Cross-Parameterization and Compatible Remeshing of 3D Models

Vladislav Kraevoy     Alla Sheffer
Department of Computer Science
University of British Columbia
X-Parameterization

- X-Parameterization: mapping between surface models (meshes)
- Applications:
  - Properties transfer
  - Morphing
  - Editing – e.g. blending
  - Template fitting
X-Parameterization

• Requirements
  – Bijectivity (one-to-one)
  – Feature correspondence
    – vertex to vertex
  – Low distortion
Compatible Remeshing

- Compatible meshes – meshes with identical connectivity
- Required by many applications of X-parameterization
  - Morphing
  - Editing
- Prerequisite: X-parameterization
Previous work: X-parameterization

- Most use heuristics - can fail
- Base mesh [Lee et al. 1999, Michikawa et al. 2001, Praun et al. 2001, Schreiner et al. 04]
  - Segment meshes into triangular patches (same connectivity)
  - Map patches to base triangles
  - [Praun et al. 01] - given a base mesh robustly construct segmentation
  - [Schreiner et al. 04] – more later
Previous work: Compatible remeshing

- Mutual tessellation [Alexa 2000, Schreiner et al. 04]
  - Intersect meshes in parameter domain

- Regular base mesh refinement [Lee et al. 1999, Praun et al. 2001]
  - Remesh with subdivision connectivity

- Both methods: output meshes much larger (~x10) than input
  - For accurate approximation
Technique Goals

• X-Parameterization
  – Bijective
  – Exact feature vertex correspondence
  – Low distortion (preserve shape)
  – Minimal user input: models + feature vertices

• Compatible remeshing
  – Closely approximate the input models
  – Similar (order of magnitude) number of elements as input

• Efficient & robust
2. Low distortion, bijective X-parameterization

3. Compatible remeshing

1. Common base mesh construction
   - Provably correct for genus 0

Input: models + features
Initial X-Parameterization

• Map each patch to corresponding base triangle
  – Shape preserving parameterization
    [Floater 2003]
  – Guarantee bijectivity

• Given a base mesh previous methods employed similar techniques
• Not good enough…
Distortion & Artifacts

- Badly shaped patches = high distortion
- Artifacts: map polyline patch boundaries to straight lines
- Need additional processing to reduce distortion
X-Parameterization: Smoothing

- Optimize (smooth) base mesh parameterization
  - Move vertices between base mesh triangles
- Unconstrained parameterization [Khodakovsky et al., 2003]
  - Solve (repeatedly) a global linear system - far from trivial...
  - No guarantee of bijectivity
- Our approach
  - Treat base as set of overlapping domains
  - Iteratively optimize mapping within each domain
Overlapping domain structure

- Domain per base triangle
  - Contains base triangle + 3 adjacent
  - Mapped to equilateral triangle in 2D
    - Parameterize all vertices mapped to those triangles onto the domain
Smoothing

- For each base triangle \( b \)
  - Create corresponding overlapping domain
  - Compute new location in equilateral domain for each vertex on \( b \)
    - Based on neighbors
    - Shape preserving [Floater 2003]
  - Map vertices back to base mesh
    - Find base triangle & compute coordinates

- Repeat
- Adjacency assumption
Smoothing Framework

- Drastically reduce distortion
  - Improves patch shape + relaxes boundaries
- Preserve bijectivity
- Generic framework – can replace the shape preserving objective function by any other
Results

Normal transfer
40K/40K faces
59 sec

Texture transfer
80K/7K faces
56 sec

Motion transfer
joint result with [Sumner & Popovic 04]
Compatible Remeshing

• Previous:
  – overlay or subdivision
  – increase mesh size by order of magnitude

• Idea:
  – Use connectivity of one model ("source") as basis
  – Map to second model ("target") using X-parameterization
  – Improve target approximation
    • local modifications
Mesh Improvement

- X-parameterization between target \& approximation
  - Use intrinsic map between source and target approximation

- Use for
  - Computing approximation error
    - Error = distance between target vertex \& its map
    - Conservative
  - Local modifications
Compatible Remeshing

- Operations:
  - Smoothing
  - Refinement
- Smoothing
  - Use overlapping domains framework
  - Error based relocation formula
- Refine edges based on error
- Iterate

40K/40K faces – 49K faces
Results

Morphing
28K/8K - ZK

Blending

3K + 4K = 3.5K

3K + 3.5K = 4.5K
Summary

- Robust method for constrained X-parameterization
  - Input: two (or more) models + corresponding feature vertices
  - Automatic construction of base mesh
    - Provably correct for genus=0
    - For genus > 0 works most of the time

- New framework for low-distortion parameterization on base mesh
  - Independent of patch structure
  - Can be applied with different objective functions
  - Efficient
Summary

• New compatible remeshing scheme
  – Closely approximate input
  – Small output mesh
    • ~20% more triangles

• Future
  – Using smoothing framework for unconstrained parameterization
<table>
<thead>
<tr>
<th></th>
<th>Our method</th>
<th>[Schreiner et al. 04]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base Construction</td>
<td></td>
<td>More robust for genus &gt; 0 (add extra feature vertices)</td>
</tr>
<tr>
<td>Smoothing</td>
<td>2 min 64K (overlapping domains)</td>
<td>2 hours 64K (overlay smoothing)</td>
</tr>
<tr>
<td>Constraints</td>
<td>Exact constraints</td>
<td>Relax constraints to reduce distortion</td>
</tr>
<tr>
<td>Remeshing</td>
<td>output: ~x1.2 triangles as source input (parameterization based scheme)</td>
<td>output: ~x8 triangles as inputs (typical) (mutual tessellation)</td>
</tr>
</tbody>
</table>
Movie !!!

Cross-Parameterization and Compatible Remeshing of 3D Models