Folded Paper Geometry from 2D Pattern and 3D Contour

Damien Rohmer, Marie-Paule Cani, Stefanie Hahmann, Boris Thibert

Grenoble & Lyon University, INRIA, France
Folded Papers are rare in video-games & CG Movies

- Few available modeling tools!
  - Non smooth
  - Isometry preserving

Goal:

2D pattern + 3D boundary → Real pictures
Related Work

- Physically based modeling

  Cloth simulators
  Thin plates from folds
  Specific spring-mass system

  [Choi, Ko; TOG 02]
  [English, Bridson; TOG 08]
  [Thomaszewski et al.; CGF 09]
  [Burogno et al.; C&A 06]
  [Kang et al.; CASA 09]

- Slow, Smooth surface
- Folds are user defined
- Folds along existing edges
Related Work

• Geometric approaches

Developable construction

Mesh deformation

Procedural generation

Restricted to the convex hull

Slow, smooth surface

Limited range of deformation

[Frey; CAD 04]
[Rose et al.; SGP 07]
[Tang, Chen; TVCG 09]
[Popa et al.; CGF 09]
[Decaudin et al.; CGF 06]
Our Key Idea

• New subdivision improving **length preservation**
• Automatic generation of **folding curves**

3D

Pattern

Input

Preserved isometry
Overview

• Divide & Conquer approach
  – **Localize** one fold
  – Compute optimal 3D **profile**
  – Divide

Input = 2D pattern + 3D boundary curve

Subdivision steps ...

Folding curve

Final folded surface
Recursive subdivision

• Input
  – 2D Pattern = convex polygon
  – 3D Boundary = 3D polyline

• Algorithm
  1. Localize fold curve
  2. Split into two separated parts
  3. Restart at 1. on the two parts

Loop until isometry is reached
Localizing fold line: straight line

- **Localize** = Find *good* pair of vertices

**Case 1: \( L = L_0 \)**

=> 2D line mapped in 3D straight line

**Case 2: \( L < L_0 \): 3D profile is not a straight line!**
Localizing fold line: curved folds

- **Localize** = Find pair of vertices with least compression

**Case 2: \( L < L_0 \)**

- profile = **cubic polynomial**
  - **precise**: good approximation of conical section
  - **robust**: does not oscillate
  - **fast**: limited degrees of freedom
Computing folding profile

**Goal:** Improve length preservation

=> Find the **best** profile **improving** length preservation

Several possible curves

Error in length

\[ E = \sum (L - L_0)^2 \]
Computing folding profile

Goal: Improve length preservation
   => Find the best profile improving length preservation

before subdivision:
   E0 = 1.0

• Optimization = non linear minimization E=f(curve)
• 6 degrees of freedom per curve (2 tangents)
• Curve is considered if E1<E0
Final surface

2D Delaunay triangulation

3D mapping

Computed 3D curves
Results: Band strip

Input

Subdivision

Our textured result
Results: comparison to real sheet

Input

Subdivision

Our result

Real example
Results: folded paper

Input

Subdivision

Spring-mass

Real paper

Our result
Results: complex folded paper

Input

Spring-mass

Real paper

Subdivision

Our result
Results: complex folded paper
Results: Real time capture
# Results: Residual error

<table>
<thead>
<tr>
<th></th>
<th>Error Length</th>
<th>Error Angle</th>
<th>Error Area</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.09 (0.21)</td>
<td>0.21 (0.25)</td>
<td>1.12 (2.2)</td>
<td>&lt;0.1s</td>
</tr>
<tr>
<td></td>
<td>1.28 (2.5)</td>
<td>0.16 (1.4)</td>
<td>2.52 (18.3)</td>
<td>&lt;0.1s</td>
</tr>
<tr>
<td></td>
<td>0.21 (0.25)</td>
<td>0.35 (1.9)</td>
<td>2.89 (22.8)</td>
<td>0.2s</td>
</tr>
<tr>
<td></td>
<td>1.12 (2.2)</td>
<td>2.52 (18.3)</td>
<td>8.0 (18)</td>
<td>0.6s</td>
</tr>
</tbody>
</table>

\[
E_{\text{length}} = \sum_i (L_i^0 - L_i)^2 ;
E_{\text{angle}} = \sum_i (\alpha_j^0 - \alpha_j)^2 ;
E_{\text{area}} = \sum_k (a_k^0 - a_k)^2
\]
Results: Extension to metal material

Input

Our result
Results: Robustness to extended/compressed 3D boundary

- 2D pattern
- Artificial compression
- 3D curve
- Flat surface
- Artificial extension
- 3D curve
- Plausible folds
Limitations

- Input
- Self collision
- Error residual
- Static only
Conclusion

New subdivision algorithm
  • creates paper looking surface
  • almost isometry preserving

Main ideas:
  • Localize the folds: least compression bw vertices
  • Find the best profile: minimizing length error

+ Fast
+ Non smooth surface
+ Adapted mesh
Thank you