Overview
Surface parameterization
- Introduction
- Parameterization as a graph embedding problem [Floater 97]
- Distortion minimizing parameterization using linear methods [Desbrun 02]
- Angle based flattening [Sheffer 05]

Introduction
- Given a 3D mesh, find mapping to 2D mesh, with one-to-one correspondence between triangles
  - Flatten mesh, embed mesh in 2D
  - Mapping is piecewise linear

Applications
- Texturing
- Remeshing
- Geometry images
  - Uniform remeshing in parameter domain

Applications
- Smooth surface fitting
Desired features

- One-to-one (bijectivity)
  - No overlaps, fold-overs
- Preservation of intrinsic geometry ("distortion minimization")
  - Angles
  - Arc lengths
  - Area
- Efficient computation
  - Linear least squares optimization preferred over non-linear optimization

Overview

- Introduction
- Parameterization as a graph embedding problem [Floater 97]
- Distortion minimizing parameterization using linear methods [Desbrun 02]
- Angle based flattening [Sheffer 05]

Graph embedding

- Meshes are graphs
  - Vertices are nodes
  - Edges are arcs
  - Embedded in 3D
- Restrict ourselves to meshes with one boundary
  - No holes, disc topology
- These meshes are planar graphs
  - Can be embedded in a 2D plane such that
    - Each vertex is mapped to some point in 2D
    - Each edge connects its two vertices
    - There are no intersection of edges (except at vertices)
  - Embedding is called a plane graph

Examples

- Note: boundary of embedding is user specified

Discussion

Advantages
- Guarantees no self intersections, bijectivity
- Fast

Disadvantages
- Requires user specified boundary
- Based purely on mesh/graph connectivity
  - Does not take into account geometry
  - Does not attempt to preserve angles, arc lengths, areas
  - May lead to large distortions
## Overview
- Introduction
- Parameterization as a graph embedding problem [Floater 97]
- Distortion minimizing parameterization using linear methods [Desbrun 02]
- Angle based flattening [Sheffer 05]

## Distortion minimization
- Attempt to preserve angles, areas
- Discrete conformal parameterization
  - Angle preserving
- Discrete authalic parameterization
  - Area preserving

## DCP
- Discrete conformal parameterization

## Near optimal parameterizations

## Discussion
**Advantages**
- Fast, linear problem
- Minimize discrete angle, area distortion

**Disadvantages**
- Linear weights may be negative
- May lead to overlap

## Overview
- Introduction
- Parameterization as a graph embedding problem [Floater 97]
- Distortion minimizing parameterization using linear methods [Desbrun 02]
- Angle based flattening [Sheffer 05]
Angle based flattening

- Observation: angles define 2D triangulation up to rigid transformation and uniform scaling
- Idea: two step procedure
  - Determine angles (optimization)
  - Convert to 2D coordinates (reconstruction)
- Optimization directly measures distortion of angles in triangles

Energy function

\[ E(\alpha) = \sum_{i=1}^{3} \sum_{j=1}^{3} \frac{1}{w_{ij}} (a_{ij} - B_{ij})^2 \]

Constraints

- Triangle validity
  \[ \forall v \in T, \quad C_{\text{tri}}(v) = a_{ij} + a_{ji} + a_{jk} - \pi = 0 \]
- Planarity
  \[ \forall v \in V_{\text{tri}}, \quad C_{\text{pl}}(v) = \sum_{(i,j,k) \in v} a_{ij} - 2\pi = 0 \]
- Reconstruction: edges shared by neighboring triangles have same length

Optimization procedure

- Challenging, since constraints are non-linear
- Details in the paper
  - Sequential linearly constrained programming
  - Hierarchy
- Advantage: guarantees that there are no flips

Texture atlases

- So far, discussed parameterization of disc-like objects
- Generalization to arbitrary objects using segmentation into patches
  - Patches form an atlas: are non overlapping and cover the object completely
  - Each patch is disc-like
  - Parameterize each patch separately

Segmentation

- Feature detection
- Region growing

Examples

- Patch parameterization using variant of DCP

[Levy 02]
Next time

- Registration