Project On Parameterization

Xuan Li

Stony Brook University

xuanli2@cs.stonybrook.edu

December 7, 2017

Noam Aigerman, Yaron Lipman (2015)
Orbifold Tutte Embeddings
ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH Asia 2015, 34(6)

Noam Aigerman, Yaron Lipman (2016)
Hyperbolic Orbifold Tutte Embeddings
ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH Asia 2016, 35(6)

Rohan Sawhney, Keenan Crane (2017)
Boundary First Flattening
ACM Transactions on Graphics (TOG) (Under Review)

Overview

Project Conponents

Orbifold Embedding
Generalization of Tutte embedding to sphere-type mesh.
I explore two kinds of background geometry: Euclidean and hyperbolic.

Boundary First Flattening
Based on differential geometry.
Conformal map.
Handle curvature and conformal factors directly.
With cone technique, can be generalize to meshes with slices.

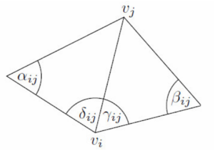
Tutte Embedding

The embedding problem is the following full-rank linear system:

$$\sum_{v_j \in N(v_i)} w_{ij}(z_j - z_i) = 0, \quad v_i \in V - B$$

$$z_i = z_i^0, \quad v_i \in B$$

where $w > 0$.
Harmonic Map: $w_{ij} = \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2}$



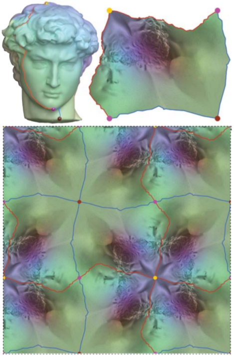
Tutte Embedding Equivalence

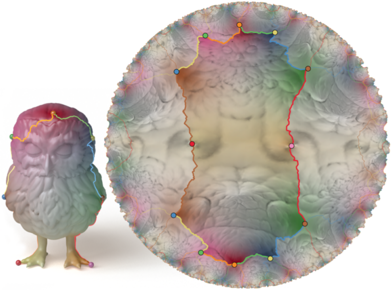
$$\min_{\Phi} \quad E(z) = \frac{1}{2} \sum_{(i,j)\in E} w_{ij} d(z_i, z_j)^2$$

$$s.t \quad z_i = z_i^0, \qquad v_i \in B$$

Orbifold

Tiled space defined by a basic tile.
Points differed by a transition between copies are equivlence.
Transition: rigid motions.





Euclidean Orbifold Tutte Embedding

Select some cones and cut the mesh into a disk, we solve:
For $v_i \in \bar{V} - \bar{B}$:

$$\sum_{v_j \in N(v_i)} w_{ij}(z_j - z_i) = 0$$

For $(v_i, v_{i'})$ boundary pair,

$$\sum_{v_j \in N(v_i)} w_{ij}(z_j - z_i) + \sum_{v_j \in N(v_{i'})} w_{i'j} R_{i'i}(z_j - z_{i'}) = 0$$
$$R_{i'i} z_{i'} - z_i = t_{ii'}$$

For $v_i \in \bar{C}$,

$$z_i = z_i^0$$

Hyperbolic Tutte Embedding

$$\min_{\Phi} E(z),$$
$$s.t \quad z_i = m_{i'i}(z_{i'}), \qquad v_i \in \bar{B} - \bar{C}$$
$$z_i = z_i^0, \qquad v_i \in \bar{C}$$

The gradient of the energy is given by

$$\nabla_{z_i} E = \sum j \in N_i w_{ij} \nabla_{z_i} d^2(z_i, z_j) + \sum j \in N_{i'} w_{i'j} \nabla_{z_i} d^2(z_i, m_{i'i}(z_j))$$

Poisson Problems

Dirichlet-type condition:

$$\begin{aligned} \Delta a &= \phi \qquad on\ M \\ a &= g, \quad on\ \partial M \end{aligned} \tag{1}$$

Riemann-type condition:

$$\begin{aligned} \Delta a &= \phi \qquad on\ M \\ \frac{\partial a}{\partial n} &= h, \quad on\ \partial M \end{aligned} \tag{2}$$

Discrete version (Riemann):

$$\begin{bmatrix} A_{II} & A_{IB} \\ A_{BI} & A_{BB} \end{bmatrix} \begin{bmatrix} a_I \\ a_B \end{bmatrix} = \begin{bmatrix} \phi_I \\ \phi_B - h \end{bmatrix} \tag{3}$$

Conversion between Dirichlet and Riemann

Given h, we simply solve the Poisson equation and set

$$g = \Lambda_\phi^* h = a_B, \tag{4}$$

Given g, we can convert it to h by

$$h = \Lambda_\phi g = \phi_B - A_{IB}^T A_{II}^{-1}(\phi_I - A_{IB} g) - A_{BB} g \tag{5}$$

Cherrier Formula

For a conformal map $f : M \to \tilde{M}$, we have

$$\begin{aligned} \Delta u &= K - e^{2u}\tilde{K} \quad on\ M \\ \frac{\partial u}{\partial n} &= k - e^{u}\tilde{k} \quad on\ \partial M \end{aligned} \tag{6}$$

where u is conformal factor, K,$\tilde{K}$ are Gauss curvature of the source surface and the target surface, $k, \tilde{k}$ is geodesic curvature of the source surface and the target surface.

Discrete Cherrier Formula

$$\begin{aligned} Au &= \Omega - \tilde{\Omega} \quad on\ Int\ M \\ h &= k - \tilde{k} \quad on\ \partial M \end{aligned} \tag{7}$$

where $\Omega_i = 2\pi - \sum_{ijk \in F} \theta_i^{jk}$ defined on interior vertices, $k_i = \pi - \sum_{ijk \in F} \theta_i^{jk}$ defined on boundary vertices, which is the exterior angle at v_i. Note that Ω is defined to be zero at boundary vertices.

Algorithm

The pipeline of the BFF algorithm is as follows:
1) If boundray conformal factor u is known, we compute bounadry target k by:

$$\tilde{k} = k - \Lambda_{\Omega} u.$$

If target boundary $\tilde{k}$ is known, we compute boundary conformal factor u by::

$$u = \Lambda_{\Omega}^{*}(k - \tilde{k}).$$

2) Define new boundary edge lengths as:

$$l_{ij}^{*} = e^{\frac{u_i + u_j}{2}} l_{ij}.$$

With $\tilde{k}$, which is exterior angles, we can integrate these two data into a closed loop.
3) Extend the loop into interior conformally.

Loop Integration

In step 2, usually, directly integrating $\tilde{k}$ over l_{ij}^* won't give a closed loop, we should change edge length a little. So we solve the following quadratic optimization problem:

$$\begin{aligned} \min_{\tilde{l}} \quad & ||\tilde{l} - l^*||_2^2 \\ s.t. \sum_{ij \in \partial M} & \tilde{l}_i j \tilde{T}_{ij} \end{aligned} \tag{8}$$

Conformal Interpolation

Two choice:
1) Use harmonic map on both components
2) Use harmonic map on one componet, and minimize conformal energy on the other component. Linear problem.

Video Demo

GitHub: xuan-li
(https://github.com/xuan-li/GraphicsProject)