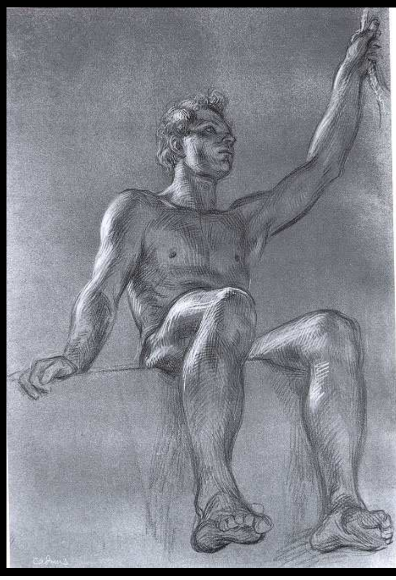
Non-photorealistic Rendering

EECS 487



Cadmus

3D Computer Graphics Today

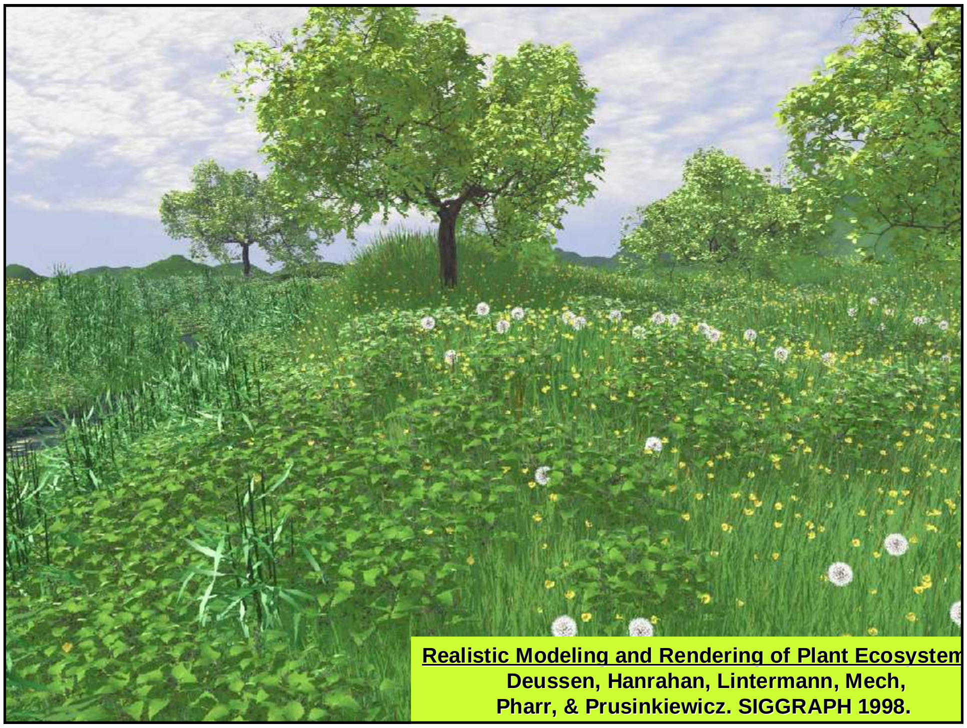
- Miraculous performance leaps
- Stunning price cuts
- Curiously low impact
  - Games
  - Movies

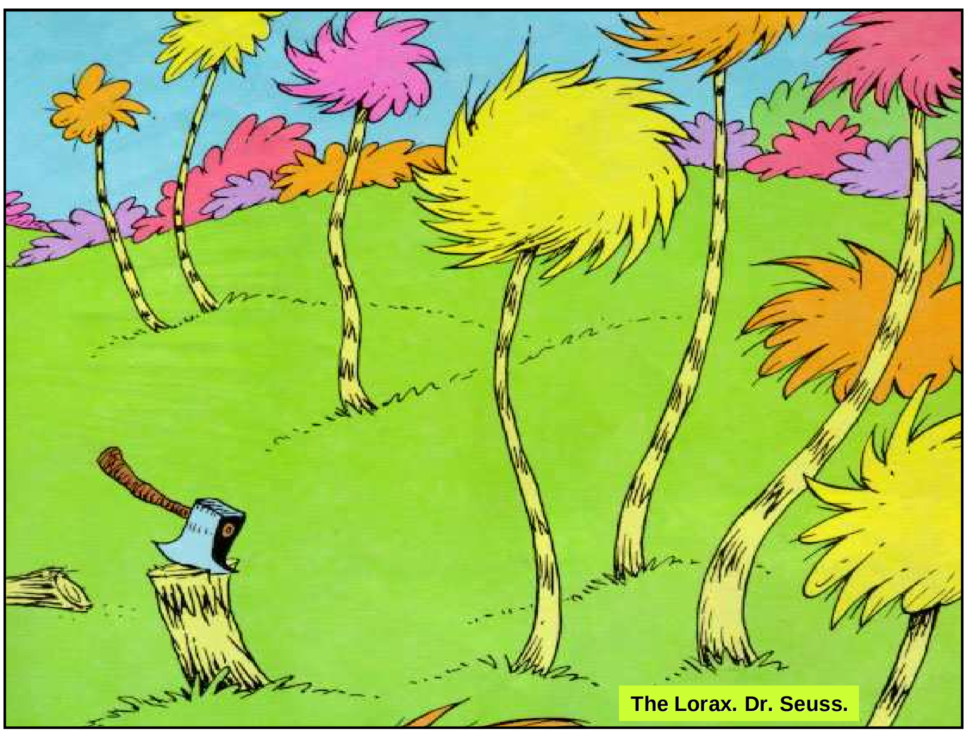
Problem: Content Creation

Available tools are difficult (Maya, 3DS…)

- Evolved from CAD (precise modeling)
- Requires special skills
- Geared toward trained experts
- Realism – no stylization or abstraction

Realism is expensive!


Realistic Modeling and Rendering of Plant Ecosystem
Deussen, Hanrahan, Lintermann, Mech,
Pharr, & Prusinkiewicz. SIGGRAPH 1998.


The Lorax. Dr. Seuss.

Non-photorealistic rendering (NPR)

Extreme reduction of details

Selective enhancement

Stylization and abstraction

- Complexity is suggested

Proposal: Model by Drawing

Draw shape *and* style

Permit abstraction / stylization

Stroke-based NPR

Potential Advantages

- Gain abstraction, stylization
- Re-use drawing skills
- Re-use existing images
- Re-use existing shapes
- Fast, lightweight modeling
- New applications, users
  education, architecture, design, animation, advertising, games…

Research Challenges: NPR

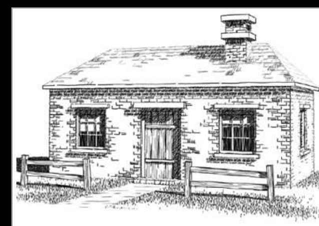
Stroke generation
- Levels of detail
- Temporal coherence
- Pattern synthesis

Media simulation
Direct user control

Picture element: pixel or stroke?

Previous work

Technical Illustration [Saito 90]



Pen & Ink [Winkenbach 94]

Watercolor [Curtis 97]

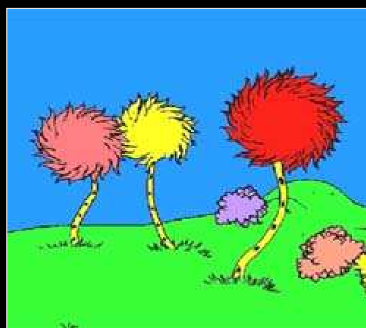
Paint [Hertzmann 98]

Previous work

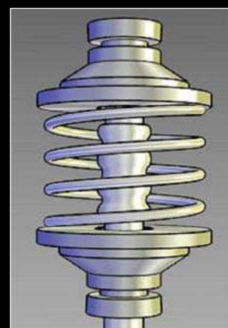
Painterly rendering for
3D models [Meier 96]

Painterly rendering for
video [Litwinowicz 97]

Previous work



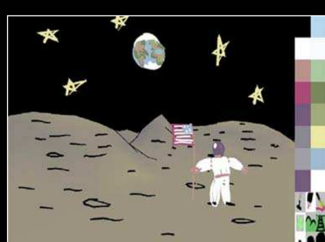
[Kowalski 99]



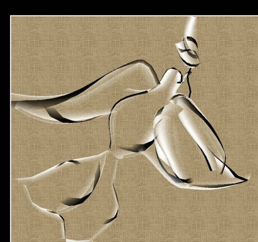
[Gooch 98]

[Praun 01]

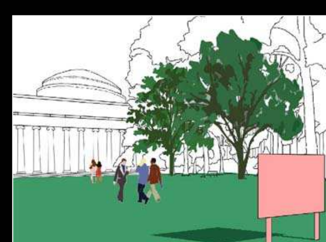
Previous work



[Cohen 00]

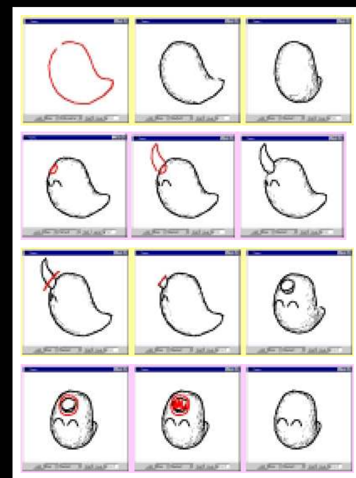


[Bourguignon 01]



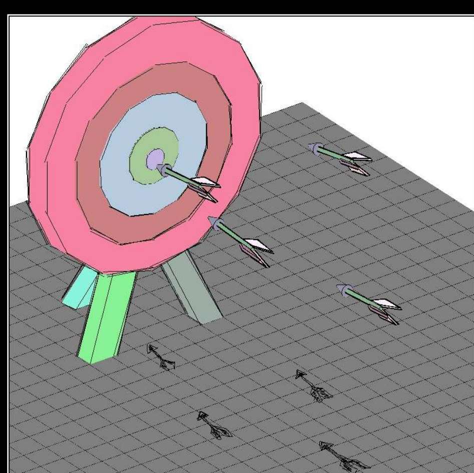
[Tolba 01]

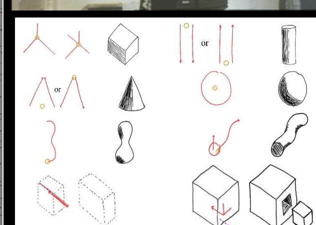
Previous work



Teddy: A Sketching Interface for 3D Freeform Design. Igarashi, Matsuoka & Tanaka. SIGGRAPH 1999.

Previous work





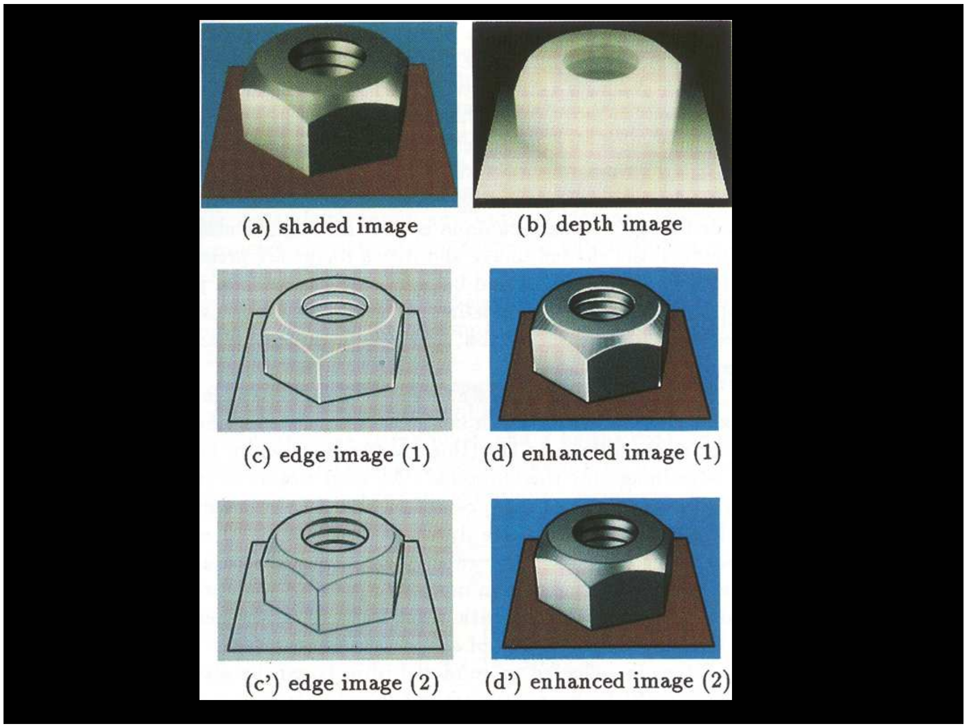
SKETCH: An Interface for Sketching 3D Scenes. Zeleznik, Herndon & Hughes. SIGGRAPH 1996.

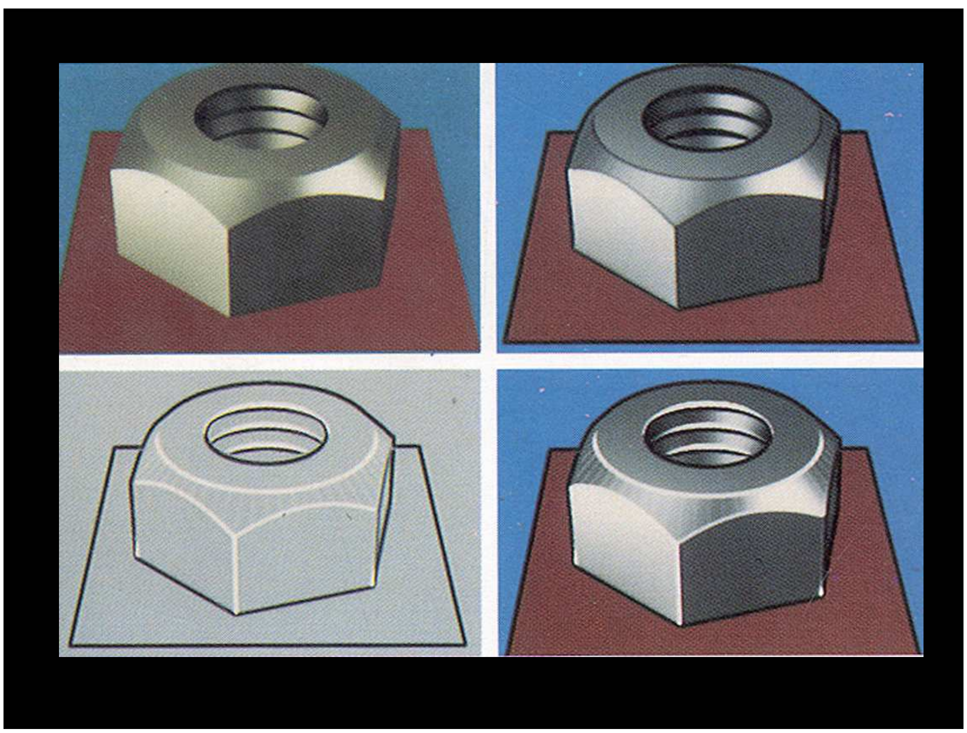
Talk overview

- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graftals
- WYSIWYG NPR
- Coherent stylized silhouettes

Technical illustration

- Saito and Takahashi, Siggraph 90
- Purpose: render 3D models in styles that are more “comprehensible”
- Method:
  - Render various intermediate images
  - Do image-processing operations on them
  - Combine the results


(a) shaded image
(b) depth image
(c) edge image (1)
(d) enhanced image (1)
(c') edge image (2)
(d') enhanced image (2)



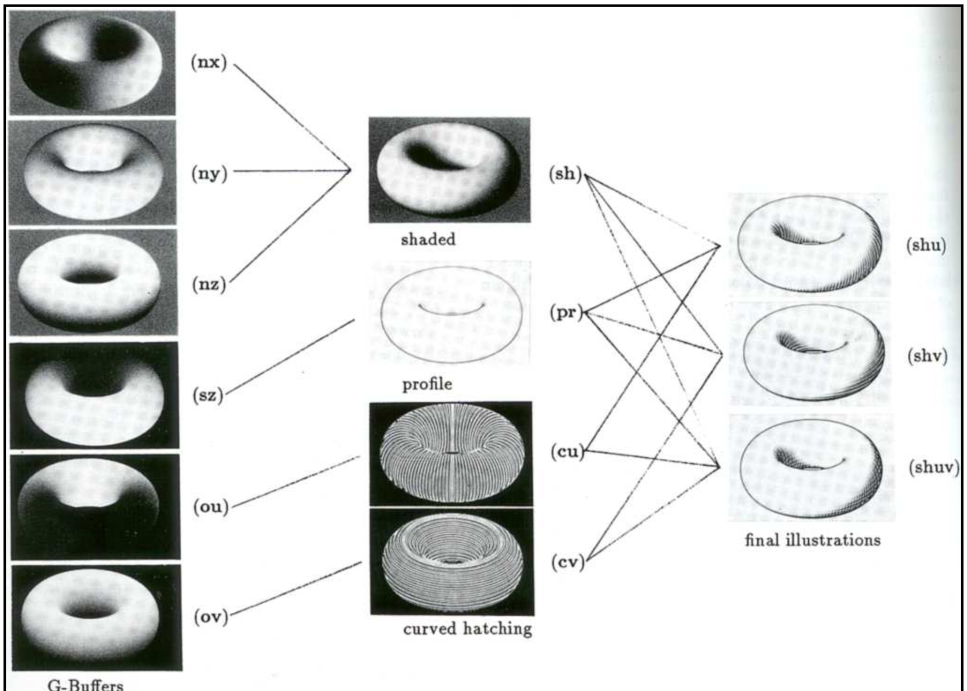
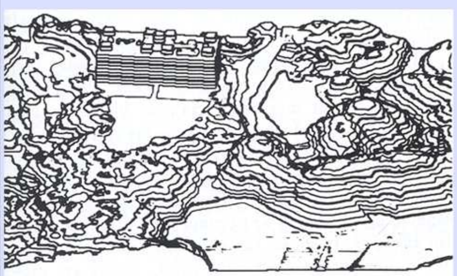
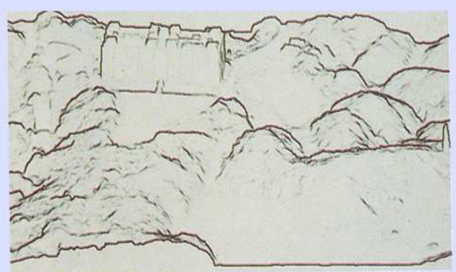


Fig.9 Process of drawing illustrations.

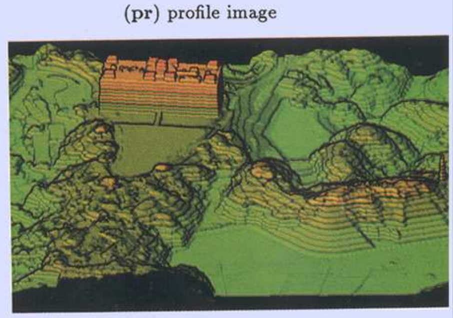


(cn) contour image



(pr) profile image

(mx2) combination of (pr) and (sh)



(mx4) combination of four enhanced images

Problem

Parameters need careful tuning for good results

Talk overview

- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graftals
- WYSIWYG NPR
- Coherent stylized silhouettes

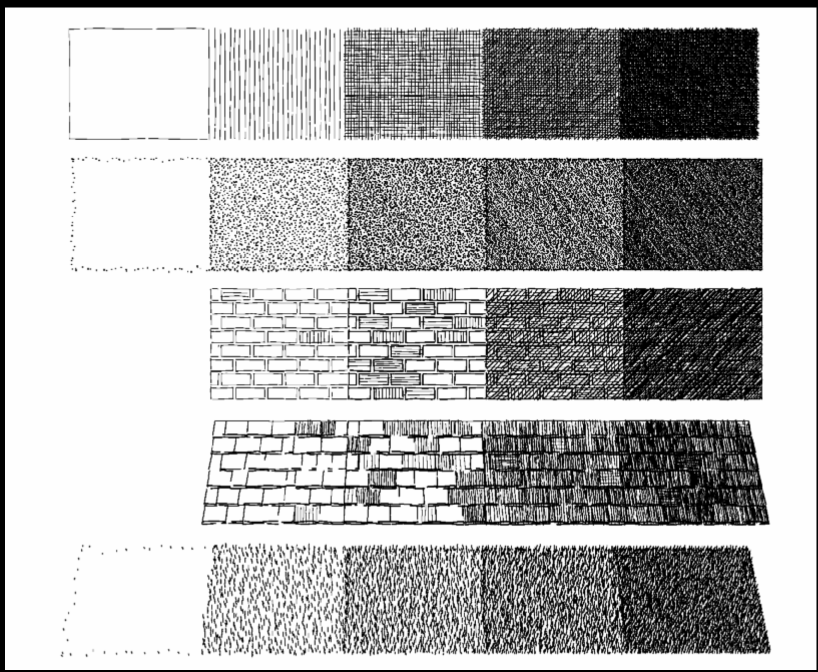
Pen and Ink

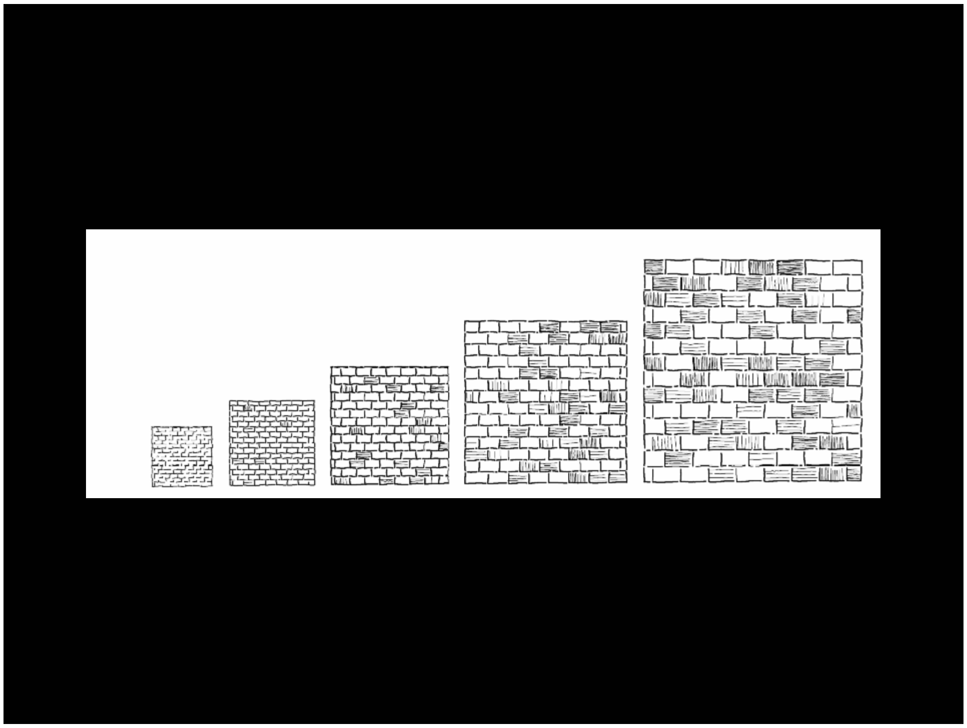
Winkenbach and Salesin, Siggraph 94

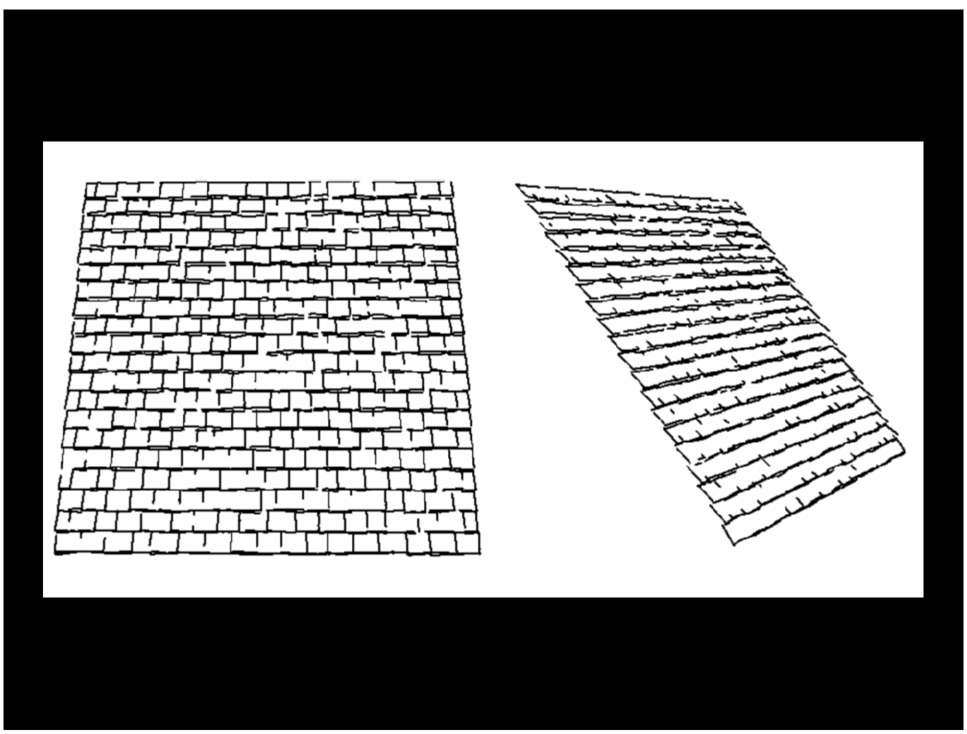
Purpose: render 3D models as pen & ink drawings

Method:

- annotate model with procedural “textures”
- Render tonal “reference image”
- Use it to guide pen and ink textures



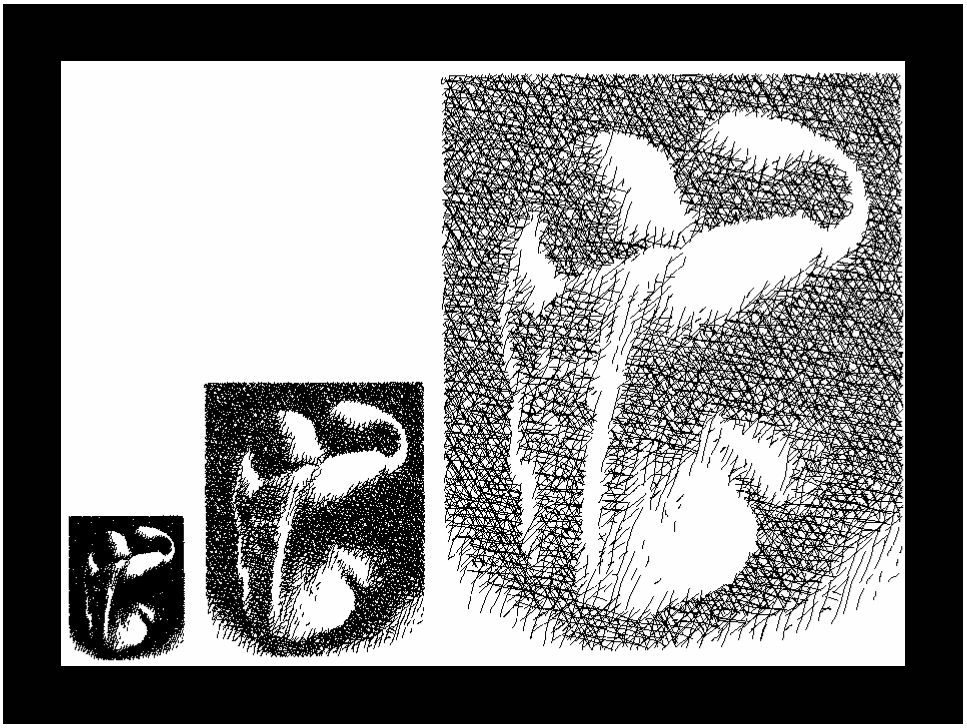


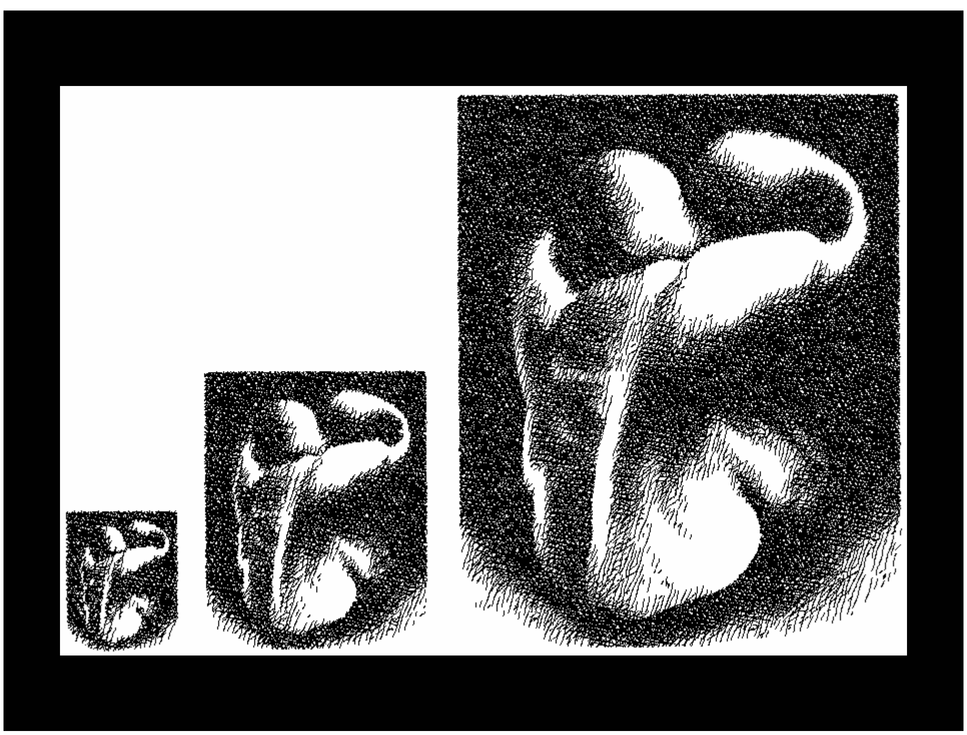


Pen and Ink

Salisbury, Anderson, Lischinski and Salesin, Siggraph 96

Purpose: define a scale-independent representation for pen & ink images

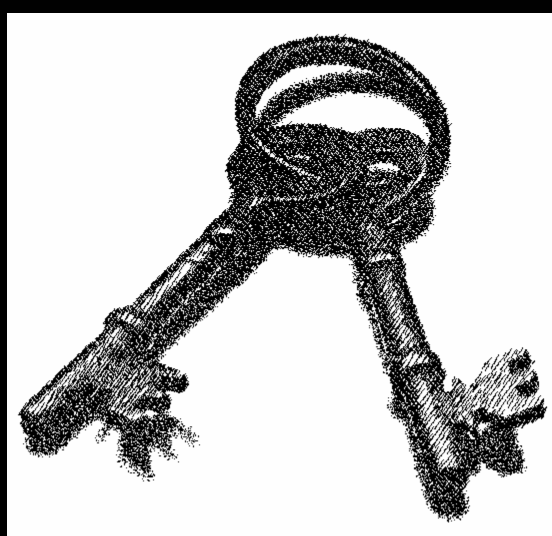


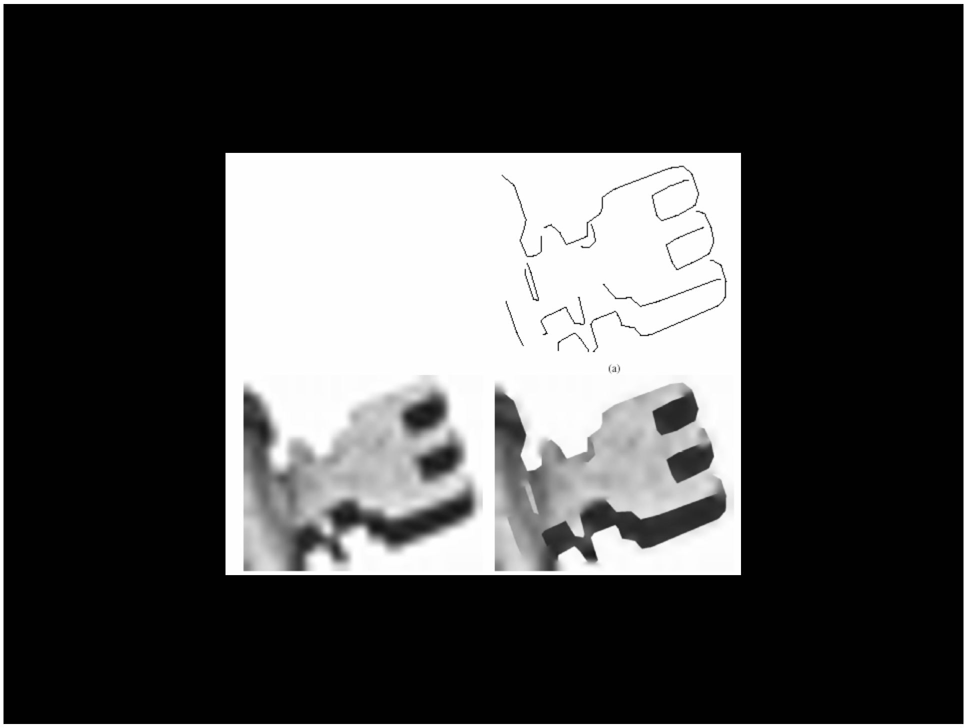


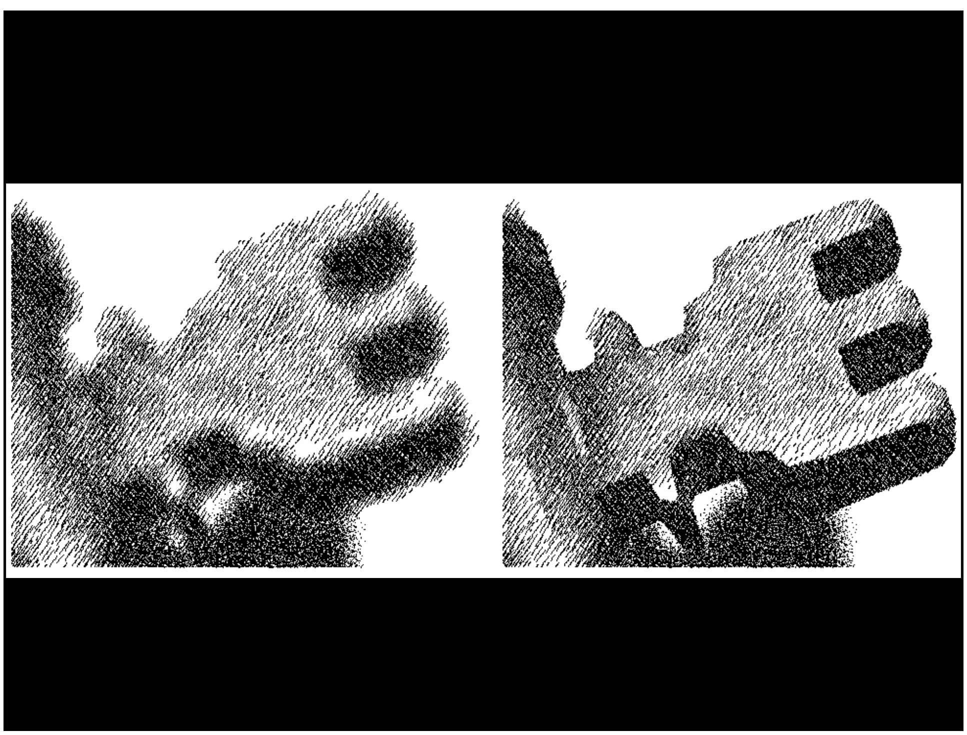
Salisbury et al., cont'd

Method:

- Store lo-res greyscale image annotated with discontinuities
- filter greyscale image to desired size, run stroke generation algorithm on it




(a)



Problems

Only produces still images

- Would not provide temporal coherence

What's the application?

Talk overview

Technical illustration

Pen & ink

Painterly rendering

Silhouette detection

Graftals

WYSIWYG NPR

Coherent stylized silhouettes

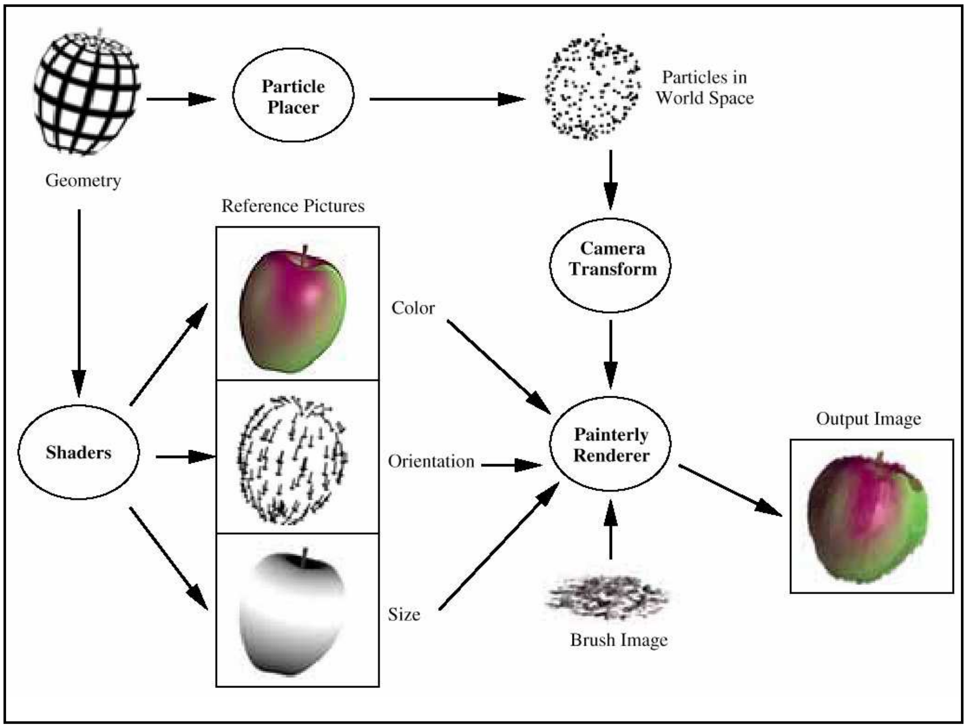
Painterly rendering

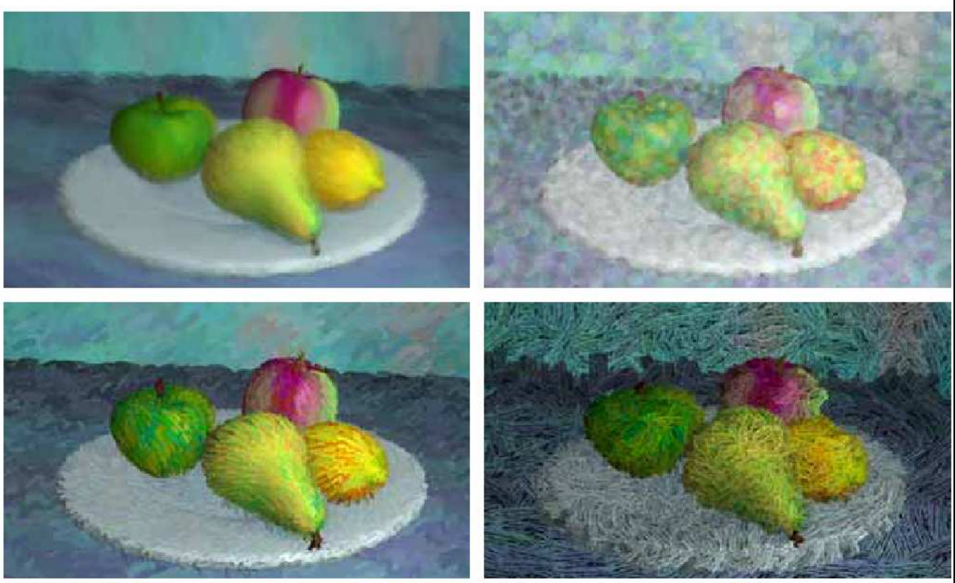
Meier, Siggraph 96

Problem: produce animations in a “painterly” style with temporal coherence of strokes

Method:

- Populate surfaces with stroke “particles”
- Render with the help of reference images


Geometry
Particle Placer
Particles in World Space
Reference Pictures
Camera Transform
Color
Shaders
Orientation
Painterly Renderer
Output Image
Size
Brush Image



video

Problem

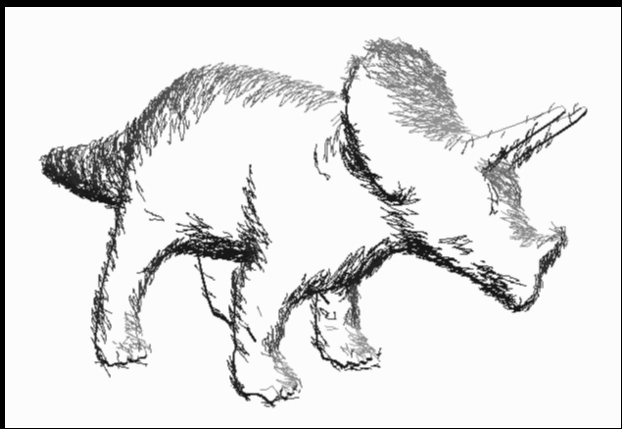
Particles have fixed distribution

- Need prescribed camera path

Talk overview

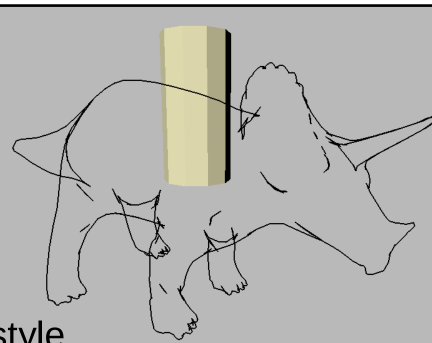
- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graftals
- WYSIWYG NPR
- Coherent stylized silhouettes

Silhouette detection



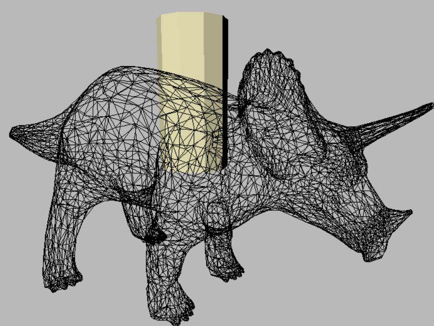
Real-Time Nonphotorealistic Rendering. Markosian, Kowalski, Trychin, Bourdev, Goldstein, & Hughes. SIGGRAPH 1997.

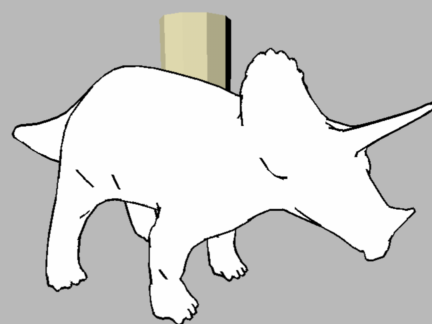
Applications



Visualization

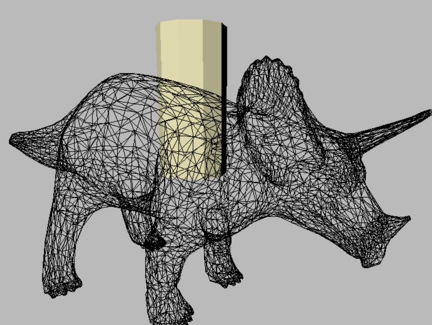
Fast, simple “line drawing” style

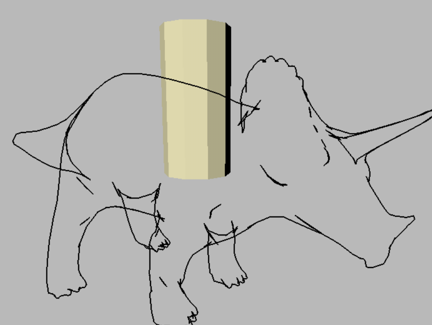




Observation: silhouette edges are

- sparse
- connected in long chains
- temporally coherent





Randomized silhouette detection

Check a fraction of edges.

- Find one, find whole chain

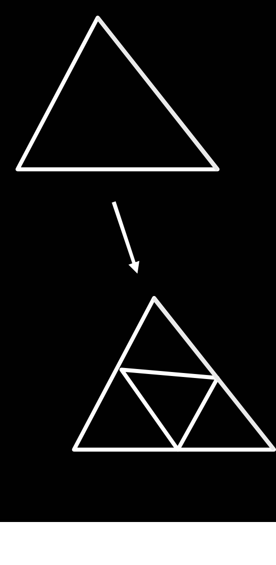
Check old silhouettes

Analysis

For fixed probability:
check $O(\sqrt{n})$ edges

Refinement scheme:

- silhouette chains "persist"
- mesh edges quadruple
- silhouette edges double



Proof:

n: number of edges in mesh
s: edges in given silhouette chain
c: number of edges to check

note $s = \beta\sqrt{n}$
take $c = \alpha\sqrt{n}$

P(miss the chain) $= \left(\frac{n-s}{n}\right)^c$

$$= \left(1 - \frac{s}{n}\right)^c = \left(1 - \frac{\beta\sqrt{n}}{n}\right)^{\alpha\sqrt{n}} = \left(1 - \frac{\beta}{\sqrt{n}}\right)^{\alpha\sqrt{n}} < e^{-\alpha\beta}$$

P(hit the chain) $> 1 - e^{-\alpha\beta}$

Example

Suppose at coarsest level mesh has 128 edges, and we want to detect a chain of 8 edges w/ probability *p* = 0.95

Then $\beta \approx 0.707$

We must take $\alpha = -\log(1 - p)/\beta \approx 4.24$

Deterministic schemes

Hierarchical methods:
pre-computed spatial data structure

Illustrating Smooth Surfaces.
Hertzmann & Zorin. SIGGRAPH 2000.

Silhouette Clipping.
Sander, Gu, Gortler, Hoppe, & Snyder. SIGGRAPH 2000.

Comparison

Randomized:

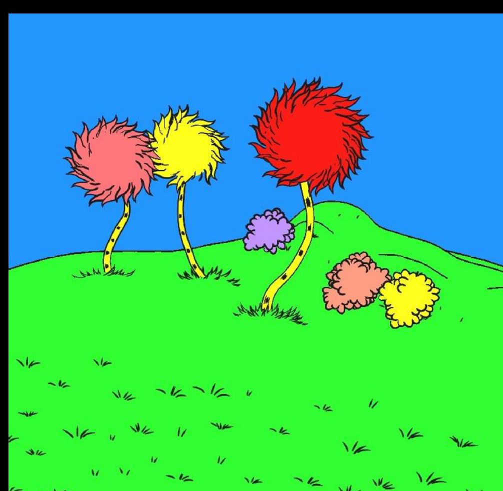
- Simple
- Effective
- Small silhouettes come in late

Deterministic:

- Requires pre-process
- Not for animated models

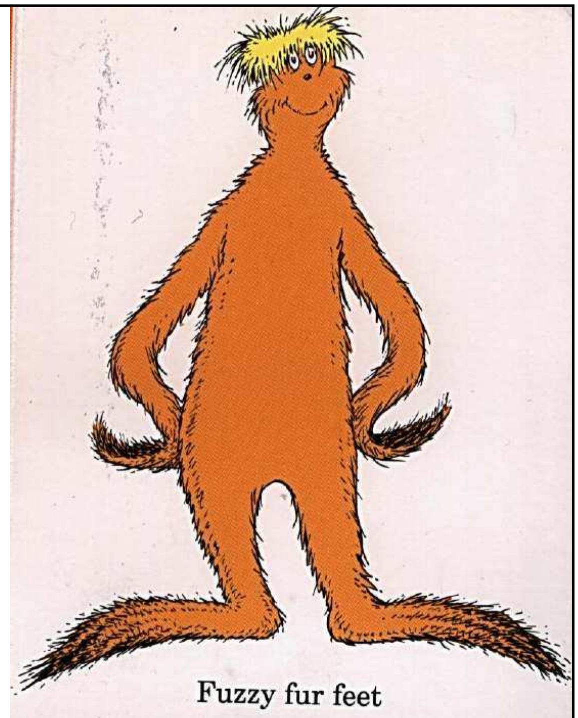
Talk overview

- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graftals
- WYSIWYG NPR
- Coherent stylized silhouettes



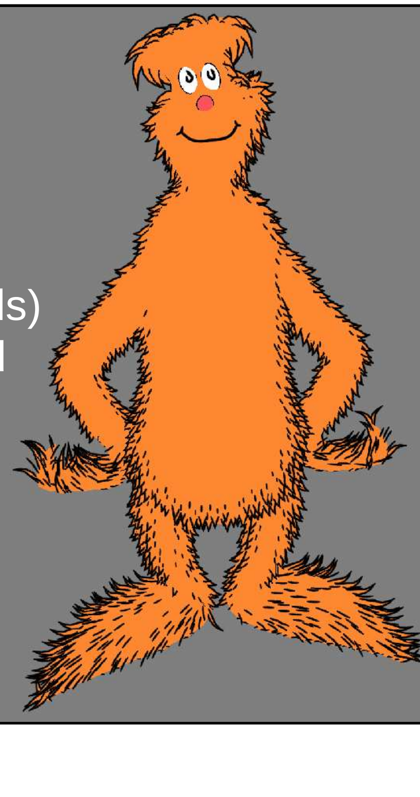
Art-based Rendering of Fur, Grass and Trees. Kowalski, Markosian, Northrup, Bourdev, Barzel, Holden & Hughes. SIGGRAPH 1999.

Dr. Seuss



Graftal textures

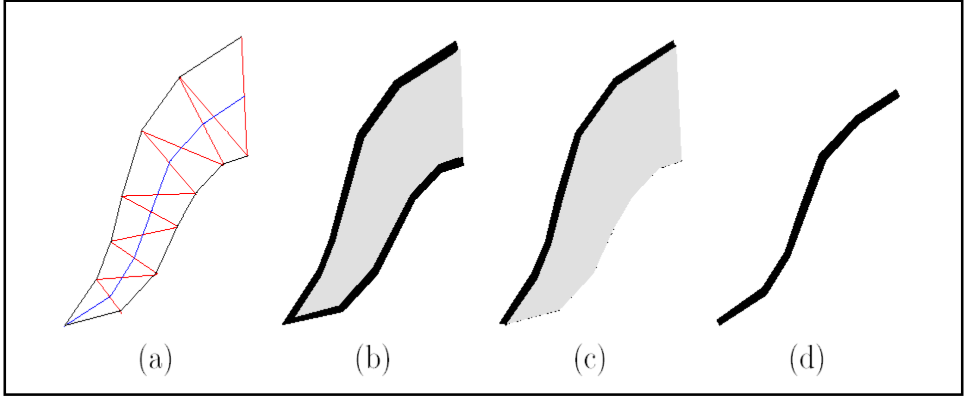
Detail elements (graftals) generated as needed



Graftals

Oriented in local frame

Can choose level of detail



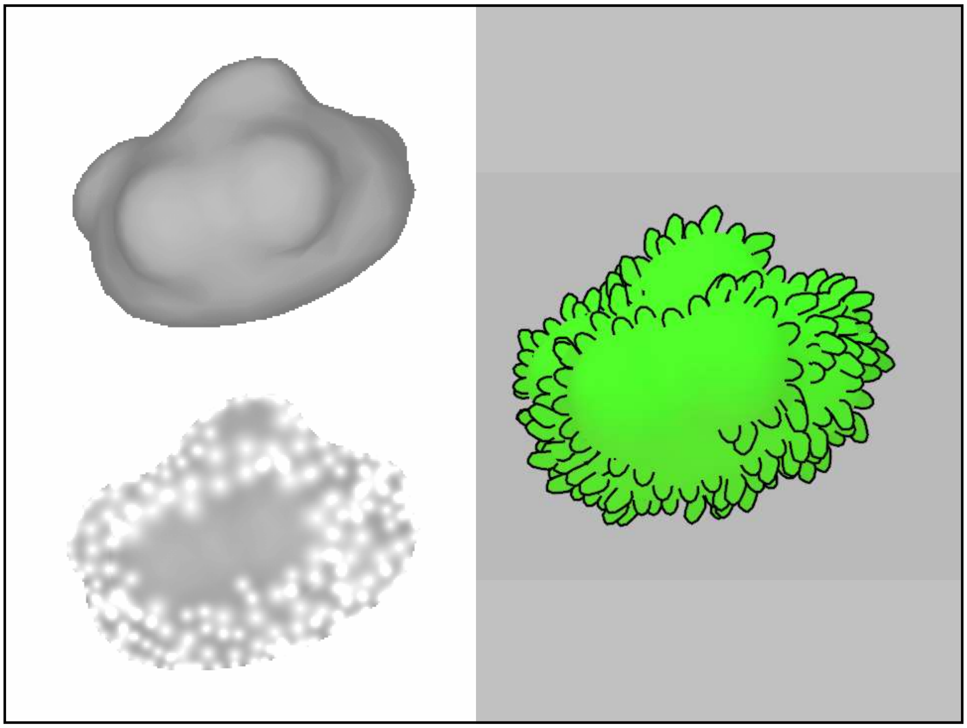
Needed for placement of graftals:

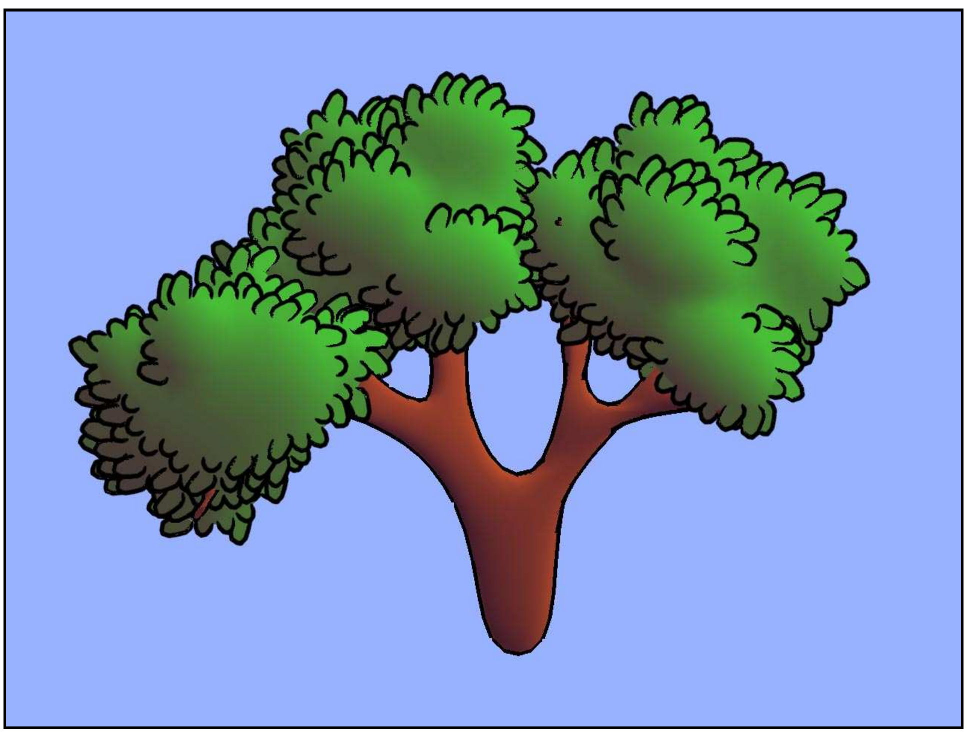
Controlled *screen-space* density

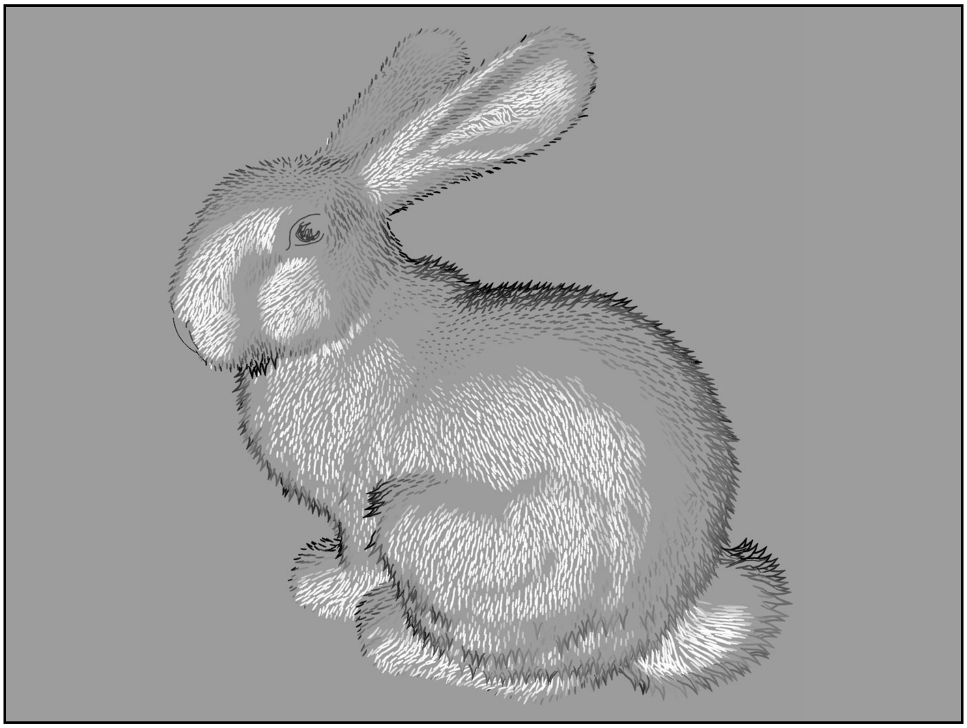
Placement on surfaces

Controlled placement (e.g. at silhouettes)

Persistence of graftals







Problems

Graftal textures defined in code

- hard to edit
- how to extend with UI?

Coherence

- Graftals popping in/out
- Better at low frame rates!

Art-based Rendering w/ Continuous Levels of Detail. Markosian, Meier, Kowalski, Holden, Northrup, & Hughes. NPAR 2000.

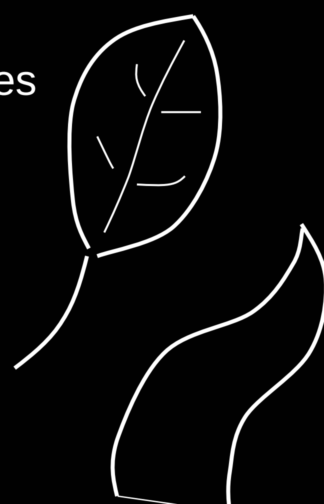
Basic graftals

Collection of drawing primitives
- triangle strips / fans

Canonical vertices

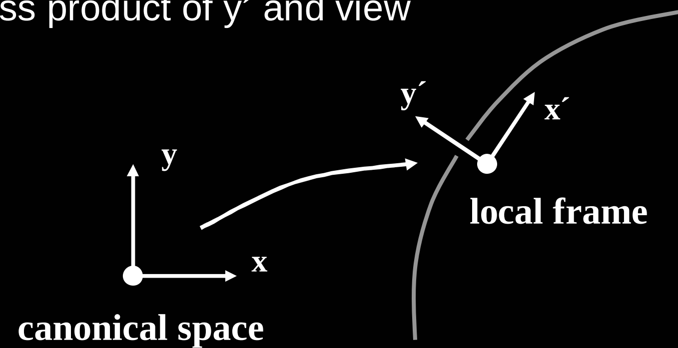
Local coordinate frame

Tuft: hierarchy of graftals



The local frame

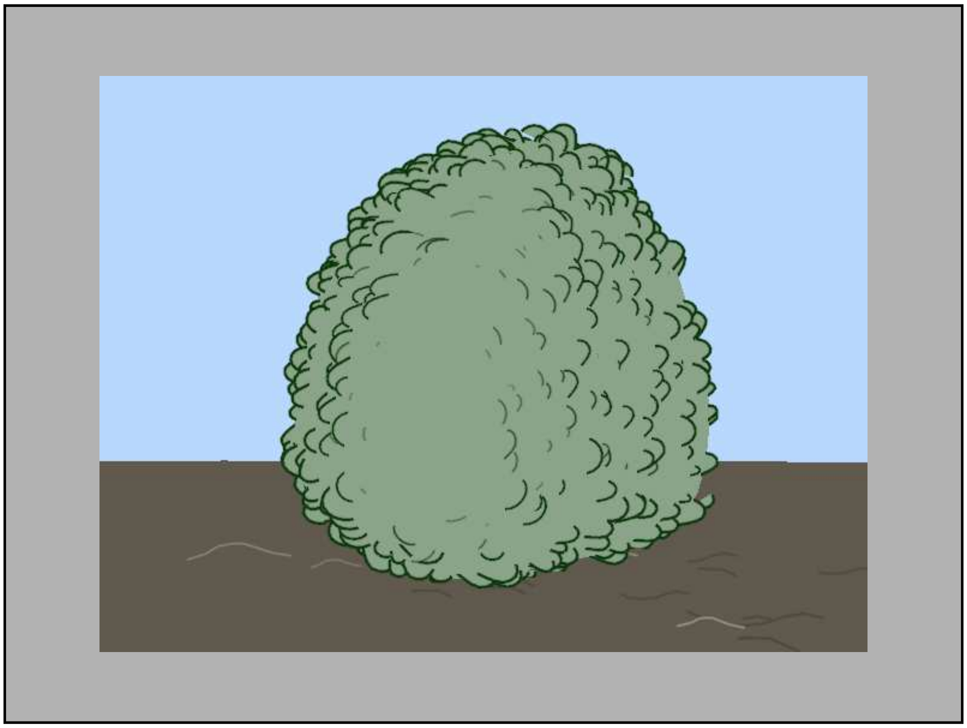
Base position (e.g. on surface)
y´ (e.g. surface normal)
x´ (e.g. cross product of y´ and view vector)



Placement and duplication

Designer creates a few “example graftals”
Duplicates generated on surfaces
- explicitly
- procedurally

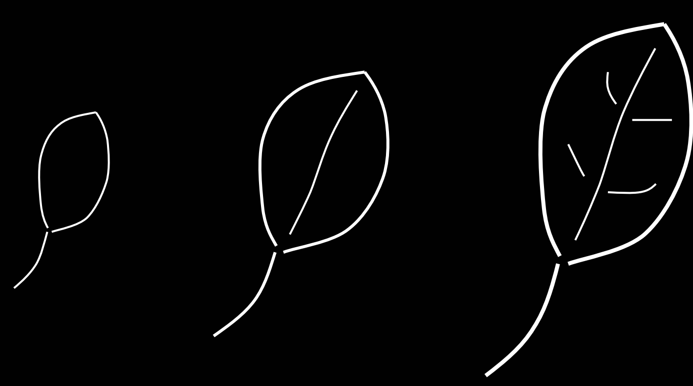
Random variation



Level of detail (LOD)

Graftal computes current LOD

Draws primitives that exceed threshold

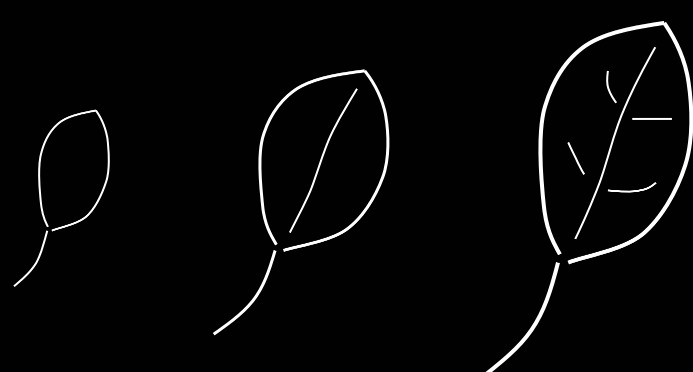


Computing LOD

LOD derived from:

- apparent size
- orientation
- elapsed time

σ: ratio of current size to "rest" size

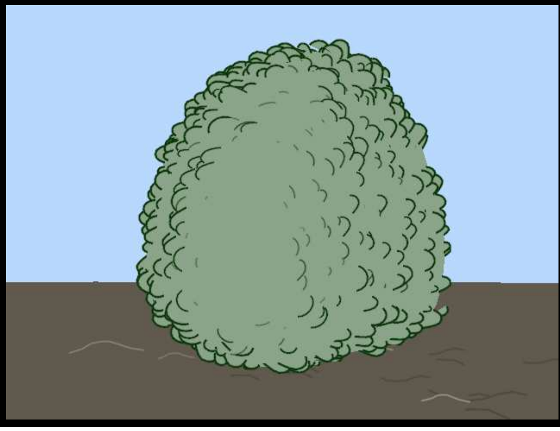


$\sigma = .7$ $\sigma = 1$ $\sigma = 1.4$

Orientation

Value used to selectively suppress LOD

E.g.: 1 - |v · n|



Movie

Discussion

Coherence: much better!

Slower

Introducing / removing elements

- Fading & thinning work well
- Growing looks creepy

LOD mechanism too inflexible

Need direct UI

Talk overview

Technical illustration

Pen & ink

Painterly rendering

Silhouette detection

Graftals

WYSIWYG NPR

Coherent stylized silhouettes

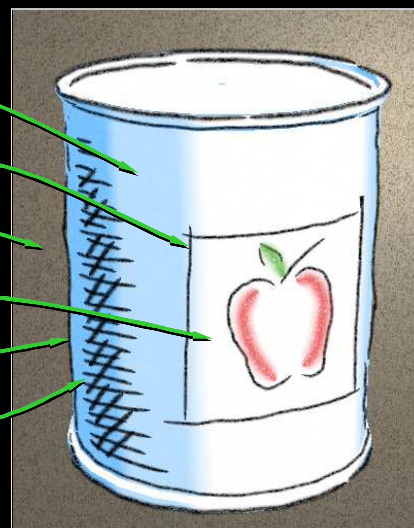
WYSIWYG NPR: Drawing Strokes Directly on 3D Models. Kalnins, Markosian, Meier, Kowalski, Lee, Davidson, Webb, Hughes & Finkelstein. SIGGRAPH 2002.

Contributions

- Direct user-control for NPR
- Better silhouettes
- New media simulation
- Stroke synthesis by example
- Hatching with LODs

Overview of Components

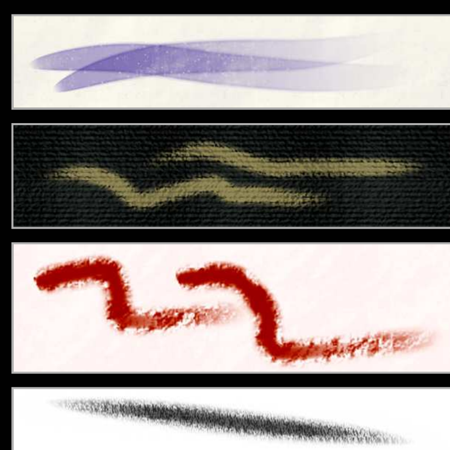
- Base Coat
- Brush Style
- Paper Effect
- Decals
- Outlines
- Hatching



Brush Style

Per stroke:

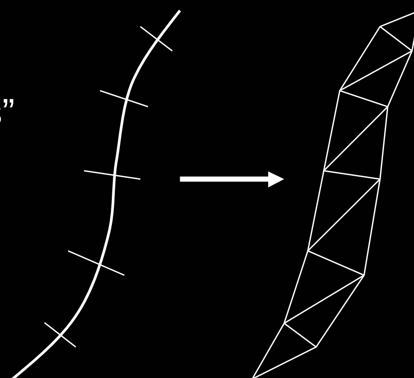
- Color
- Width
- Paper effect



Rendered as triangle strips.

Strokes in OpenGL

Based on "Skeletal strokes"
Hsu *et al.*, UIST '93



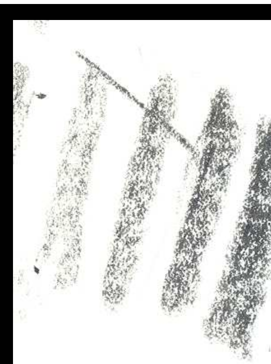
Paper Effect

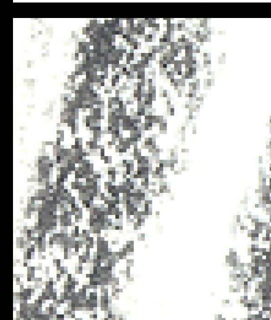
Height field texture:

- Peaks catch pigment
- Valleys resist pigment

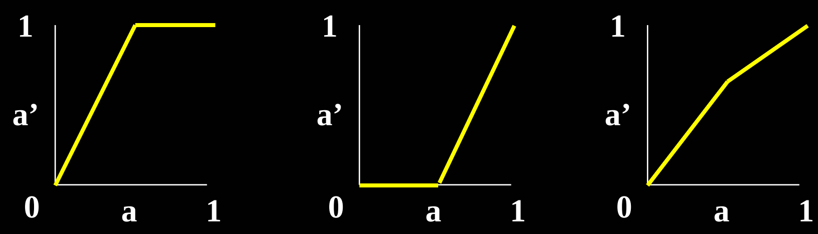
Implementation:

- Pixel shader

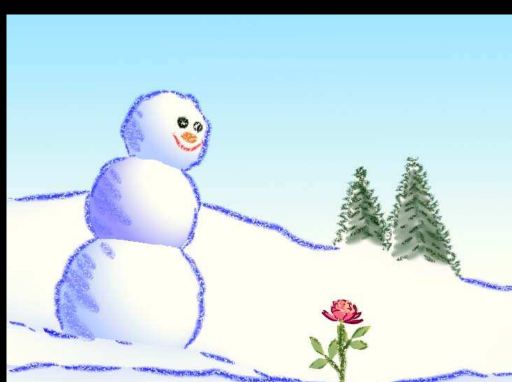


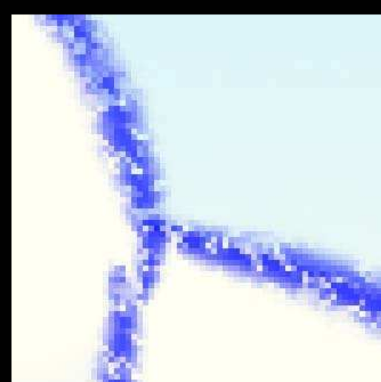


Re-map alpha with a
“paper texture” heightfield



peak valley intermediate





Hatching: LOD

video

Discussion

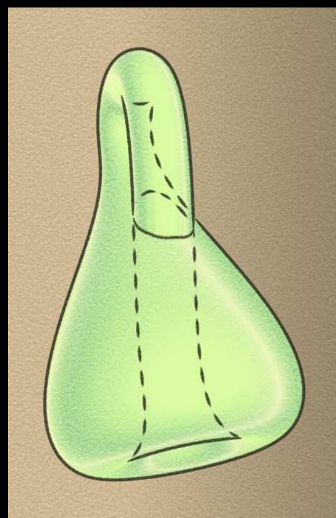
Huge benefit from user-control
Wide range of effects
Interactive rates

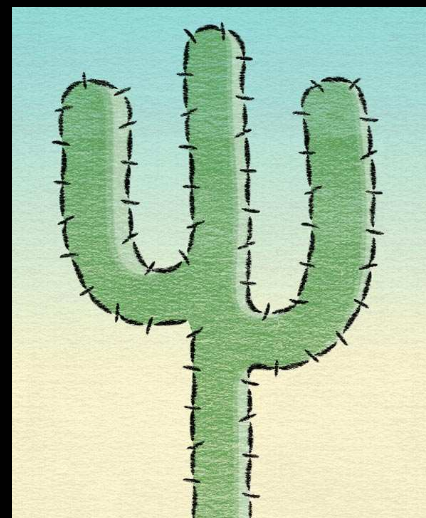
Future work

- Stroke patterns / synthesis
- Stroke behavior
- Graftals / LOD
- Silhouette coherence

Talk overview

Technical illustration
Pen & ink
Painterly rendering
Silhouette detection
Graftals
WYSIWYG NPR
Coherent stylized silhouettes





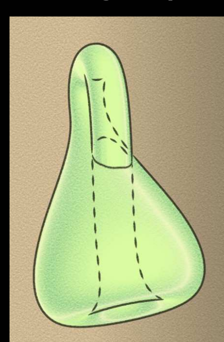
Coherent Stylized Silhouettes.
Kalnins, Davidson, Markosian & Finkelstein. SIGGRAPH 2003.

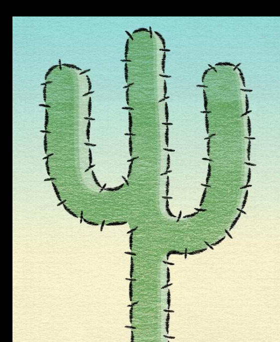
Goals

Coherence for stylized silhouettes

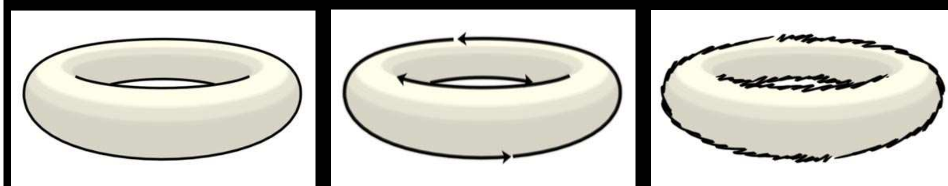
Balance competing objectives:

- Coherence in 3D
- 2D arc-length parameterization



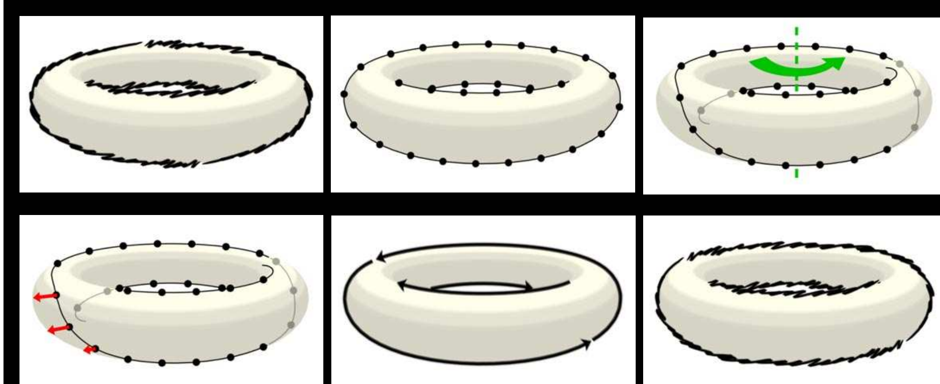


Terms



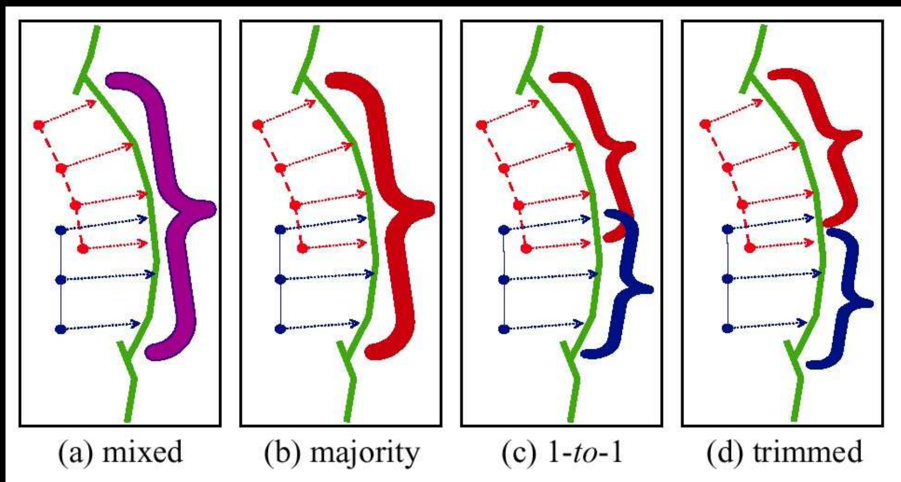
silhouette paths   brush paths   strokes

Overview of process



Each sample records parameter value

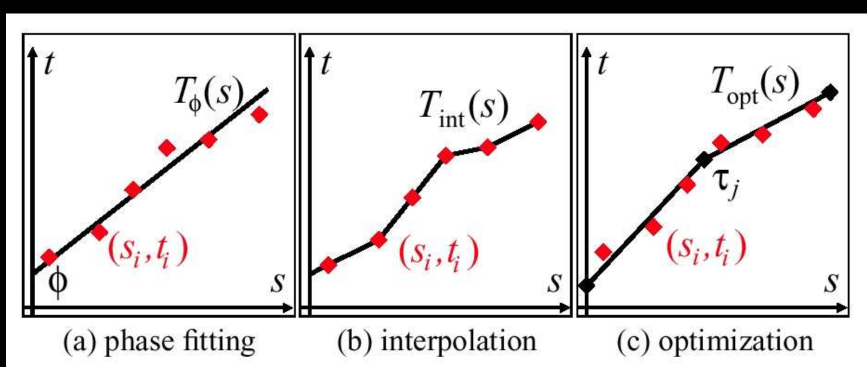
Brush path generation



Brush path parameterization

s: arc-length parameter along brush path
t: stylization parameter

E.g.: t = ks (k is "stretch factor")



Optimization

Minimize "energy" that measures

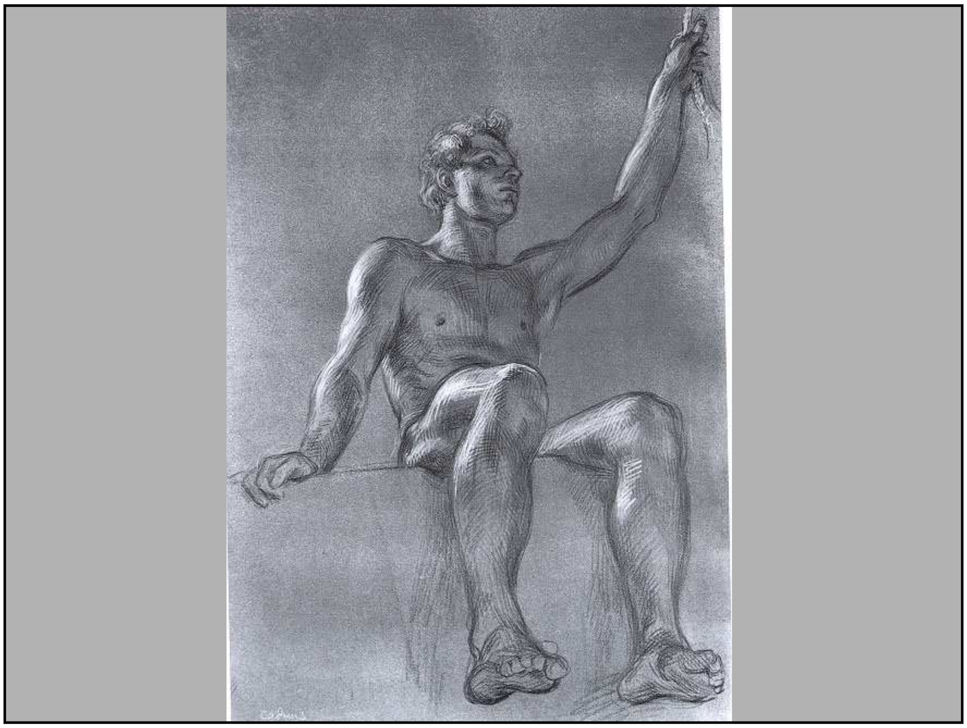
- Deviation from votes
- Deviation from scaled arc-length
- Bending

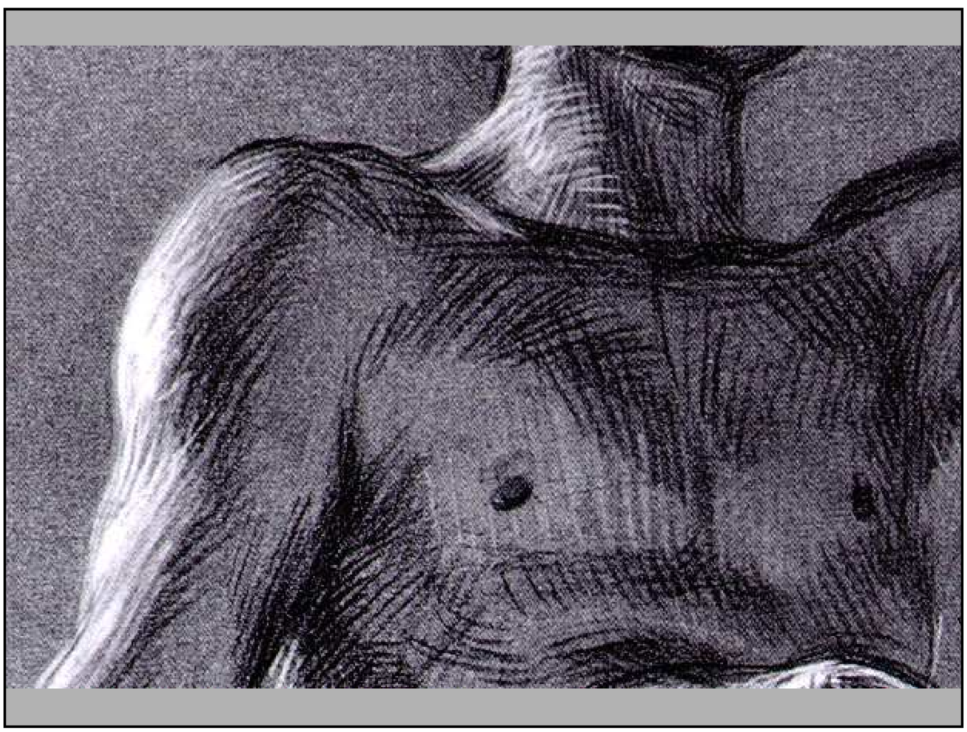
Energy terms are quadratic

- Differentiate, solve system of equations

$$E_{votes} = \frac{1}{n} \sum_{i=1}^{n} [T_{opt}(s_i) - t_i]^2 \qquad (2)$$

video





$$E = E_{votes} + \omega_\rho E_\rho + \omega_b E_{bend} + \omega_h E_{heal} \quad (1)$$

$$E_{votes} = \frac{1}{n}\sum_{i=1}^{n}[T_{opt}(s_i) - t_i]^2 \quad (2)$$

$$E_\rho = \frac{1}{m}\sum_{j=1}^{m}[\hat{\tau}_{ave} - \hat{\tau}_j]^2 \quad (3)$$

$$E_{heal} = \sum_{k}[T_{opt}(s_k) - t_{ave}]^2 \quad (5)$$

$$E_{bend} = \frac{1}{m}\sum_{j=1}^{m-2}[\tau_{j+2} - 2\tau_{j+1} + \tau_j]^2 \quad (4)$$