Geometry Representations

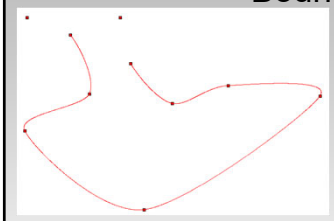
© Wolfgang Heidrich & Alla Sheffer

How to represent shape (2D or 3D)?

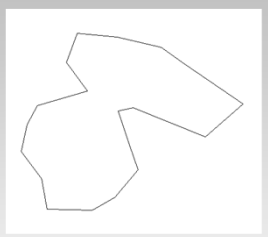
© Alla Sheffer

Geometry – Curves/2D shapes

- Boundary representations



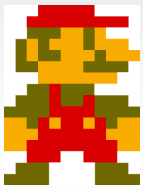
Freeform (aka smooth)



Discrete: polylines

- Alternatives:

Primitive based



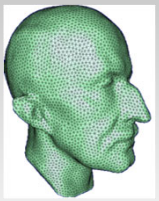
Pixel based

© Alla Sheffer

How to represent 3D shape?

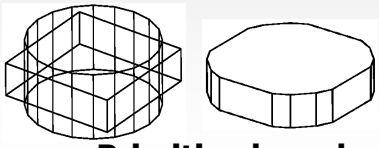
- Most common: Boundary representations

Freeform – smooth surface

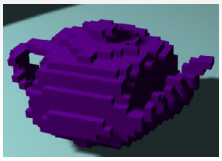


Mesh – polygonal surface

- Alternative: Volumetric representations



Primitive based



Voxel based

© Alla Sheffer

Modeling Geometry

Approaches:

- Fixed set of primitives
  - *Curves: lines, circles, rectangles…*
  - *Surfaces: spheres, cylinders…*
  - *Hard to assemble arbitrary (smooth) geometry*
- Freeform curves/surfaces
  - *Single representation for arbitrarily complex geometry*
  - *Curves and surfaces as functions with built-in smoothness properties*
  - *Bézier curves, splines*
- Discrete: meshes, points, splats

© Alla Sheffer

Curves: Explicit vs. Implicit vs. Parametric

© Wolfgang Heidrich & Alla Sheffer

Curves & Surfaces as Explicit Functions

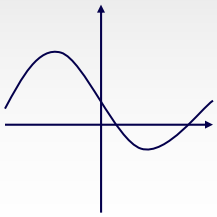
Curves:

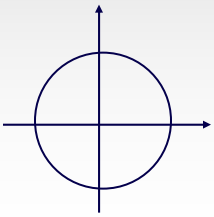
$$y = F(x)$$

Surfaces:

$$z = F(x, y)$$

Examples:





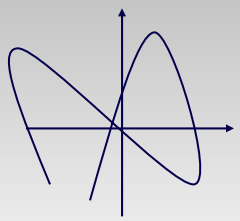
Not a function in Cartesian coord.,

$$y = \pm\sqrt{1 - x^2}$$

© Alla Sheffer

Curves & Surfaces as Explicit Functions

Not representable as a function:

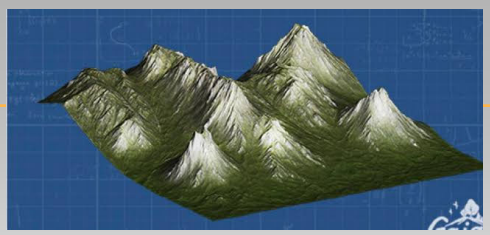


Limitations of explicit functions:

- Cannot model every curve in 2D
- No true 3D curves possible
  - *All curves confined to a plane*

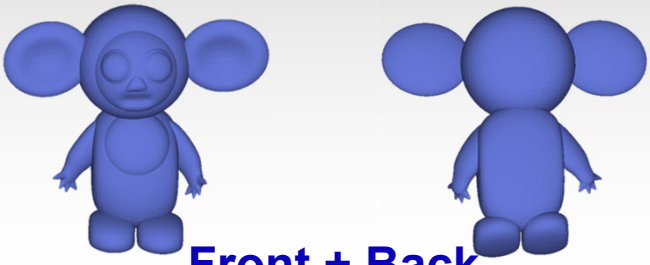
© Alla Sheffer

Explicit Functions

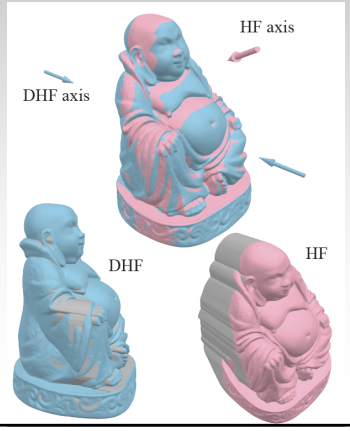


Uses

- Graphs (one Y per X, one Z per (X,Y)
- Terrain modeling (no caves)
- Complex shapes as Boolean combinations (intersections) of explicits



Front + Back



© Alla Sheffer

Curves & Surfaces as Implicit Functions

Curves

$$F(x, y) = 0$$

Surfaces

$$F(x, y, z) = 0$$

Interpretation for curves:

- Iso-lines (contours) in a terrain

Property:

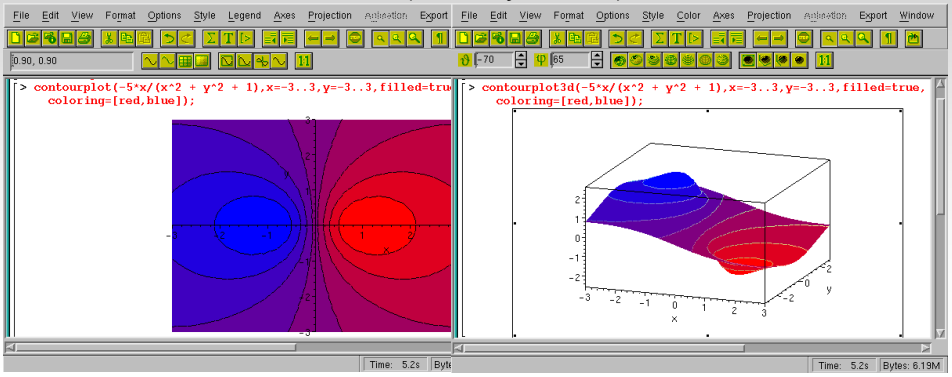
- If F is continuous, implicit curves and surfaces are always closed or extend to infinity

© Alla Sheffer

Curves & Surfaces as Implicit Functions

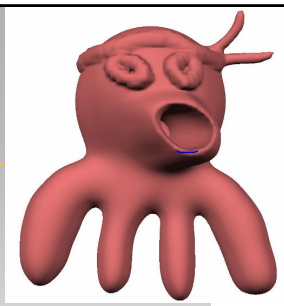
Examples:

- $x^2 + y^2 - 1 = 0$
- $-5x/(x^2 + y^2 + 1) = c$



© Alla Sheffer

Curves & Surfaces as Implicit Functions



Conversion:

- Explicit to implicit: trivial $y - F(x) = 0$
- Implicit to explicit: hard
  - *Solving for y involves root finding!*

Useful for many settings including modeling & machine learning

Limitations:

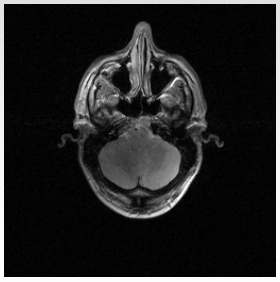
- Curves only in 2D
  - *Non degenerate implicits in 3D are surfaces*
- Often unintuitive
- Relatively costly to render (typically use ray tracing)

© Alla Sheffer

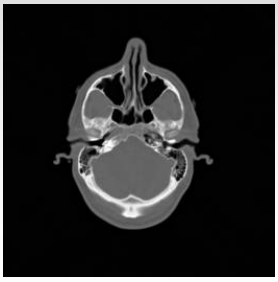
Implicit Functions in Medical Imaging

Data:

- CT & MRI scanners produce volume of *density* values F(x,y,z)
- Individual features (bone surface, brain surface) are iso-surfaces of the volume: F(x,y,z)=c



MRI

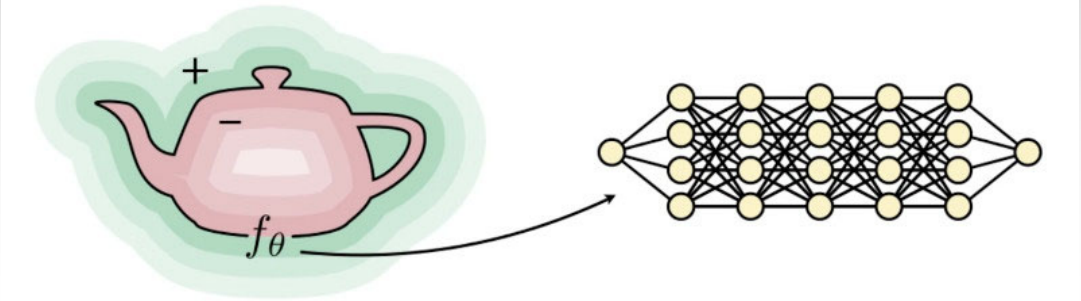


CT

© Alla Sheffer

Implicit Functions in ML

- Allow for regular grid representation
  - *Simplifies encoding*
- Or direct functional encoding
  - *Learning values at locations in space*



© Alla Sheffer

Curves & Surfaces as Parametric Functions

Concept:

- Curve as function of artificial "time" parameter t

2D curve:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} F_x(t) \\ F_y(t) \end{pmatrix} =: F(t); F : \mathcal{R} \mapsto \mathcal{R}^2$$

3D curve:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} F_x(t) \\ F_y(t) \\ F_z(t) \end{pmatrix} =: F(t); F : \mathcal{R} \mapsto \mathcal{R}^3$$

© Alla Sheffer

Curves & Surfaces as Parametric Functions

Curve example:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos t \\ \sin t \\ t \end{pmatrix}$$

Surfaces (in 3D):

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} F_x(s,t) \\ F_y(s,t) \\ F_z(s,t) \end{pmatrix} = F(s,t); F : \mathcal{R}^2 \mapsto \mathcal{R}^3$$

© Alla Sheffer

Curves & Surfaces as Parametric Functions

This works in arbitrary dimensions!

- Curves:

$$\mathbf{x} = F(t); F : \mathcal{R} \mapsto \mathcal{R}^d$$

- Surfaces:

$$\mathbf{x} = F(s,t); F : \mathcal{R}^2 \mapsto \mathcal{R}^d$$

- Hypersurfaces:

$$\mathbf{x} = F(\mathbf{t}); F : \mathcal{R}^n \mapsto \mathcal{R}^d; n < d$$

Notation:

- Bold variables ($\mathbf{t}$, $\mathbf{x}$) denote vectors, while italics denote scalars (t, d).

© Alla Sheffer

Parametric Curves

Advantage:

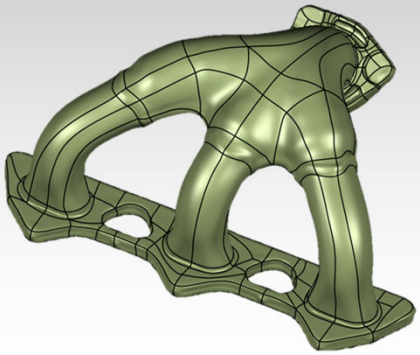
- Arbitrary curves in arbitrary dimensions

Parametric functions:

- High degree of control
  - *"easy" to create from scratch*
- Efficient Rendering
- Storing Signals
  - *Use same parametric representation*

Use case

- Most 3D assets
  - *Engineering, games, movies…*



© Alla Sheffer

Parametric Curves

Challenges:

- Control?
  - *Try to find a formula for a specific curve you have in mind?*
- Encoding?
  - *Arbitrary mathematical functions?*

Solution:

- Restrict yourself to specific class of functions

© Alla Sheffer

Spline Curves

Description = basis functions + coefficients

$$F(t) = \sum_{i=0}^{n} P_i B_i(t) = (x(t), y(t))$$

$$x(t) = \sum_{i=0}^{n} P_i^x B_i(t)$$

$$y(t) = \sum_{i=0}^{n} P_i^y B_i(t)$$

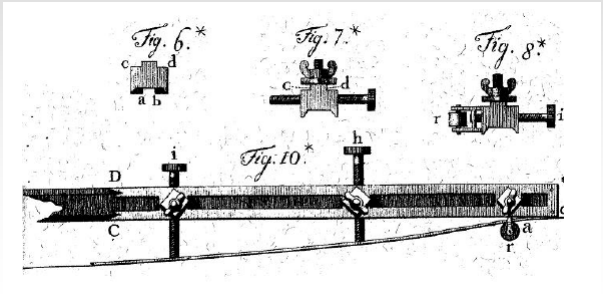
- Same basis functions for all coordinates

© Alla Sheffer

Splines: parametric curves over geometric basis

Geometric meaning of coefficients

- Approximate/interpolate set of positions, derivatives, etc..



© Alla Sheffer

Example: Curves in PowerPoint

© Alla Sheffer

Parametric Spline Curves

Commonly used classes:

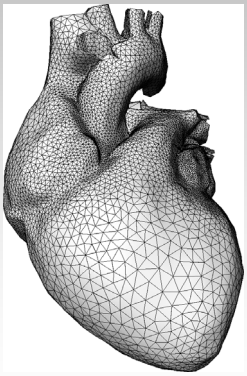
- Polynomials
  - Bézier, Hermite
- Piecewise polynomials
  - *B-splines*
- Rational and piecewise-rational curves
  - *Rational Bézier curves, rational B-splines (NURBS)*
- Neural: not strictly splines – no geometric meaning, “indirect” coefficients

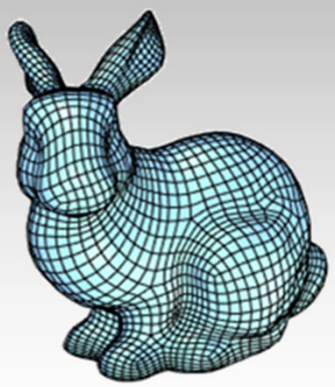
© Alla Sheffer

Discrete representations

© Wolfgang Heidrich & Alla Sheffer

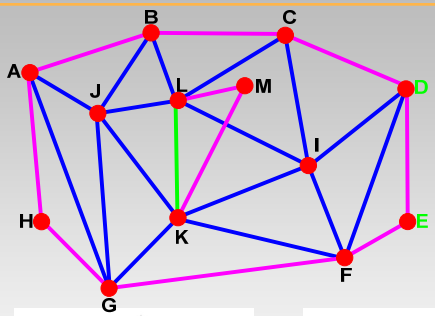
Meshes





© Alla Sheffer

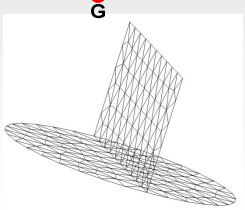
Meshes



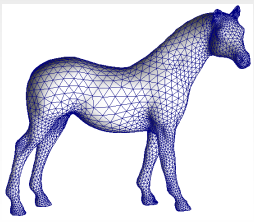
Mesh: straight-line graph embedded in R^3

Boundary edge: adjacent to exactly *one* face
Regular edge: adjacent to exactly *two* faces
Singular edge: adjacent to more than two faces

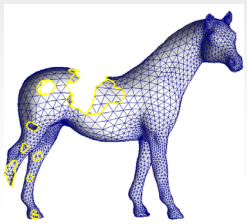
Closed mesh: mesh with no boundary edges
Manifold mesh: mesh with no singular edges



Non-Manifold



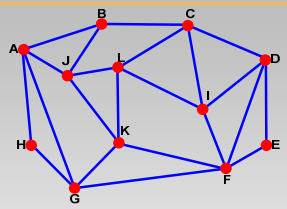
Closed Manifold



Open Manifold

© Alla Sheffer

Topology



v =12
f = 14
e = 25
c = 1
g = 0
b = 1

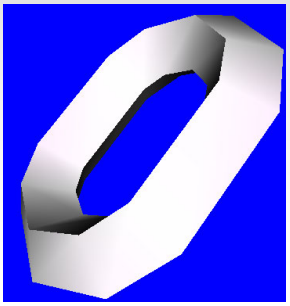
Genus of graph: *half* of *maximal* number of closed paths that do *not* disconnect the graph (number of "holes")

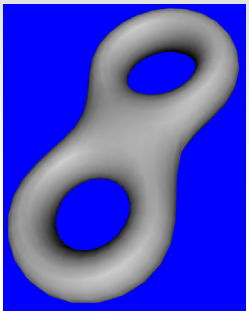
Genus(sphere)= 0
Genus(torus) = 1

Euler-Poincare Formula

v+f-e = 2(c-g)-b

v = # vertices c = # conn. comp
f = # faces g = genus
e = # edges b = # boundaries

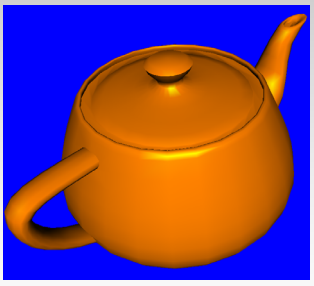


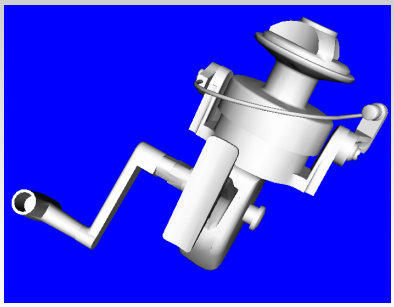


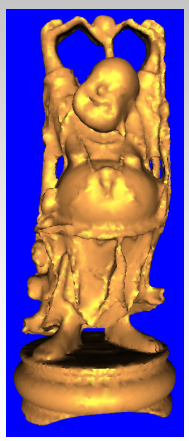
© Alla Sheffer

Topology Quiz

What can you say about the genus of these meshes ?







28

© Alla Sheffer

Meshes

Most common representation for graphics applications

Pro

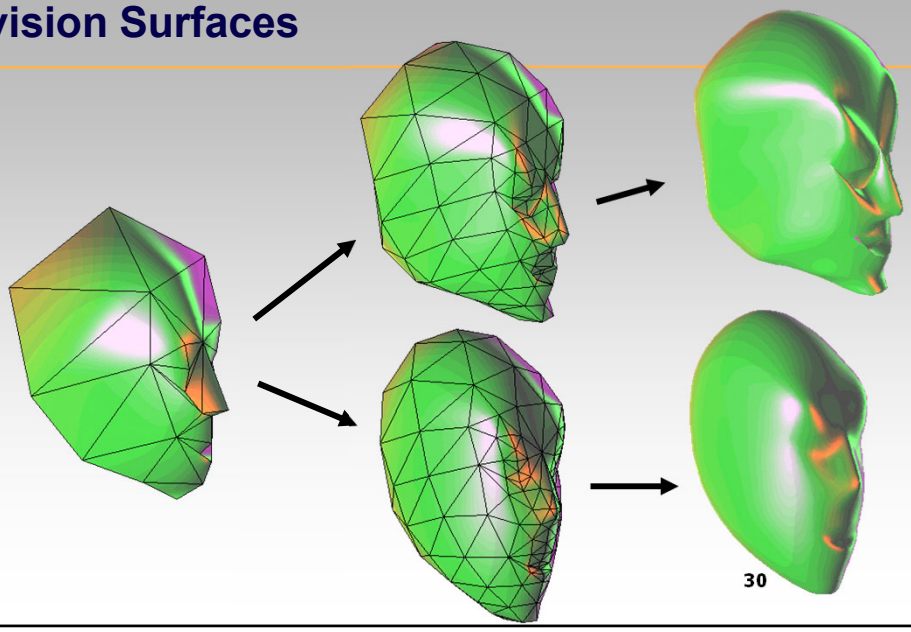
- Easy rendering
- Easy manipulation
- Explicit connectivity
- Preserves properties under processing operations
- Supports sharp features

Cons

- Hard to learn on (irregular connectivity)
- Hard to change genus/topology
- Not smooth

© Alla Sheffer

Subdivision Surfaces



© Alla Sheffer

Subdivision surfaces

Common for high-end graphics

Pro

- Easy rendering
- Explicit connectivity
- Preserves properties under processing operations
- Supports sharp features

Cons

- (Less) hard to learn on (semi-regular connectivity)
- Hard to change genus/topology
- (Harder) manipulation/creation

© Alla Sheffer

Point Clouds

Shape (outer surface) as collection of points (sometimes with normals)

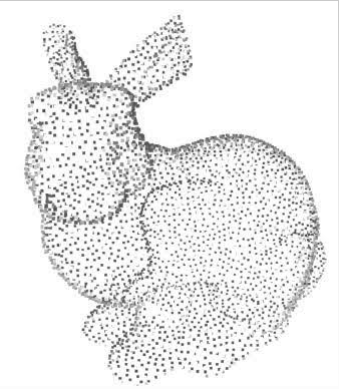
Raw/natural output of many capture/scanning methods/systems

Pro

- Easy to learn on
- Native

Cons

- No connectivity
  - Hard to manipulate
  - Rendering cloud not same as rendering shape
  - Hard to preserve density

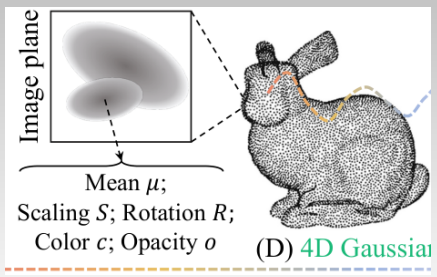


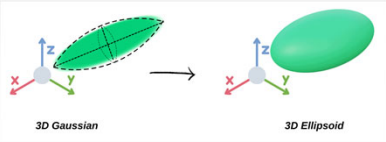
© Alla Sheffer

Splats/Gaussian Splats

Shape (outer surface) as collection of ellipsoids

- Can have varying density (Gaussian)
- Can have outward normal + other properties





Pro

- Easy to learn on
- Has more power than point clouds
- “Hole free”: well suited for rendering

Cons

- No connectivity
  - Hard to manipulate
  - Hard to preserve density

© Alla Sheffer