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!
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 (e.g., paper, canvas, stone, etc.)
- Direct user control (e.g., let artist’s use it)
- Picture element: pixel or stroke?

NPR History

Technical Illustration [Saito 90]
Pen & Ink [Winkenbach 94]
Watercolor [Curtis 97]
Paint [Hertzmann 96]

Painterly rendering for 3D models [Meier 96]
Painterly rendering for video [Litvinowicz 97]
NPR History


In more detail...

- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graffals
- 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

Problem

- Parameters need careful (manual) tuning for good results
In more detail...

- 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
Problems

- Only produces still images
  - Would not provide temporal coherence
- What’s the application?

In more detail...

- Technical illustration
- Pen & ink
- Painterly rendering
- Silhouette detection
- Graffals
- WYSIWYG NPR
- Coherent styled 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
Problem

- Particles have fixed distribution
  - Need prescribed camera path

In more detail...

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

Silhouette detection

Applications
- Visualization
- Fast, simple “line drawing” style

Observation
- Silhouette edges are
  - sparse
  - connected in long chains
  - temporally coherent

Randomized silhouette detection
- Algorithm
  - Check a fraction of edges
  - Find one, then whole chain
  - Check old silhouettes
  - Loop back
- Characteristics
  - Simple
  - Effective
  - Small silhouettes come in late

Analysis
- $n$: number of edges in mesh
- $s$: edges in given silhouette chain
- $c$: number of edges to check
- $P(\text{miss the chain}) = \left(\frac{n-s}{n}\right)^c$
- $P(\text{hit the chain}) > 1 - c^{-\alpha}$

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 $\alpha = -\log(1 - p)$ = 4.24
In more detail...

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


Dr. Seuss

Graftal Textures

- Detail elements (graftals) generated as needed

Scene without graftals
Scene with graftals

Problems
- Where do you put graftals?
  - Placement
  - Screen-space density
- Graftal textures defined in code
  - Hard to edit
  - How to extend with UI?
- Coherence
  - Graftals popping in/out
  - (Better at low frame rates!)

Placement and duplication
- Designer creates a few “example graftals”
- Duplicates generated on surfaces
  - Explicitly
  - Procedurally
- Random variation

Local frame
- Base position (e.g., on surface)
- y’ (e.g., surface normal)
- x’ (e.g., cross product of y’ and view vector)

Levels of detail
- Graftal computes current LOD
- Draws primitives that exceed threshold
Levels of detail

- Graftal computes current LOD
- Draws primitives that exceed threshold

Levels of detail

- LOD derived from:
  - apparent size
  - orientation
  - elapsed time
Orientation

- Value used to selectively suppress LOD
- E.g.: $1 - |v \cdot n|$
**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

**Paper Effect**

- Re-map alpha with a “paper texture” height-field

- Peak
- Valley
- Intermediate

**Hatching: LOD**

- Examples of hatching with varying detail levels.
WYSIWYG NPR

- Movie

Discussion

- Huge benefit from user-control
- Wide range of effects
- Interactive rates
- Future work
  - Stroke patterns / synthesis
  - Stroke behavior
  - Graftals / LOD
  - Silhouette coherence

In more details...

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

Goals

- Coherence for stylized silhouettes
- Balance competing objectives:
  - Coherence in 3D
  - 2D arc-length parameterization

Terms

silhouette paths  brush paths  strokes

Coherent Stylized Silhouettes
Overview of process

Each sample records parameter value

Brush path generation

(a) mixed  (b) majority  (c) 1-to-1  (d) trimmed

Brush path parameterization

- \(s\): arc-length parameter along brush path
- \(t\): stylization parameter
- E.g.: \(t = k s\) (\(k\) 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

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

Coherent Stylized Silhouettes

- [Movie]
\[ E = E_{\text{rootes}} + \omega_p E_p + \omega_h E_{\text{head}} + \omega_i E_{\text{head}} \]  

(1)

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

(2)

\[ E_p = \frac{1}{m} \sum_{j=1}^{m} [i_{\text{ave}} - i_j]^2 \]  

(3)

\[ E_{\text{head}} = \sum_{k} [T_{\text{opt}}(s_k) - t_{\text{ave}}]^2 \]  

(5)

\[ E_{\text{head}} = \frac{1}{m} \sum_{j=1}^{w-2} [r_{j+2} - 2r_{j+1} + r_j]^2 \]  

(4)