“Virtual Hairy Brush for Painterly Rendering”

-Cale Scholl
Motivation

- Okami is a PS2 game with amazing artistic visuals
- Not only cell-shaded, but made to resemble an ink painting
- First and only of its kind so far
Goals

- Design art program that simulates ink painting (sumi-e)
- Create artwork similar to that seen in PS2 game Okami
Example 1
Example 2
Implementation Theory

- Drawing surface is copied into framebuffer every frame
- Only the brush is rendered
Example 3
Implementation Theory

- Some contours are texture-mapped brush strokes
- Too difficult to render brush-stroke contours in real time
Modeling Breakdown

- Brush Tip
- Ink
- Paper
Brush Tip

- Quality Parameters
- Writing Primitives
- Parametric Model
Quality Parameters (Shape)

- e – degree of elasticity
- tre – threshold for splitting
- v – relocation factor
- re – rotation coefficient
- ie – elongation coefficient
Quality Parameters (Texture)

- ren – color rendering control factor
- n – ink diffusion distance factor
Writing Primitives

- Fundamental unit of virtual brush is writing primitive
- Writing primitive can split dynamically into several writing primitives
- Writing primitive described by NURBS surface
Parametric Model

- the bottom control circle
- the middle control ellipse
- the middle control axis
- the tip control line
Bottom Control Circle C

- C := (cen, r, cori)
- cen – coordinates of C’s center
- r – C’s radius
- cori – C’s orientation (unit vector)
Tip Control Line $L[i]$

- $L[i] := (\text{len}[i], \text{mid}[i], \text{lori}[i])$
- $\text{len}[i]$ – length of $L[i]$
- $\text{mid}[i]$ – $L[i]$’s midpoint
- $\text{lori}[i]$ – $L[i]$’s orientation (unit vector)
Middle Control Axis A[i]

- Cubic B-spline with interpolated key points P[i][1], P[i][2], ...
- cc[i][j] – coords of P[i][j]
- col[i][j] – RGB color of P[i][j]
- wet[i][j] – P[i][j]’s degree of wetness
- wv[i][j] – wetness changing vector
- cv[i][j] – color changing vector
- wr[i][j] – wetness changing factor
- cr[i][j] – color changing factor
Middle Control Ellipse $E[i]$

- $E[i] := (a[i], b[i], loc[i], eori[i])$
- $a[i]$ – length of $E[i]$’s major axis
- $b[i]$ – length of $E[i]$’s minor axis
- $loc[i]$ – $E[i]$’s relative position along middle control axis $A[i]$
- $eori[i]$ – $E[i]$’s minor axis’s orientation
Modeling Breakdown

- Brush Tip
- Ink
- Paper
Ink

Fig. 6. Radiation (left) and vector (right) ink distribution modes.
\[
\begin{align*}
\delta_v \text{wet}_Q &= \text{wet}_{P_{i,j}} \times ((\text{cc}_Q - \text{cc}_{P_{i,j}}) \cdot \text{wv}_{i,j}), \\
\delta_v \text{col}_Q &= \text{col}_{P_{i,j}} \times ((\text{cc}_Q - \text{cc}_{P_{i,j}}) \cdot \text{cv}_{i,j}),
\end{align*}
\]
\[
\begin{align*}
\delta_r \text{wet}_Q &= \text{wet}_{P_{i,j}} \times ||cc_Q - cc_{P_{i,j}}|| \times wr_{i,j}, \\
\delta_r \text{col}_Q &= \text{col}_{P_{i,j}} \times ||cc_Q - cc_{P_{i,j}}|| \times cr_{i,j}.
\end{align*}
\]
Ink Texture Function

\[
\begin{align*}
\text{wet}_Q & \triangleq \text{wet}_{P_{i,j}} + \delta_r \text{wet}_Q + \delta_v \text{wet}_Q, \\
\text{col}_Q & \triangleq \text{col}_{P_{i,j}} + \delta_r \text{col}_Q + \delta_v \text{col}_Q,
\end{align*}
\]
Modeling Breakdown

- Brush Tip
- Ink
- Paper
Paper Quality Parameters

- sm – degree of smoothness
- ab – absorbing ability
- pw – diffusion control factor
- dry – drying factor
Drawing Methods

- Dipping State
- 6-DOF Input
- Dynamic Adjustments
- Writing Process
Dipping State

\[
\begin{align*}
\text{col}_{i,l} &= \text{ink color}, \\
(l &= k, k + 1, \ldots, n_i), \\
\text{col}_{i,l} &= \frac{\|\text{ce}_{i,k} - \text{ce}_{i,l}\| \times \text{col}_{i,1} + \|\text{ce}_{i,l} - \text{ce}_{i,1}\| \times \text{col}_{i,k}}{\|\text{ce}_{i,k} - \text{ce}_{i,1}\|}, \\
(l &= 2, 3, \ldots, k - 1), \\
\text{col}_{i,1} &= 0.
\end{align*}
\]

\[
\begin{align*}
\text{wet}_{i,l} &= \text{the degree of wetness}, \\
(l &= k, k + 1, \ldots, n_i), \\
\text{wet}_{i,l} &= \frac{\|\text{ce}_{i,k} - \text{ce}_{i,l}\| \times \text{wet}_{i,1} + \|\text{ce}_{i,l} - \text{ce}_{i,1}\| \times \text{wet}_{i,k}}{\|\text{ce}_{i,k} - \text{ce}_{i,1}\|}, \\
(l &= 2, 3, \ldots, k - 1), \\
\text{wet}_{i,1} &= 0.
\end{align*}
\]
Drawing Methods

- Dipping State
- 6-DOF Input
- Dynamic Adjustments
- Writing Process
6-DOF Input

Fig. 11. The six degrees of freedom of an input sample.
<table>
<thead>
<tr>
<th>User Action</th>
<th>Corresponding Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moving the mouse</td>
<td>Generate the ((x^t, y^t)) coordinates</td>
</tr>
<tr>
<td>Scrolling the middle wheel of the mouse</td>
<td>Generates the (z^t) coordinate</td>
</tr>
<tr>
<td>Pressing key “q” by the little finger</td>
<td>Reduces the value of (d^t)</td>
</tr>
<tr>
<td>Pressing key “w” by the ring finger</td>
<td>Increases the value of (d^t)</td>
</tr>
<tr>
<td>Pressing key “e” by the middle finger</td>
<td>Reduces the value of (q^t)</td>
</tr>
<tr>
<td>Pressing key “r” by the index finger</td>
<td>Increases the value of (q^t)</td>
</tr>
<tr>
<td>Pressing key “c” by the thumb</td>
<td>Reduces the value of (r^t)</td>
</tr>
<tr>
<td>Pressing key “v” by the thumb</td>
<td>Increases the value of (r^t)</td>
</tr>
<tr>
<td>Pressing a key while the left button is pressed</td>
<td>The key is pressed twice</td>
</tr>
<tr>
<td>Pressing a key while the right button is pressed</td>
<td>The key is pressed four times</td>
</tr>
</tbody>
</table>

Fig. 12. The 3-button-mouse and keyboard hybrid input method.
Virtual Position

\[ S^t = \text{wei}_{\text{sam}} \times \text{Sam}^t + (1 - \text{wei}_{\text{sam}}) \times M^t. \]  \hspace{2cm} (9)

\( M^t \) comes from HB’s displacements in its last few sampling intervals \( S^{t-1}, S^{t-2}, \ldots, S^{t-n} \):

\[ M^t = \text{Vel}^t \times dT + S^{t-1}, \]  \hspace{2cm} (10)

\[ \text{Vel}^t = \frac{\sum_{k=1}^{el} \left( \text{wei}_k \times \frac{s^{t-k} - s^{t-k-1}}{dT} \right)}{\sum_{k=1}^{el} \text{wei}_k}. \]  \hspace{2cm} (11)

\( el = 4; \, \text{wei}_1 = 4, \, \text{wei}_2 = 2, \, \text{wei}_3 = 1, \, \text{wei}_4 = 1; \, \text{and} \, \text{wei}_{\text{sam}} = 0.6. \)
Drawing Methods

- Dipping State
- 6-DOF Input
- Dynamic Adjustments
- Writing Process
Dynamic Adjustments

- Inner Stress
- Pressure on Paper
- Current Active Point
- Deformation of Middle Control Axis
- Recovery of Middle Control Axis
- Wetness adjustment
- Adjustment of Middle Control Ellipse
- Adjustment of Tip Control Line
- Splitting of Virtual Hairy Brush
- Ink Flowage between Writing Primitives
Inner Stress

\[
\begin{align*}
\text{his}^t_i & \triangleq \text{hsd}^t_i + \text{hdd}^t_i, \\
\text{hsd}^t_i & \triangleq \frac{a^t_i}{b^t_i} \times \text{len}^t_i, \\
\text{hdd}^t_i & \triangleq \frac{\sum_{j=2}^{n_i} (\| \text{cc}^{0}_{i,j} - \text{cc}^t_{i,j} - \text{mc}^t \| \times \| \text{cc}^t_{i,j} - \text{cc}^t_{i,j-1} \|)}{\sum_{j=2}^{n_i} \| \text{cc}^t_{i,j} - \text{cc}^t_{i,j-1} \|}, \\
\text{mc}^t & \triangleq \text{cc}^{0}_{i,1} - \text{cc}^t_{i,1}.
\end{align*}
\]

\[
\text{str}^t_i \triangleq (1 - e) \times \frac{n_i}{\sum_{k=1}^{n_i} \text{wet}^t_{i,k}} \times \frac{S^t_i \times \text{hei}^t_i}{3} \times \text{his}^t_i \times \frac{\pi \times a^t_i \times b^t_i}{4}.
\]
Pressure on Paper

\gamma_i^t \triangleq \| Vel^t \| \times str_i^t \times sm \times e,
\[
\begin{align*}
\text{col}_{i,j}^t &= \frac{\text{col}_{i,j-1}^t \times \|\text{cc}_{i,j}^t - \text{cc}_{i,j-1}^t\| + \text{col}_{i,j+1}^t \times \|\text{cc}_{i,j}^t - \text{cc}_{i,j-1}^t\|}{\|\text{cc}_{i,j+1}^t - \text{cc}_{i,j-1}^t\|}, \\
\text{wet}_{i,j}^t &= \frac{\text{wet}_{i,j-1}^t \times \|\text{cc}_{i,j}^t - \text{cc}_{i,j-1}^t\| + \text{wet}_{i,j+1}^t \times \|\text{cc}_{i,j}^t - \text{cc}_{i,j-1}^t\|}{\|\text{cc}_{i,j+1}^t - \text{cc}_{i,j-1}^t\|}.
\end{align*}
\]
Deform of Middle Control Axis

\[ dis = e \times \gamma_i^t. \]

NOT IMPLEMENTED
Recov of Middle Control Axis

- Additional vertical displacement

\[ \gamma_i^t \times |sz^{t+1} - sz^t| \times tr(sz^{t+1} - sz^t), \]

NOT IMPLEMENTED
Wetness adjustment

\[ wet_{i,j \pm s}^{t+1} = wet_{i,j \pm s}^t - ab \times \gamma_i^t \times \frac{1}{2^{s+d}}, \]

where \( d = \begin{cases} 
2(s \in \mathbb{N}) \\
1(s = 0) 
\end{cases} \) and \((j \pm s) \in [1, n_i]\).
\[
\begin{align*}
\text{rot}_i^t &= re \times \gamma_i^t \times (\text{Vel}_i^t \cdot \text{eori}_i^t), \\
\text{inc}_i^t &= ie \times \gamma_i^t \times \|\text{Vel}_i^t \times \text{eori}_i^t\|.
\end{align*}
\]

\[
l_0c_{i}^{t+1} = v \times l_0c_{i}^{t} + (1 - v) \times \frac{\sum_{j=1}^{n_i-1} \frac{\sum_{k=1}^{j} \|cc_{i,k+1}^{t+1} - cc_{i,k}^{t+1}\|}{\sum_{k=1}^{n_i-1} \|cc_{i,k+1}^{t+1} - cc_{i,k}^{t+1}\|}}{n_i}.
\]

NOT IMPLEMENTED
Adjustment of Tip Control Line

- $L[i].\text{len}$ increases by $\text{inc}_i^t$
- $L[i].\text{lori}$ rotates by $\text{rot}_i^t$

- Tip control line defines shape of end of stroke when brush leaves paper

NOT IMPLEMENTED
Splitting of Virtual Hairy Brush

$$k = \left[ \frac{str^t_i}{tre} \right]$$

$$\begin{align*}
  a^t_{i,j} &= \frac{a^t_i}{\sqrt{k}} \\
  b^t_{i,j} &= \frac{b^t_i}{\sqrt{k}} \\
  len^t_{i,j} &= \frac{len^t_i}{k}
\end{align*}$$

($j = 1, 2, \ldots, k$).

For each $P_{i,l}(l = 1, 2, \ldots, n_i)$ in $WP_i$, there is a corresponding key point $P^j_{i,l}$ in $WP^j_i(j = 1, 2, \ldots, k)$. The coordinates of $P_{i,l}$ and $P^j_{i,l}$ have the following relationship:

$$ce_{P^j_{i,l}} = ce_{P_{i,l}} + S_{P^j_{i,l}}$$

where $S_{P^j_{i,l}}$ is determined by

$$S_{P^j_{i,l}} = \text{rand} \times (str^t_i - tre) \times tr(str^t_i - tre),$$

and $\text{rand}$ is a random unit vector in the 3D space.

NOT IMPLEMENTED
Ink Flowage between WPs

\[ \text{I}_{i,k}^{t+1} = \text{I}_{i,k}^t \times (1 - p_w) + \frac{\sum_{j=1, j \neq i}^m \sum_{l=1}^{n_j} (\text{I}_{j,l}^t \times \text{tr}(\eta - \|\text{cc}_{j,l}^t - \text{cc}_{i,k}^t\|))}{\sum_{j=1, j \neq i}^m \sum_{l=1}^{n_j} \text{tr}(\eta - \|\text{cc}_{j,l}^t - \text{cc}_{i,k}^t\|)} \times p_w \quad (i = 1, 2, \ldots, m; \ k = 1, 2, \ldots, n_i). \]
Drawing Methods

- Dipping State
- 6-DOF Input
- Dynamic Adjustments
- Writing Process
The Writing Process

- Determine cross section of each WP
  
  NOT IMPLEMENTED

- Each current active point is linear interpolation of 2 closest key points

\[
\begin{align*}
\text{col}_{u,v} & = \frac{\text{col}_{1} - \text{col}_{2} \times \| \text{cc}_{1} - \text{cc}_{2} \|}{\| \text{cc}_{1} - \text{cc}_{2} \|}, \\
\text{wet}_{u,v} & = \frac{\text{wet}_{1} - \text{wet}_{2} \times \| \text{cc}_{1} - \text{cc}_{2} \|}{\| \text{cc}_{1} - \text{cc}_{2} \|}.
\end{align*}
\]
Alternative Implementation

- Cross section currently represented by fixed 21-pixel grid
- All DOF except x,y are fixed
- Current active point is fixed
- Brush’s parametric deformation is fixed
The Writing Process (2)

- Apply texture function to pixels on ink mark boundary

\[
\begin{align*}
\text{wet}_{u,v} &= \text{wet}_{u,v} \times (1 + \| \text{cc}_{u,v} - \text{cc}_{u,v} \| \times \text{wr}_{u,v} + (\text{cc}_{u,v} - \text{cc}_{u,v} \cdot \text{wv}_{u,v}), \\
\text{col}_{u,v} &= \text{col}_{u,v} \times (1 + \| \text{cc}_{u,v} - \text{cc}_{u,v} \| \times \text{cr}_{u,v} + (\text{cc}_{u,v} - \text{cc}_{u,v}) \cdot \text{cv}_{u,v}).
\end{align*}
\]

- Inner pixels are linearly interpolated

\[
\begin{align*}
\text{col}_{\tau} &= \frac{\text{col}_{u,v} \times \| \text{cc}_{u,v} - \text{cc}_{\tau} \| + \text{col}_{u,v} \times \| \text{cc}_{\tau} - \text{cc}_{u,v} \|}{\| \text{cc}_{u,v} - \text{cc}_{u,v} \|}, \\
\text{wet}_{\tau} &= \frac{\text{wet}_{u,v} \times \| \text{cc}_{u,v} - \text{cc}_{\tau} \| + \text{wet}_{u,v} \times \| \text{cc}_{\tau} - \text{cc}_{u,v} \|}{\| \text{cc}_{u,v} - \text{cc}_{u,v} \|}.
\end{align*}
\]
The Writing Process (3)

- Wetness on paper

\[ \text{wet}_{t+1}^{u,v} = (1 - ab) \times \text{wet}_t^{u,v} + ab \times \text{wet}_t^{\tau_{u,v}} - \text{dry}. \]

- Wetness dispersion due to saturation

\[ \text{wet}_{t+1}^{i,j} = \text{wet}_t^{i,j} + \frac{(1 - \text{wet}_t^{i,j}) \times \text{tr}(1 - \text{wet}_t^{i,j})}{q_{u,v}^t} \times (\text{wet}_t^{u,v} - 1) \times \text{tr}(\text{wet}_t^{u,v} - 1), \]

\[ q_{u,v}^t = \sum_{g=u-1}^{u-1} \sum_{h=v-1}^{v+1} (1 - \text{wet}_t^{g,h}) \times \text{tr}(1 - \text{wet}_t^{g,h}), \]
The Writing Process (4)

- Rendering of current ink mark

\[
\text{col}_{\lambda_{u,v}}^{t+1} = \text{col}_{\tau_{u,v}}^{t} \times C_{r_{u,v}}^{t} + \text{col}_{\lambda_{u,v}}^{t} \times (1 - C_{r_{u,v}}^{t}),
\]

\[
\begin{cases} 
P\{C_{r_{u,v}}^{t} = 0\} = 1 - \min(\text{ren} \times ab \times \gamma_i^t, 1), \\
P\{C_{r_{u,v}}^{t} = \text{wet}_{\lambda_{u,v}}^t\} = \min(\text{ren} \times ab \times \gamma_i^t, 1),
\end{cases}
\]
Improvements

- Implementing dynamic cross section allows all other aspects (such as brush’s parametric deformation) to be implemented.
- Could store cross section as vector of x,y-pairs that are displacements from the current active point.
Improvements (2)

- When updating wetness of paper, every pixel on screen is iterated over.
- Using sparse matrix to store only wet pixels could drastically improve performance.