Improving the Parameterization of Approximate Subdivision Surfaces

Lei He\textsuperscript{1}, Charles Loop\textsuperscript{2}, Scott Schaefer\textsuperscript{1}
\textsuperscript{1}Texas A&M University
\textsuperscript{2}Microsoft Research
Subdivision Surface
Subdivision Surface
Subdivision Surface
Subdivision Surface
Motivation

Toy Story © Disney / Pixar

Geri’s Game © Pixar Animation Studios
Motivation
Motivation
Problem of Subdivision Surfaces

- Expensive to do the exact evaluation
- Hard to fit hardware tessellation
Approximate Subdivision Surfaces

- Loop, C. and Schaefer, S. 2008, *Approximate Catmull-Clark subdivision surfaces with bicubic patches*
Approximate Subdivision Surfaces

• Loop, C. and Schaefer, S. 2008,  
  *Approximate Catmull-Clark subdivision surfaces with bicubic patches*

• Ni, T., Yeo. Y.I., Myles, A, and Peters, J. 2008,  
  *GPU smoothing of quad meshes*

• Myles, A, Ni, T., and Peters, J. 2008,  
  *Fast Parallel construction of smooth surfaces from meshes with tri/quad/pent facets*

• Loop, C., Schaefer, S., Ni, T., and Castaño, I. 2009,  
  *Approximating subdivision surfaces with Gregory patches for hardware tessellation*
Where we are?

✓ Subdivision Surface

✓ Texturing Subdivision Surface

✓ Approximate Subdivision Surface

✗ Texturing Approximate Subdivision Surface
Ptex

Burley B., and Lacewell D. 2008
Ptex: Per-face texture mapping for production rendering
Subdivision surface
Gregory surface
With reparameterization
Reparameterization

Original
Reparameterization

With reparameterization
Reparameterization

\[ p(u, v) \quad g(u, v) \quad s(u, v) \quad g(s(u, v)) \]
Reparameterization

\[ p(u, v) \rightarrow g(u, v) \rightarrow s(u, v) \rightarrow g(s(u, v)) \]
Reparameterization

\[
\min_s \int_{u=0}^{1} \int_{v=0}^{1} \left| p_i(u, v) - g_i(s_i(u, v)) \right|^2
\]
Reparameterization

- Optimizing on 2D texture space instead of 3D

$$\min_{s_i} \sum_j \left| (u_j, v_j) - s_i(u_{k(j)}, v_{k(j)}) \right|^2 dp_j$$

$k(j)$ gives the index of the closest point $g(u_k, v_k)$ to $p(u_j, v_j)$. Uniform grid of size $17^2$ for $(u_j, v_j)$ and $5000^2$ for $(u_k, v_k)$. 
Reparameterization

Bicubic function
Reparameterization

Original
Reparameterization

Rational bicubic function
Reparameterization

• Optimizing on 2D texture space instead of 3D

\[
\min_{s_i} \sum_j \left| (u_j, v_j) - s_i(u_{k(j)}, v_{k(j)}) \right|^2 dp_j
\]

k(j) gives the index of the closest point g(u_k,v_k) to p(u_j,v_j). Uniform grid of size 17^2 for (u_j,v_j) and 5000^2 for (u_k,v_k).

• Geometry Independence

every single patch requires an individual optimization to obtain the fitting function s
Geometry Independence
Multiple Extraordinary Vertices

\[
\begin{pmatrix}
0,0,c_0^3 & 0,c_1^3,c_2^3 \\
c_1^3,0,c_2^3 & c_3^3,c_3^3,c_4^3 \\
\bar{c}_1^5,0,c_2^5 & \bar{c}_3^5,c_3^5,c_4^5 \\
1,0,c_0^5 & 1,c_1^5,c_2^5
\end{pmatrix}
\begin{pmatrix}
0,\bar{c}_1^4,c_2^4 \\
c_3^4,\bar{c}_3^4,c_4^4 \\
\bar{c}_1^6,\bar{c}_3^6,c_4^6 \\
1,\bar{c}_1^6,c_2^6
\end{pmatrix}
\]

\[c_i = 1 - c_i\]
Multiple Extraordinary Vertices

\[ C = \begin{pmatrix}
(0, 0, c_0^n) & (0, c_1^n, c_2^n) & (0, \frac{2}{3}, 1) & (0, 1, 1) \\
(c_1^n, 0, c_2^n) & (c_3^n, c_3^n, c_4^n) & (\frac{1}{3}, \frac{2}{3}, 1) & (\frac{1}{3}, 1, 1) \\
(\frac{2}{3}, 0, 1) & (\frac{2}{3}, \frac{1}{3}, 1) & (\frac{2}{3}, \frac{2}{3}, 1) & (\frac{2}{3}, 1, 1) \\
(1, 0, 1) & (1, \frac{1}{3}, 1) & (1, \frac{2}{3}, 1) & (1, 1, 1)
\end{pmatrix} \]
## Results

<table>
<thead>
<tr>
<th>Valence</th>
<th>$c_0^n$</th>
<th>$c_1^n$</th>
<th>$c_2^n$</th>
<th>$c_3^n$</th>
<th>$c_4^n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.0759</td>
<td>0.1014</td>
<td>0.6548</td>
<td>0.2105</td>
<td>0.9240</td>
</tr>
<tr>
<td>5</td>
<td>0.3296</td>
<td>0.0850</td>
<td>0.7913</td>
<td>0.3223</td>
<td>0.8958</td>
</tr>
<tr>
<td>6</td>
<td>0.4188</td>
<td>0.0759</td>
<td>0.8259</td>
<td>0.3636</td>
<td>0.8768</td>
</tr>
<tr>
<td>7</td>
<td>0.4830</td>
<td>0.0686</td>
<td>0.8487</td>
<td>0.3949</td>
<td>0.8595</td>
</tr>
<tr>
<td>8</td>
<td>0.5293</td>
<td>0.0629</td>
<td>0.8645</td>
<td>0.4189</td>
<td>0.8451</td>
</tr>
<tr>
<td>9</td>
<td>0.5634</td>
<td>0.0586</td>
<td>0.8757</td>
<td>0.4372</td>
<td>0.8335</td>
</tr>
<tr>
<td>10</td>
<td>0.5890</td>
<td>0.0553</td>
<td>0.8840</td>
<td>0.4515</td>
<td>0.8242</td>
</tr>
</tbody>
</table>

Valence-dependent precomputed parameters
Results

Gregory surface

Original
Results

Gregory surface

With reparameterization
Results

Original  Our method
Results

Gregory surface  Original
Results

Gregory surface  With reparameterization
Results

Original

Our method
Results

Gregory surface

Original
Results

Gregory surface

With reparameterization
Results

Original  Our method
Subdivision surface
With reparameterization
Conclusion

• Handles arbitrary number of extraordinary vertices per patch

• Geometry independent reparameterization

• Simple, efficient implementation
Thanks!