Mesh Parameterization

Richard Liu

September 23, 2021

Introduction

Surface parameterization: construct a (bijective) map between two surfaces with similar topology

- Roots in cartography: how to make accurate map of Earth?

Mesh parameterization: construct a map between a triangular mesh and another surface (most often 2D plane)

Mesh Parameterization Applications

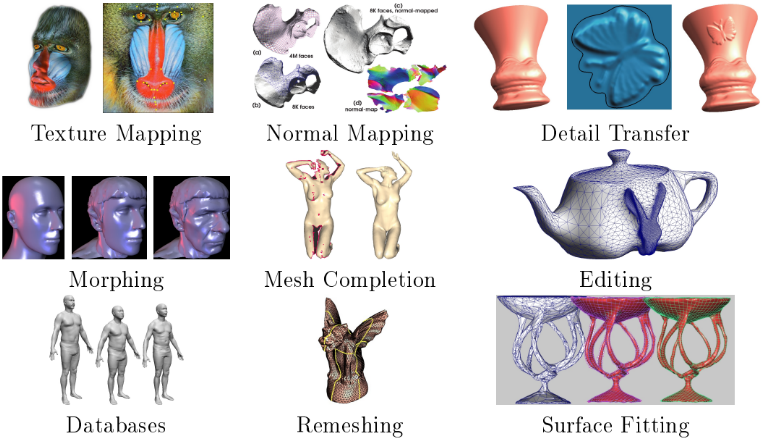


Figure: Parameterization Applications

Mathematical Framework

Given a function $f : X \to Y$,

Definition

f is **injective** or **one-to-one**, if $\forall x, x' \in X, f(x) = f(x') \Rightarrow x = x'$.

Mathematical Framework

Given a function $f : X \to Y$,

Definition

f is **injective** or **one-to-one**, if $\forall x, x' \in X, f(x) = f(x') \Rightarrow x = x'$.

Definition

f is **surjective** or **onto**, if $\forall y \in Y, \exists x \in X$ s.t. $y = f(x)$.

Mathematical Framework

Given a function $f : X \to Y$,

Definition

f is **injective** or **one-to-one**, if $\forall x, x' \in X, f(x) = f(x') \Rightarrow x = x'$.

Definition

f is **surjective** or **onto**, if $\forall y \in Y, \exists x \in X s.t. y = f(x)$.

Definition

f is **bijective** if f is both injective and surjective. Equivalently, f is bijective iff it is **invertible**.

Mathematical Framework

Definition

Let $\Omega \subset \mathbb{R}^2$ be a **simply connected** (without holes) region. Let $f : \Omega \to \mathbb{R}^3$ be continuous and injective. The image of f is called a **surface**

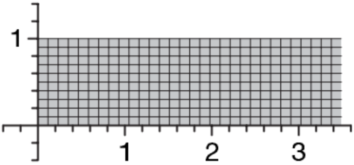
$$S = f(\Omega) = \{f(u, v) : (u, v) \in \Omega\}$$

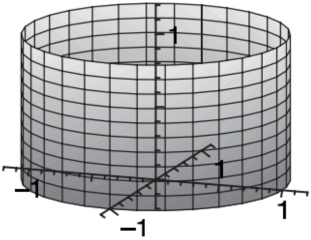
We say that f is a **parameterization** of S over the **parameter domain** Ω.

Note: By construction, $f : \Omega \to S$ is trivially surjective. In practice injectivity is often what we care about.

Mathematical Framework

Example

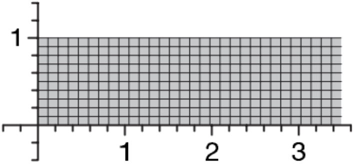


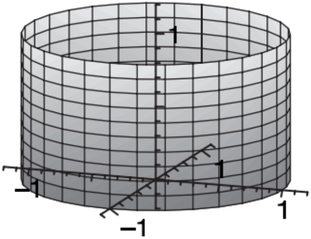


- Parameter domain: $\Omega = \{(u, v) \in \mathbb{R}^2 : u \in [0, 2\pi), v \in [0, 1]\}$

Mathematical Framework

Example

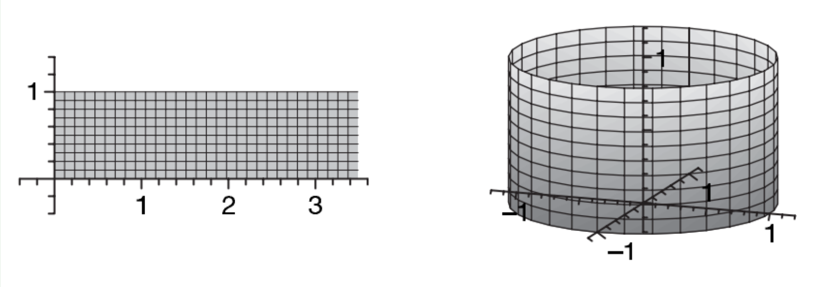




- Parameter domain: $\Omega = \{(u, v) \in \mathbb{R}^2 : u \in [0, 2\pi), v \in [0, 1]\}$
- Surface: $S = \{(x, y, z) \in \mathbb{R}^{:} x^2 + y^2 = 1, z \in [0, 1]\}$

Mathematical Framework

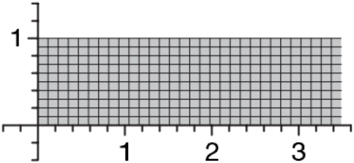
Example

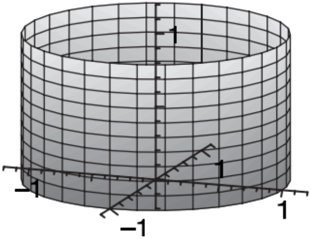


- Parameter domain: $\Omega = \{(u, v) \in \mathbb{R}^2 : u \in [0, 2\pi), v \in [0, 1]\}$
- Surface: $S = \{(x, y, z) \in \mathbb{R}^{:} x^2 + y^2 = 1, z \in [0, 1]\}$
- Parameterization: $f(u, v) = (\cos u, \sin u, v)$

Mathematical Framework

Example





- Parameter domain: $\Omega = \{(u, v) \in \mathbb{R}^2 : u \in [0, 2\pi), v \in [0, 1]\}$
- Surface: $S = \{(x, y, z) \in \mathbb{R}^{:} x^2 + y^2 = 1, z \in [0, 1]\}$
- Parameterization: $f(u, v) = (\cos u, \sin u, v)$
- Inverse: $f^{-1}(x, y, z) = (\arccos x, z)$

Mathematical Framework

Remark

A parameterization $f : \Omega \to S$ is never unique. Given any bijection $\gamma : \Omega \to \Omega$, $g = f \circ \gamma$ is a parameterization of S over Ω.

Mathematical Framework

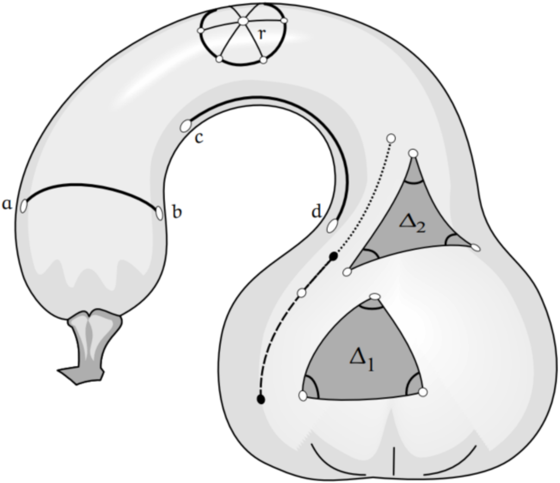
We can use f for deriving some key **intrinsic surface properties**, or properties that are independent of how the surface sits in space (extrinsic geometry).

Mathematical Framework

We can use f for deriving some key **intrinsic surface properties**, or properties that are independent of how the surface sits in space (extrinsic geometry).

Another perspective: everything that is **knowable** to a tiny observable living on the surface (e.g. humans on the Earth)

Mathematical Framework



[1.9] *The* **intrinsic geometry** *of the surface of a crookneck squash:* **geodesics** *are the equivalents of straight lines, and triangles formed out of them may possess an angular excess of either sign, depending on how the surface bends:* $\mathcal{E}(\Delta_1) > 0$ *and* $\mathcal{E}(\Delta_2) < 0$.

Mathematical Framework

Definition

A parameterization $f : \Omega \subset \mathbb{R}^2 \to S \subset \mathbb{R}^3$ is **regular** if the tangent vectors $f_u = \frac{\partial f}{\partial u}$ and $f_v = \frac{\partial f}{\partial v}$ are always linearly independent.

Note: f_u, f_v are functions from $\mathbb{R}^2$ to $\mathbb{R}^3$ and span the local tangent plane.

Mathematical Framework

Definition

Given a regular parameterization f, the **surface normal** n_f is defined as

$$n_f = \frac{f_u \times f_v}{||f_u \times f_v||}$$

Mathematical Framework

Definition

Given a regular parameterization f, the **surface normal** n_f is defined as

$$n_f = \frac{f_u \times f_v}{||f_u \times f_v||}$$

Note: regularity is required for n_f to be nonzero everywhere.

Mathematical Framework

Definition

Given a regular parameterization f, the **surface normal** n_f is defined as

$$n_f = \frac{f_u \times f_v}{||f_u \times f_v||}$$

Note: Regularity is required for n_f to be nonzero everywhere.
Note: The surface normal is always **independent** of the parameterization, making it an intrinsic property.

Mathematical Framework

We can also apply f towards deriving the **first and second fundamental forms.** They are fundamental precisely because they determine the key metric properties of a surface, such as the **gaussian curvature**, **mean curvature**, and **surface area**.

Mathematical Framework

Definition

Given parameterization f, the **first fundamental form** is defined as

$$\mathrm{I}_f = \begin{pmatrix} f_u \cdot f_u & f_u \cdot f_v \\ f_v \cdot f_u & f_v \cdot f_v \end{pmatrix} = \begin{pmatrix} E & F \\ F & G \end{pmatrix}$$

Mathematical Framework

Definition

Given parameterization f, the **first fundamental form** is defined as

$$\mathrm{I}_f = \begin{pmatrix} f_u \cdot f_u & f_u \cdot f_v \\ f_v \cdot f_u & f_v \cdot f_v \end{pmatrix} = \begin{pmatrix} E & F \\ F & G \end{pmatrix}$$

Area of a Surface

Given parameterization $f : \Omega \to S$, the area $A(S)$ can be found

$$A(S) = \int_\Omega \sqrt{\det(\mathrm{I}_f)} dudv$$

Mathematical Framework

Definition

Given a twice-differentiable parameterization f, the **second fundamental form** is defined as

$$\text{II}_f = \begin{pmatrix} f_{uu} \cdot n_f & f_{uv} \cdot n_f \\ f_{uv} \cdot n_f & f_{vv} \cdot n_f \end{pmatrix} = \begin{pmatrix} L & M \\ M & N \end{pmatrix}$$

Mathematical Framework

Definition

The Gaussian curvature is

$$K = \det(\mathrm{I}_f^{-1}\mathrm{II}_f) = \frac{\det \mathrm{II}_f}{\det \mathrm{I}_f} = \frac{LN - M^2}{EG - F^2}$$

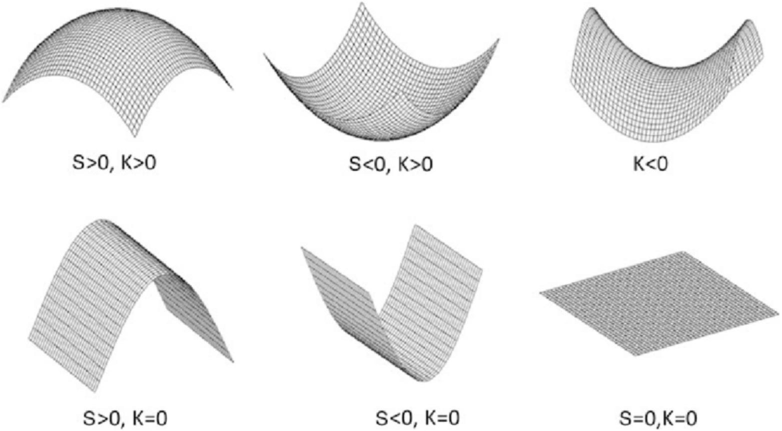
Mathematical Framework

Definition

The mean curvature is

$$S = \frac{1}{2}\text{trace}(\mathrm{I}_f^{-1}\mathrm{II}_f) = \frac{LG - 2MF + NE}{2(EG - F^2)}$$

Mathematical Framework



Mathematical Framework

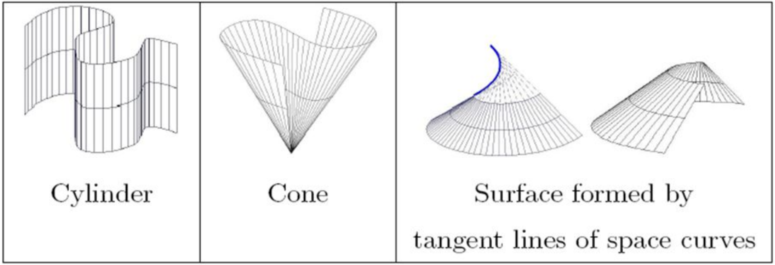
Definition

A surface S is **developable** if $\forall p \in S$, $K(p) = 0$, i.e. the Gaussian curvature is 0 everywhere on S.

Mathematical Framework

Developable Surface

Three types of developable surfaces



Mathematical Framework

Definition

The **Jacobian** of parameterization f is the 3 x 2 matrix of partial derivatives of f.

$$J_f = (f_u, f_v)$$

Mathematical Framework

Definition

The **Jacobian** of parameterization f is the 3 x 2 matrix of partial derivatives of f.

$$J_f = (f_u, f_v)$$

Mathematical Framework

Definition

For any $m \times n$ matrix J, the **singular value decomposition** (SVD) is given by

$$J = U \Sigma V^T$$

where Σ is an $m \times n$ diagonal matrix, and U and V are $m \times m$ and $n \times n$ orthonormal matrices, respectively.

Mathematical Framework

Definition

For any $m \times n$ matrix J, the **singular value decomposition** (SVD) is given by

$$J = U \Sigma V^T$$

where Σ is an $m \times n$ diagonal matrix, and U and V are $m \times m$ and $n \times n$ orthonormal matrices, respectively.

By the above, the SVD of the Jacobian is

$$J_f = U \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} V^T$$

where σ_1, σ_2 are the **singular values**.

Mathematical Framework

There is an easier way to get the singular values of the Jacobian.

Mathematical Framework

There is an easier way to get the singular values of the Jacobian.

Remark

We can write the first fundamental form as

$$\mathrm{I}_f = J_f^T J_f = \begin{pmatrix} f_u^T \\ f_v^T \end{pmatrix} \begin{pmatrix} f_u & f_v \end{pmatrix}$$

It is clear I_f is **symmetric**.

Mathematical Framework

There is an easier way to get the singular values of the Jacobian.

Remark

We can write the first fundamental form as

$$\mathrm{I}_f = J_f^T J_f = \begin{pmatrix} f_u^T \\ f_v^T \end{pmatrix} \begin{pmatrix} f_u & f_v \end{pmatrix}$$

It is clear I_f is **symmetric**.

Thus the eigenvalues of I_f are given by

$$\lambda_{1,2} = \frac{1}{2}((E+G) \pm \sqrt{4F^2 + (E-G)^2}$$

.

Mathematical Framework

Remark

For a matrix A, the singular values are the square roots of the eigenvalues of $A^T A$.

Mathematical Framework

Remark

For a matrix A, the singular values are the square roots of the eigenvalues of $A^T A$.

The singular values of J can be found using the eigenvalues of I_f

$$\sigma_1 = \sqrt{\lambda_1}$$
$$\sigma_2 = \sqrt{\lambda_2}$$

Mathematical Framework

Remark

For a matrix A, the singular values are the square roots of the eigenvalues of $A^T A$.

The singular values of J can be found using the eigenvalues of I_f

$$\sigma_1 = \sqrt{\lambda_1}$$
$$\sigma_2 = \sqrt{\lambda_2}$$

σ_1 and σ_2 tell us **everything** about the **metric distortion** induced by the parameterization.

Properties of Parameterizations

Parameterizations induce distortion in **lengths**, which can be further divided into distortion in **angles** and distortion in **areas**.

Properties of Parameterizations

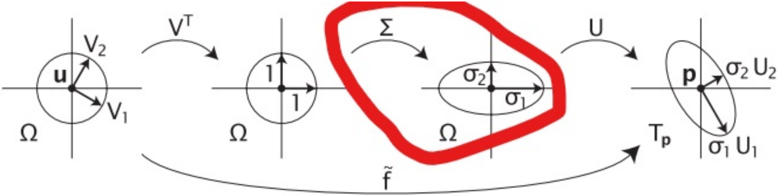


Figure: SVD Decomposition of mapping $\tilde{f}$

Properties of Parameterizations

Definition

A parameterization is **conformal**, or **angle-preserving**, when the singular values of the Jacobian are equal, i.e. $\sigma_1 = \sigma_2$.

Properties of Parameterizations

Definition

A parameterization is **conformal**, or **angle-preserving**, when the singular values of the Jacobian are equal, i.e. $\sigma_1 = \sigma_2$.

Definition

A parameterization is **equiareal/authalic**, or **area-preserving**, when the singular values of the Jacobian multiply to 1, i.e. $\sigma_1 \sigma_2 = 1$.

Properties of Parameterizations

Definition

A parameterization is **conformal**, or **angle-preserving**, when $\sigma_1 = \sigma_2$.

Definition

A parameterization is **equiareal/authalic**, or **area-preserving**, when $\sigma_1 \sigma_2 = 1$.

Definition

A parameterization is **isometric**, or **length-preserving** iff it is conformal and equiareal, i.e. $\sigma_1 = \sigma_2 = 1$.

Properties of Parameterizations

So can we always find an isometric parameterization to the plane?

Properties of Parameterizations

So can we always find an isometric parameterization to the plane?

NOPE

Properties of Parameterizations

Theorem

(Gauss, 1827) Globally isometric parameterizations (from 3D to 2D) only exist for developable *surfaces (i.e. $K = 0$ everywhere)*

Properties of Parameterizations

So how to find the "best" parameterization?

Properties of Parameterizations

So how to find the "best" parameterization?

Take bivariate non-negative function $E : \mathbb{R}_+^2 \to \mathbb{R}_+$ that takes local distortion measures σ_1 and σ_2, and has minimum defined according to objective.

$$E(f) = \int_\Omega E(\sigma_1(u, v), \sigma_2(u, v)) dudv / A(\Omega)$$

Properties of Parameterizations

So how to find the "best" parameterization?

Take bivariate non-negative function $E : \mathbb{R}_+^2 \rightarrow \mathbb{R}_+$ that takes local distortion measures σ_1 and σ_2, and has minimum defined according to objective.

$$E(f) = \int_\Omega E(\sigma_1(u,v), \sigma_2(u,v)) du dv / A(\Omega)$$

e.g. E global minimum at $(1,1)$ = isometry objective

e.g. E minimal values along (x,x) for $x \in \mathbb{R}_+$ = conformal objective

Discrete Setting

Now let's consider triangle meshes specifically, which can be considered **piecewise linear surfaces**.

Discrete Setting

Definition

A **mesh** is a triangulation $M = (V, E, F)$, where $V = \{v_i\} \subset \mathbb{R}^3$, $E = \{e_{ij}\}$, and $F = \{f_{ijk}\}$ are the vertex, edge, and face sets, respectively. More formally, edge e_{ij} represents the convex hull between vertices v_i and v_j (i.e. line segment), and face f_{ijk} is the convex hull of non-collinear points v_i, v_j, v_k.

Mesh Parameterization Properties

We already mentioned **conformal** and **equiareal** maps. Another important property for applications to meshes is **bijectivity**.

Mesh Parameterization Properties

We already mentioned **conformal** and **equiareal** maps. Another important property for applications to meshes is **bijectivity**.

e.g. For texture mapping, want to be able to annotate parts of the texture with reference to unique region of surface

Mesh Parameterization Properties

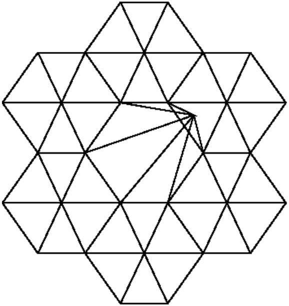
Definition

A mesh parameterization is **locally injective** if no triangles change orientation ("flip" or "fold over") during the parameterization.

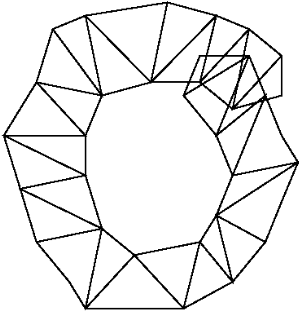
Definition

A mesh parameterization is **globally bijective** if it is locally injective and the boundary of the parameterization does not intersect itself.

Mesh Parameterization Properties



Triangle Flip



Boundary Intersection

Mesh Parameterization Properties

In general, mesh parameterization methods can be characterized by the following set of properties:

- Distortion minimized: $\{$angle (conformal), area (equiareal), distance (isometric)$\}$
- Boundary: $\{$fixed, free$\}$
- Bijectivity: $\{$global, local$\}$

Mesh Parameterization Properties

In general, mesh parameterization methods can be characterized by the following set of properties:

- Distortion minimized: $\{$**angle (conformal)**, area (equiareal), distance (isometric)$\}$
- Boundary: $\{$fixed, free$\}$
- Bijectivity: $\{$global, local$\}$

Fixed Boundary Methods

Boundary-based, or **barycentric** mappings all follow the same general procedure.

Fixed Boundary Methods

Boundary-based, or **barycentric** mappings all follow the same general procedure.

1. Choose the **shape of the boundary** of the parameter domain and the **distribution** of the parameter points around the boundary.

Fixed Boundary Methods

Boundary-based, or **barycentric** mappings all follow the same general procedure.

1. Choose the **shape of the boundary** of the parameter domain and the **distribution** of the parameter points around the boundary.
2. Compute **barycentric coordinates** for the interior vertices

Fixed Boundary Methods

Boundary-based, or **barycentric** mappings all follow the same general procedure.

1. Choose the **shape of the boundary** of the parameter domain and the **distribution** of the parameter points around the boundary.
2. Compute **barycentric coordinates** for the interior vertices
3. Solve a **linear system** based around minimizing the spring energy of the mesh

Fixed Boundary Methods

Barycenteric coordinates are simply a way of representing an interior point in a polygon (typically triangle) as a linear combination of its vertices.

Fixed Boundary Methods

Barycenteric coordinates are simply a way of representing an interior point in a polygon (typically triangle) as a linear combination of its vertices.

Definition

For a point x in the interior of a triangle $f_{ijk} = \{v_i, v_j, v_k\}$, values λ_i, λ_j, λ_k are **barycentric coordinates** of x with respect to the vertices of f_{ijk} if:

1. $x = \lambda_i v_i + \lambda_j v_j + \lambda_k v_k$
2. $\lambda_i + \lambda_j + \lambda_k = 1$

Fixed Boundary Methods

Barycenteric coordinates are simply a way of representing an interior point in a polygon (typically triangle) as a linear combination of its vertices.

Definition

For a point x in the interior of a triangle $f_{ijk} = \{v_i, v_j, v_k\}$, values λ_i, λ_j, λ_k are **barycentric coordinates** of x with respect to the vertices of f_{ijk} if:

1. $x = \lambda_i v_i + \lambda_j v_j + \lambda_k v_k$
2. $\lambda_i + \lambda_j + \lambda_k = 1$

Note: above definition can be easily generalized to n-gons, but barycentric coordinates are **only unique** when x has 3 neighbors.

Fixed Boundary Methods

Fixed boundary methods primarily differ on how to construct the barycentric coordinates, and how to deal with the boundary.

Typically want to choose a **convex** parameter domain. Why?

Fixed Boundary Methods

Fixed boundary methods primarily differ on how to construct the barycentric coordinates, and how to deal with the boundary.

Typically want to choose a **convex** parameter domain. Why?

Theorem

Tutte (1963) For a parameterization $f : \Omega \to S$ constructed by fixing the boundary and computing positive *barycentric coordinates for the interior vertices, if Ω is* convex, *then f is bijective.*

Fixed Boundary Methods

Tutte embeddings. Tutte first to introduce the above-described framework into the mesh parameterization context with his seminal work on straight-line embeddings of planar graphs.

Fixed Boundary Methods

Tutte embeddings. Tutte first to introduce the above-described framework into the mesh parameterization context with his seminal work on straight-line embeddings of planar graphs.

- $\lambda_{ij} = 1/|N_i|$ defined uniformly (**not** barycentric)
- Guarantee **bijectivity** under certain constraints
- **No** guarantee of distortion minimization

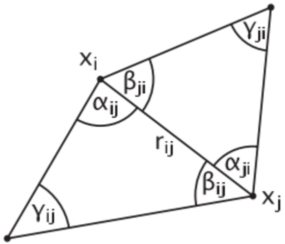
Fixed Boundary Methods

Harmonic parameterization. Eck et al.'s method makes use of **harmonic coordinates**, or **cotangent weights** (very famous).

Fixed Boundary Methods

Harmonic parameterization. Eck et al.'s method makes use of harmonic coordinates, or **cotangent weights** (well known).

$$w_{ij} = \cot \gamma_{ij} + \cot \gamma_{ji}$$



Fixed Boundary Methods

Harmonic parameterization. Eck et al.'s method makes use of harmonic coordinates, or **cotangent weights** (very famous).

- Minimize **harmonic energy** ($\triangle f(u, v) = 0$)

Fixed Boundary Methods

Harmonic parameterization. Eck et al.'s method makes use of harmonic coordinates, or **cotangent weights** (very famous).

- Minimize **harmonic energy** ($\triangle f(u, v) = 0$)
- Harmonic condition **weaker** than conformal

Fixed Boundary Methods

Harmonic parameterization. Eck et al.'s method makes use of harmonic coordinates, or **cotangent weights** (very famous).

- Minimize **harmonic energy** ($\triangle f(u, v) = 0$)
- Harmonic condition **weaker** than conformal
- Weights can be negative when angles are obtuse $\Rightarrow$ non-bijective parameterization

Fixed Boundary Methods

Other Coordinates.

- Wachspress coordinates (Wachspress 1975)
- Mean value coordinates (Floater 2003)

Fixed Boundary Methods

Pros

- Weights can be computed for every interior vertex even if neighbors not coplanar or more than 3 vertex neighbors

Fixed Boundary Methods

Pros

- Weights can be computed for every interior vertex even if neighbors not coplanar or more than 3 vertex neighbors
- Linear complexity

Fixed Boundary Methods

Cons

- High distortion when surface boundary highly non-convex
- Often no "natural" way of distributing parameter points along the boundary.

Cons

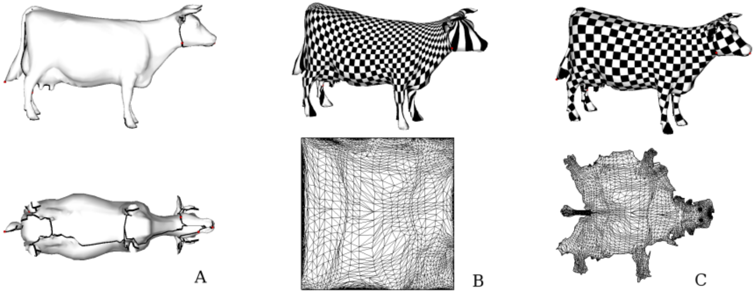


Figure 4.1: A: a mesh cut in a way that makes it homeomorphic to a disk, using the *seamster* algorithm [Sheffer and Hart, 2002]; B: Tutte-Floater parameterization obtained by fixing the border on a square; C: parameterization obtained with a free-boundary parameterization [Sheffer and de Sturler, 2001].

Fixed Boundary Methods

Workarounds

- **Virtual boundary:** augment 3D boundary with extra triangles (Lee et al. 2002)

Fixed Boundary Methods

Workarounds

- **Virtual boundary:** augment 3D boundary with extra triangles (Lee et al. 2002)
- **Scaffolding:** similar idea, but iteratively remeshes virtual boundary based on some distortion energy (Jiang et al. 2017)

Fixed Boundary Methods

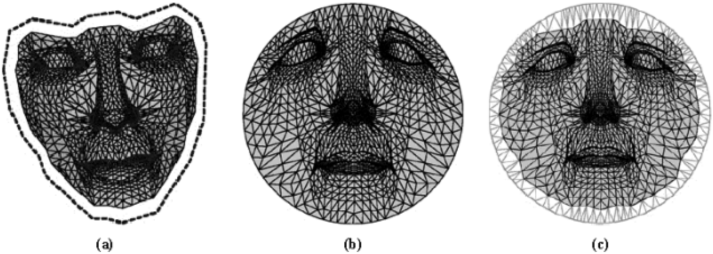


Fig. 3.6 (a) Adding a virtual boundary to the original mesh. (b) Shape Preserving [32] parameterization of the original mesh. (c) Parameterization of the original mesh and its virtual boundary [74]. The virtual boundary vertices are fixed, allowing the real boundary vertices to move.

Free Boundary Methods

LSCM. (Levy et al. 2002) The least squares conformal maps method seeks to minimize the following **conformal energy**

$$E_{LSCM} = E_C = \frac{1}{2}\int_S ||f_v - \text{rot}_{90}(f_u X)||^2 dp = \frac{(\sigma_1 - \sigma_2)^2}{2}$$

Free Boundary Methods

LSCM. (Levy et al. 2002) The least squares conformal maps method seeks to minimize the following **conformal energy**

$$E_{LSCM} = E_C = \frac{1}{2}\int_S ||f_v - \mathrm{rot}_{90}(f_u X)||^2 dp = \frac{(\sigma_1 - \sigma_2)^2}{2}$$

Intuition: the gradient vectors f_u and f_v are **orthogonal** and **have the same norm**.

Free Boundary Methods

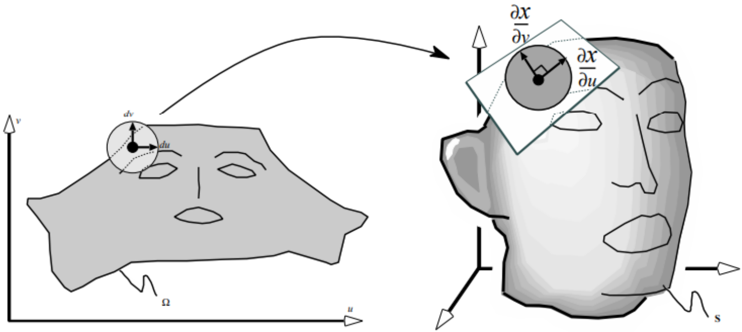


Figure 4.9: A conformal parameterization transforms an elementary circle into an elementary circle.

Free Boundary Methods

DCP. (Desbrun et al 2002) Discrete conformal parameterization minimizes the **dirichlet energy**.

Definition

Given a parameterization $f : \Omega \subset \mathbb{R}^2 \rightarrow S \subset R^3$, the **Dirichlet energy** measures the integral of the squared norm of the gradients.

$$E_D = \frac{1}{2} \int_S ||f_u||^2 + ||f_v||^2 dp$$

Free Boundary Methods

DCP. Discrete conformal parameterization (Desbrun et al 2002) minimizes the **dirichlet energy**.

Definition

Given a parameterization $f : \Omega \subset \mathbb{R}^2 \rightarrow S \subset R^3$, the **Dirichlet energy** measures the integral of the squared norm of the gradients.

$$E_D = \frac{1}{2} \int_S ||f_u||^2 + ||f_v||^2 dp$$

The Dirichlet energy can also be expressed in terms of the singular values σ_1, σ_2 of the Jacobian

$$E_D = \frac{\sigma_1^2 + \sigma_2^2}{2}$$

Free Boundary Methods

Using the singular value definitions, we can easily see that **DCP and LSCM are equivalent methods**.

$$E_D - E_C = \sigma_1 \sigma_2 = \det(J) = \frac{\text{Area}(\Omega)}{\text{Area}(S)}$$

Free Boundary Methods

Using the singular value definitions, we can easily see that **DCP and LSCM are equivalent methods**.

$$E_D - E_C = \sigma_1 \sigma_2 = \det(J) = \frac{\text{Area}(\Omega)}{\text{Area}(S)}$$

Recall: Ω is the parameter domain (2D) and S is the surface (3D). So Dirichlet and conformal energies are the same up to a fixed boundary (choice of pinned vertices) in the parameter domain.

LSCM/DCP. Properties

- Require two pinned vertices to avoid trivial solution (heuristic: two diameter vertices)

Free Boundary Methods

LSCM/DCP. Properties

- Require two pinned vertices to avoid trivial solution (heuristic: two diameter vertices)
- LSCM energy a **flawed** metric: scaled by area of parameter domain (dependent on pinned vertices)

Free Boundary Methods

LSCM/DCP. Properties

- Require two pinned vertices to avoid trivial solution (heuristic: two diameter vertices)
- LSCM energy a **flawed** metric: scaled by area of parameter domain
- **No** guarantee of local or global bijectivity

Free Boundary Methods

LSCM/DCP. Properties

- Require two pinned vertices to avoid trivial solution (heuristic: two diameter vertices)
- LSCM energy a **flawed** metric: scaled by area of parameter domain
- **No** guarantee of local or global bijectivity
- Linear (fast) and empirically lower distortion than fixed boundary methods

Free Boundary Methods

LSCM/DCP. Extensions

- Spectral conformal parameterization (Mullen et al. 2008): find solution to minimizing conformal energy **without** needing to pin vertices

Free Boundary Methods

LSCM/DCP. Extensions

- Spectral conformal parameterization (Mullen et al. 2008): find solution to minimizing conformal energy **without** needing to pin vertices
    - Find Fiedler vector solution u from $L_c u = \lambda B u$

Free Boundary Methods

LSCM/DCP. Extensions

- Spectral conformal parameterization (Mullen et al. 2008): find solution to minimizing conformal energy **without** needing to pin vertices
- Hierarchical LSCM (Ray and Levy 2003): Speed-up using hierarchical solver

MIPS. (Hormann and Greiner 2000) First method to compute natural boundary. Minimizes the *Dirichlet energy per parameter-space area*

$$K_F(J_T) = ||J_T||_F||J_T^{-1}||_F = \frac{\sigma_1^2 + \sigma_2^2}{\sigma_1 \sigma_2} = \frac{\mathrm{trace}(\mathrm{I}_T)}{\det J_T}$$

Free Boundary Methods

MIPS. (Hormann and Greiner 2000) Iteratively move each vertex to reduce energy, checks for flips, and checks for boundary overlaps.

MIPS. Properties

- Nonlinear (slow)
- Global bijectivity

Free Boundary Methods

Angle Based Flattening. (Sheffer and de Sturler 2000) Based on the observation: a planar triangulation is defined by the corner angles of triangles (up to similarity).

Unlike previous methods, problem is defined in **angle space**.

Free Boundary Methods

Angle Based Flattening. (Sheffer and de Sturler 2000) Minimize the objective

$$D(\alpha_i) = \sum_{i=1}^{3T} (\alpha_i - \beta_i)^2$$

where β_i are the known 3D angles and α_i are the unknown 2D angles.

Free Boundary Methods

Angle Based Flattening. (Sheffer and de Sturler 2000) Require constraints on 2D angles for "valid triangulation"

Free Boundary Methods

Angle Based Flattening. (Sheffer and de Sturler 2000) Require constraints on 2D angles for "valid triangulation"

- All angles positive
- Angles in each triangle sum to π
- Sum of angles around each vertex is 2π
- Edges shared by adjacent triangles have same length

Angle Based Flattening. Properties

- Locally bijective (but not global)

Free Boundary Methods

Angle Based Flattening. Properties

- Locally bijective (but not global)
- Non-linear (slow) and unstable for large meshes

Free Boundary Methods

Angle Based Flattening. Extensions

- Zayer et al (2003): Enforce convex boundaries on parameter domain $\Rightarrow$ global bijectivity

Free Boundary Methods

Angle Based Flattening. Extensions

- Zayer et al (2003): Enforce convex boundaries on parameter domain ⇒ global bijectivity

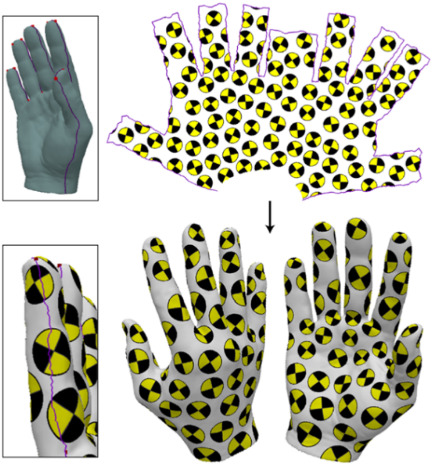
Free Boundary Methods

Angle Based Flattening. Extensions

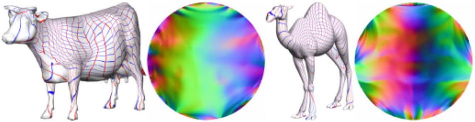
- Zayer et al (2003): Enforce convex boundaries on parameter domain $\Rightarrow$ global bijectivity
- Kharevych et al (2006): Introduce cone singularities $\Rightarrow$ global parameterization. Continuous up to translation and rotation, except at singularities.

Free Boundary Methods

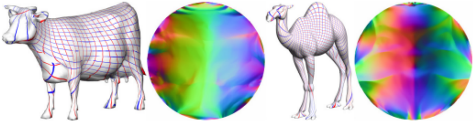
Angle Based Flattening. Extensions



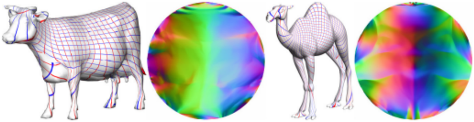
Comparisons



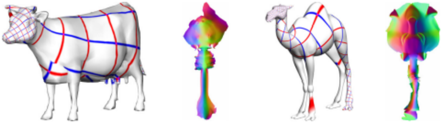
Parameterization with uniform weights [128] on a circular domain.



Parameterization with harmonic weights [28] on a circular domain.

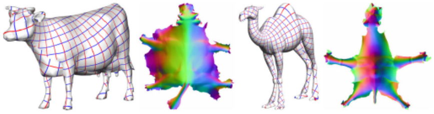


Parameterization with mean value weights [33] on a circular domain.

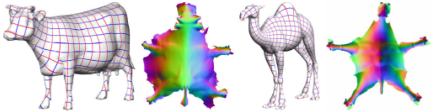


Parameterization with LSCM [79].

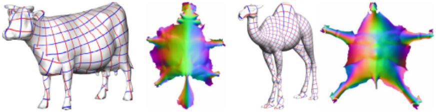
Comparisons



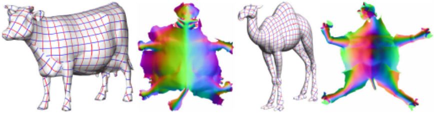
Parameterization with MIPS [54].



Parameterization with ABF++ [118].



Parameterization with circle patterns [62].



Stretch minimizing parameterization [107].

Conclusion

We have **only just scratched the surface** of mesh parameterization methods, and even left out a lot of newer conformal methods.

Conclusion

We have **only just scratched the surface** of mesh parameterization methods, and even left out a lot of newer conformal methods.

- Ricci flows
- Circle packing
- Discrete conformal equivalence
- Cone singularities
- Etc...

Resources

Mesh Parameterization: Theory and Practice (2008)

Mesh Parameterization Methods and Their Applications