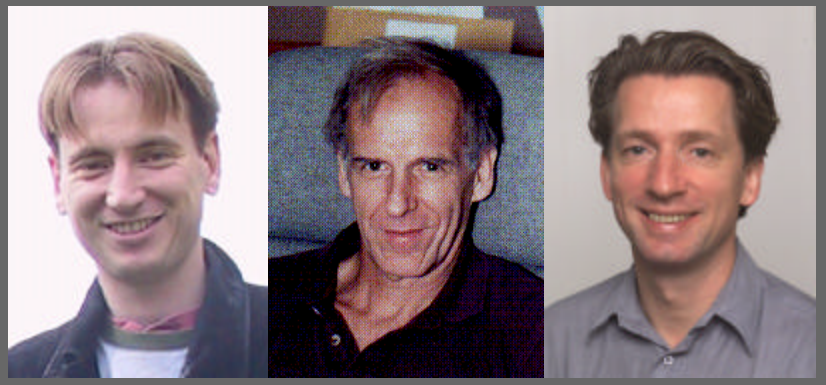
Compression of Geometry Datasets

Mathieu Desbrun (USC)
Ron DeVore (SC)
Peter Schröder (Caltech)

Project

The team

- principals
- students
  - grad, ugrad
- collaborators
  - industry
  - national labs

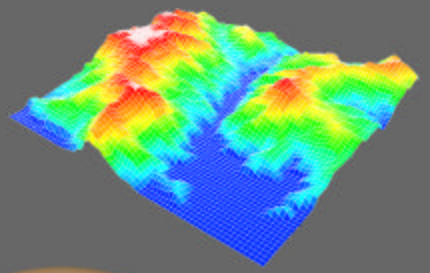


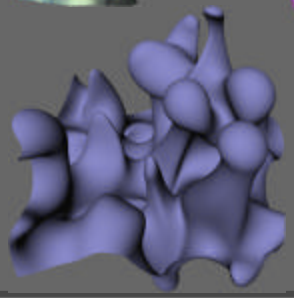
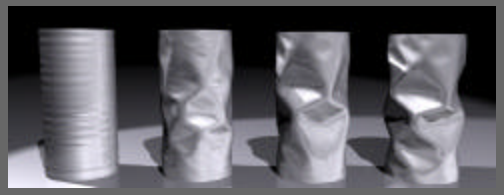
Desbrun DeVore Schröder

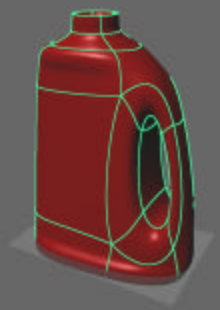
Geometry Compression

Motivation

- surface compression
  - most of it in meshes





What's the Problem?

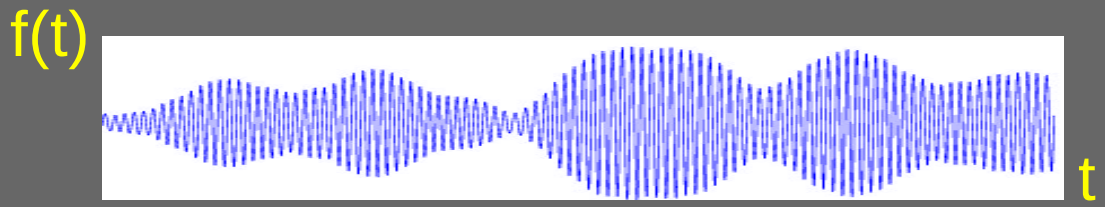
Standard tools insufficient

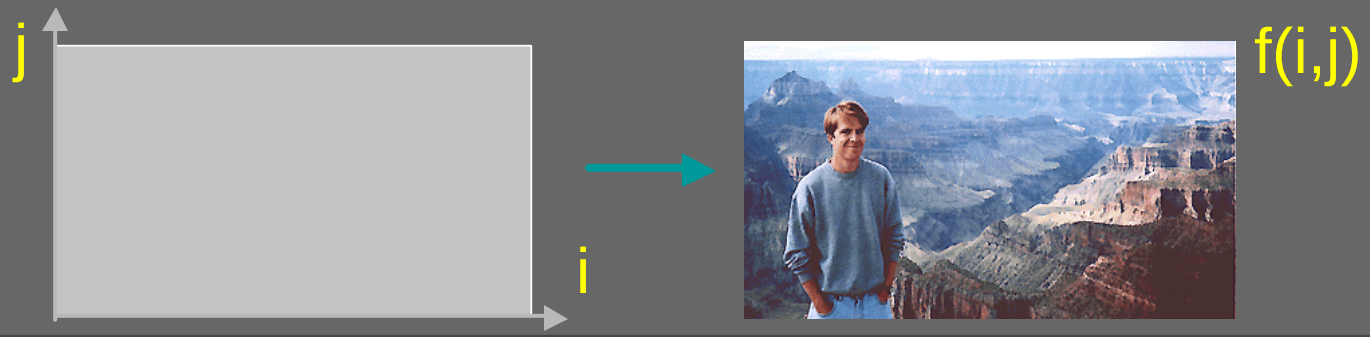
- geometry is not just an image
  - even in functional setting may want to account for 3D nature
- what are the right error measures?
- important geometric constraints
  - ridges, visibility, structure in data,...

Sound, Images, Video

Signals as functions

- basis for sampling, quantization, transform,...





Geometry

Geometry as function?

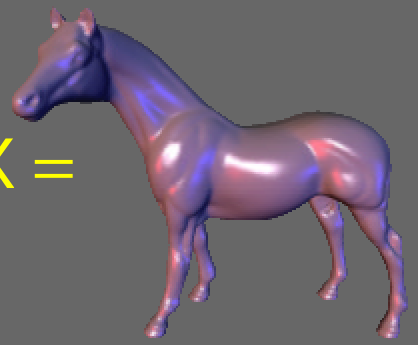
- what is the independent variable?

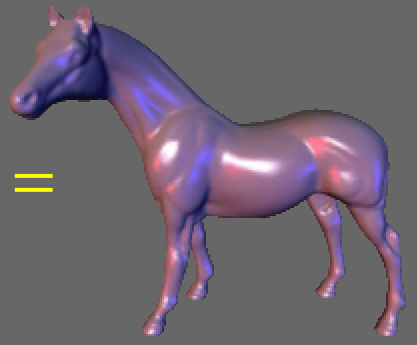
$= f(x) \quad x \in X?$

- why do we care?

Functional Description

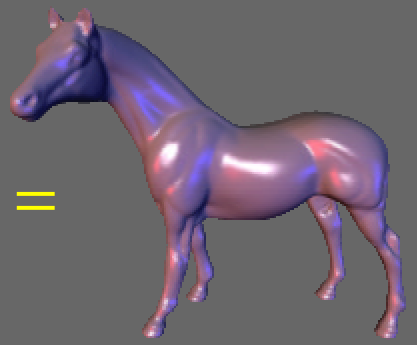
Trivial function

$X =$ 

$f(X) = X =$ 

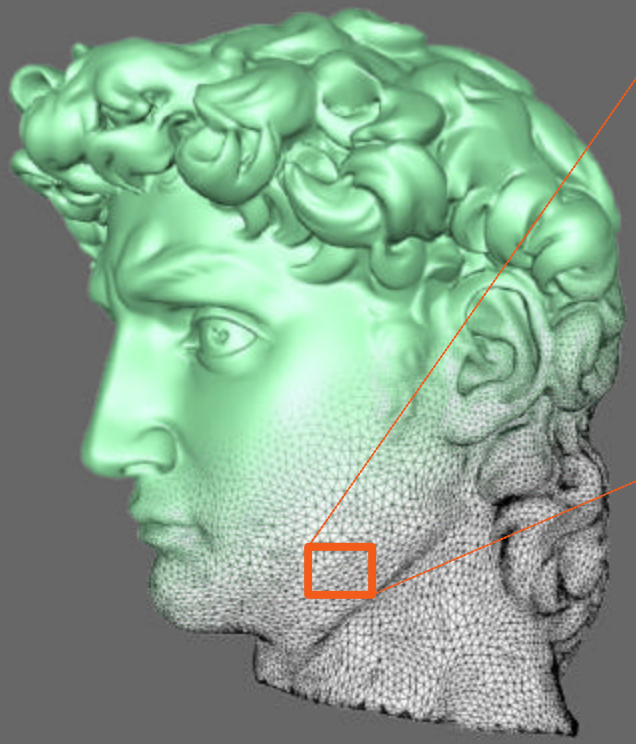
Trivial domain

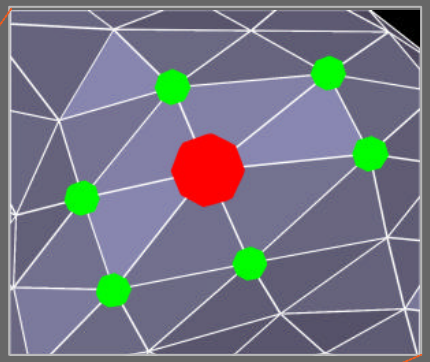
$X =$

$f(X) =$ 

What's the Challenge?

Example format

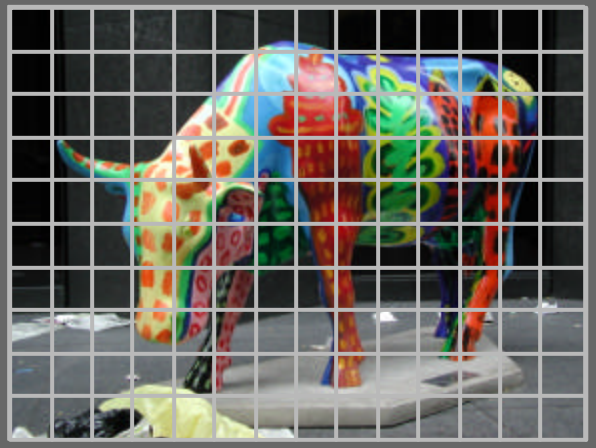


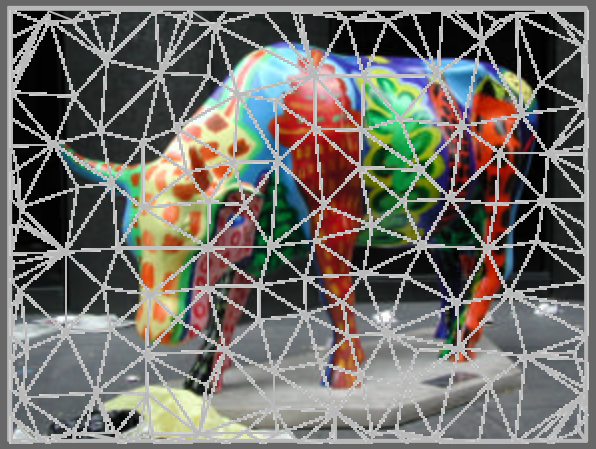


Indexed face set
e.g., VRML, MPEG-4

Vertices	Faces
v1 (x1;y1;z1)	f1 (v1;v3;v2)
v2 (x2;y2;z2)	f2 (v4;v3;v1)
v3 (x3;y3;z3)	f3 (v4;v1;v5)
v4 (x4;y4;z4)	f4 (v1;v6;v5)
v5 (x5;y5;z5)	f5 (v6;v1;v7)
v6 (x6;y6;z6)	f6 (v2;v7;v1)

Image

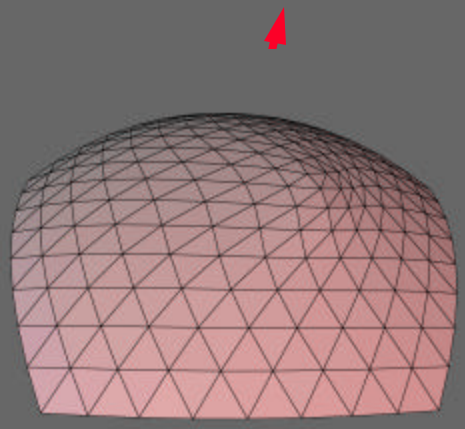




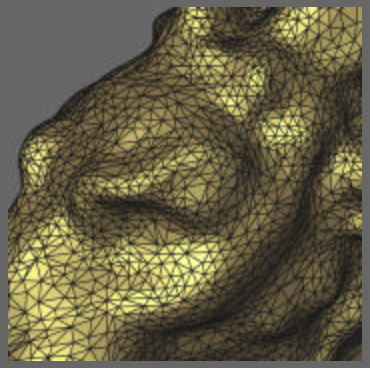
Key Observation

What is in a mesh?

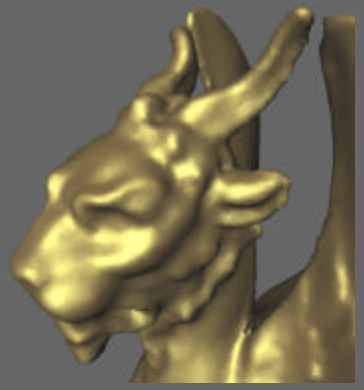
connectivity + vertices



sample locations

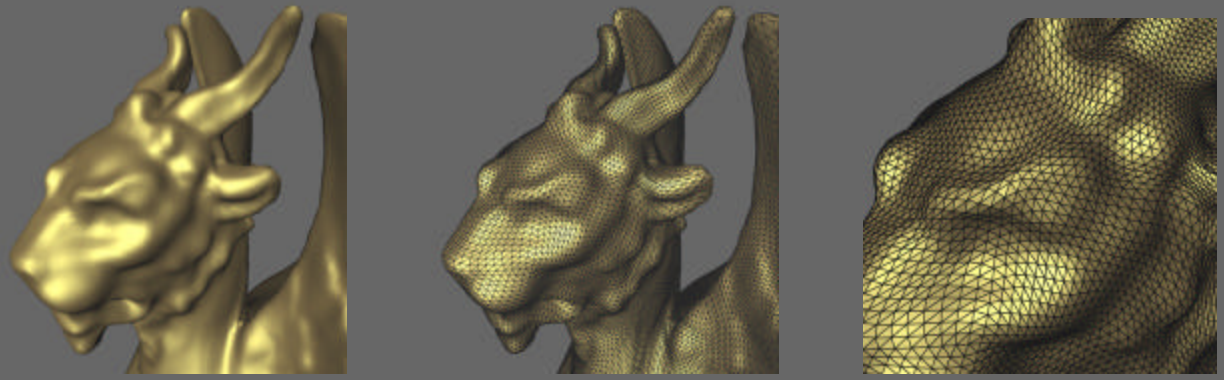


geometry

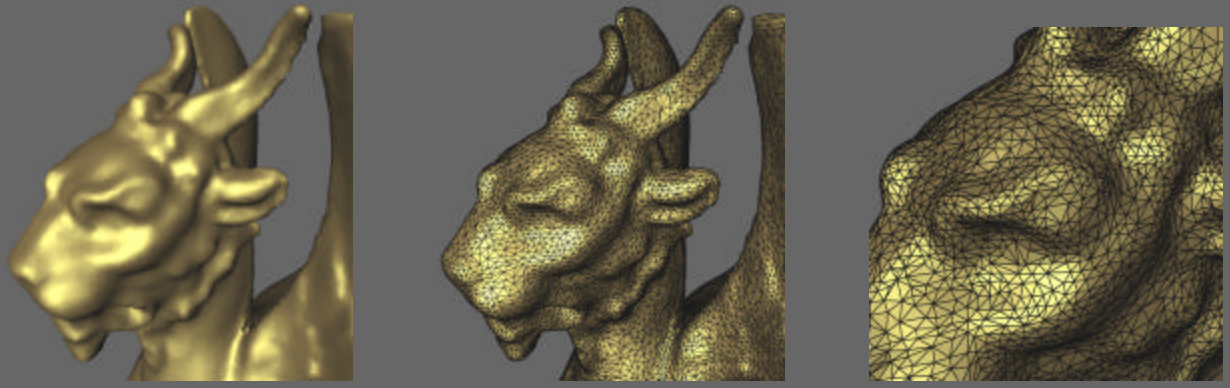


3D Surface

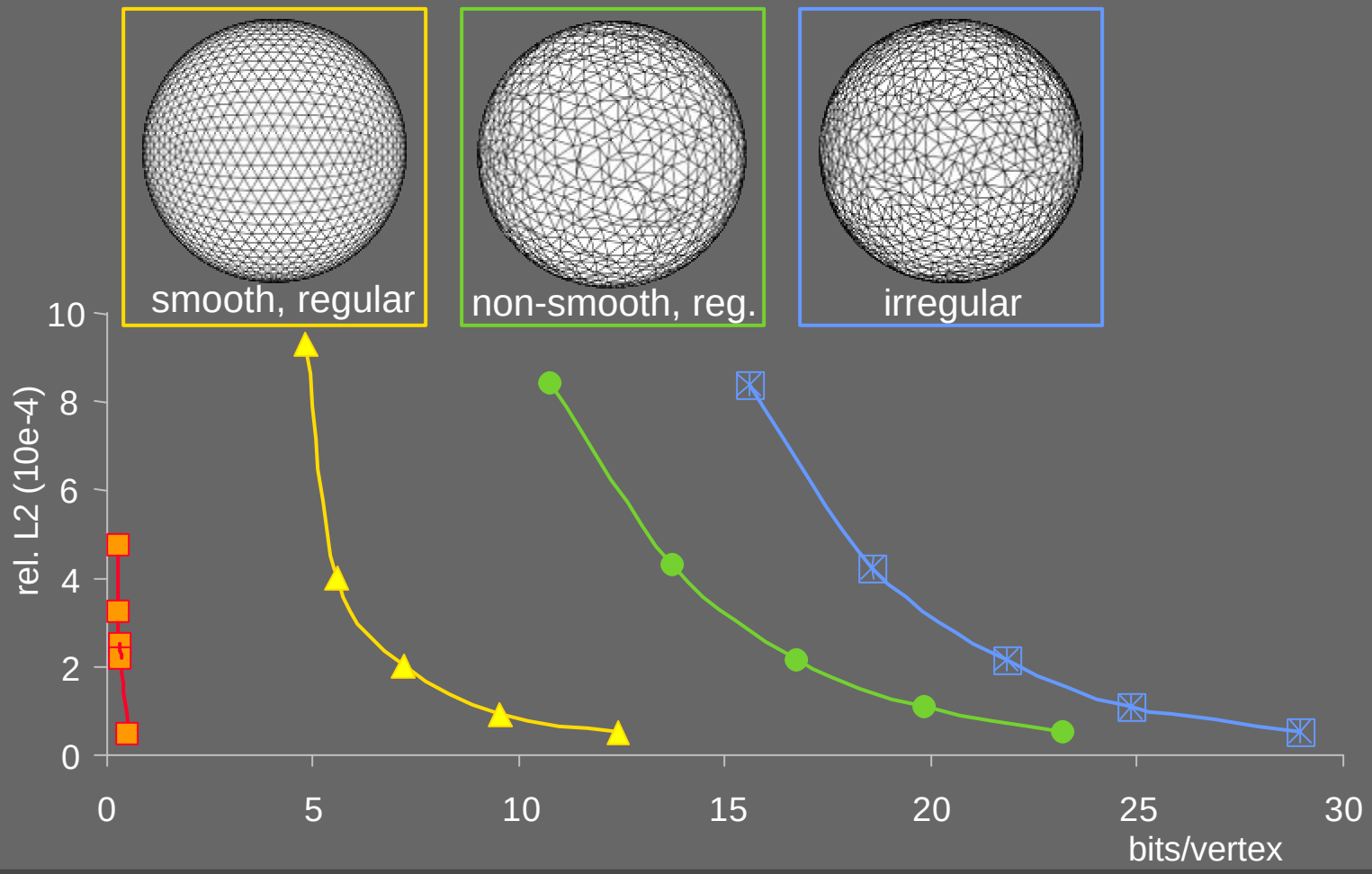
smooth,
regular



irregular



An Example



Compression of Geometry Datasets

Some Basic Questions

Sampling geometry

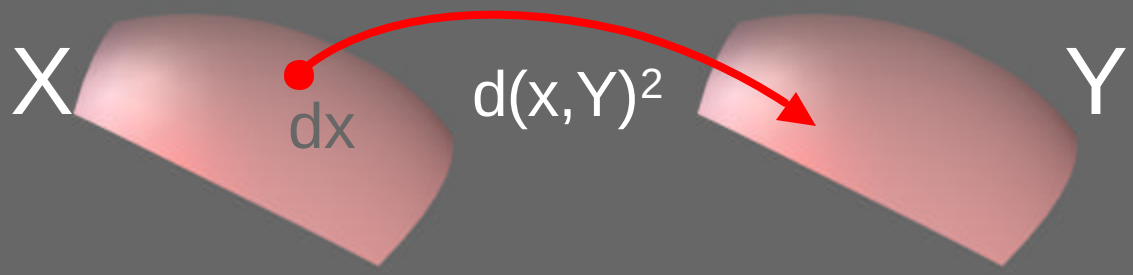
- how? what's a good sampling?
- what's a good parameterization?
- is there a sampling theorem?
  - curvatures?
- entropy of a surface?!
  - NOT of a particular parameterization

Care About Geometry!

Best geometry for given rate

- change parameterization
- geometric error: L_2 average

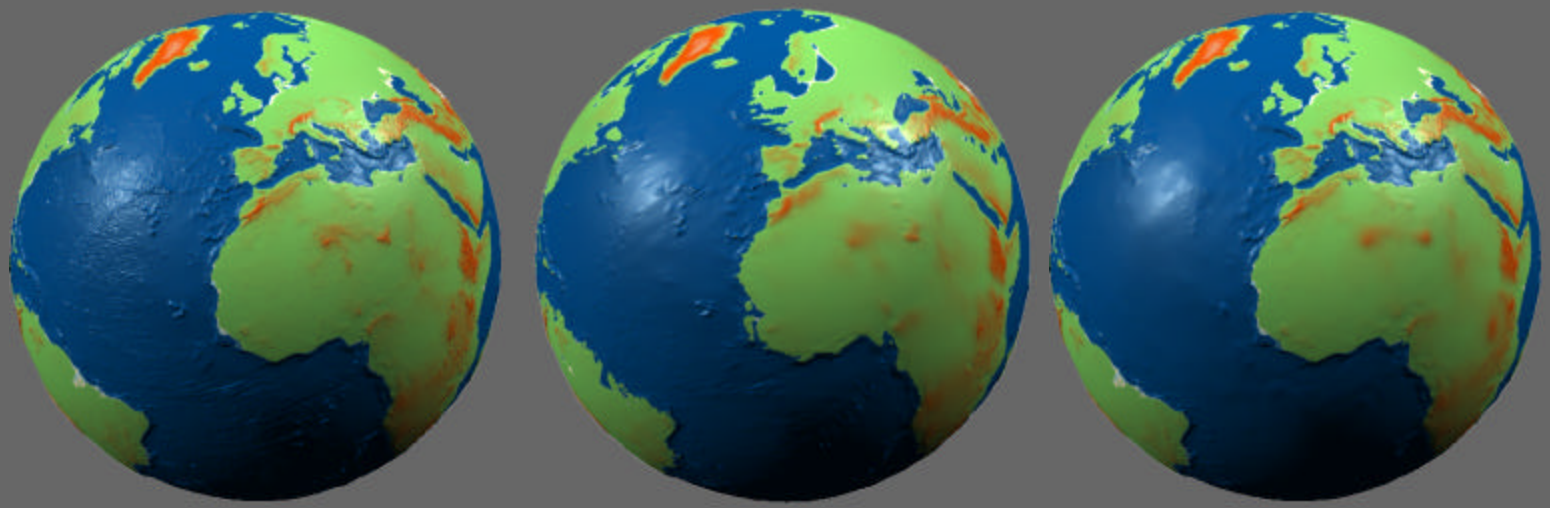
$$d_{NS}^2(X, Y) = \frac{1}{|X|} \int d(x, Y)^2 \, dx$$



Enough?

Distance between sets?

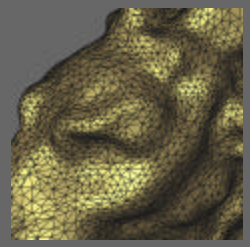
- need more
  - e.g., local slope

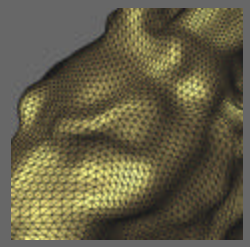


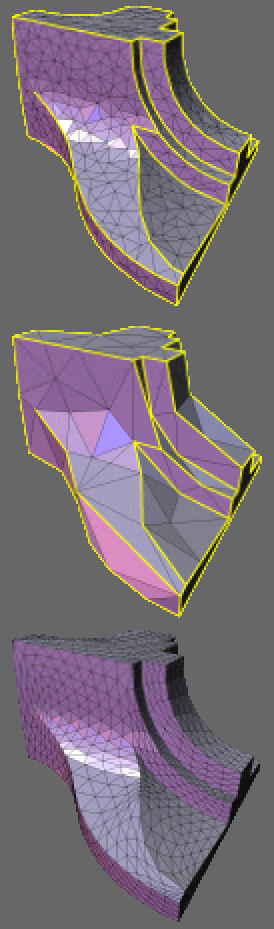
Remeshing

Parameterize & resample

- find base domain
- 1-4 split & resample



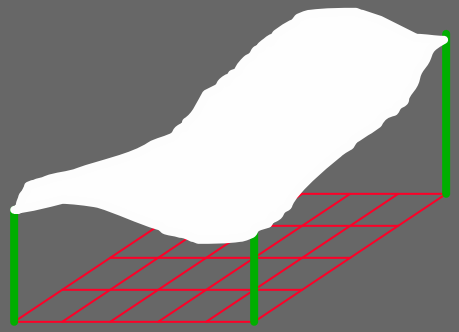




Surfaces

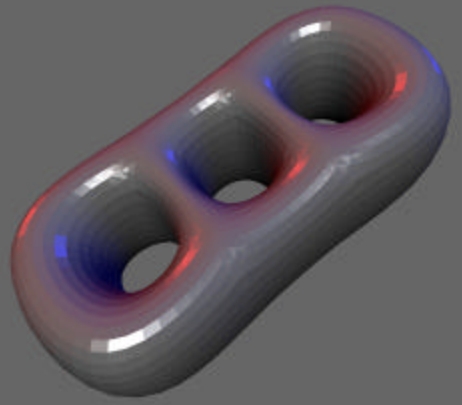
Height fields, terrains

- f(u,v)
- one float / vertex



3d surfaces

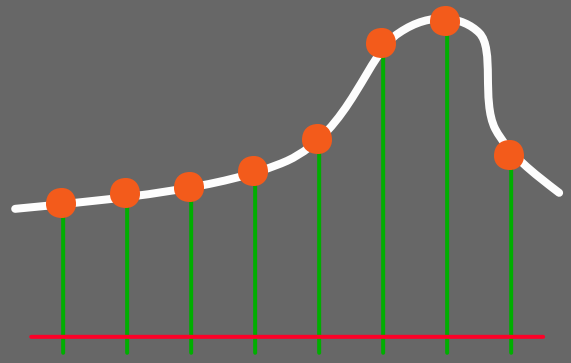
- x(u,v),y(u,v),z(u,v)
- three floats / vertex



Curves

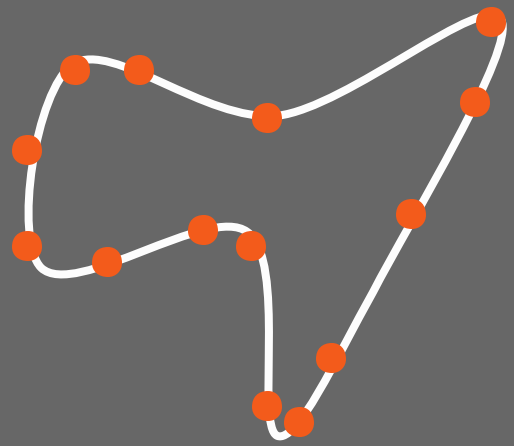
Function

- f(u)
- one float / vertex



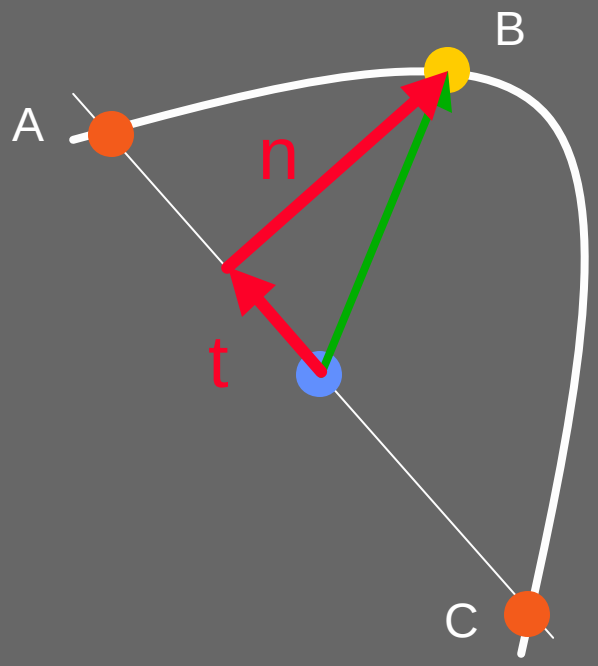
Curve

- x(u), y(u)
- two floats / vertex

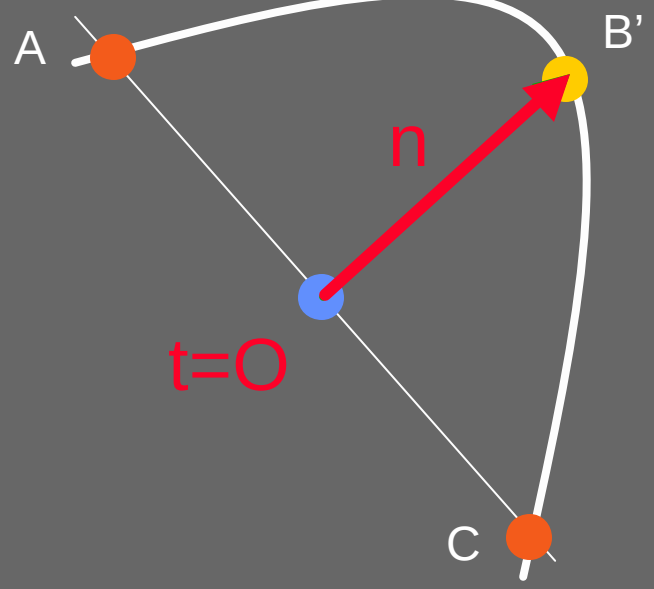


Local Detail (x,y)

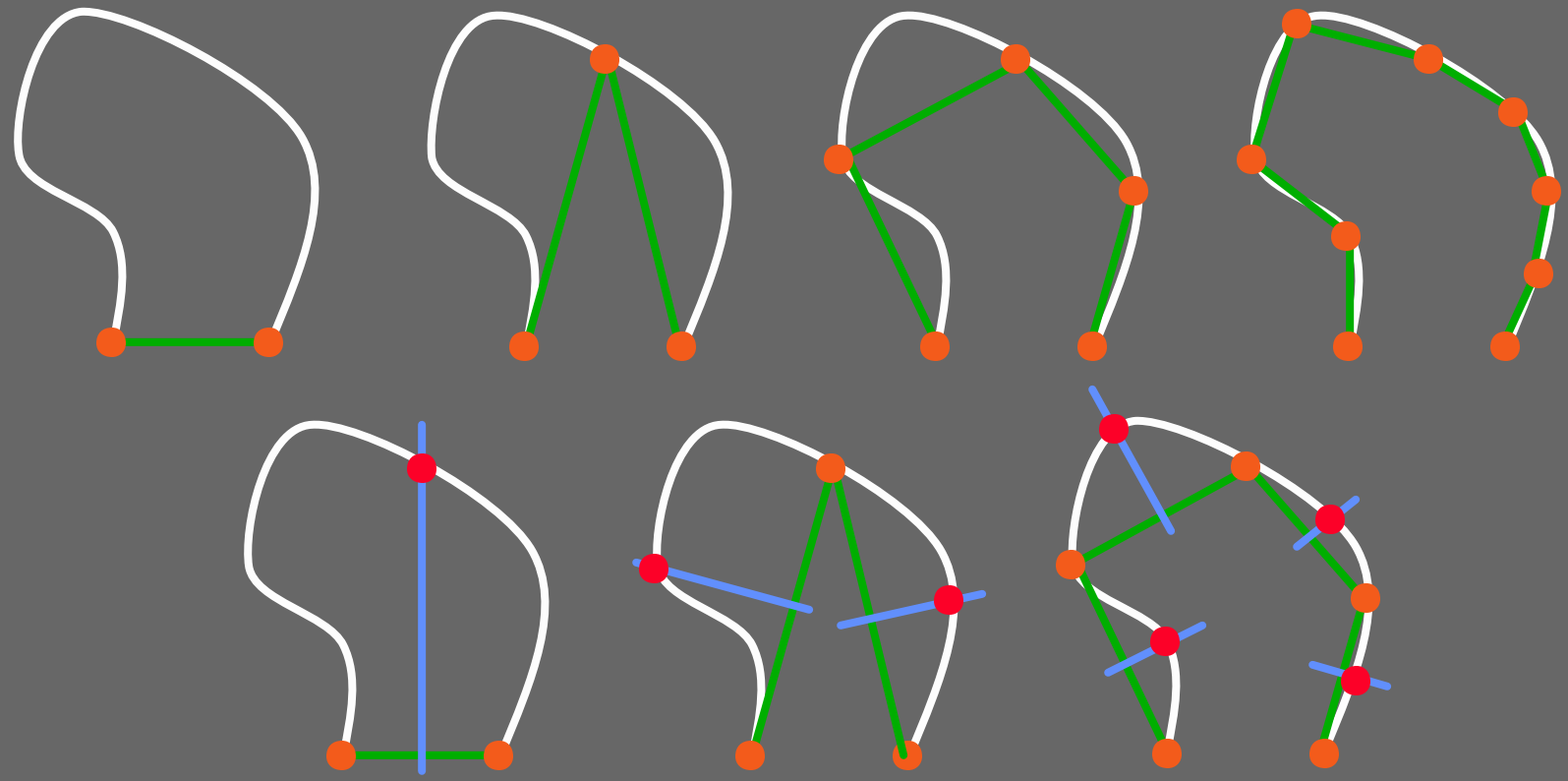
(t,n) stored



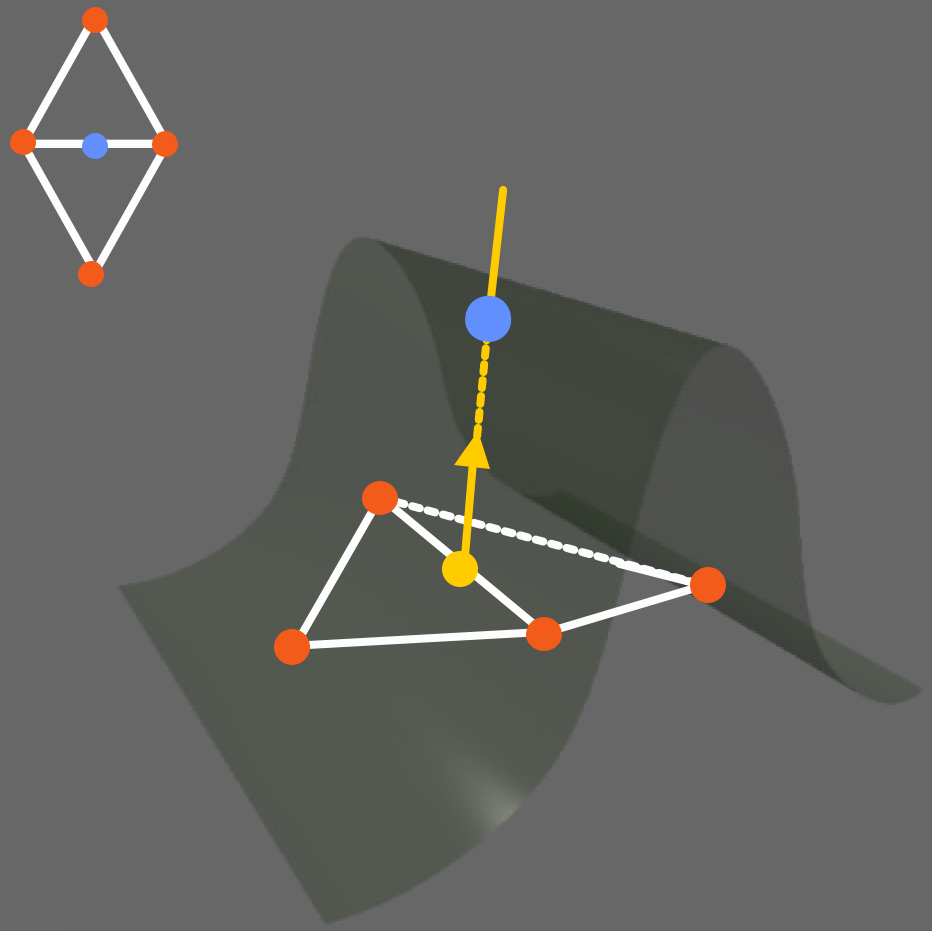
One float!



Simple Algorithm



Piercing



Storage

- one float/vertex

Naïve algorithm:

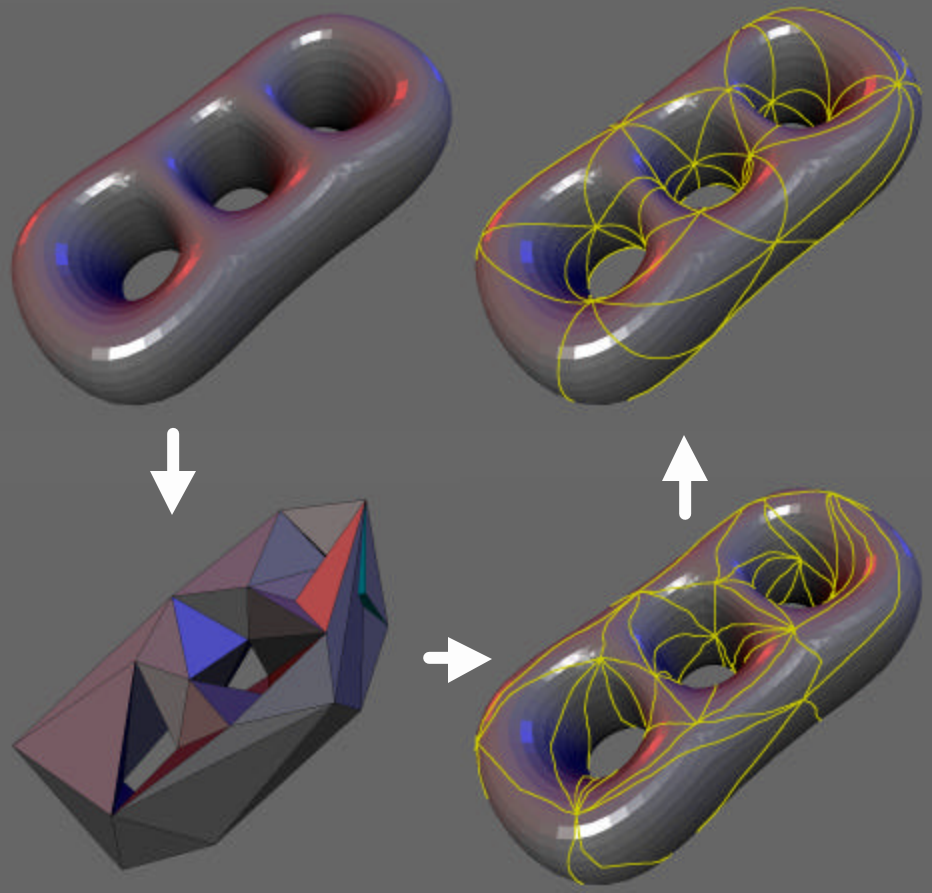
- start with coarse mesh
- pierce recursively

Patch Layout

Base mesh
- simplification

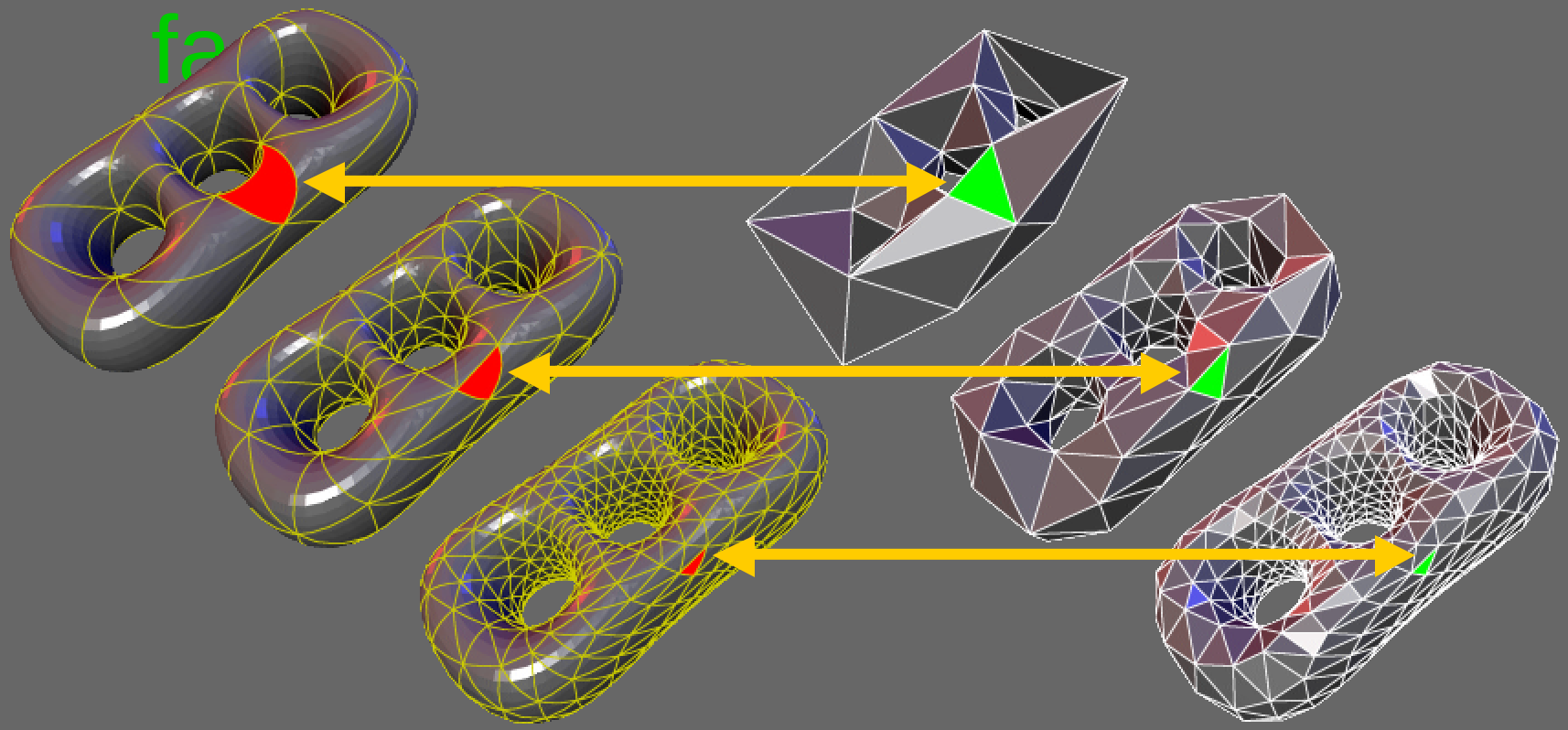
Boundaries
- propagated

Global vertices
- relocated

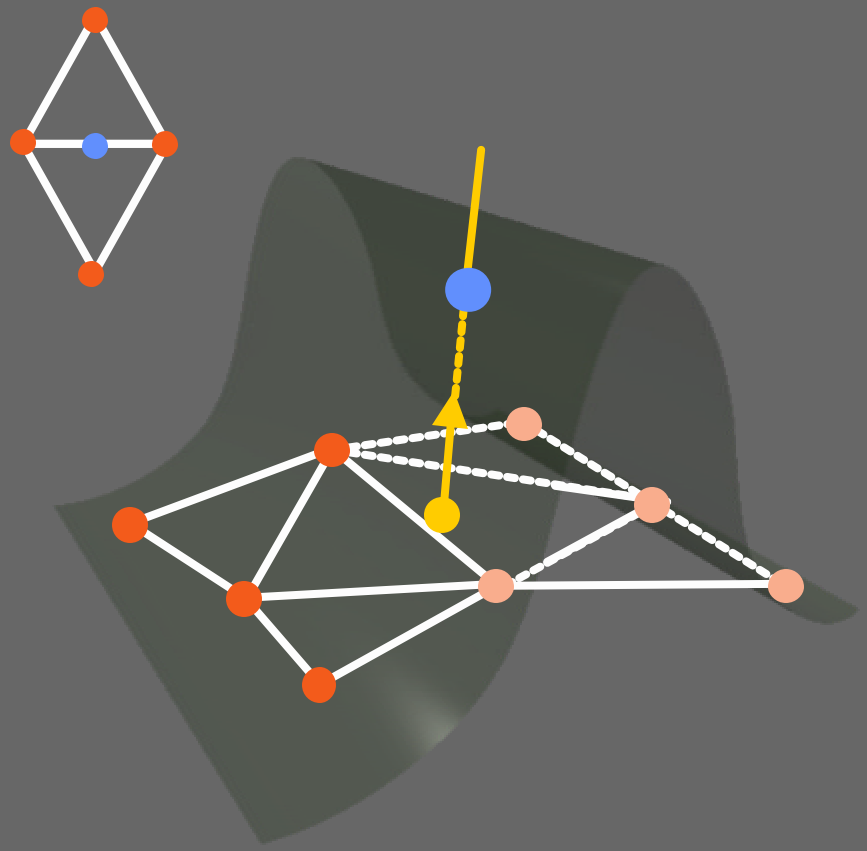


Remeshing Procedure

Correspondence: patches

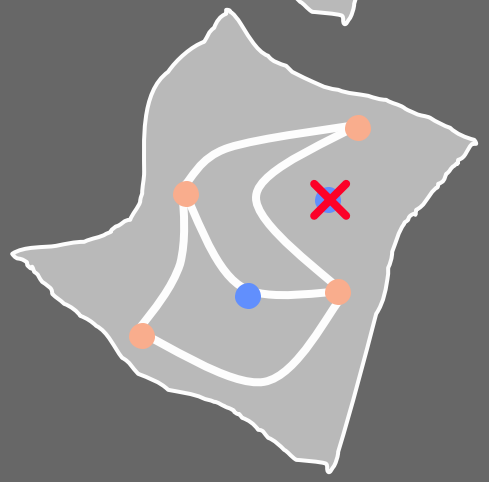


Piercing & Patches

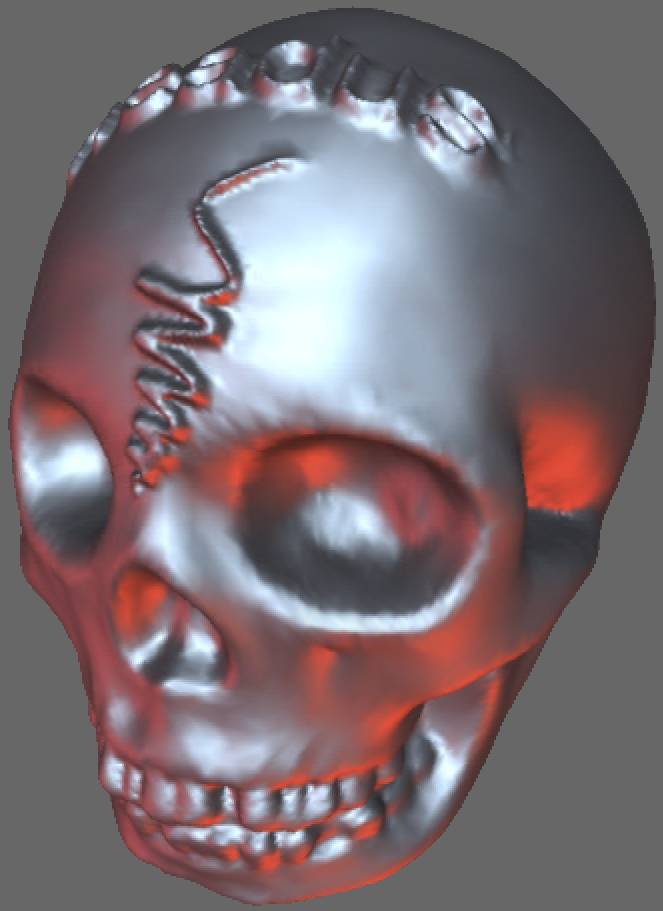


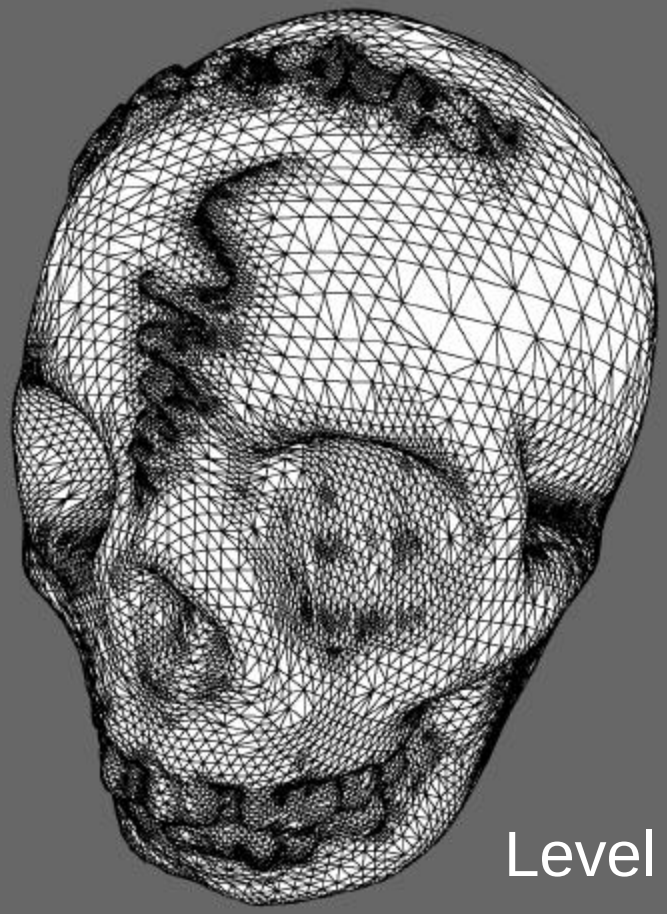
Canonical:

Exception:



Adaptive Remeshing



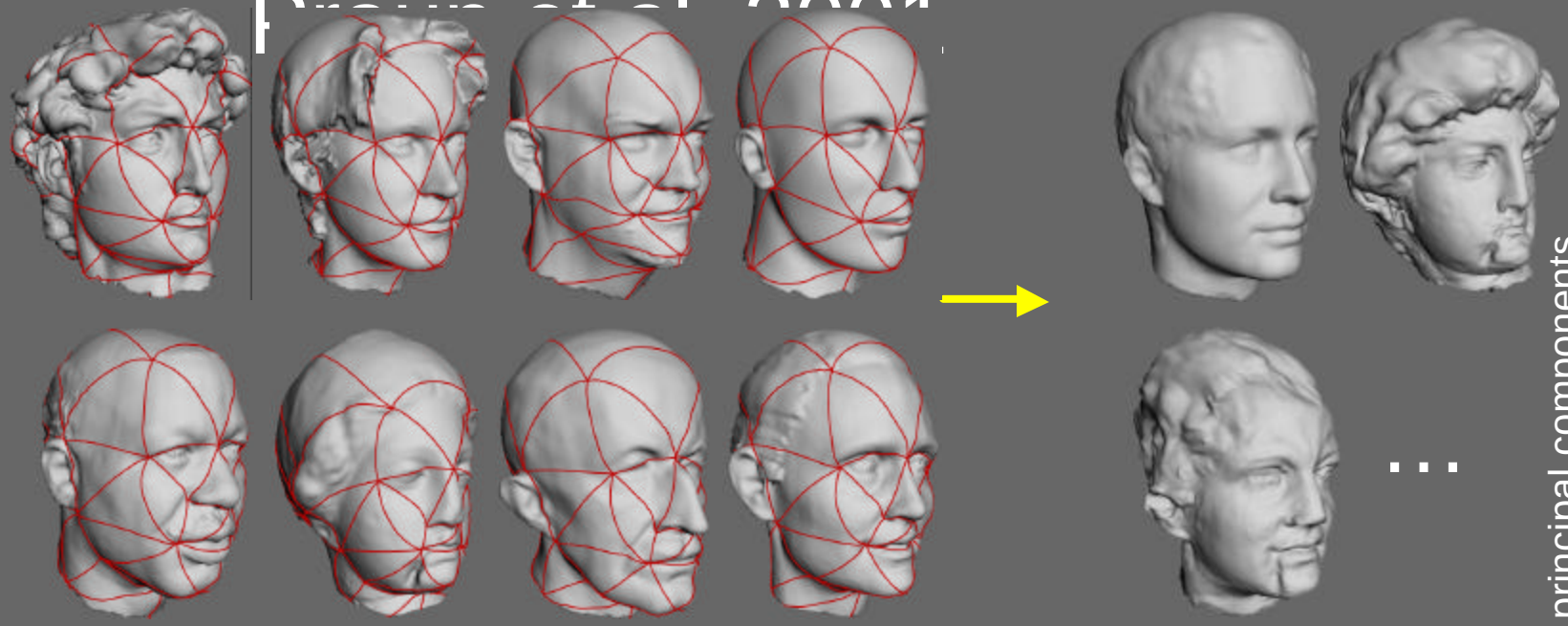


Level 10

Multiple Models

Consistent mesh parameterizations

Braun et al. 2001



Entropy Coding

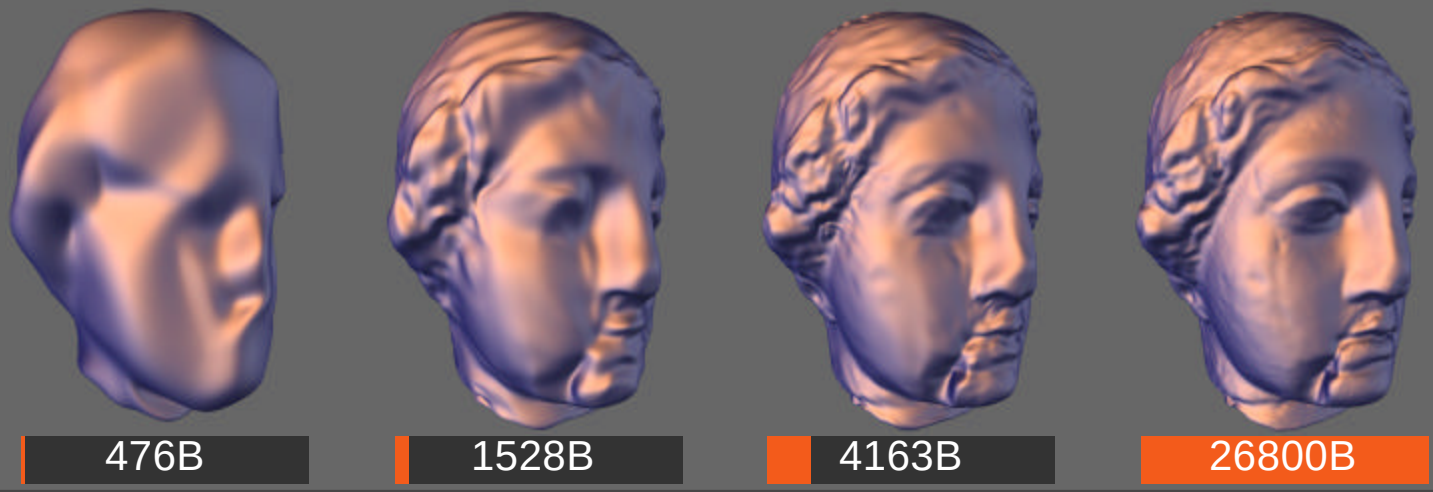
Transform coding

- wavelets successful for images
  - how to make them work for surfaces?
    - how to even construct them…
    - properties… vanishing moments?
- never mind…
  - curvelets, ridgelets, etc.

Surface Compression

Geometry compression

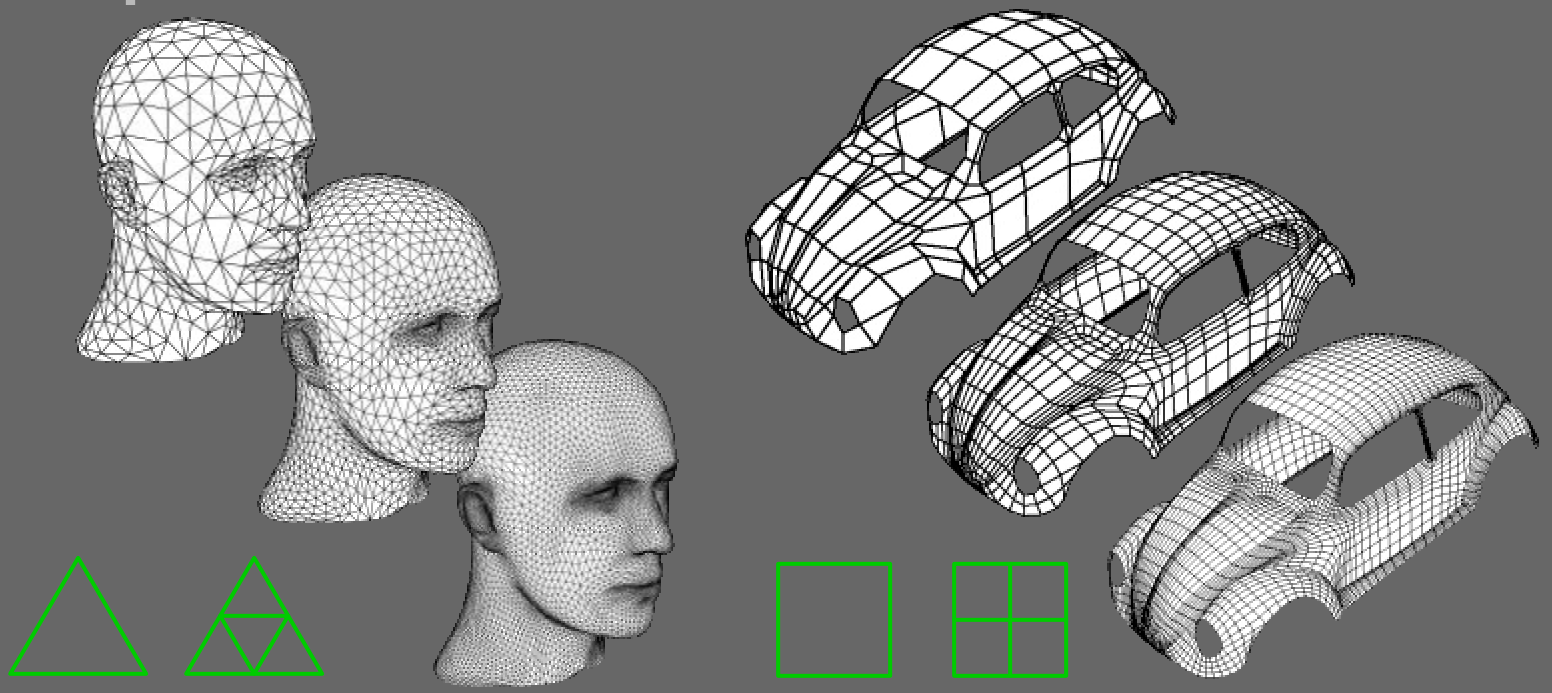
- methods for “large” geometry
- progressivity critical



Compression of Geometry Datasets

What is Subdivision?

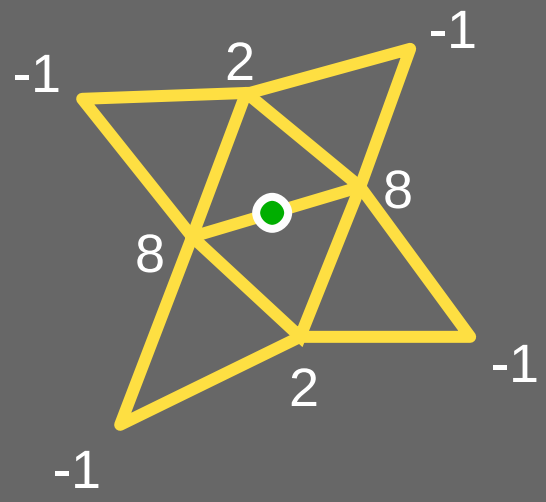
Smooth surfaces as the limit of a sequence of refinements

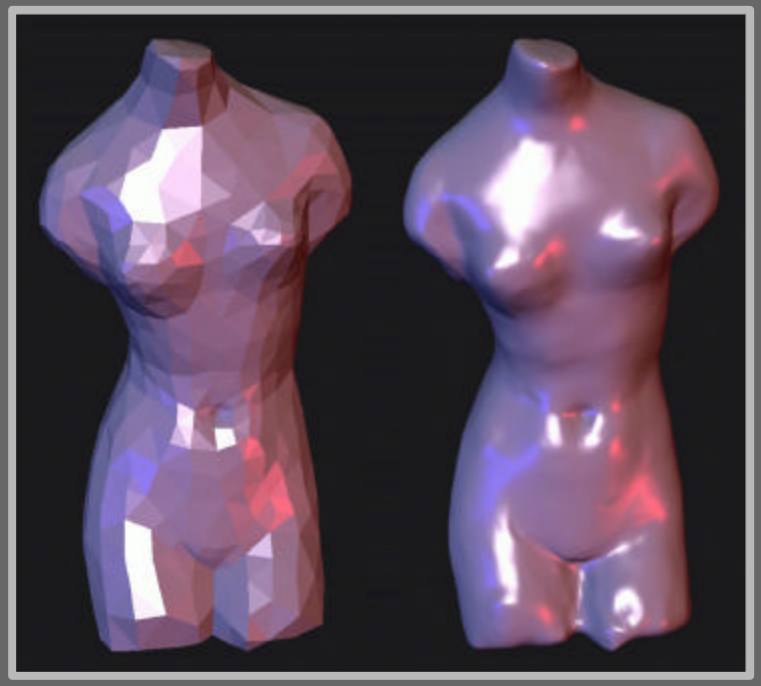


Interpolating

Keep old points, insert new ones

- affine combination of nearby points

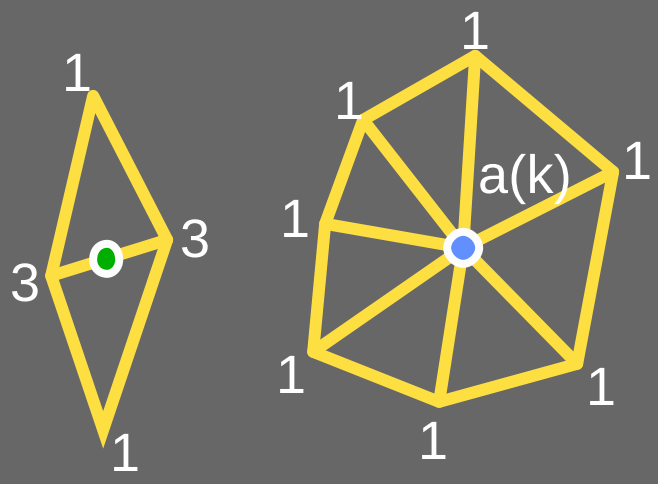


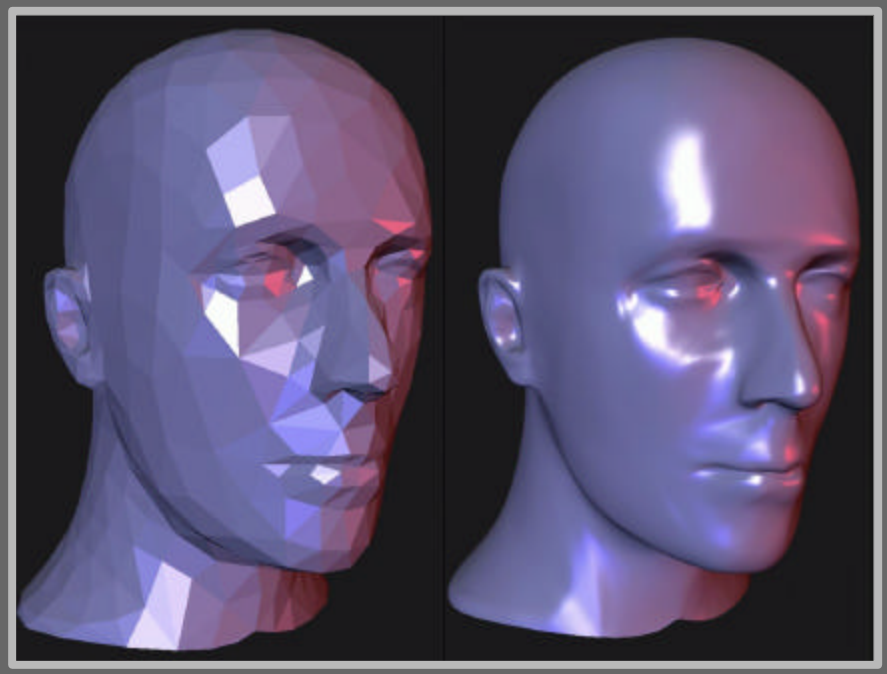


Approximating

Insert new, smooth new and old

- generalizes splines

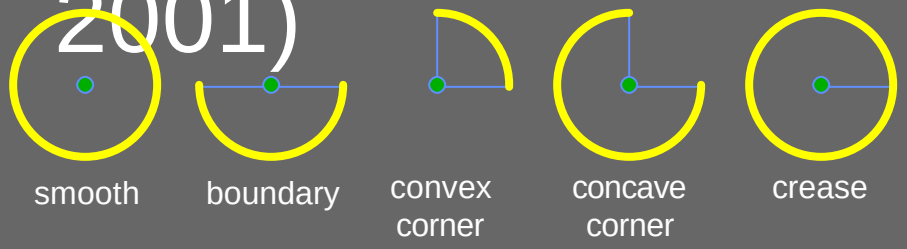


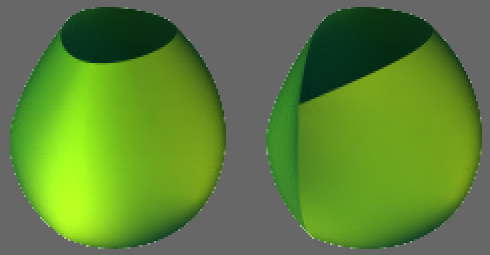


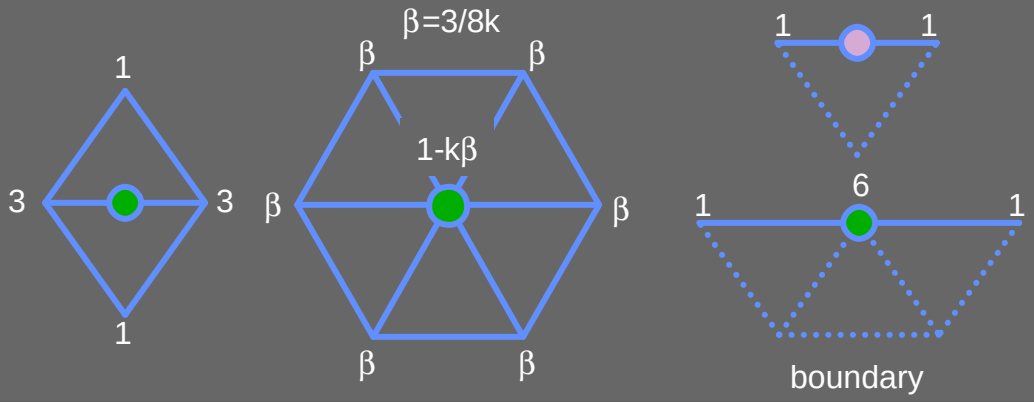
Loop Scheme

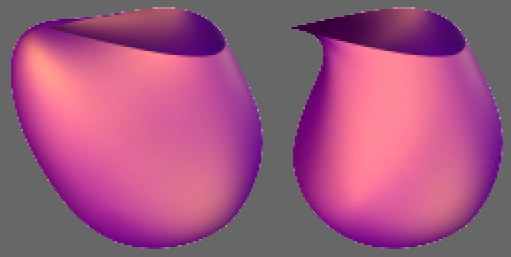
Generalizes quartic box splines

- simple rules (Biermann et al. 2001)





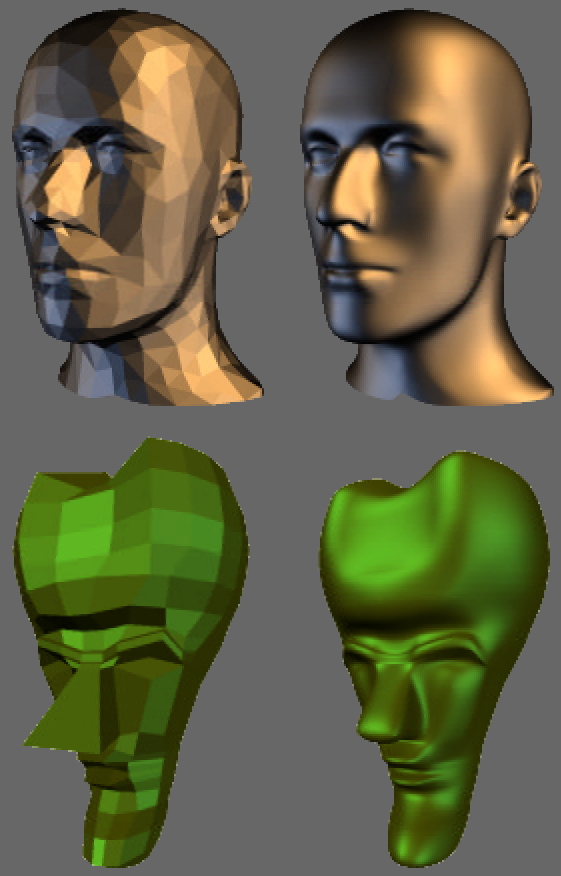




Why Subdivision

Many advantages

- arbitrary topology
- scalable
- wavelet connection
- easy to implement
- efficient



Successive Scales

- Add details in every subdivision step
  - subdivide then displace

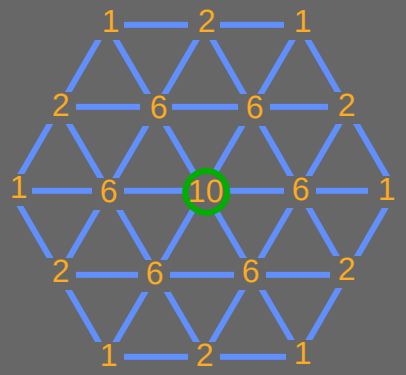
Loop Wavelets

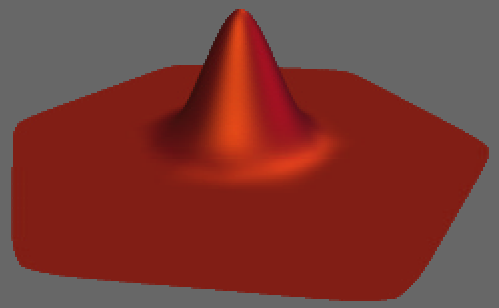
Refinement relations

- subdivision yields scaling functions

$$p^{j+1} = Sp^{j}$$

$$\varphi(t) = \sum_{k \in Z^2} s_k \varphi(2t - k)$$





scaling function

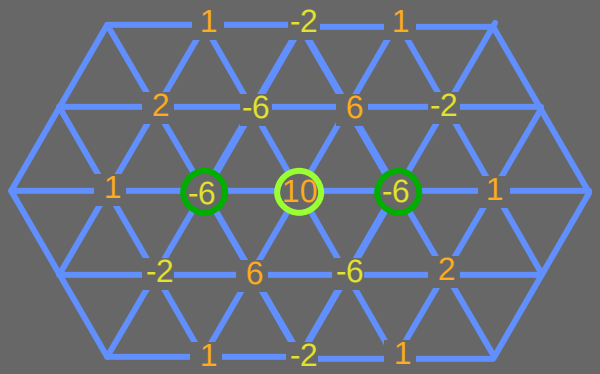
- primal scaling function fixed
  - find completion

$$p^{j+1} = (SQ)\begin{pmatrix} p^{j} \\ d^{j} \end{pmatrix}$$

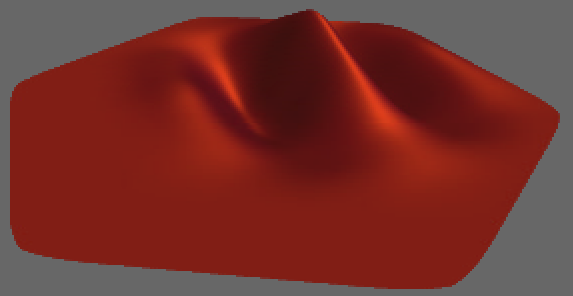
Completion

Completion yields wavelet filters

- want finite reconstruction filters
- QMF for Q, but: $(SQ)^{-1}$ not finite



$$\psi(t) = \sum_{k \in Z^2} d_k \varphi(2t - k)$$



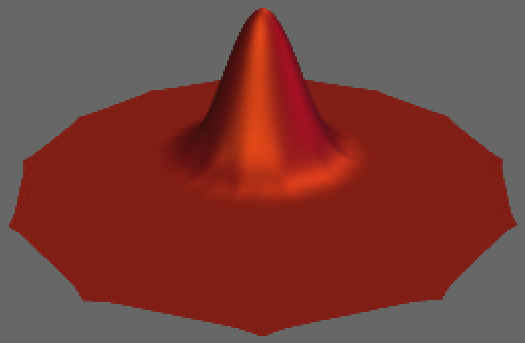
wavelet

1 of 3 different orientations

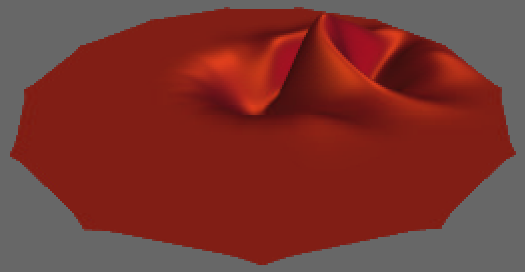
Loop Wavelets

Irregular vertices

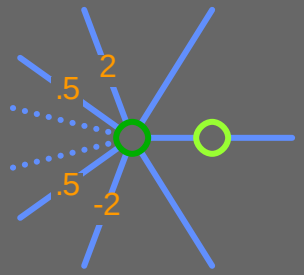
- split filter around irregular vertex
  - results in good conditioning
  - boundaries: extension



Scaling function, valence 13



Wavelet function, valence 13



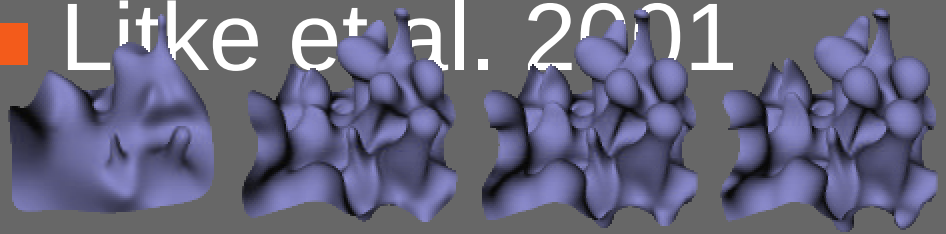
Some Basic Questions

Constructing function systems

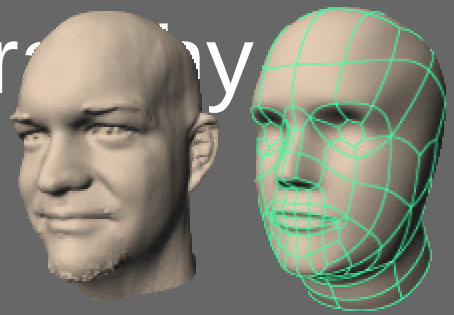
- how to take geometry into account?
  - approximation on/of manifolds
  - point and curve singularities
- standard tools essentially based on Fourier analysis
  - very limited usefulness

Surface Fitting

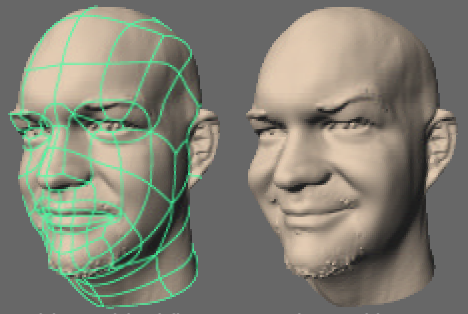
- Quasi-Interpolation for subdivision
  - compute details in hierarchy
    - LSQ optimal
    - local: no solver!
  - Litke et al. 2001



iso-surface: 1% 10% 20% 100%



original (300k) base domain

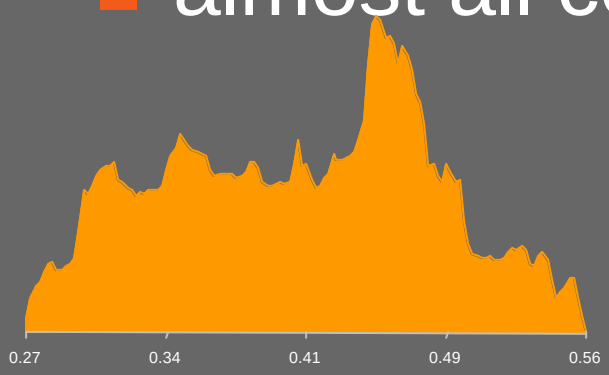


hierarchical fit animated base

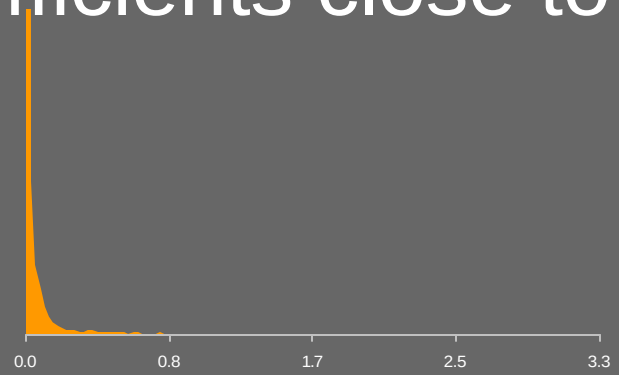
Wavelet Transform

Effect of wavelet transform

- changes distribution of coefficients
  - almost all coefficients close to zero



Vertex coordinates

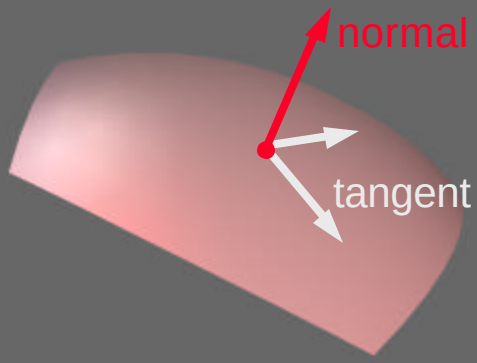


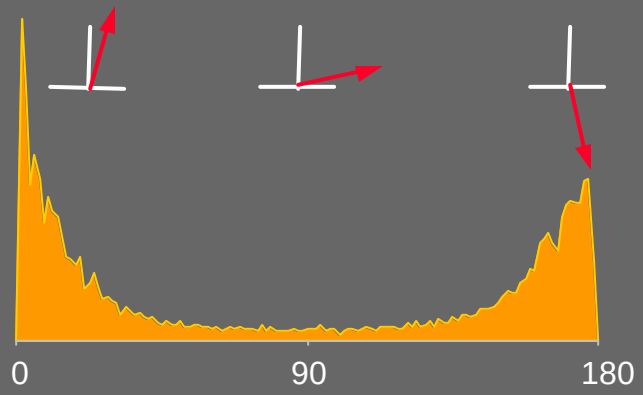
Wavelet coefficients

Local Frames

Coefficients are vector valued

- normal direction most important!





- best results with scalar quantization
- finer quantization of normal direction

Progressive Encoding

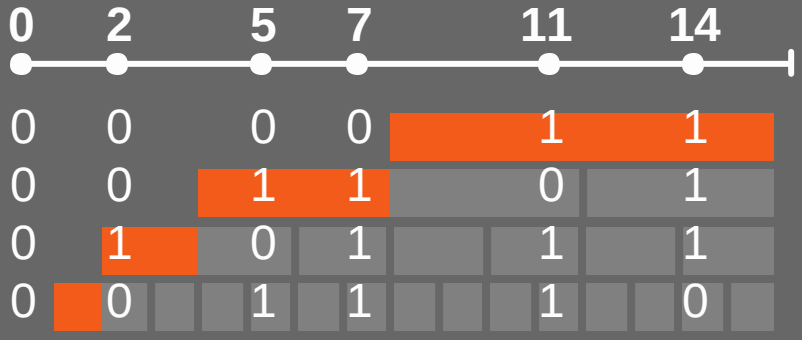
Make each bit count

- no sense in sending approximate geometry very precisely…
- sending coefficients one at a time not enough

Bitplane Encoding

Progressive compression:

- encode largest coefficients first
  - encode only significance bit
  - subsequent bits in later iterations



Send: new coefs. refinemt. bits

2
2 0 1
1 0 1 1 1
0 0 1 1 1 0

Zero-Tree

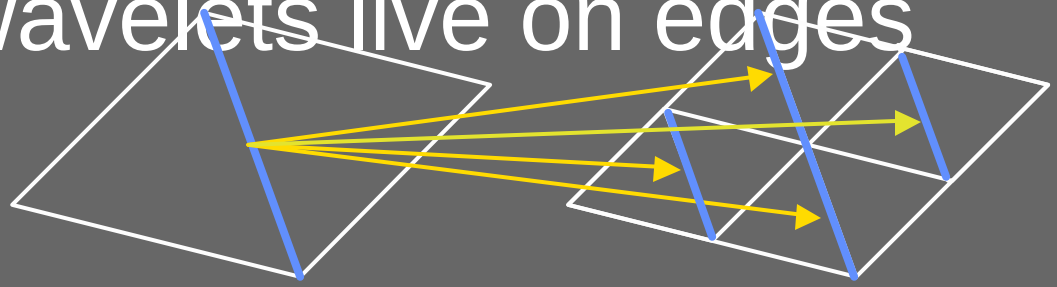
Why it works

- be smart about the location of large coefficients
  - small parents tend to have smaller children > entropy coding gain
- run state machine at encoder and decoder
  - only send synchronization bits

Zero-Tree

Need tree structure for coefficients

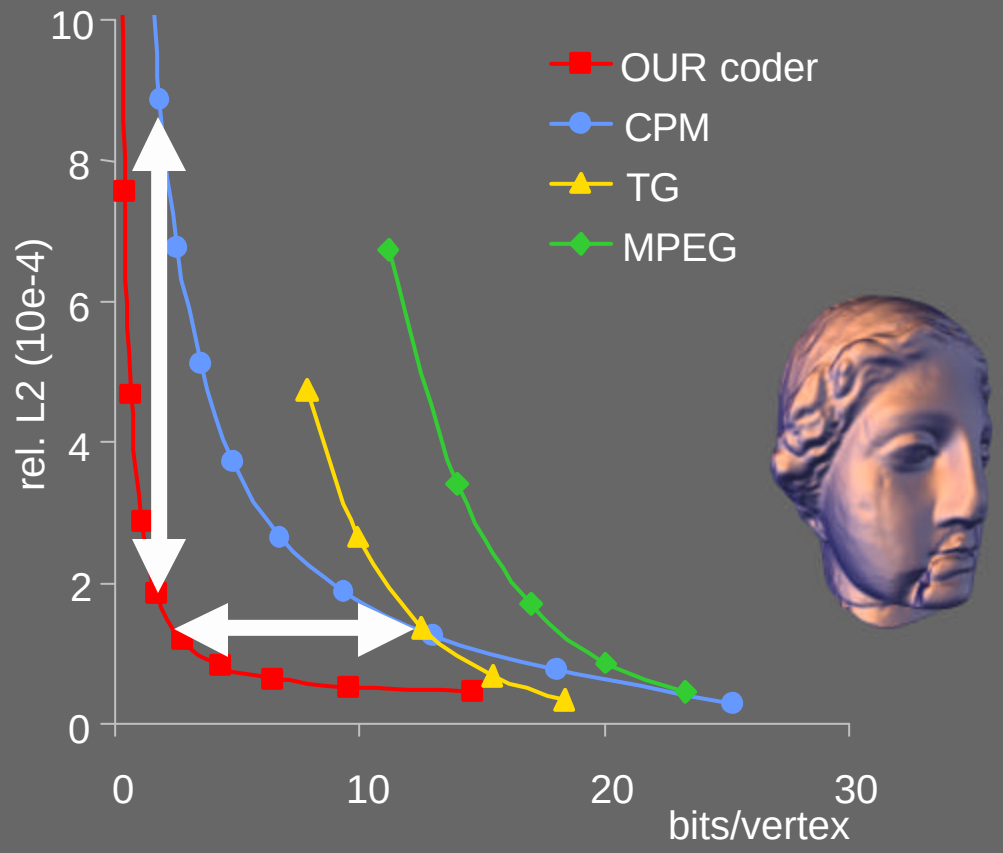
- wavelets live on edges



Test whole tree for significance

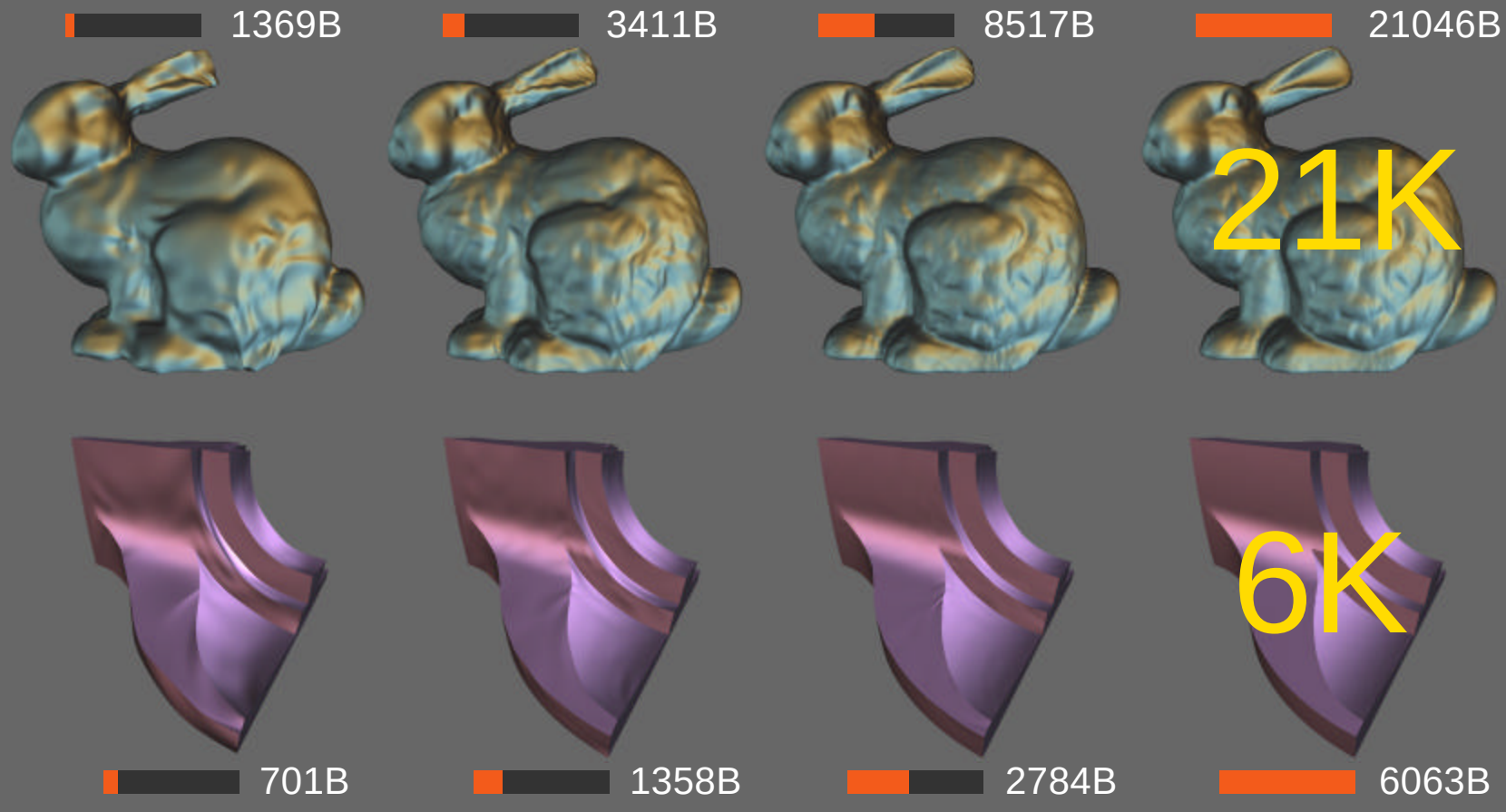
- split tree isolating significant coefs

Comparison



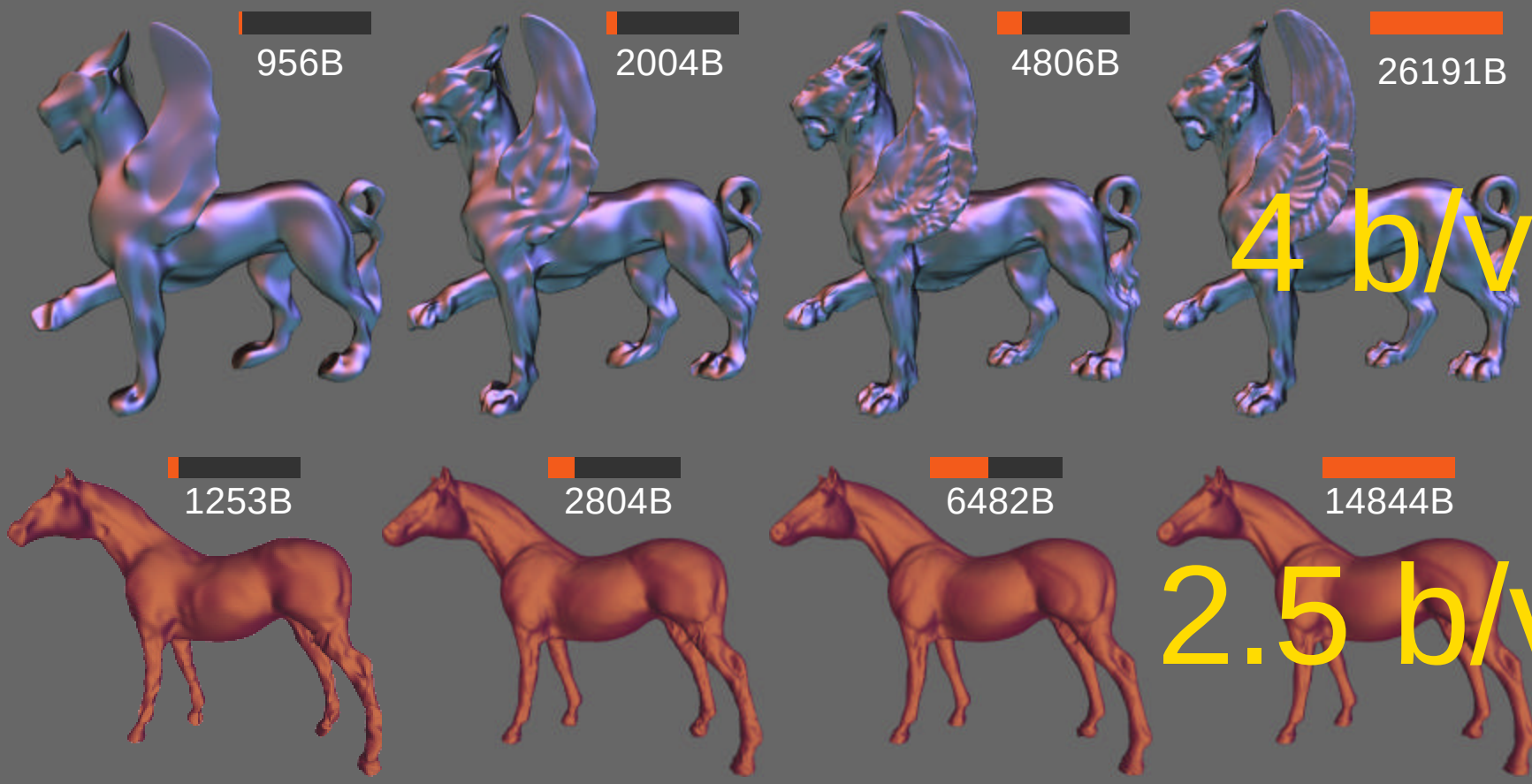
Compression of Geometry Datasets

Results III



Compression of Geometry Datasets

Khodakovsky et al.2000



Entropy Coding

Zero-tree ok, but…

- what are the statistics?
  - file sizes surprisingly small
- context coding
- different data sources
  - volumetric imaging with limited accuracy (take acquisition process into account)

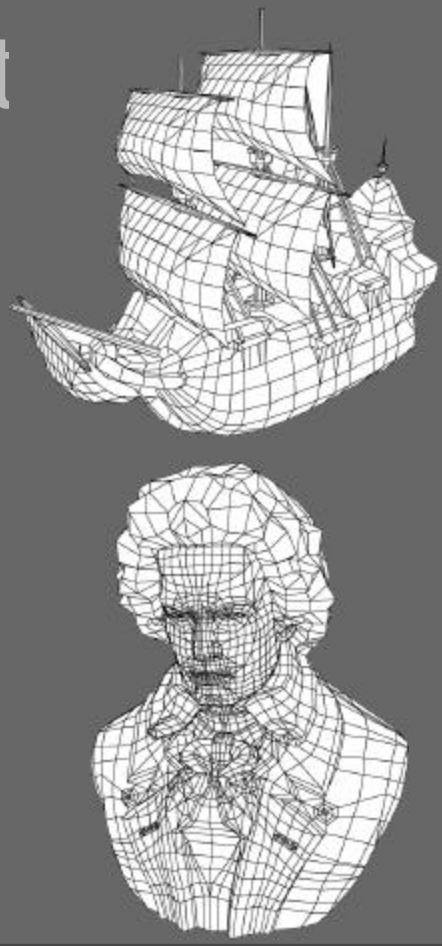
Basic Questions

Encoding of connectivit

- 2-manifolds
- triangle meshes
- arbitrary polyhedra

Long term

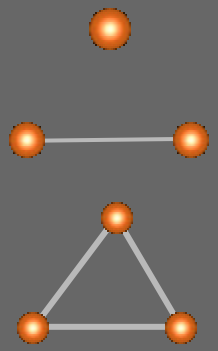
- entropy of a surface
- sampling theory

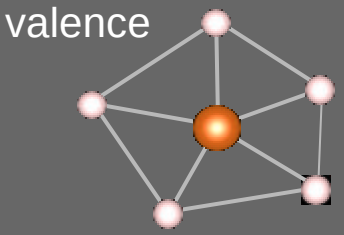


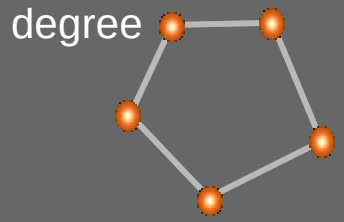
Basic Definitions

Basic elements

- V vertices
- E edges
- F faces

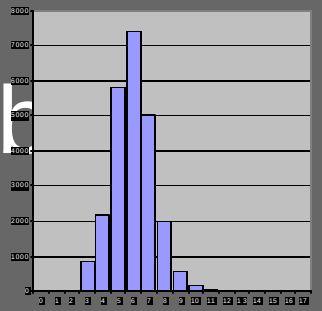


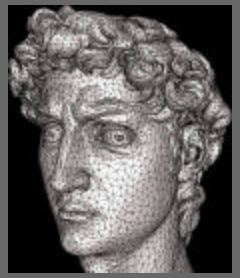




Euler characteristic

- F – E + V = 2(c-g) – h

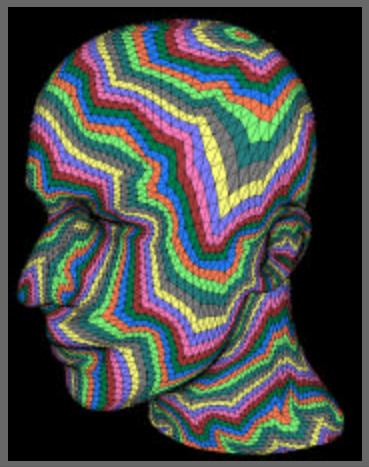


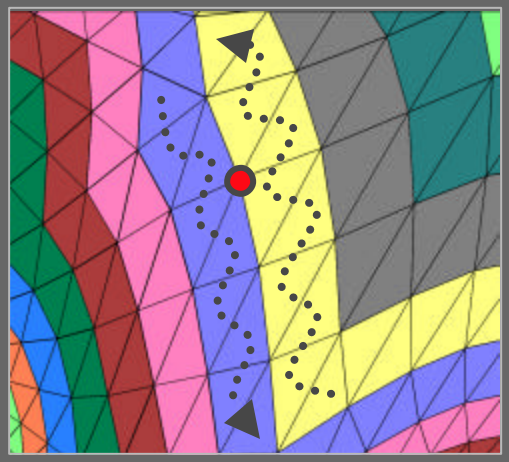


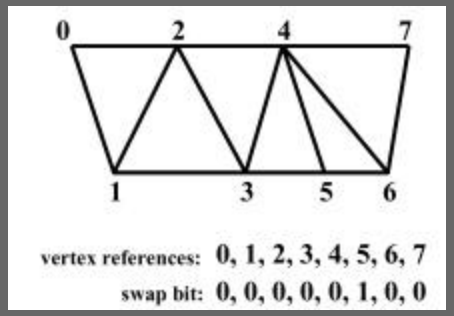
Algorithms

Generalized triangle strips

- Deering 95
  - triangle strips w. stack for reuse
  - 8-11 b/v



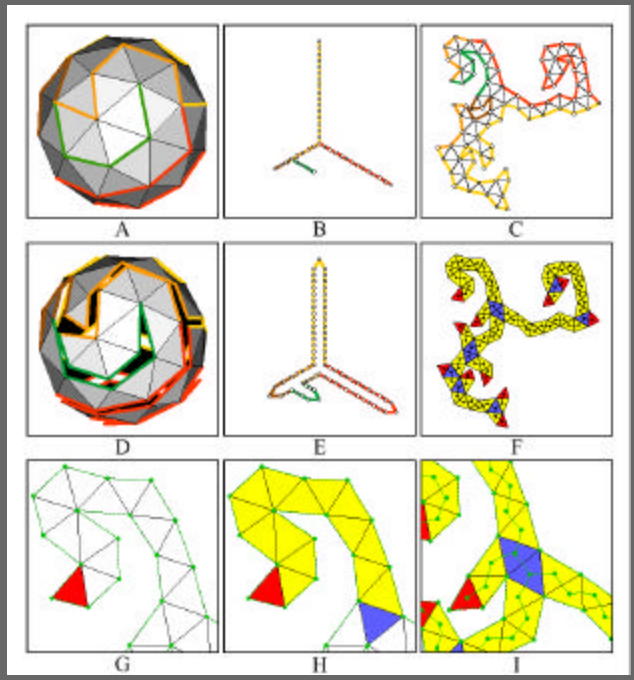




Algorithms

Spanning trees

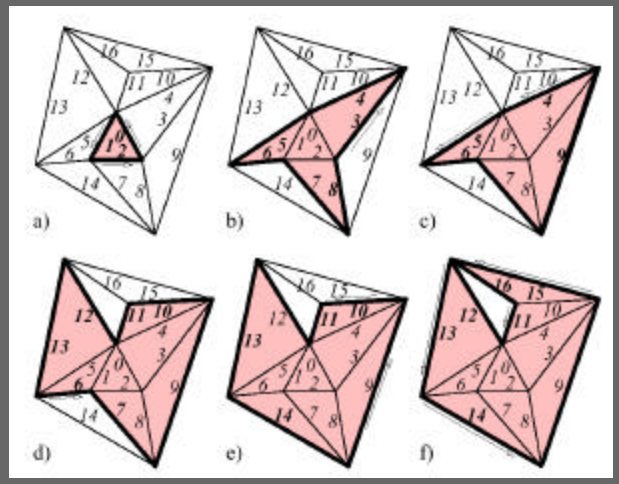
- Taubin&Rossignac 98
  - zigzag walk
  - 0/1 patterns
  - entropy code
  - 4b/v
- basis for MPEG-4



Algorithms

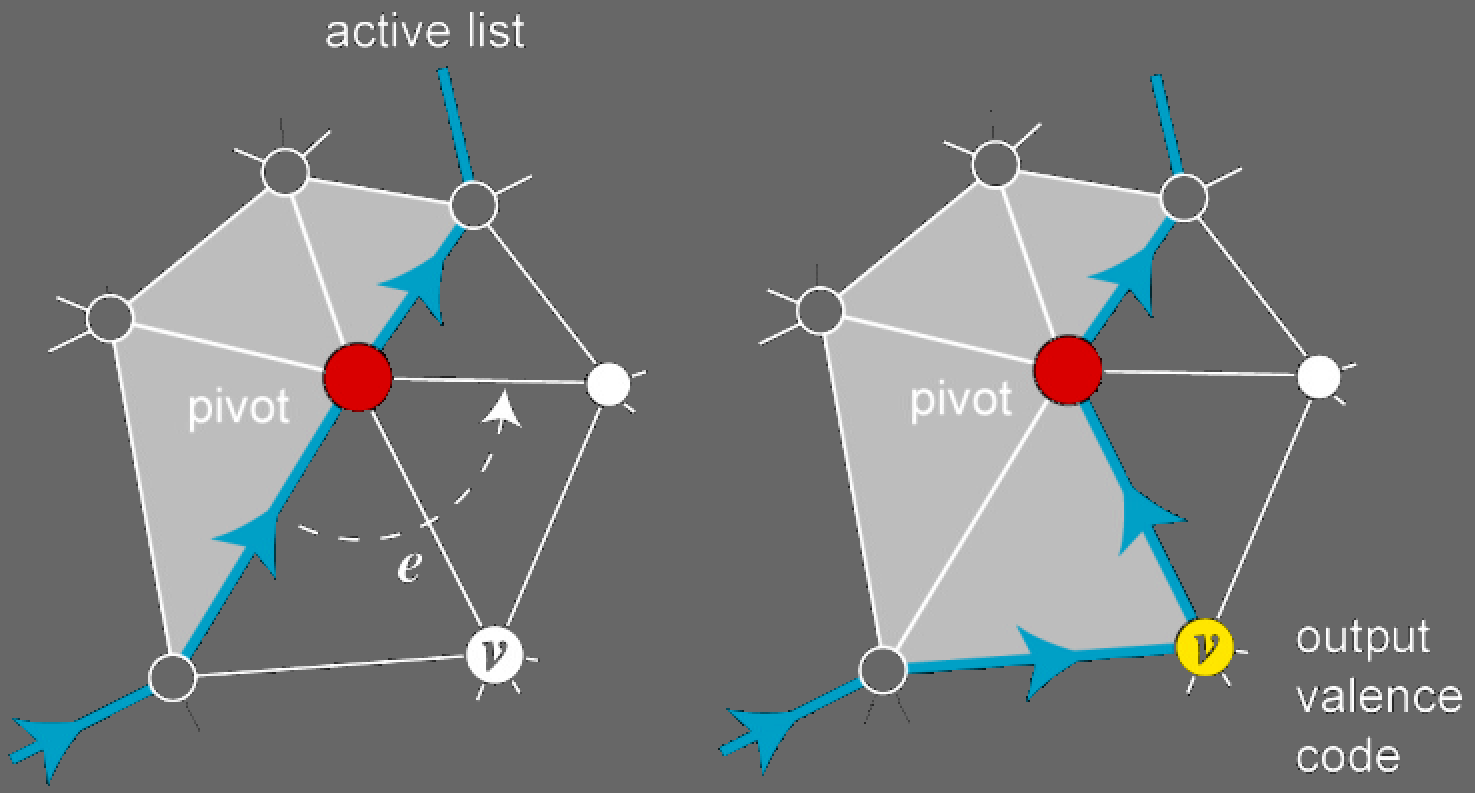
Conquer mesh

- Gumhold&Straßer 98
  - expand boundary
  - use 8 codes
  - 4.36 b/v
- Rossignac 98
  - gate based traversal



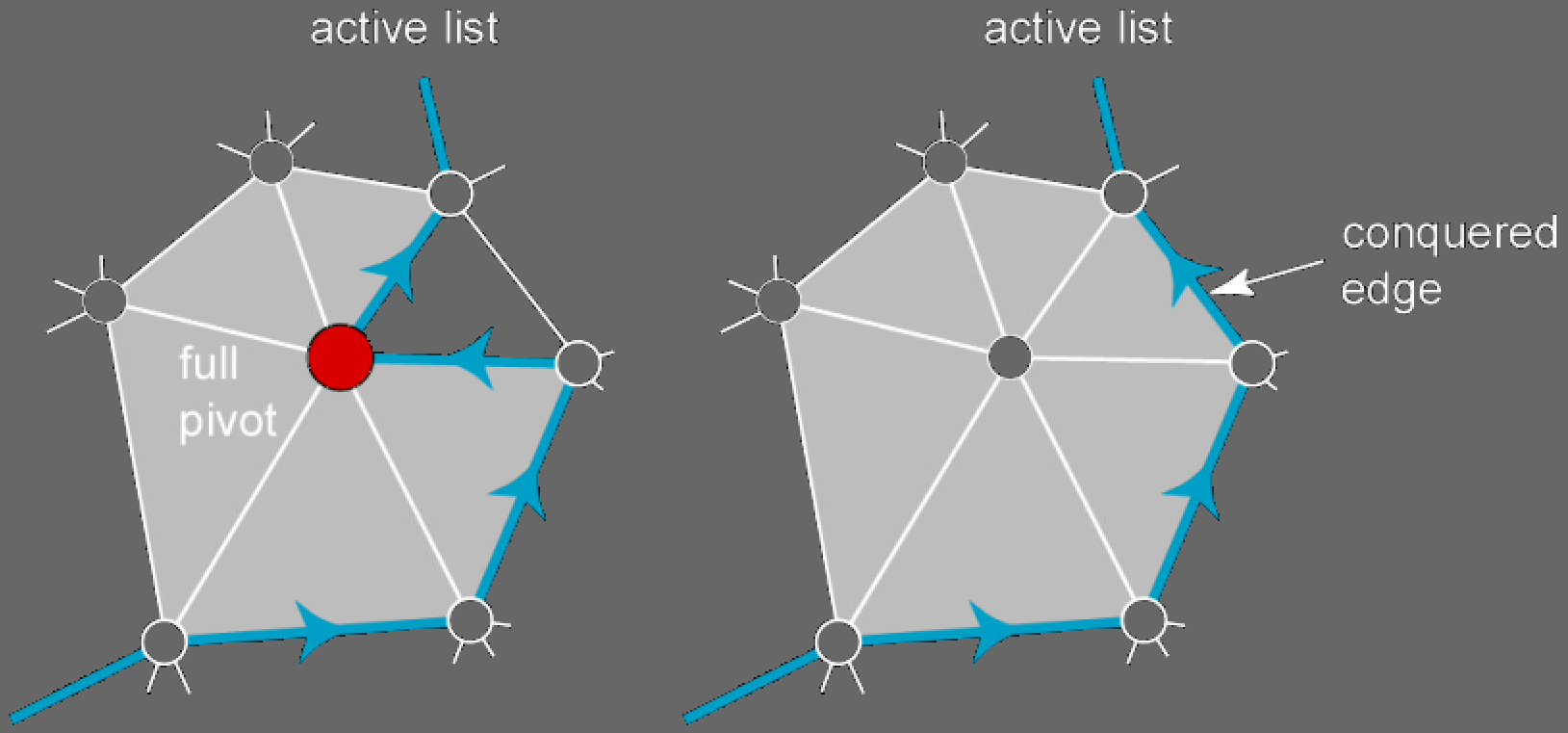
Touma&Gotsman 98

Atomic edge conquest



Touma&Gotsman 98

Full pivot vertex

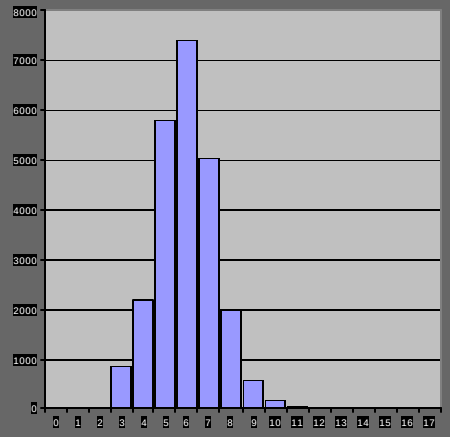


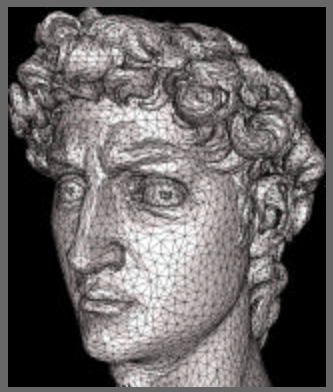
Valence Approaches

Very attractive

- exploit statistics of valences
  - arithmetic coder
- worst case distribution?
  - exponential!

$$e = \sum_i p_i \log_2(1/p_i)$$





Valence as Entropy

What is the worst case?

- maximize: $e = \sum_i p_i \log_2(1/p_i)$
- with constraints:
  - $p_i = v_i/V$: $\sum p_i = 1$
  - F – E + V = c : $\sum i p_i = 6$

$$\sum p_i \log_2(1/p_i) + l\left(\sum p_i - 1\right) + m\left(\sum i p_i - 6\right)$$

Valence as Entropy

Worstcase distribution p_i

- necessary condition

$$\log_2(p_i) = l - 1/\ln(2) + m\,i \qquad p_i = \frac{16}{27}\left(\frac{4}{3}\right)^{-i}$$

$$e = \sum_i p_i \log_2(1/p_i) = \log_2(256/27) \approx 3.24$$

Tutte census: 1962

$$\frac{2(4V+1)!}{(3V+2)!\,(V+1)!} \qquad 2^{\log_2\left(\frac{256}{27}\right)V} = 2^{3.24\,V}$$

Overall Focus

Compression of geometry data

- use good error metrics
- pay attention to structure
- settings: functional, manifold
- areas: terrain, reverse engineering, scientific computing
- properties: drill-down, registration

Starting Up

Project vectors

- theory
  - optimal Haussdorf approximation
    - can we meet Kolmogorov entropy?
    - what class of functions?
  - connectivity entropy
  - R/D optimal parameterization

Starting Up

Experimental

- streaming of compressed terrains
  - Ilja Friedel (grad)
    - collaboration with LLNL
- compression of evolving surfaces
  - Nathan Litke (grad)
    - collaboration with ASCI