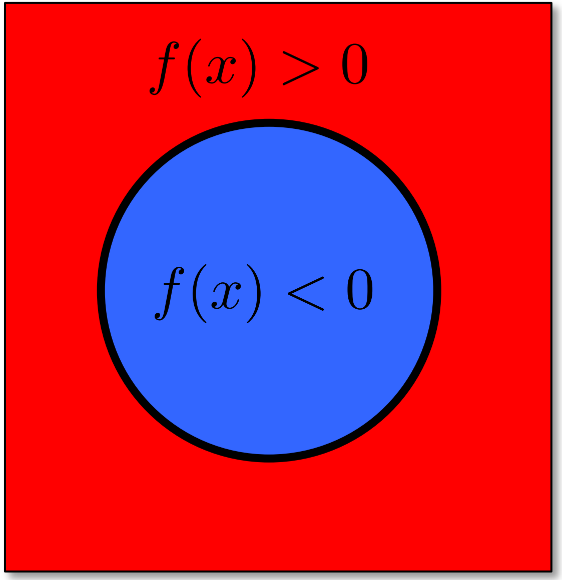
We can further define inside and outside of an object as the regions where the implicit function has different signs. Note that we can define the sign freely as both f and $-f$ represent the same implicit surface.

- Implicit curves & surfaces

$\{x \in \mathbb{R}^m | f(x) > 0\}$ Outside

$\{x \in \mathbb{R}^m | f(x) = 0\}$ Curve/Surface

$\{x \in \mathbb{R}^m | f(x) < 0\}$ Inside

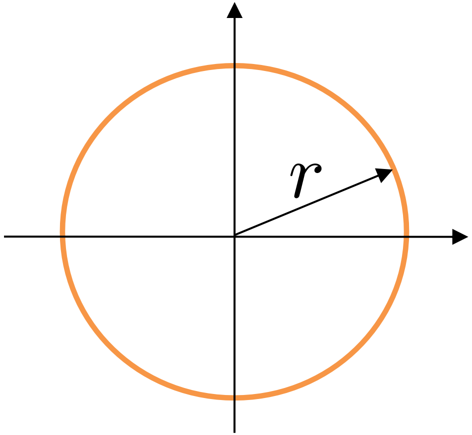


The forms of the implicit functions are quite simple for a circle and a sphere as compared to the parametric ones.

- Implicit curves & surfaces

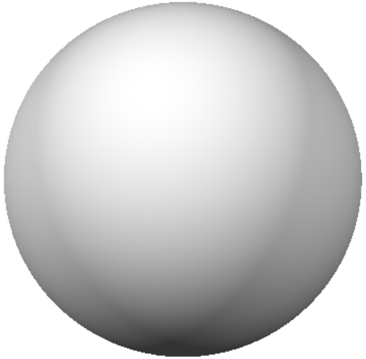
Circle

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



Sphere

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



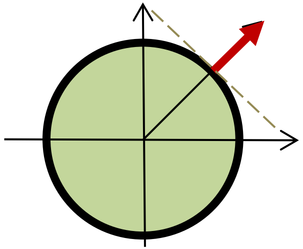
- Implicit curves & surfaces

Surface Normal

$$\nabla f(x,y,z) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right)^T$$

Sphere

$$f(x,y,z) = x^2 + y^2 + z^2 - r^2$$
$$\nabla f(x,y,z) = (2x, 2y, 2z)^T$$

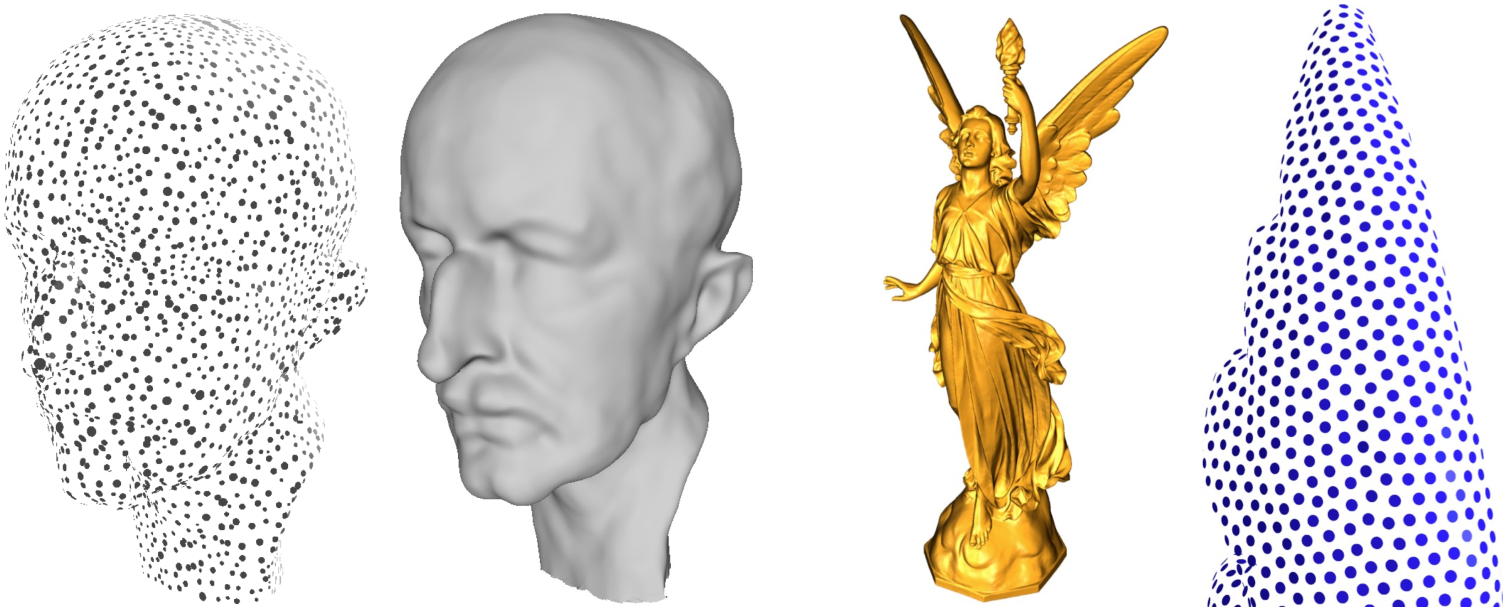


Geometry Representations

- Implicit curves & surfaces
  - + Easy to determine inside/outside
  - + Easy to determine if a point is on a curve/surface
  - + Easy to combine
  - − Hard to generate points on a curve/surface
  - − Limited set of surfaces
  - − Does not lend itself to (real-time) rendering

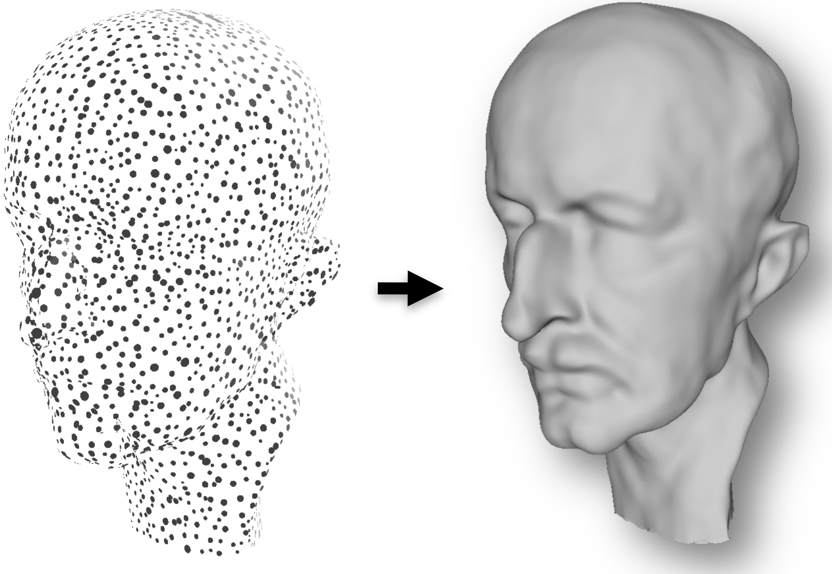
One practical representation that combines some advantages of parametric and implicit representations is point set surfaces.
A point set surface is defined in terms of a point cloud in 3D.

- Point Set Surfaces



A point cloud is a set of points possibly with point-wise attributes, e.g. position, surface normal, etc.
There are number of techniques from the approximation literature that define a point set surface.
These techniques work reliably with data acquired from the real world, so quite important in practice.

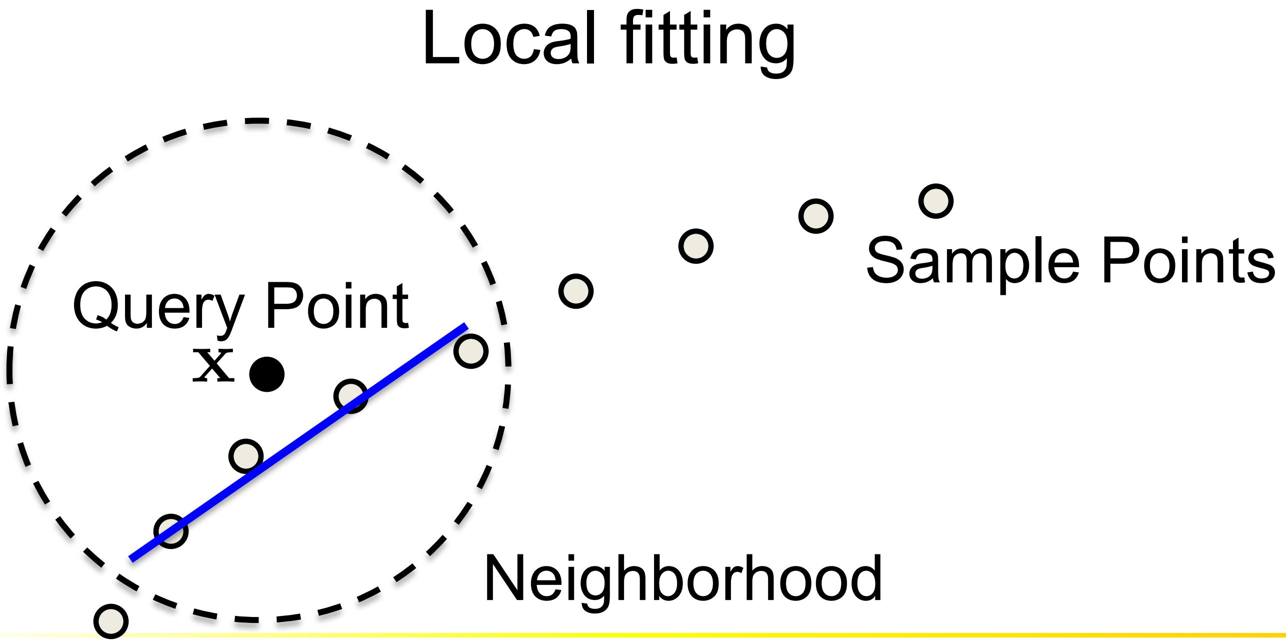
- Point Set Surfaces



Only point-wise attributes
Approximation methods
Smooth surfaces
Works on acquired data

One particular form a point set surface is defined via locally fitting a proxy surface.
We first choose a neighborhood around a query point and a simple proxy surface e.g. a line in this case.
The distance to the surface is then approximated with the distance to this local proxy surface.

- Point Set Surfaces

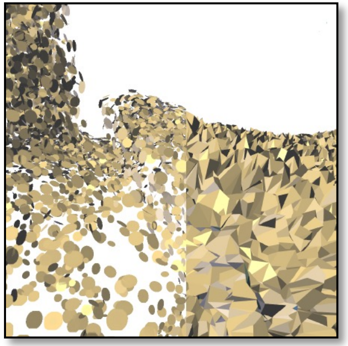


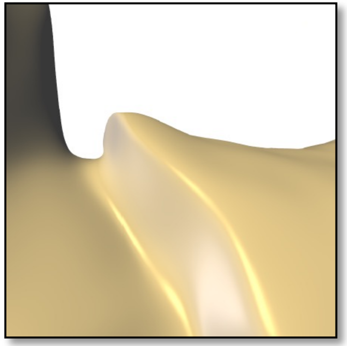
We thus get an implicit function that defines the surface.
It is also easy to generate points on the surface via projection, eliminating a limitation of implicit surfaces.

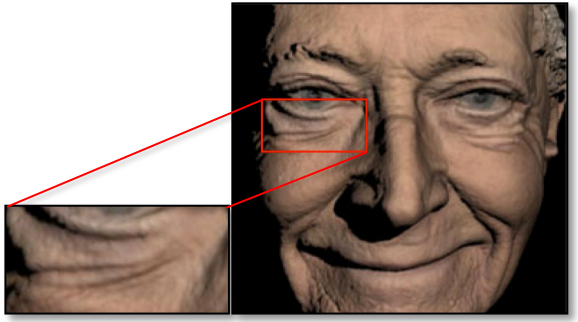
- Point Set Surfaces
  - Implicit representation & fast projection

A point set surface comes with the advantages of robustness to imperfect real-world point data, relatively fast rendering, and conversion to other surface representations such as meshes.

- Point Set Surfaces
  - Robust to noise
  - Direct rendering
  - Conversion to meshes







Geometry Representations

- Point Set Surfaces
  - \+ Easy to determine inside/outside
  - \+ Easy to determine if a point is on the curve/surface
  - \+ Easy to generate points on the curve/surface
  - \+ Suitable for reconstruction from general data
  - \+ Direct real-time rendering
  - − Not efficient to use in some modeling tasks