

# Implicit Minimal Surfaces for Bijective Correspondences

ETIENNE CORMAN, Université de Lorraine, CNRS, Inria, LORIA, France

YOUSUF SOLIMAN, Side Effects Software, Canada

ROBIN MAGNET, Inria, Université Paris Cité, France

MARK GILLESPIE, Inria, France and University of Utah, USA

We introduce an *implicit* representation of continuous, bijective, orientation-preserving maps between genus zero surfaces with or without boundary. The distortion of these maps can easily be minimized by optimizing the Ginzburg-Landau functional—a ubiquitous model in physics and differential geometry—leading to a simple algorithm for computing bijective correspondences using only standard tools of the tangent vector field toolbox. The method avoids combinatorial mesh modifications and does not require barrier functions to enforce bijectivity making it more robust to noise and simpler to implement. Moreover, the algorithm does not assume a bijective initialization and can untangle non-bijective correspondences generated by computationally cheaper methods such as functional maps. It supports the use of both landmark points and landmark curves to guide the correspondence. The key idea is that a bijection between surfaces defines a two-dimensional mapping surface sitting inside the four-dimensional product space of the two inputs, and this mapping surface can be stored implicitly as the zero set of a *complex section*—essentially a complex function defined on the product space. Now the distortion of the map can be optimized by minimizing the area of this mapping surface, which amounts to minimizing the Ginzburg-Landau functional of the complex section. We demonstrate the practical benefits of our method by comparing to state-of-the-art correspondence algorithms and show that our implicit representation offers improved stability and naturally supports constraints that are difficult to enforce with explicit map representations.

CCS Concepts: • **Computing methodologies** → **Shape analysis**; • **Mathematics of computing** → *Partial differential equations*.

## ACM Reference Format:

Etienne Corman, Yousuf Soliman, Robin Magnet, and Mark Gillespie. 2026. Implicit Minimal Surfaces for Bijective Correspondences. *ACM Trans. Graph.* 45, 4, Article 162 (July 2026), 20 pages. <https://doi.org/10.1145/3811368>

## 1 Introduction

Computing a map between two surfaces  $A$  and  $B$  is a ubiquitous problem in geometry processing. It is a prerequisite for analyzing data across collections of shapes and is essential for transferring data such as textures, segmentations, or other semantic attributes, which are crucial to a wide range of applications from studying

Authors' Contact Information: [Etienne Corman](mailto:etienne.corman@cnrs.fr), [etienne.corman@cnrs.fr](mailto:etienne.corman@cnrs.fr), Université de Lorraine, CNRS, Inria, LORIA, Vandœuvre-lès-Nancy, France; [Yousuf Soliman](mailto:yousufsoliman@sidefx.com), [yousufsoliman@sidefx.com](mailto:yousufsoliman@sidefx.com), Side Effects Software, Toronto, Canada; [Robin Magnet](mailto:robin.magnet@inria.fr), [robin.magnet@inria.fr](mailto:robin.magnet@inria.fr), Inria, Université Paris Cité, Paris, France; [Mark Gillespie](mailto:mark.gillespie81@gmail.com), [mark.gillespie81@gmail.com](mailto:mark.gillespie81@gmail.com), Inria, Palaiseau, France and University of Utah, Salt Lake City, USA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2026 ACM.

ACM 1557-7368/2026/7-ART162  
<https://doi.org/10.1145/3811368>

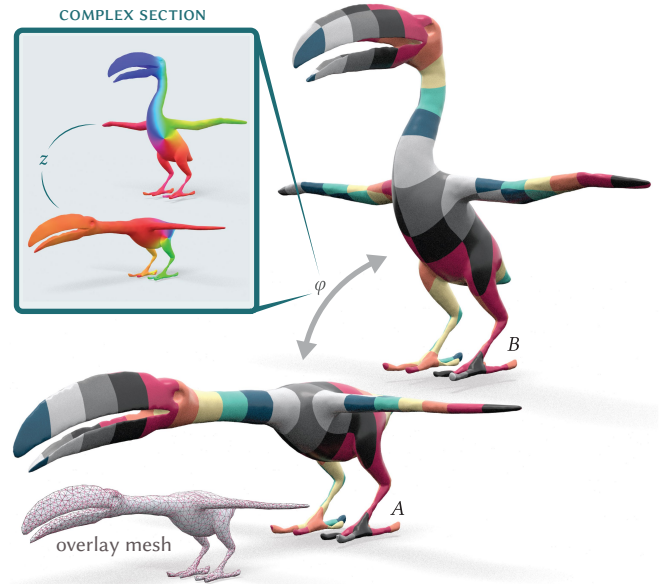


Fig. 1. We introduce a new implicit representation of maps  $\varphi : A \rightarrow B$  between triangle meshes  $A$  and  $B$ , encoding the maps as the zero sets of complex functions  $z$  on the product space  $A \times B$ . In this representation we can compute high-quality orientation-preserving bijections between  $A$  and  $B$  by minimizing a simple Ginzburg-Landau energy, without requiring any combinatorial mesh modifications, barrier functions, or a bijective initialization. Once an implicit map has been computed, it encodes not only the vertex map, but also the entire overlay mesh under the correspondence.

brain-fold variations in medical imaging to reconstructing morphological correspondences in paleontology, and many more tasks across computer graphics, computer vision, and scientific modeling.

When the surfaces share the same topological class, the goal is typically to find a map that is bijective, continuous, and aligns salient geometric or semantic features—often formalized by looking for as-isometric-as possible maps. Yet it remains highly challenging to compute such maps: the underlying distortion energies are non-convex, while continuity and bijectivity are strict constraints that are difficult to enforce during optimization.

In practice, even describing an exact map between two triangle meshes is cumbersome, yet it is necessary for evaluating a map's distortion. For every point on  $A$ , the map must specify its image on  $B$ , not only for the vertices of  $A$  but also for all interior points of its faces. The most common approach is to describe the map by *remeshing* both surfaces so that they share identical connectivity, either by intersecting the meshes [[Schmidt et al. 2020](#)] or by replacing them altogether [[Schmidt et al. 2023](#)]. However, this representation is

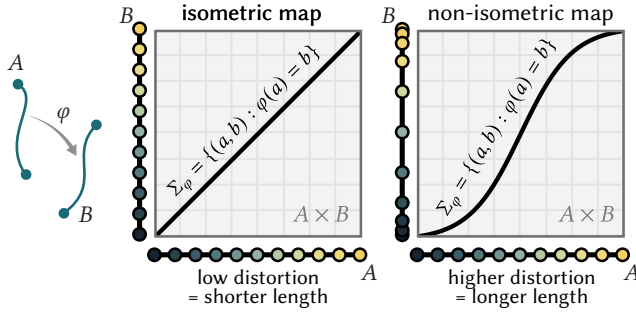


Fig. 2. To illustrate the idea behind our method, we move down a dimension and consider a map  $\varphi : A \rightarrow B$  between two curves (left). Such a map can be represented by its “graph”, i.e. the set of identified points  $\Sigma_\varphi = \{(a, b) : \varphi(a) = b\}$  inside the product space  $A \times B$  (center, right). The more that  $\varphi$  distorts the curves, the longer its graph becomes—thus, we can compute high-quality maps by minimizing the length of the graph.

deeply coupled to the mesh connectivity, whereas the map itself is a smooth, coordinate-invariant geometric object. Instead, we propose a new representation of correspondences that avoids combinatorial modifications entirely and relies only on well-established tools for processing tangent vector fields (Figure 1).

*Mappings as Surfaces in Product Space.* Any map  $\varphi : A \rightarrow B$  carves out a two-dimensional manifold  $\Sigma_\varphi$  from the product space  $A \times B$ :

$$\Sigma_\varphi := \{(p, \varphi(p)) : p \in A\}. \quad (1)$$

When  $\varphi$  is continuous and bijective each point  $p \in A$  is mapped to a unique point  $\varphi(p) \in B$ , and so the projection of  $\Sigma_\varphi$  onto  $B$  covers  $B$  exactly once. Conversely, if a two-dimensional submanifold  $\Sigma \subset A \times B$  has projections onto each input surface which cover that surface exactly once, then  $\Sigma$  represents a continuous bijection between  $A$  and  $B$ . Moreover, if the projections are orientation-preserving, then the bijection will also preserve orientation (Figure 5). And if  $\Sigma$  has minimal area, it represents a distortion-minimizing map (Figure 3). (For the mathematical expression of this distortion, and its relation to other well known distortion measures, see Section 1.1.)

We can illustrate this idea for the simpler problem of matching curves (Figure 2). In this setting, the product space is a rectangle, and a map is represented by the graph of the function  $a \mapsto \varphi(a)$ . This representation enables a direct measure of the distortion of the map: the length of the graph. The identity map, represented by a straight line, minimizes distortion, while any other mapping necessarily increases both the distortion and the length of the graph.

We thus shift the focus: rather than working with map  $\varphi$  directly, we instead compute a surface in the product manifold  $A \times B$  which implicitly encodes a distortion-minimizing correspondence.

*Minimal Surfaces and Complex Fields.* Now that we have framed the matching problem using minimal surfaces, a central question remains: *How can we effectively represent and compute a minimal surface in a four-dimensional manifold?* Fortunately a substantial body of literature addresses this problem using tools remarkably similar to those used in geometry processing. A key insight is that minimal surfaces can be characterized through the zero set of a

vector-field like object known as a *complex section*. In geometry processing, we often use the property that the zeros of a smooth vector field on a surface lie at isolated points, and the zeros of complex functions in 3D space lie along curves. Similarly, the zeros of a complex section on a 4D space lie on a two-dimensional subset.

This leads to our core representation: a map  $\varphi : A \rightarrow B$  is encoded as a complex-valued section  $z$  on the product space  $A \times B$  which is zero on the mapping surface: so  $z(a, b) = 0$  if and only if  $b = \varphi(a)$ .

The bijectivity of  $\varphi$  can also be expressed via the field  $z$ . For each fixed  $p \in A$ , the restriction  $b \mapsto z(p, b)$  must vanish at exactly one point  $b \in B$ , and vice versa for each fixed  $q \in B$ . This condition is incorporated into a special choice of connection on  $A$  and  $B$ , akin to the standard treatment of cross fields (see Section 3.1). Strictly speaking, this connection does not force the output map to be bijective—it only imposes the relaxed constraint that the net number of signed zeros is equal to one for each restriction. But we find that in practice this relaxed condition suffices to compute high-quality maps (Section 5).

Using this representation, the computation of a minimal surface becomes accessible. Minimizers of the Ginzburg-Landau functional

$$\mathcal{GL}_\varepsilon(z) := \int_{A \times B} \left( \frac{1}{2} |\nabla z|^2 + \frac{1}{4\varepsilon^2} (1 - |z|^2)^2 \right) \text{vol}_{A \times B}, \quad (2)$$

converge, as  $\varepsilon \rightarrow 0$ , to fields whose zero sets form minimal surfaces. Intuitively, if we were to normalize the field  $z$ , zeros of  $z$  would become singularities where the Dirichlet energy blows up. The parameter  $\varepsilon$  forces the field to approach unit length almost everywhere, leading to a large Dirichlet energy near the zeros. Thus, minimizing Dirichlet energy minimizes the area of the zero set. A more detailed discussion of this phenomenon is provided in Section 3.2.

The behavior of the Ginzburg-Landau functional and related energies on spaces with nontrivial topology is still an active area of mathematical research, so in this work we mostly restrict our attention to surfaces with *sphere-like topology*. However, we discuss the treatment of genus-zero surfaces with boundary in Section 4.7.

*Contributions.* In summary, we introduce a method to compute high quality continuous, bijective, orientation-preserving maps between genus zero surfaces with or without boundary, by minimizing the Ginzburg-Landau energy on a four-dimensional product space. This approach yields several advantages:

- Distortion minimization is achieved without barrier terms that prohibit non-bijective configurations, resulting in improved stability and simpler optimization;
- The method does not require a bijective initial map and can robustly “untangle” invalid or overlapping correspondences;
- In addition to landmark points, the algorithm naturally supports landmark curves, allowing points to slide along feature lines without explicit parameterization;
- The formulation is fully intrinsic, but preserves orientation;
- No combinatorial modifications to the input meshes are needed;
- The implementation relies solely on widely used operators from the tangent vector field processing toolbox. Leveraging the tensor product structure, we avoid constructing dense operators on the full product space.

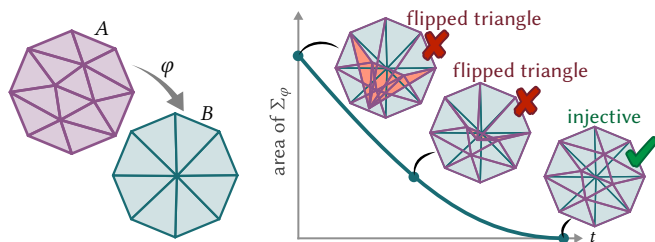


Fig. 3. As in the case of curves, a map  $\varphi$  between surfaces  $A$  and  $B$  defines a surface  $\Sigma_\varphi$  in the product space whose area encodes the distortion of  $\varphi$ . In particular, maps which flip triangles have larger surface area.

### 1.1 Area Minimization and Metric Distortion

We now give a brief analysis of our distortion measure, the area of the “graph”  $\Sigma_\varphi$  associated to a map  $\varphi$  (Equation 1). In Appendix A, we express the area of  $\Sigma_\varphi$  using the singular values  $\sigma_1, \sigma_2$  of  $d\varphi$ :

$$\text{Area}(\Sigma_\varphi) = \int_A \sqrt{(1 + \sigma_1^2)(1 + \sigma_2^2)} \text{vol}_A. \quad (3)$$

This area based distortion measure is bounded by the Dirichlet energy<sup>1</sup> and the area distortion<sup>2</sup> of  $\varphi$ , with equality (up to the addition of a constant) if and only if  $\varphi$  is conformal:

$$\int_A |\det d\varphi| \text{vol}_A \leq \text{Area}(\Sigma_\varphi) - \text{Area}(A) \leq \int_A \frac{1}{2} |d\varphi|^2 \text{vol}_A. \quad (4)$$

*Definition 1.1.* A bijective correspondence  $\varphi : A \rightarrow B$  is an *area minimizing correspondence* if  $\Sigma_\varphi$  is a minimal surface.

The bound in Equation 4 also shows that maps with low total harmonic energy [Ezuz et al. 2019b, Eq. 5] have small area, as  $\Sigma_\varphi = \Sigma_{\varphi^{-1}}$ , and thus a similar bound involving the area of  $B$  and the Dirichlet energy of  $\varphi^{-1}$  holds. Averaging the two inequalities shows that the area of  $\Sigma_\varphi$  is controlled by the sum of the Dirichlet energy of  $\varphi$  and its inverse. Other distortion energies, that may also be related, are listed in [Abulnaga et al. 2023, Table 1] and [Poya et al. 2023]. It would also be interesting to relate area minimizing correspondences to Gromov-Wasserstein distances [Mandad et al. 2017].

### 1.2 Bijective Correspondences and Homology

We conclude with a brief comment on the constraints that we will later impose on the mapping surface  $\Sigma_\varphi$ . Topological properties of  $\varphi$ , like orientation preservation and bijectivity, are reflected in the topology of  $\Sigma_\varphi$ . For instance, if we intersect  $\Sigma_\varphi$  with a “vertical” slice at a fixed  $a \in A$ , or a “horizontal” slice at a fixed  $b \in B$ ,

$$\Sigma_\varphi \cap (\{a\} \times B) = \{(a, \varphi(a))\}, \quad \Sigma_\varphi \cap (A \times \{b\}) = \{(\varphi^{-1}(b), b)\},$$

we find a single intersection point. In particular, the projection maps  $\pi_A : \Sigma_\varphi \rightarrow A$  and  $\pi_B : \Sigma_\varphi \rightarrow B$  have degree  $\pm 1$ . The orientation preservation of  $\varphi$  is encoded in the sign of the degree, which in turn is encoded in the homology class of  $\Sigma_\varphi$ . Thus we see that the homology class  $[\Sigma_\varphi] \in H_2(A \times B)$  is not arbitrary—constant maps into  $B$ , for example, induce graphs that are in the same homology class of  $A$  but do not satisfy the correct degree constraint for  $\pi_B$ .

<sup>1</sup>  $\int_A \text{Area}(\sigma) \leq 1 + \frac{1}{2}(\sigma_1^2 + \sigma_2^2)$  follows from the elementary inequality  $xy \leq \frac{1}{2}(x^2 + y^2)$   
<sup>2</sup>  $\int_A \text{Area}(\sigma) \geq 1 + |\sigma_1 \sigma_2|$  follows from the elementary inequality  $(1 + x^2)(1 + y^2) \geq (1 + |xy|)^2$

The homology class of surfaces  $\Sigma_\varphi$  arising from bijections  $\varphi$  is described explicitly in Milnor and Stasheff [1974, Theorem 11.11, p.128] and is known as the *diagonal homology class*  $[\Delta]$  in  $A \times B$  since under the identification  $A \cong B$  induced by  $\varphi$ , the graph  $\Sigma_\varphi$  is the diagonal in  $A \times A$ . When  $A$  and  $B$  are simply-connected, the intersection of  $\Sigma_\varphi$  with horizontal and vertical slices completely determines its homology class. In Section 3.4.6 we explain how we constrain the topology of  $\Sigma_\varphi$  to agree with  $[\Delta]$  on slices.

## 2 Related Work

Computing mappings between surfaces is a long-standing challenge, and many algorithms have been proposed to address it. Our method is distinguished from existing alternatives by a combination of properties: it minimizes isometric distortion while guaranteeing orientation preservation, though it does not strictly enforce bijectivity on discrete domains. Below we contrast it to several alternative families of approaches, but a comprehensive review lies beyond the scope of this paper—see the surveys by van Kaick et al. [2011]; Tam et al. [2012] and Sahillioğlu [2020] for broader context.

### 2.1 Relaxed Map Representations

*Registration.* One approach to surface mapping is to deform one surface to match the other. While these methods simplify distortion analysis [Beg et al. 2005; Sharf et al. 2006; Huang et al. 2008; Li et al. 2008b; Tam et al. 2012; Eisenberger et al. 2019, 2020], they offer limited guarantees of bijectivity. Moreover, obtaining a close fit between the surfaces can involve expensive optimization, and aligning shape features often requires user intervention.

Ezuz et al. [2019a,b] minimizes the harmonic energy using vertex coordinates as variables. Unlike our method, their mapping remains undefined within triangles. However, like ours, it supports arbitrary initialization and can untangle correspondences.

*Higher-Dimensional Relaxation.* Many works relax the bijectivity constraint by embedding the problem in higher-dimensional spaces, though this may introduce discontinuities, suboptimal distortion minimization, or orientation violations. Kantorovich’s optimal transport formulation replaces bijections with probability measures, sacrificing bijectivity and continuity. Mandad et al. [2017] and Brifault et al. [2025] mitigate the lack of continuity by regularizing the transport plan. Functional maps [Ovsjanikov et al. 2012, 2017] represent mappings as linear operators acting on function spaces. However, converting these to continuous, bijective maps remains challenging [Melzi et al. 2019; Ren et al. 2021], and orientation preservation is non-trivial in the presence of intrinsic symmetries [Ren et al. 2018, 2020; Donati et al. 2022]. Unlike exact bijective representations, functional maps enable partial mappings [Litany et al. 2016]. Solomon et al. [2012] frames correspondences as probabilistic measures, but it is subject to the same limitations. Vestner et al. [2017] share our premise of computing maps as surfaces in a product space, noting that minimal surfaces reduce distortion—though without formal characterization for surfaces. Their method approximates  $\Sigma_\varphi$  via kernel density estimation, reducing the problem to linear assignment. Consequently, their mapping is vertex-defined—with no guarantee of orientation preservation—and relies on search over the space of all possible vertex matchings.

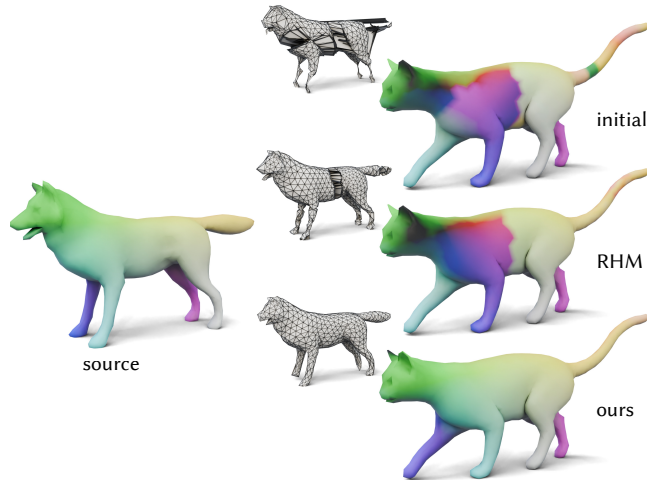


Fig. 4. The maps computed by our algorithm are constrained to preserve normal orientation of the base surface. Intrinsic algorithms (such as functional maps) often fail to distinguish between symmetric parts. Here, we initialize with a map computed via [Ren et al. 2018] that incorrectly swaps the left and right front legs. While RHM [Ezuz et al. 2019b] improves smoothness, it fails to resolve the orientation reversal. In contrast, our approach successfully untangles the map, visualized by both color and geometry transfer.

## 2.2 Strictly Bijective Maps

*Mapping via Common Domains.* A provably bijective map can be obtained by composing two bijective maps from surfaces into a common reference domain. Methods leveraging Tutte embeddings ensure bijectivity by mapping into simple reference domains: the plane [Kanai et al. 1997; Litke et al. 2005; Aigerman et al. 2014, 2015], periodic tilings [Aigerman and Lipman 2015], the sphere [Aigerman et al. 2017; Baden et al. 2018], or hyperbolic space [Tsui et al. 2013; Aigerman and Lipman 2016; Shi et al. 2016]. But distortion is typically controlled only for the surface-to-domain mapping, except for conformal maps [Li et al. 2008a; Baden et al. 2018] whose composition preserves conformality. Other work has attempted to address these limitations by mapping to geometrically closer domains [Schreiner et al. 2004], blending multiple conformal maps [Kim et al. 2011], or computing mesh intersections in the common domain to optimize distortion [Schmidt et al. 2019, 2020]. The latter achieves high accuracy but is computationally intensive and often converges to suboptimal solutions. Morreale et al. [2021] proposed using neural networks to encode surfaces, which alleviates some of the piecewise-linear limitation of existing mesh-based methods, but their method remains limited to pairs of surfaces which are both mapped into a common domain.

*Remeshing.* A straightforward approach to encode correspondences is to approximate both surfaces with a mesh of shared connectivity. Distortion minimization then reduces to a remeshing problem, often solved via coarse-to-fine refinement [Michikawa et al. 2001; Peng and Timalena 2016]. Early methods prioritized mesh approximation error [Kraevoy and Sheffer 2004; Yang et al. 2020], while later works explicitly minimized distortion using volumetric mappings [Yang et al. 2018], intrinsic remeshing [Takayama 2022],

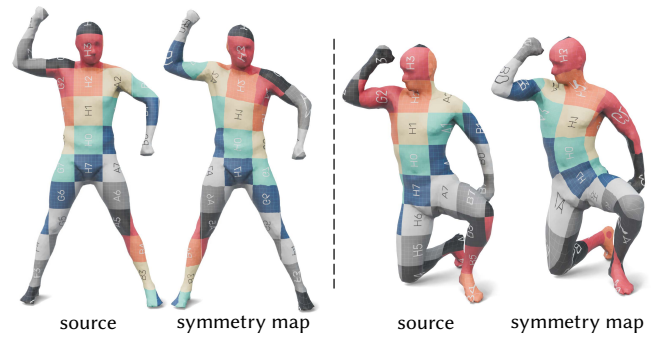


Fig. 5. Our method preserves mapping orientation by encoding surface orientation in the connection form. This enables the computation of a symmetric map from an object to itself by simply inverting the normals of the target domain.

or spherical parameterizations [Schmidt et al. 2023]. These methods guarantee bijectivity and orientation preservation but require bijective initialization, precluding untangling.

*Explicit Product Space.* Another line of work enforces geometric consistency by explicitly searching for a submanifold in the discrete product space. Windheuser et al. [2011a,b] initially formulated the search for an orientation-preserving diffeomorphism as an integer program on the product triangulation. Recent work have attempted to improve the scalability of this approach with novel formulations and solvers [Roetzer et al. 2022; Amrani et al. 2025; Roetzer and Bernard 2025]. However, these methods fundamentally need to define the connectivity of the product graph with massive constraint matrices, leading to prohibitive memory requirements. The resulting map is furthermore only combinatorial and doesn't define a continuous mapping within the faces.

In contrast, our method never explicitly meshes the product space, even though the implicit formulation encodes the geometric consistency constraints in the topology of the field's connection.

## 2.3 Surface Reconstruction in Higher Dimensions

Our algorithm builds a 2-dimensional surface embedded in a 4-dimensional space. Kohlbrenner et al. [2023] address the broader challenge of reconstructing a manifold from point samples in higher dimensions, but our setting is dramatically simplified because the surface of interest is defined directly as the zero set of an implicit function—not approximated from a sparse points set.

## 2.4 Minimal Surfaces

In the mathematical literature, the problem of computing minimal surfaces is often relaxed to computing minimal “currents” instead, which yields a convex relaxation of the problem in the space of generalized surfaces defined by Federer and Fleming [1960]. For a computational introduction to the theory, see Wang and Chern [2021]. This approach has been used in computer graphics in applications ranging from neural surface modeling [Palmer et al. 2022] to quad meshing [Palmer et al. 2024]. Unfortunately, when dealing with surfaces of codimension greater than 1, this relaxation is no

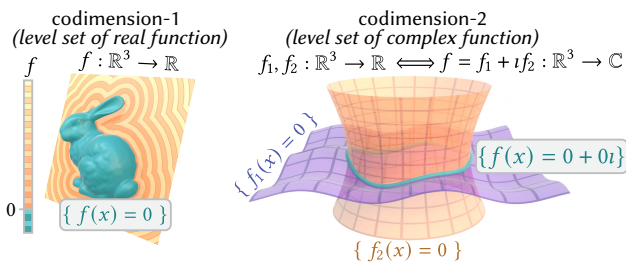


Fig. 6. *Left:* A codimension-1 object (like a 2D surface in 3D space) can be encoded as the zero level set of a real-valued function  $f$ . *Right:* A codimension-2 object (like a 1D curve in 3D space, or a 2D surface in 4D space) can be encoded as the shared zero level set of a pair of real functions  $f_1, f_2$ . Equivalently, a codimension-2 object is the zero level set of a single *complex* function  $f(x) = f_1(x) + i f_2(x)$ . This simple idea forms the basis of our implicit complex line bundle encoding (Section 3.1).

longer tight. Indeed, even when considering two-dimensional surfaces bounded by a one-dimensional loop in  $\mathbb{R}^4$ , the relaxation to currents can already produce currents which have a lower area than any valid surface [Young 1963; De Lellis 2014].

In the past twenty years, zero sets of critical points of the Ginzburg-Landau energy have been shown to be related to codimension-2 minimal surfaces [Jerrard and Soner 2002; Alberti et al. 2005; De Philippis and Pigati 2024; Canevari et al. 2023], with very recent work showing that a generalization of the Ginzburg-Landau energy known as the *self-dual Yang-Mills-Higgs functional*<sup>3</sup> provides a phase-field approximation of the codimension-2 area functional [Pigati and Stern 2021; Parise et al. 2024]. These results are formulated in the language of complex line bundles, which provides the tools to generalize codimension-1 implicit surfaces represented as the level sets of real valued functions to codimension-2 implicit surfaces represented as the zero sets of complex sections with prescribed topology. This makes complex line bundles the perfect tool for our problem.

### 3 Background

In Sections 3.1 and 3.2 we explain more background about the underlying smooth mathematical theory, and we review the discretizations that we use in Sections 3.3 and 3.4. But the practically-minded reader can jump to Section 4 for a concrete description of our algorithm.

#### 3.1 Implicit Representation via Complex Line Bundles

Our algorithm searches for a two-dimensional surface belonging to a specified homology class inside of a four-dimensional space (Figure 6). Here we describe how these surfaces can be represented using mathematical objects known as *complex line bundles*.

*Warmup.* As a simpler example, consider a codimension-two subsets of a surface  $S$ , *i.e.* sets of oriented points. We could try to represent these points as the shared zero level set of a pair of functions  $f_1, f_2 : S \rightarrow \mathbb{R}$ , or equivalently as the zero level set of a single complex function  $f(x) = f_1(x) + i f_2(x)$ . But not every collection of points can be represented as such a zero set. For instance, if  $f$  is a projection

<sup>3</sup>it additionally includes a connection as a free variable of the optimization, penalizing the  $L^2$ -norm of its curvature in addition to the usual Ginzburg-Landau energy

onto the  $xy$ -plane then its zero level set is the set of signed intersections between the surface and a ray shot up from the origin. As shown in the inset, such a point set will always have even size, and adding up the sign of each intersection point always yields a sum of 0. Indeed, the same is true for all complex functions, not just projections: for any smooth function  $f : S \rightarrow \mathbb{C}$ , the signs of the zeros sum to 0.

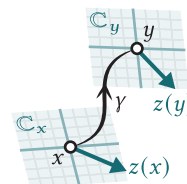
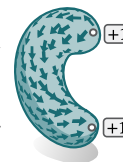
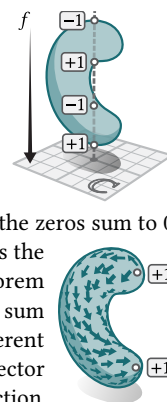
Alternatively, we could encode our point set as the zeros of a vector field. The Poincaré-Hopf theorem then guarantees that the indices of the zeros must sum to the Euler characteristic  $\chi(S)$ , providing a different constraint on our zero set. So even though any vector field can *locally* be represented as a complex function, the space of vector fields and the space of complex functions have different global structures—and different constraints on their zeros.

The space of vector fields and space of complex functions and are both examples of *complex line bundles*. Formally, a complex line bundle on a manifold  $M$  is a space which associates a copy  $\mathbb{C}_x$  of the complex plane to each point  $x \in M$ , and locally looks like the product  $M \times \mathbb{C}$ . But its global topological structure may be different.

The analogue of a vector field on a general complex line bundle is a smooth mapping sending points  $x \in M$  to complex values  $z(x) \in \mathbb{C}_x$ . Such mappings are known as *sections*. If  $M$  is a  $d$ -dimensional manifold, then the zero set of a section is an oriented submanifold of dimension  $d - 2$ . Just as the zeros of a complex function or vector field sum to a fixed constant, the zeros of any section of a complex line bundle on  $M$  always lie in a particular *homology class* of  $H_{d-2}(M; \mathbb{Z})$ , determined by the bundle’s curvature.

*Connections and Curvature.* Sections can be studied using *connections*, just as in the case of vector fields. A connection  $\nabla$  on a complex line bundle is a differential operator which provides a way of “parallel transporting” a value  $z(x) \in \mathbb{C}_x$  along a path  $\gamma$  from  $x$  to  $y$  to obtain a value  $z(y) \in \mathbb{C}_y$ . However, parallel transport along different paths can result in different values in  $\mathbb{C}_y$ . Indeed, parallel transporting  $z(x)$  along a closed loop may produce a different value  $z$  upon returning to  $x$ . The failure to close up is quantified by the curvature 2-form  $\Omega^\nabla$ . See [Milnor and Stasheff 1974, Appendix C] for more details.

*Chern Classes.* The curvature 2-form is particularly important because it characterizes the topological class of the line bundle—and thus the homology class of our zero sets. The first Chern class of a complex line bundle is the cohomology class  $c_1 := [\frac{1}{2\pi} \Omega^\nabla] \in H^2(M)$ . It encodes the homology class of our zero sets in the following sense: for any closed 2-dimensional subset  $S \subseteq M$ , the net number of signed intersections with the  $(d-2)$ -dimensional zero set of a generic section is precisely  $\frac{1}{2\pi} \int_S \Omega^\nabla$ . Remarkably, these values are always integers, and do not depend on the specific section or connection that we started from. The curvature of *any* connection on the complex line bundle always lies in the same cohomology class. So by making an appropriate choice of complex line bundle structure we can control the topological class of the zero set of any section and can thus enforce the homology constraints of Section 1.2 by construction.



*Connection Laplacians.* Just as the ordinary Laplacian measures the smoothness of scalar functions, connections can also be used to measure the smoothness of a section  $z$  on a complex line bundle. The connection Dirichlet energy

$$\mathcal{E}^D(z) = \frac{1}{2} \int_M |\nabla z|^2 \text{vol}_M \quad (5)$$

then gives rise to the *connection Laplacian*  $L^\nabla$  as the gradient of this Dirichlet energy. When  $M = A \times B$  is the product of two surfaces  $A$  and  $B$ , the Dirichlet energy can be expressed in a factorized form

$$\mathcal{E}^D(z) = \int_A \left( \int_B |\nabla z|_B^2 \text{vol}_B \right) \text{vol}_A + \int_B \left( \int_A |\nabla z|_A^2 \text{vol}_A \right) \text{vol}_B, \quad (6)$$

where  $|\nabla z|_A^2$  (resp.,  $|\nabla z|_B^2$ ) denotes the norm of the covariant derivative of the section restricted to  $A$  (resp.,  $B$ ).

*Zero Set Distribution.* The placement of zeros of a smooth section in a complex line bundle is very closely related to the curvature of the connection. Experiments show that in low energy states of the connection Laplacian, zeros concentrate in regions of high curvature [Knöppel et al. 2013; Weißmann et al. 2014]. Recent theoretical work supports this intuition by showing that when you start with random sections whose zeros are distributed according to the curvature, the corresponding heat flow will connect these random zeros to the ground state zero set [Nicolaescu and Savale 2017]. We use this intuition to design line bundles where curvature directs the zero set, producing high quality and controllable correspondences.

### 3.2 Implicit Area Minimization

The Ginzburg-Landau functional provides a way of minimizing the area of a surface encoded via a complex section, reducing the problem of bijective mapping to the computation of smooth sections on the product mesh. We briefly describe the smooth theory here, and later provide the discrete algorithm in Section 4.2.

*Allen-Cahn Energy.* Before introducing the full functional used in our method, we review the main ideas in the simpler codimension-1 case. Here a surface is represented as the zero set of a real-valued function  $u : M \rightarrow \mathbb{R}$  and we consider the Allen-Cahn energy

$$\mathcal{A}C_\varepsilon(u) := \int_M \frac{1}{2} |du|^2 + \frac{1}{4\varepsilon^2} (1 - u^2)^2 \text{vol}_M. \quad (7)$$

The main contribution of the energy comes from the double well potential on the right which is minimized when  $u \equiv \pm 1$ . The Dirichlet energy, on the other hand, prevents discontinuous jumps from a region where  $u \equiv +1$  to a region where  $u \equiv -1$ . However, in the limit as  $\varepsilon \rightarrow 0$  the minimizers  $u_\varepsilon$  converge to function that has a jump across a minimal surface [Modica and Mortola 1977].

*Ginzburg-Landau Energy.* The Ginzburg-Landau functional on a complex line bundle closely resembles the Allen-Cahn equation

$$\mathcal{G}L_\varepsilon(z) := \int_M \frac{1}{2} |\nabla z|^2 + \frac{1}{4\varepsilon^2} (1 - |z|^2)^2 \text{vol}_M, \quad (8)$$

The unit norm penalty is now minimized when  $z \in S^1$ , and so we call it the *circular well potential*. In the  $\varepsilon \rightarrow 0^+$  limit minimizers produce  $S^1$ -valued harmonic maps away from a codimension-2 set of zeros. Compared to the scalar valued case, the relationship between the

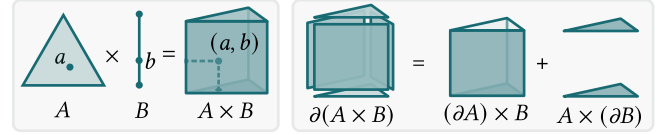


Fig. 7. *Left:* Geometrically, the product space  $A \times B$  is the result of extruding  $A$  along  $B$ . Algebraically, points of  $A \times B$  are pairs  $(a, b)$  for  $a \in A$  and  $b \in B$ . *Right:* The boundary operator obeys a product rule on product cells.

Ginzburg-Landau functional and the area of its zero set is more subtle [Parise et al. 2024], but in certain situations the zeros of minimizers are known to form minimal surfaces [Lin and Riviere 1999; Canevari et al. 2023]. The intuition to keep in mind is (1) that  $\varepsilon$  controls the interface width between the zero set and the set where  $|z| \approx 1$  and (2) since the energy blows up as  $\varepsilon \rightarrow 0^+$  the blow up should happen on a set that is area minimizing.

### 3.3 Cell Complexes and Product Meshes

In our setting, the domain  $M = A \times B$  is the product of two surfaces. This section introduces the construction of a discrete product space, and the next section presents the discretization of complex line bundles required to evaluate the discrete Ginzburg-Landau energy.

Mathematically, a  $d$ -dimensional mesh is an object called a *cell complex*. Just as a polygon mesh is made up of zero-dimensional vertices, one-dimensional edges, and two-dimensional faces, a cell complex  $M = (M^0, M^1, \dots, M^d)$  is made up of  $k$ -dimensional  $k$ -cells  $\sigma_i^k \in M^k$  for all  $k = 0, \dots, d$ . When the index  $k$  is unnecessary we follow the usual convention for meshes and call the vertex set  $V_M := M^0$ , the edge set  $E_M := M^1$ , and the face set  $F_M := M^2$ .

The boundary  $\partial\sigma_i^k$  of a  $k$ -cell  $\sigma_i^k$  is a collection of oriented  $(k-1)$ -cells, which we write as a formal sum  $\partial\sigma_i^k = \sum_j \sigma_j^{k-1}$ . The boundary operator can be written as a matrix  $\partial_k \in \mathbb{Z}^{M^{k-1} \times M^k}$  where  $(\partial_k)_{j,i}$  is 1 if  $\sigma_j^{k-1}$  appears in  $\partial\sigma_i^k$  with positive orientation,  $-1$  if it appears with negative orientation, and zero otherwise. The transpose of these boundary operators  $d_{k-1} := \partial_k^T$  are the discrete exterior derivatives in discrete exterior calculus [Desbrun et al. 2005], and we denote the space discrete differential  $k$ -forms by  $\Omega^k(M; \mathbb{R})$ .

*Product Spaces.* Given two topological spaces  $A$  and  $B$ , the product  $A \times B$  is the set of ordered pairs  $(a, b)$  for  $a \in A$  and  $b \in B$  (Figure 7). Similarly, if we have a pair of triangle meshes  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$ , their product space is a four-dimensional cell complex whose cells are products of cells from  $A$  and  $B$ . Its vertex set is precisely the set  $V_{A \times B} = V_A \times V_B$  of pairs of vertices, while its edge set is  $E_{A \times B} = E_A \times V_B \cup V_A \times E_B$ , its face set is  $F_{A \times B} = V_A \times F_B \cup E_A \times E_B \cup F_A \times V_B$ . In general its set of  $k$ -cells is

$$(A \times B)^k = \bigcup_{i+j=k} A^i \times B^j. \quad (9)$$

The boundary operator on  $A \times B$  obeys a product rule (Figure 7)

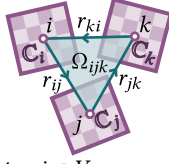
$$\partial(\sigma_i^k \times \sigma_j^l) = (\partial\sigma_i^k) \times \sigma_j^l + (-1)^k \sigma_i^k \times (\partial\sigma_j^l), \quad (10)$$

which we can write in matrix notation as

$$\partial_k^{A \times B} = \sum_{i+j=k} \partial_i^A \otimes \mathbb{I}_j^B + (-1)^i \mathbb{I}_i^A \otimes \partial_j^B. \quad (11)$$

### 3.4 Discrete Complex Line Bundles

We use the discrete complex line bundles with connections defined by Knöppel and Pinkall [2016]. A discrete complex line bundle on a cell complex  $M = (V, E, F, \dots)$  consists of:



- (1) a copy  $\mathbb{C}_i$  of the complex plane for each vertex  $i \in V$ ,
- (2) a connection  $r_{ij} \in \mathbb{C}$  of unit norm for each oriented edge  $ij \in E$ , satisfying  $r_{ji} = r_{ij}^{-1}$ . These complex numbers act as discrete parallel transport maps  $r_{ij} : \mathbb{C}_i \rightarrow \mathbb{C}_j$ ,
- (3) a curvature  $\Omega_{ijk} \in \mathbb{R}$  for each face  $ijk \in F$ , that satisfies  $d\Omega = 0$  (trivially true on a surface), and is compatible with the connection: for each face  $ijk \in F$ , we have  $r_{ki}r_{jk}r_{ij} = \exp(i\Omega_{ijk})$ .

Just as in the smooth setting, the topological class of a discrete complex line bundle (and the implicit surfaces it can represent) are determined by the curvature 2-form  $[\frac{1}{2\pi}\Omega] \in H^2(M)$ . Since the parallel transport maps only determine  $\Omega$  modulo  $2\pi$ , we have the freedom to modify the topological class of the bundle without changing the connection—in Section 4.1, we use this ability to construct an appropriate bundle for our mapping problem.

Often, a discrete connection is represented a rotation angle  $\rho_{ij}$  for each oriented edge, with the parallel transport maps  $r_{ij} = e^{i\rho_{ij}}$ .

**Sections and Zeros.** A discrete section  $z$  is an assignment of a complex number  $z_i \in \mathbb{C}_i$  to each vertex  $i$ . Given a section  $z$ , the rotation of  $z$  along edge  $ij$  is measured by the *angular one-form*

$$\omega_{ij}^z := \arg\left(\frac{z_j}{r_{ij}z_i}\right) \in [-\pi, \pi). \quad (12)$$

If  $\omega_{ij}^z = -\pi$ , then  $z$  has a zero along edge  $ij$ . In the generic case where  $\omega_{ij}^z \in (-\pi, \pi)$ , Knöppel and Pinkall define the index 2-form

$$\text{ind}^z := \frac{1}{2\pi}(d\omega^z + \Omega), \quad (13)$$

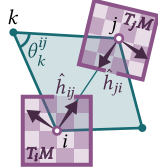
and prove that  $\text{ind}_{ijk}^z$  is always an integer, which gives the sum of the indices of all zeros within face  $ijk$ . In contexts where the section  $z$  is clear, we drop the superscript and refer to  $\omega_{ij}$  and  $\text{ind}_{ijk}$ .

Note that  $[\text{ind}^z] \equiv [\frac{1}{2\pi}\Omega]$  as cohomology classes in  $H^2(M; \mathbb{R})$ , as  $d\omega^z$  is a closed discrete 2-form. Since  $\text{ind}^z$  encodes the zeros of  $z$ , we conclude that for any section  $z$  of the bundle, the zero set always lies in the same homology class—just as in the smooth setting.

**3.4.1 The Discrete Levi-Civita Connection.** One important example of a discrete complex line bundle is the discrete Levi-Civita connection on the tangent bundle of a mesh  $M$ , which Knöppel et al. [2013] construct by “flattening” a neighborhood of each vertex as follows: let  $\theta_i^{jk}$  be the corner angle of triangle  $ijk$  at vertex  $i$ , and let  $\Theta_i := \sum_{ijk > i} \theta_i^{jk}$ . Multiplying the angles around  $i$  by a factor of  $\frac{2\pi}{\Theta_i}$  yields scaled angles  $\tilde{\theta}_i^{jk} := \frac{2\pi}{\Theta_i} \theta_i^{jk}$  that sum to  $2\pi$ , allowing us to identify the tangent space of vertex  $i$  with the complex plane  $T_iM \cong \mathbb{C}$ . Then each halfedge  $ij$  leaving vertex  $i$  can be assigned a unit complex number  $\hat{h}_{ij} \in T_iM$  giving its direction in this tangent space. The discrete Levi-Civita connection maps  $T_iM$  to  $T_jM$  while preserving the direction of edge  $ij$ , which can be written

$$r_{ij}^{LC} := -\hat{h}_{ji}\hat{h}_{ij}^{-1}, \quad (14)$$

after choosing any coordinates on  $T_iM$  and  $T_jM$ .



To complete the complex line bundle, we assign curvature to each face  $ijk$  based on its rescaled corner angles

$$\Omega_{ijk}^{LC} = \tilde{\theta}_i^{jk} + \tilde{\theta}_j^{ki} + \tilde{\theta}_k^{ij} - \pi. \quad (15)$$

These curvatures are compatible with  $r^{LC}$  and satisfy the Gauss-Bonnet theorem  $\sum \Omega_{ijk}^{LC} = 2\pi\chi(M)$ .

**3.4.2 Trivial Connections.** Starting from any initial connection  $r^0$  with curvature  $\Omega^0$ , we can modify the curvature by solving a Poisson equation. If we are given another discrete 2-form  $\Omega \in \Omega^2(M; \mathbb{R})$  on a simply connected triangle mesh  $M$  that has the same integral, i.e.,

$$\sum_{ijk \in F} \Omega_{ijk}^0 = \sum_{ijk \in F} \Omega_{ijk}, \quad (16)$$

we can find a discrete 1-form  $\alpha \in \Omega^1(M; \mathbb{R})$  that satisfies

$$d_1\alpha = \Omega - \Omega^0. \quad (17)$$

Using the Hodge decomposition, this amounts to solving a Poisson equation on the dual mesh<sup>4</sup>. The discrete connection

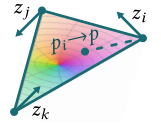
$$r_{ij} = e^{i\alpha_{ij}} r_{ij}^0 \quad (18)$$

is then compatible with the prescribed curvature 2-form  $\Omega$ . If we concentrate all of the prescribed curvature on a finite number of points this is the trivial connections algorithm of Crane et al. [2010]. We use this procedure in Section 4.1 to construct a connection with the correct curvature on each input surface, and in Section 4.4 to compute an initialization from a noisy correspondence.

**3.4.3 Finite Element Space.** To define continuous maps between triangle meshes, we need to be able to evaluate a section  $z$  not only at mesh vertices, but also inside triangles. To do so, we use the finite element space for complex line bundles on surfaces constructed by Knöppel and Pinkall [2016] and Liu et al. [2016]. In our setting, the basis function for vertex  $i$  within triangle  $ijk$  can be expressed as:

$$\phi_i(p) = b_i(p) \exp\left[-i\left(\int_{p_i \rightarrow p} \rho_{ijk}\right)\right], \quad (19)$$

where  $b_i(p)$  is the barycentric coordinate of point  $p$  with respect to vertex  $i$ ,  $\rho_{ijk}$  is the connection 1-form interpolated using Whitney interpolation [Whitney 1957; Desbrun et al. 2005], and  $p_i \rightarrow p$  is the straight line connecting  $i$  to  $p$ .



Using this finite element space on a triangle mesh  $S$ , Knöppel et al. [2013, §6.1.1] give discretizations of the connection Laplacian  $L_S^\nabla \in \mathbb{C}^{V \times V}$  and the mass matrix  $M_S^\nabla \in \mathbb{C}^{V \times V}$ , which depend on the parallel transport maps  $r_{ij}$  and face curvatures  $\Omega_{ijk}$ . We describe the full constructions in the supplementary material.

**3.4.4 Product Space Finite Elements.** Recall that the complex field  $z : A \times B \rightarrow \mathbb{C}$  is defined at each vertex and stored as a complex matrix  $Z$  of dimensions  $|V_A| \times |V_B|$ . In order to interpolate these values over the 4-cells which make up the product space  $A \times B$ , we define basis functions by taking tensor products of the basis functions  $\phi^A$  and  $\phi^B$  defined on surfaces  $A$  and  $B$  respectively.

<sup>4</sup>letting  $L^- = d_1 *^{-1} d_1^\top$  we have that  $\alpha = *^{-1} d_1^\top \beta$  where  $\beta$  solves  $L^- \beta = \Omega - \Omega^0$

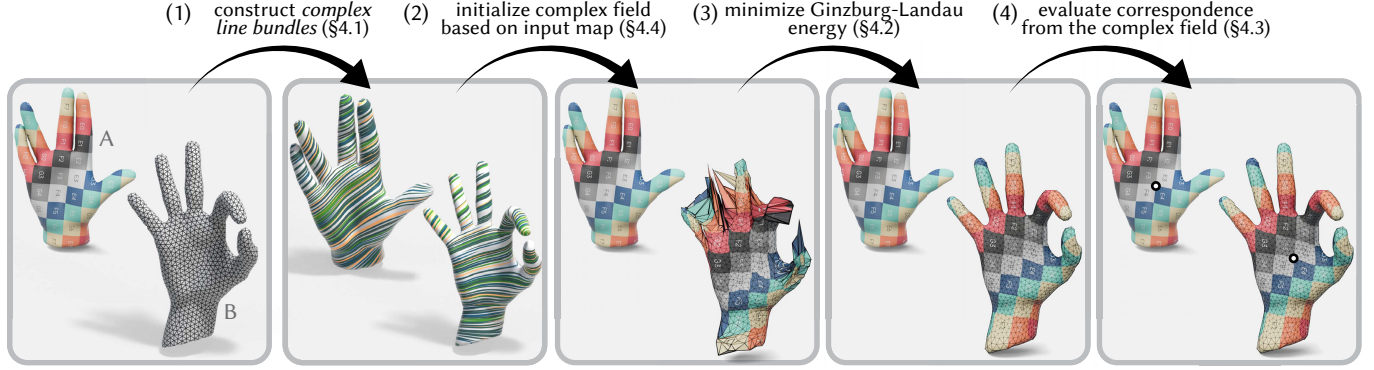


Fig. 8. Our algorithm proceeds in four steps. We begin with a pair of meshes  $A$  and  $B$  (left). In step (1), we compute *complex line bundle* structures on  $A$  and  $B$ , which together define a complex line bundle structure on the product space  $A \times B$  (Section 4.1). In step (2), we initialize our complex field based on a (potentially low quality) input map (Section 4.4). In step (3) we optimize the map by minimizing the discrete Ginzburg-Landau energy of our field (Section 4.2). And finally in step (4) we evaluate the mapping at any desired points by identifying the zero set of  $z$  when interpolated via finite elements.

Any 4-cell  $\sigma_{ijk,abc}^4 \in F_{A \times B}$  is the product of triangles  $ijk \in F_A$  and  $abc \in F_B$ . The interpolated field at a point  $(p, q) \in \sigma_{ijk,abc}^4$  is:

$$z(p, q) = \sum_{\substack{u \in \{i,j,k\} \\ v \in \{a,b,c\}}} Z_{u,v} \phi_u^A(p) \phi_v^B(q). \quad (20)$$

Note that if we fix  $q$  and allow  $p$  to vary, we recover the standard surface interpolant on  $A$  up to a global rotation.

This interpolation scheme enables the definition of a connection Laplacian and the precise localization of zeros within each cell.

**3.4.5 Connection Laplacian.** We define a connection Laplacian on the product space by discretizing the Dirichlet energy (Equation 6). The discrete Dirichlet energy of a discrete section  $Z \in \mathbb{C}^{V_A \times V_B}$  on  $A \times B$  can be written in terms of left and right multiplication by the connection Laplacian and mass matrix defined on each manifold:

$$\mathcal{E}^D(Z) = \frac{1}{2} \langle Z, L_A^\nabla Z (M_B^\nabla)^\top \rangle_{\mathbb{C}} + \frac{1}{2} \langle Z, M_A^\nabla Z (L_B^\nabla)^\top \rangle_{\mathbb{C}}. \quad (21)$$

Here we write  $\bullet^\top$  for the ordinary transpose,  $\bullet^\dagger$  for the conjugate transpose, and use the inner product  $\langle A, B \rangle_{\mathbb{C}} = \text{Re tr}[A^\dagger B]$  for complex matrices. For a full derivation, see Appendix B.

Similarly, the connection mass matrices  $M_A^\nabla \in \mathbb{C}^{V_A \times V_A}$ ,  $M_B^\nabla \in \mathbb{C}^{V_B \times V_B}$  determine the  $L^2$  energy of product space sections via:

$$\int_{A \times B} |z|^2 \text{vol}_{A \times B} = \langle Z, M_A^\nabla Z (M_B^\nabla)^\top \rangle_{\mathbb{C}}, \quad (22)$$

and the scalar mass matrices  $M_A \in \mathbb{R}^{V_A \times V_A}$ ,  $M_B \in \mathbb{R}^{V_B \times V_B}$  determine the  $L^2$  energy of real functions  $F \in \mathbb{R}^{V_A \times V_B}$  on  $A \times B$ :

$$\int_{A \times B} f(a, b)^2 \text{vol}_{A \times B} = \langle F, M_A F (M_B)^\top \rangle_{\mathbb{R}}. \quad (23)$$

**3.4.6 Encoding the Correct Homology Class.** In order to compute bijections between  $A$  and  $B$ , we must prescribe the right curvature on the product space  $A \times B$ . Recall that the map  $\varphi : A \rightarrow B$  is defined implicitly: for any vertex  $i \in V_A$ , the image  $\varphi(i)$  is the zero of the section  $z$  restricted to the slice  $\{i\} \times B$ . This defines a section  $z^{(i)}$  on  $B$  where  $z^{(i)}(p) = z(i, p)$  for any point  $p \in B$ . Since vertex  $i$  should map to a single point on  $B$ , this restricted section must

contain exactly one zero. Recalling that the total number of signed zeros on a the slice is  $\frac{1}{2\pi} \int \Omega$  (Section 3.1), this implies the curvature of the bundle restricted to any copy of  $B$  must integrate to  $2\pi$ . By symmetry, the curvature of the bundle restricted to any slice on  $A$  must also integrate to  $2\pi$ . To satisfy these constraints we start with two separate base connections on  $A$  and  $B$ , with curvatures  $\Omega^A$  and  $\Omega^B$  each integrating to  $2\pi$ , and we combine them to form a product connection on  $A \times B$ . The curvature associated to this resulting connection restricts to  $\Omega^A$  on any horizontal slice and to  $\Omega^B$  on any vertical slice, as desired. Formally, when  $A$  and  $B$  are topological spheres, this product connection and curvature precisely encode the diagonal class  $[\Delta]$ , which contains all bijections (Section 1.2).

## 4 Algorithm

As illustrated in Figure 8, our algorithm for computing correspondences has four key steps:

- (1) Design a discrete connection on  $A \times B$  such that each slice  $A$  and  $B$  of the product mesh contains one zero (Section 4.1).
- (2) Initialize the complex field  $Z$  from an input map (Section 4.4).
- (3) Minimize the discrete Ginzburg-Landau energy of  $Z$  (Section 4.2).
- (4) Evaluate the correspondences by identifying the zero set of  $z$  after interpolation using a finite element basis (Section 4.3).

To extend the algorithm's applicability, we introduce several straightforward modifications: support for surfaces with boundaries (Section 4.7), support for point or curve landmarks (Section 4.6), and a simple coarse-to-fine acceleration strategy (Section 4.5).

See the supplemental material for comprehensive pseudocode.

**Notation.** Throughout,  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$  are closed genus zero triangle meshes, with edge lengths  $\ell_A : E_A \rightarrow \mathbb{R}$  and  $\ell_B : E_B \rightarrow \mathbb{R}$ . We write the discrete section on  $A \times B$  as a matrix  $Z \in \mathbb{C}^{V_A \times V_B}$ . We write  $\bullet^\top$  for the ordinary transpose,  $\bullet^\dagger$  for the conjugate transpose, and use the inner products  $\langle A, B \rangle_{\mathbb{C}} = \text{Re tr}[A^\dagger B]$  for complex matrices, and  $\text{tr}[A^\top B]$  for real matrices. On each mesh  $S$  we let  $L_S^\nabla, M_S^\nabla \in \mathbb{C}^{V_S \times V_S}$  be the connection Laplacian and mass matrix for complex sections on  $S$  (Section 3.4.5), and let  $M_S \in \mathbb{R}^{V_S \times V_S}$  be the real diagonal mass matrix for scalar functions.

### 4.1 Constructing Surface Connections

We begin by finding connections on  $A$  and  $B$  with total curvature  $2\pi$ , which will allow us to find sections on  $A$  or  $B$  with a single zero—see [Section 3.4.6](#) for more discussion of the curvature constraint.

We can construct such a connection on each mesh by taking advantage of the  $2\pi$  ambiguity in the curvature of a discrete complex line bundle. We start with the trivial bundle  $\mathbb{C}_i := \mathbb{C}$  and trivial parallel transport maps  $r_{ij}^0 := 1$ . We set the initial curvature  $\Omega^0$  to zero everywhere except on an arbitrarily chosen face  $f \in F$  where we set  $\Omega_f^0 := 2\pi$ . This is a discretization of the skyscraper bundle [[Knöppel 2020](#), §7.2], which has the correct total curvature, but contains no geometric information. To obtain a more meaningful connection, we solve a linear system as in [Section 3.4.2](#) to obtain a curvature of half the Gaussian curvature  $\Omega_{ijk}^{LC}$  on each face ([Equation 15](#)), i.e. we solve for a new connection  $r_{ij}$  with curvature  $\Omega_{ijk} = \frac{1}{2}\Omega_{ijk}^{LC}$ .

Together, these connections on  $A$  and  $B$  implicitly define a product connection on  $A \times B$ . Fortunately, as we will see below, all of the relevant expressions factor into terms depending only on  $A$  or  $B$ , so we do not have to assemble this full product-space connection.

**REMARK 1.** Any connection with the same curvature can in principle be used instead. In [Appendix D](#) we give two alternative constructions more directly related to tangent vectors and the Levi-Civita connection.

### 4.2 Discrete Ginzburg-Landau Minimization

We discretize the Ginzburg-Landau energy, [Equation 8](#), via a finite-element Dirichlet energy and a lumped discretization of the circular well penalty term. This energy can be minimized with L-BFGS using the expressions for the energy and gradient provided below.

*Dirichlet Energy.* As explained in [Section 3.4.5](#), the Dirichlet energy on  $A \times B$  can be assembled from the connection Laplacians and mass matrices of  $A$  and  $B$  as follows

$$\mathcal{E}^D(Z) = \frac{1}{2} \left\langle Z \overline{M}_B^\nabla, L_A^\nabla Z \right\rangle_{\mathbb{C}} + \frac{1}{2} \left\langle \overline{M}_A^\nabla Z, Z L_B^\nabla \right\rangle_{\mathbb{C}}. \quad (24)$$

*Circular Well Potential.* We discretize the penalty term at the vertices, weighted by their barycentric dual volumes in  $A \times B$ :

$$\mathcal{W}(Z) = \sum_{(i,j) \in V_A \times V_B} (1 - |Z_{i,j}|^2)^2 M_{i,i} M_{j,j}. \quad (25)$$

To write  $\mathcal{W}$  in matrix notation, we define the norm deviation matrix  $U \in \mathbb{R}^{V_A \times V_B}$  by  $U_{i,j} = |Z_{i,j}|^2 - 1$ , so that  $\mathcal{W}(Z) = \langle U, M_A U M_B^\top \rangle_{\mathbb{R}}$ . This vertex-based discretization minimizes the stencil size—and computational cost—of function and gradient evaluations.

*Ginzburg-Landau Energy.* The discrete energy of a section  $Z$  is

$$\begin{aligned} \mathcal{GL}_\lambda(Z) &:= \mathcal{E}^D(Z) + \frac{\lambda}{4} \mathcal{W}(Z) \\ &= \frac{1}{2} \left\langle Z \overline{M}_B^\nabla, L_A^\nabla Z \right\rangle_{\mathbb{C}} + \frac{1}{2} \left\langle \overline{M}_A^\nabla Z, Z L_B^\nabla \right\rangle_{\mathbb{C}} \\ &\quad + \frac{\lambda}{4} \langle U M_B, M_A U \rangle_{\mathbb{R}}, \end{aligned} \quad (26)$$

where the variable  $\lambda$  is called the *Ginzburg-Landau parameter*. Its gradient can also be written as the following matrix in  $\mathbb{C}^{V_A \times V_B}$ :

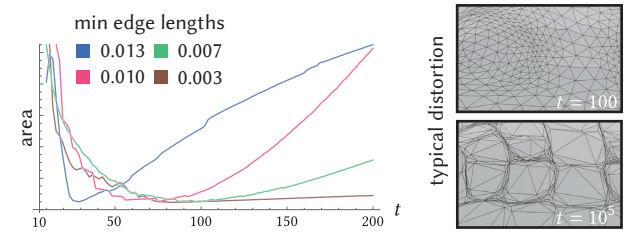
$$\nabla_Z \mathcal{GL}_\lambda = L_A^\nabla Z \left( M_B^\nabla \right)^\top + M_A^\nabla Z \left( L_B^\nabla \right)^\top + \lambda \left( M_A U M_B^\top \right) \odot Z, \quad (27)$$

where  $\odot$  denotes the element-wise product.

The choice of Ginzburg-Landau parameter significantly influences the quality of the correspondence. If  $\lambda$  is below the minimal eigenvalue of  $L_{A \times B}^\nabla$ , then the only critical point of the energy is  $z \equiv 0$  ([Lemma C.1](#)). But if  $\lambda$  is too large, the circular well potential repels zeros away from the vertices. And since the Dirichlet term carries a comparatively low weight, the field  $z$  tends to remain close to its initial value during optimization.

One simple strategy is to set  $\lambda$  relative to the smallest eigenvalue  $\lambda_{A \times B}$  of  $L_{A \times B}^\nabla$ . In the smooth setting, this eigenvalue decomposes as  $\lambda_{A \times B} = \lambda_A + \lambda_B$ , where  $\lambda_A$  and  $\lambda_B$  are the smallest eigenvalues of the connection Laplacians on  $A$  and  $B$ . While this equality only holds approximately in the discrete setting due to differing mass matrix discretizations (see [Appendix C](#) for details), it remains a good estimate. Experimentally, we find  $\lambda \approx 100\lambda_{A \times B}$  provides near-optimal results across a variety of meshes ([Figure 9](#)).

For challenging initializations, starting from a smaller value of  $\lambda$  is often necessary to smooth the initial correspondences and match surface features correctly. For example, in [Figures 4](#) and [23](#), the optimization is performed in two stages: first, with  $\lambda = 10\lambda_{A \times B}$  to align large-scale features, and second, with  $\lambda = 100\lambda_{A \times B}$  to refine the correspondences and address high-frequency matching.



**Fig. 9.** An experimental search for the optimal choice of Ginzburg-Landau parameter  $\lambda = t \lambda_0$  where  $\lambda_0 = \lambda_A + \lambda_B$ . Since the area depends on the distortion between the surfaces, to consistently analyze the optimal parameter choice we normalize the area by the maximal area measured for each example separately. Large values of  $\lambda$  cannot produce smooth correspondences unless the mesh has a sufficiently high resolution, as indicated by having a small minimum edge length.

### 4.3 Evaluating the Correspondence

By construction ([Section 4.1](#)), for any vertex  $v \in V_B$ , the  $v$ -th column of  $Z$  is a section  $z^{(v)}$  on  $A$ —simply denoted  $z$  below—with a single zero (see [Figure 10](#)). To locate this zero, we first compute the integer-valued index 2-form  $\text{ind}^z$  using [Equation 13](#). This form takes a non-zero value on exactly one face  $ijk \in F_A$ , where the zero lies.

We then compute the barycentric coordinates  $(b_j, b_k)$  of the zero within  $ijk$  ([Section 3.4.3](#)) by solving the following system:

$$(1 - b_j - b_k) |z_i| + b_j |z_j| e^{i(b_k \Omega + \omega_{ij})} + b_k |z_k| e^{-i(b_j \Omega + \omega_{ki})} = 0. \quad (28)$$

Directly solving [Equation 28](#) with Newton's method is unlikely to yield the solution within triangle  $ijk$ . Instead, we adopt a homotopy continuation approach, and deform the geometry from a flat triangle ( $t = 0$ ) to the target curvature  $\Omega_{ijk}$  ( $t = 1$ ), tracking the zero as it moves. We use Newton's method to find the zero at each time step ([Figure 11](#)). Pseudocode is provided in the supplementary material.

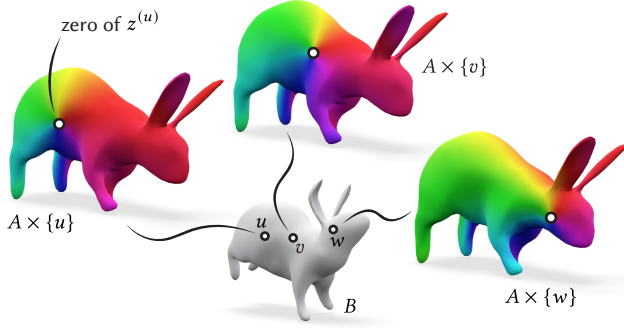


Fig. 10. To evaluate the map at a vertex  $u \in V_B$ , we take the  $u$ 'th column of  $Z$  as a section  $z^{(u)}$  on  $A$ . We can visualize this section as a complex function in a chart that covers all but a single face of  $A$ . Now the image  $\varphi(u)$  of vertex  $u$  is given by the location of the zero on  $A$ .

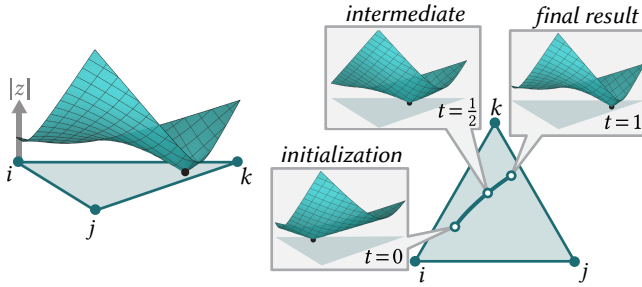


Fig. 11. *Left*: In order to evaluate the correspondence, we have to find a root of the interpolated section  $z$  within some triangle  $ijk$ . But the interpolant is nonlinear and non-convex, so solving directly with Newton's method might not find the desired root. *Right*: Instead, we interpolate the face curvature from flat at  $t = 0$ —in which case the interpolant becomes linear—to the full curvature  $\Omega_{ijk}$  at  $t = 1$ , solving with Newton's method at each time step.

We stop the zero from “escaping” by ensuring that the zero never hits an edge while interpolating, i.e.  $\omega_{ij}^z(t), \omega_{jk}^z(t), \omega_{ki}^z(t) \in (-\pi, \pi)$  for  $t \in [0, 1)$ . This is achieved via the following linear interpolation:

$$\omega_{ij}^z(t) = \omega_{ij}^z + (1-t)\frac{1}{3} \left( \Omega_{ijk} - 2\omega_{ij}^z + \omega_{jk}^z + \omega_{ki}^z \right) \quad (29)$$

$$\omega_{jk}^z(t) = \omega_{jk}^z + (1-t)\frac{1}{3} \left( \Omega_{ijk} + \omega_{ij}^z - 2\omega_{jk}^z + \omega_{ki}^z \right) \quad (30)$$

$$\omega_{ki}^z(t) = \omega_{ki}^z + (1-t)\frac{1}{3} \left( \Omega_{ijk} + \omega_{ij}^z + \omega_{jk}^z - 2\omega_{ki}^z \right) \quad (31)$$

$$\Omega_{ijk}(t) = t\Omega_{ijk} \quad (32)$$

One can check that  $\omega_{ij}^z(t) \in (-\pi, \pi)$  for  $0 \leq t < 1$  and that the index stays fixed, so the zero must remain inside triangle  $ijk$ :

$$\forall t, \quad \omega_{ij}^z(t) + \omega_{jk}^z(t) + \omega_{ki}^z(t) + \Omega_{ijk}(t) = 2\pi \text{ind}_{ijk}^z. \quad (33)$$

**Edge-Edge Intersections.** The section encodes not only vertex locations, but also the entire embedding of the edges (Figures 1 and 12). The intersection of an edge of  $A$  and an edge of  $B$  may be calculated by finding a zero inside an “edge-edge” face of the product space. Since these faces have zero curvature, the zero can be found by solving a single quadratic equation, which is presented in Appendix E.

**Evaluation Inside a Face.** We can also evaluate the correspondence for any point  $p$  within triangle  $ijk \in F_B$  using the basis functions in Section 3.4.3. Using the complex weights  $\phi_\bullet(p)$  for each vertex  $i, j, k$  of  $B$  given by Equation 19, we define a section  $z^{(p)}$  on  $A$  as the linear combination of the columns of  $Z$  corresponding to vertices  $i, j, k$ . Solving for its zero as above yields the image of  $p$  on  $A$ .

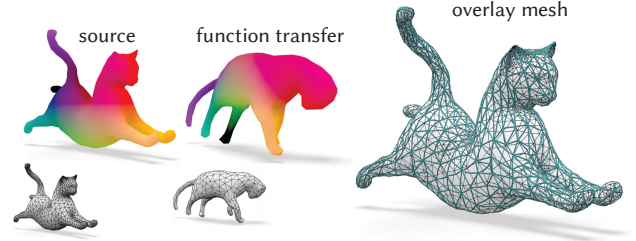


Fig. 12. Our algorithm produces bijective and continuous mappings even when the meshes exhibit substantially different connectivity and sampling. The zero set of the field may place multiple vertices of  $A$  within the same face of mesh  $B$ , while preserving low distortion and bijectivity.

#### 4.4 Initialization

Our method can start from a (potentially noisy) input function given as vertex-face maps  $\varphi : V_A \rightarrow F_B$  and  $\psi : V_B \rightarrow F_A$ . To use these in our algorithm, we must convert them into an initial complex section  $z : A \times B \rightarrow \mathbb{C}$  whose zero set approximates the graphs of  $\varphi$  and  $\psi$ .

**Strategy.** We use the principle that low-energy states of a connection Laplacian concentrate their zeros in regions of high curvature (Section 3.1). In particular, we construct a connection  $r^{\varphi, \psi}$  on  $A \times B$  that concentrates all its curvature exactly along the graphs of  $\varphi$  and  $\psi$ . We then take our initial section  $z$  to be the smallest eigenvector of the associated Laplacian. As this bundle lies in the same topological class as the bundle from Section 4.1, this section serves as a valid initialization for the following Ginzburg-Landau minimization.

**Construction.** We build  $r^{\varphi, \psi}$  slice-by-slice using the trivial connections algorithm from Section 3.4.2. For each vertex  $i_A \in V_A$ , we compute a connection  $r^{i_A}$  on  $B$  that concentrates  $2\pi$  curvature into the single face  $\varphi(i_A)$ . Then, for every edge of the form  $i_A \times e_B$  in the product space, we assign  $r_{i_A \times e_B}^{\varphi, \psi} = r_{e_B}^{i_A}$ . Symmetrically, for each  $i_B \in V_B$ , we run trivial connections on  $A$ , concentrating curvature in face  $\psi(i_B)$  and assign this connection to product space edges of the form  $e_A \times i_B$ . This fully defines a connection on the product space whose curvature is concentrated to the graph of the input maps. The smallest eigenvector of the associated Laplacian acts as a good representation of the input correspondences (see Figure 19).

**Laplacian.** The above connection does not factor as the product of independent connections on  $A$  and  $B$ , so it would normally require assembling the full four-dimensional finite element Laplacian. Instead, we consider a simpler discretization of the Dirichlet energy using a lumped discretization for the outer integrals of Equation 6 while retaining a finite-element discretization for the inner integrals over the 2D slices. Explicitly, for every vertex  $i_A \in V_A$ , let  $L_B^{V, i_A}$  be

the finite element Laplacian of  $r^{i_A}$  on  $B$ , and let  $z^{(i_A)}$  be the restriction of section  $Z$  to the slice. Using similar notation for the slices fixed at  $i_B$ , we write the Dirichlet energy of the connection  $r^{\varphi, \psi}$  as:

$$\mathcal{E}_{\varphi, \psi}^D(Z) = \sum_{v \in V_A} (M_A)_{v,v} \langle L_B^{\nabla, v} z^{(v)}, z^{(v)} \rangle_{\mathbb{C}} + \sum_{v \in V_B} (M_B)_{v,v} \langle L_A^{\nabla, v} z^{(v)}, z^{(v)} \rangle_{\mathbb{C}}, \quad (34)$$

and so the blocks in the product-space connection Laplacian are given by the surface connection Laplacians on the individual slices weighted by the dual area of the vertex on the complementary mesh.

The smallest eigenvalue can then be computed via an iterative matrix-free solver such as LOBPCG [Knyazev et al. 2007].

*Initialization from Distributions.* Rather than starting from a map  $\varphi : V_A \rightarrow F_B$ , one can start from *distributions* on  $B$  associated to the vertices of  $A$ , provided e.g. by a functional map or optimal transport plan. The procedure is almost exactly the same: the only change is that for each vertex of  $A$  we spread curvature over  $B$  proportional to its density instead of concentrating all curvature in one face  $\varphi(i_A)$ .

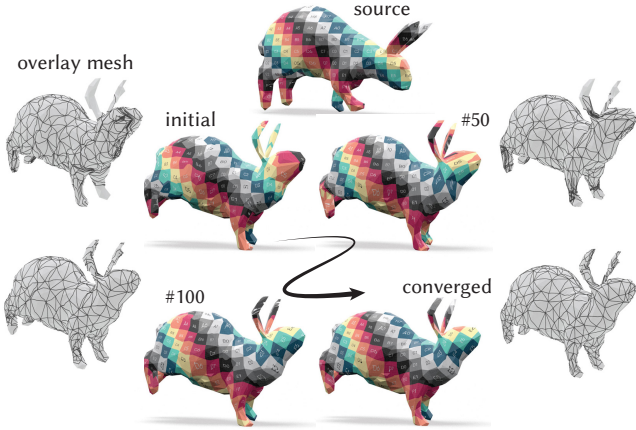


Fig. 13. High quality correspondences can be computed even on very coarse tessellations. On this example, with  $|V_A| = 474$  and  $|V_B| = 502$ , the solution is computed in 21 seconds. The robustness to tessellation density is a consequence of the finite-element representation of the implicit surface: the complex section  $z$  smoothly extends to the interior of the faces and does not require fine sampling to represent a continuous bijection.

#### 4.5 Multiresolution Hierarchy

Our method can be trivially accelerated using a multiresolution hierarchy of meshes. We first compute an optimal complex section  $z^{(0)}$  on coarsened meshes  $A^{(0)}$  and  $B^{(0)}$  by minimizing the Ginzburg-Landau energy. To transfer this solution to fine meshes  $A$  and  $B$ , we distinguish between two cases depending on the bundle structure.

*Direct Upsampling.* For the surface connections constructed in Section 4.1, the fibers  $\mathbb{C}_i$  are globally identified with the complex plane. This allows us to transfer the complex values of the section directly without explicit change of basis (see Figure 14). We upsample  $z^{(0)}$  by closest point projection: for every vertex pair  $(i_A, i_B)$  in

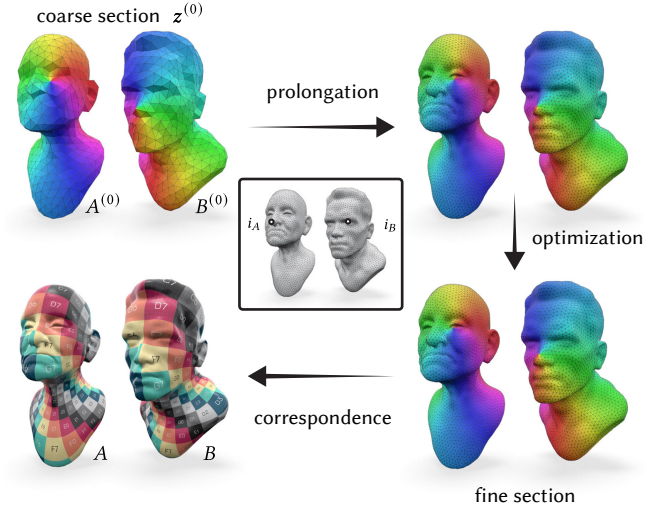


Fig. 14. Our implicit representation of the map can be prolonged to a finer mesh by closest point interpolation of the complex section. Thus we can initialize our optimization from a coarse solution, which already provides a very good approximation of the desired map. To visualize the sections, we plot the complex values on the slices  $\{i_A\} \times B$  and  $A \times \{i_B\}$ , and to visualize the correspondence we transfer a conformal parameterization from  $B$ .

the fine product mesh  $A \times B$ , we find the closest geometric points  $p_A \in A^{(0)}$  and  $p_B \in B^{(0)}$  and evaluate  $z^{(0)}$  at  $(p_A, p_B)$ , using interpolation defined in Section 3.4.4. This upsampled section is then used as initialization for the fine-scale Ginzburg-Landau minimization, or for the iterative eigenvector solver. In practice, this can provide a 12× speedup in computation time (Section 6). Note that for this transfer to be as accurate as possible, the face selected in Section 4.1, which concentrates the curvature in the coarse mesh, should contain the corresponding face in the fine mesh.

*Geometric Initialization.* When using a more general connection, e.g. based on tangent vector fields (Appendix D), direct transfer of complex values might not be possible. In these cases, we instead transfer the map directly: we first extract the coarse correspondences  $\varphi^{(0)}$  and  $\psi^{(0)}$  from the section  $Z^{(0)}$ , as described in Section 4.3. We then construct vertex-to-face maps between the fine meshes  $A$  and  $B$  by composing these coarse correspondences with closest point projections, and initialize the fine level as in Section 4.4.

#### 4.6 Landmarks

The Ginzburg-Landau energy can be modified to accommodate a user-specified set of landmarks  $\mathcal{L} = \{(l_i^A, l_i^B)\}_{i=1}^P$  (Figure 15). Since the map sends a point  $a \in A$  to  $b \in B$  if and only if  $z(a, b) = 0$ , we can encourage this behavior by penalizing non-zero values at the landmarks. Hence, we consider the specifications as soft-constraints.

We replace the standard circular well potential with a spatially varying version, defined by a non-negative potential function  $V : A \times B \rightarrow \mathbb{R}$  having isolated zeros at  $\mathcal{L}$ :

$$\mathcal{W}(Z) := \frac{1}{4} \int_{A \times B} (V(p) - |z(p)|^2)^2 \text{vol}_{A \times B}, \quad (35)$$

We construct  $V$  using Gaussians centered at the landmarks:

$$V(p) := \min_{(p_A, p_B) \in \mathcal{L}} \left( 1 - \exp \left( -\frac{1}{2\sigma_A^2} d_A(p, p_A)^2 - \frac{1}{2\sigma_B^2} d_B(p, p_B)^2 \right) \right), \quad (36)$$

where  $\sigma_A, \sigma_B > 0$  determine the kernel width, and  $d_A(\cdot, \cdot)$  and  $d_B(\cdot, \cdot)$  denote the geodesic distance on  $A$  and  $B$ , respectively. Finer granularity over the pinning potential can be achieved by specifying a landmark dependence choice of  $\sigma_A$  and  $\sigma_B$ .

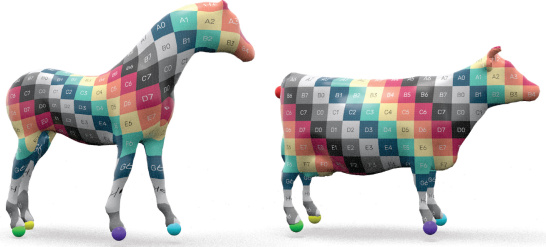


Fig. 15. Landmarks are specified using a pinning potential that has local minima at the specified points in the product mesh.

This modification of the Ginzburg-Landau functional is known as *singularity pinning* in the physics literature<sup>5</sup>, and Aftalion et al. [2001] proved that in the small  $\varepsilon$  limit that zeros of the Ginzburg-Landau minimizers (in a two-dimensional background) are “pinned” at the local minimizers of  $V$ . For this attraction to be effective in practice, the width of the Gaussian kernel must be sufficiently large. Since  $V$  modifies the energy gradient only locally, a zero far outside the well, where  $V \approx 1$ , will not be affected by the potential.

The pinning potential can also be modified to support unparameterized curve to curve correspondences (Figure 16). For a pair of curves  $\gamma_A \subset A$  and  $\gamma_B \subset B$  we modify  $V$ , and replace the point-to-point distances with point-to-curve distances  $d_A(p, \gamma_A)^2$  and  $d_B(p, \gamma_B)^2$ . The resulting contribution encourages the section to vanish along the two-dimensional surface  $\gamma_A \times \gamma_B \subset A \times B$ .

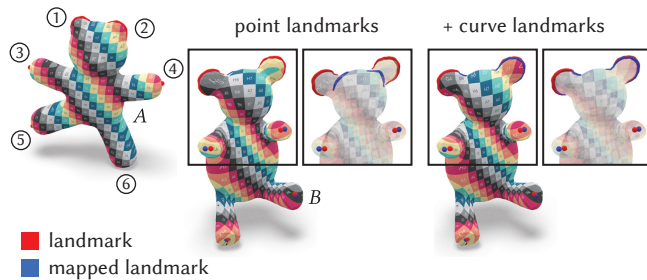


Fig. 16. Enforcing curve to curve correspondences where points are allowed to slide along the specified curves is notoriously challenging. Our implicit representation can elegantly handle these as soft-constraints by modifying the circular well potential. The red points and curves are the specified landmarks, and the blue points and curves visualize their image in  $B$  under the computed correspondence.

<sup>5</sup>it has been observed experimentally that the relevant singularities configurations are related to material defects that can be encoded in  $V$

## 4.7 Surfaces with Boundary

In Figure 17 we compute a map between two topological disks by filling in each boundary component to obtain topological spheres. To ensure that the boundaries are mapped correctly we also include a curve-to-curve singularity pinning potential for them. For simplicity we fill the boundaries with triangle fans, although a more isotropic mesh of the boundary disk would provide a better finite element space, and may be necessary for finely tessellated boundaries.

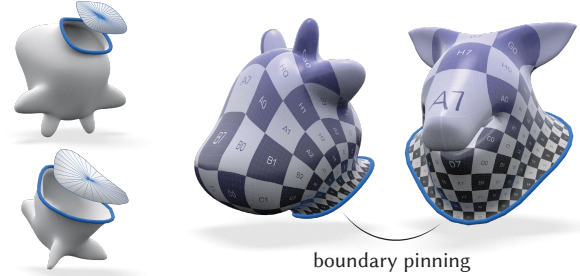


Fig. 17. Our method handles surfaces with boundary by filling in a disk and pinning the boundaries together.

## 4.8 Intrinsic Triangulations

Since the Ginzburg-Landau functional is defined solely in terms of intrinsic quantities, we can improve accuracy by using the intrinsic Delaunay triangulation [Bobenko and Springborn 2007]. While our discretization is already robust to triangulation quality and sampling density (Figure 13), using the intrinsic Delaunay triangulation produces higher quality maps even on severe examples (Figure 18). And intrinsic Delaunay triangulations improve the PDE-based geodesic distance approximation used to constrain landmarks. We represent intrinsic triangulations with the integer coordinates of Gillespie et al. [2021], and applied this intrinsic preprocessing throughout. Intrinsic Delaunay refinement [Sharp et al. 2019; Gillespie et al. 2021] may offer additional improvements, especially on overly coarse inputs.

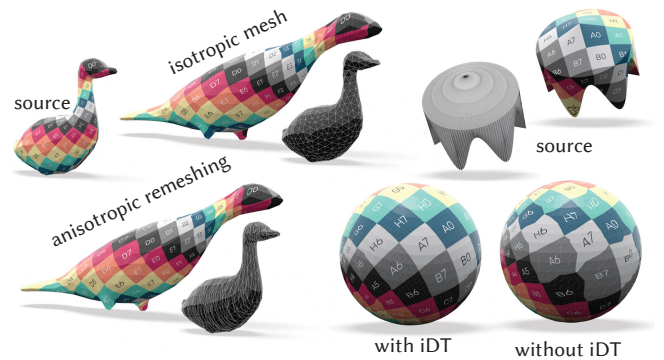


Fig. 18. Our method is robust to triangulation quality, finding the same correspondence when the source is triangulated with an isotropic or a highly anisotropic mesh (left). Using the intrinsic Delaunay triangulation (iDT) yields smooth correspondences even in the most extreme situations (right).

## 5 Results and Evaluation

In this section, we demonstrate the fundamental features and applications of our approach and compare with state-of-the-art techniques for computing bijective correspondences. Note that surface correspondences are general purpose tools, so our method has further applications beyond the basic operations we consider here (mesh transfer, surface interpolation, *etc.*).

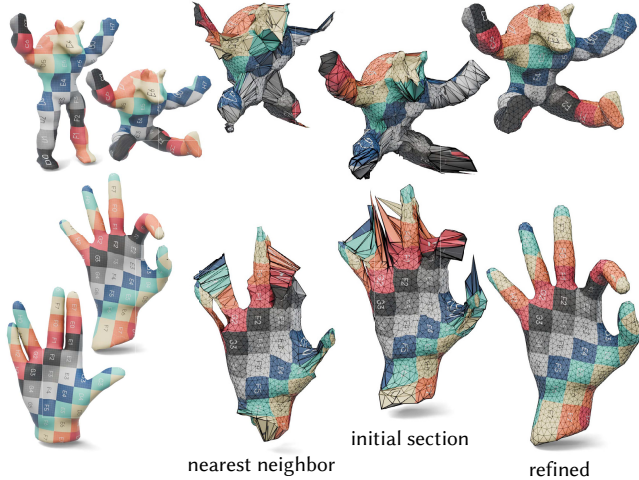


Fig. 19. Correspondences obtained from closest-point initialization, visualized with texture transfer (*left*). Our initialization algorithm finds an initial complex section whose zero set approximates the graph of the nearest neighbor correspondences. Minimizing the Ginzburg-Landau energy refines this section to produce a high-quality correspondence (*right*). Correspondences are visualized using both geometry and texture transfer.

### 5.1 Landmark-Free Correspondences

Many correspondence algorithms require landmarks or other user-specified constraints to guide the optimization. Our method can compute high-quality correspondences even in the absence of landmarks, relying only on the extrinsic geometry of the input surfaces. Given two surfaces  $A$  and  $B$ , we first scale and rigidly align them. Then we compute the nearest-neighbor map which assigns to each vertex  $v_A \in V_A$  the closest face of  $B$ , and vice versa.

This raw nearest-neighbor map is often noisy and discontinuous, but our initialization procedure (Section 4.4) effectively regularizes it, naturally smoothing the correspondence. As shown in Figure 19, this step alone can help to correct the rough geometric input, producing a more coherent initial map even before the energy minimization begins. Optimizing the Ginzburg-Landau energy refines this map into a high-quality correspondence, as seen on the right of Figure 19.

Due to its simplicity and effectiveness, we use this closest-point initialization on all examples (including those with landmark constraints) unless otherwise stated. Figures 12 to 14, 18 and 19 show landmark-free correspondences computed between pairs of nearly isometric shapes differing by large deformations, while Figure 20 shows landmark-free correspondences computed between similar

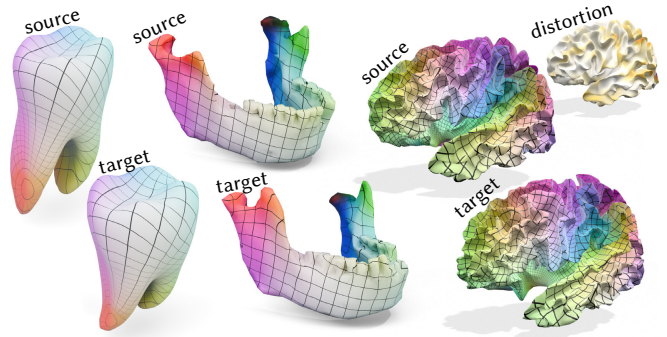


Fig. 20. Landmark-free correspondences computed on a variety of non-isometric biological shapes.

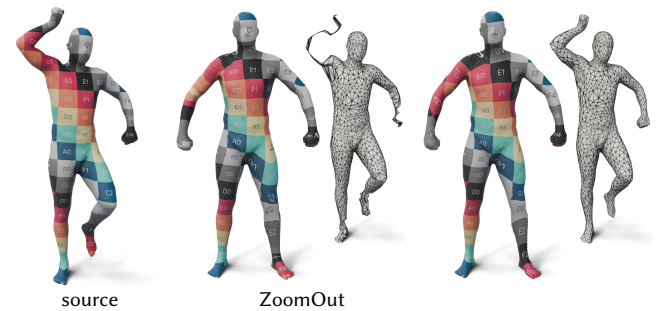


Fig. 21. Spectral methods, such as Functional Maps, usually fail preserve thin shape features. A ZoomOut algorithm [Melzi et al. 2019] initialized with landmarks produces a map that appears smooth (center). Visualizing coordinate transfer however reveals that the arms of the human shape have collapsed. Using this map as initialization, our algorithm (right) is able to correctly match the limbs thanks to its built-in topology and orientation preserving constraints.

but non-isometric biological shapes. Despite the lack of supervision, our method produces smooth, minimal-distortion maps that correctly untangle and align salient geometric features.

### 5.2 Untangling Surface Maps

Since the Ginzburg-Landau functional is well-defined regardless of whether the section  $z$  encodes a valid bijection, our method can also repair invalid correspondences produced by other algorithms.

In particular, thanks to its structural orientation preservation, our method can correct correspondences exhibiting severe orientation reversals due to intrinsic symmetries (Figure 4). We can also compute symmetric self-maps (Figure 5), applying an orientation-reversing constraint that effectively the identity map from the search space.

The method is also well-suited for fixing local geometric collapse: a common artifact in spectral shape matching algorithms (e.g. functional maps), where thin structures shrink to points. Figure 21 shows a typical example of a collapse from the ZoomOut algorithm [Melzi et al. 2019], appearing despite the use of initial landmarks. In these cases, our minimization drives the zero set to expand and cover the target surface, recovering a bijection from a degenerate input.

### 5.3 Curve-to-Curve Correspondences

Like many shape correspondence algorithms, our method supports landmarks that constrain the map at isolated points (Figure 15). But many applications require more flexible landmarks, like mapping a curve on surface  $A$  to a corresponding curve on surface  $B$  without fixing the exact pointwise map. These constraints arise naturally when matching surfaces with sharp feature curves like boundaries or creases (Figures 16 and 17). And they often appear in geometric morphometrics, under the name “semilandmarks”, as curves are easier to identify than specific points [Gunz and Mitteroecker 2013].

A naïve, albeit common, approach is to sample points along each curve and enforce explicit point-to-point correspondences. However, this arbitrarily fixes the parameterization between the curves; even the “natural” choice of arc-length parameterization often twists the map, leading to poor interior correspondences (Figure 22). Our implicit framework, on the other hand, handles such curve-to-curve correspondences with ease—the singularity pinning potential attracts the zeros (and therefore the mapping surface) to the two-dimensional patch  $\gamma_A \times \gamma_B \subset A \times B$  traced out by the pair of corresponding curves, without enforcing any preferential parameterization. The correspondence between the curves then emerges according to what is energetically favorable. Importantly, the two input curves do not need to have the same number of sample points. Finally, we note that extending this approach to mixed point-to-curve constraints is straightforward.

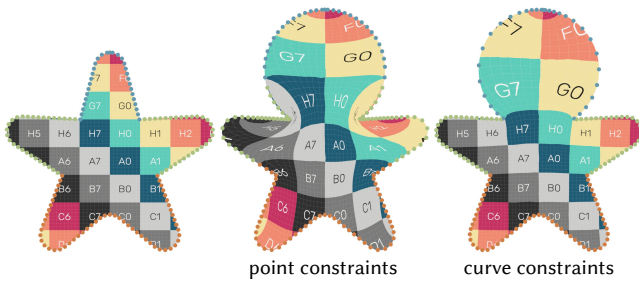


Fig. 22. Enforcing curve-to-curve correspondences can be challenging, and many algorithms rely on constraining the correspondences of each point on the curve. Matching curves based on curvilinear coordinates, however, may introduce significant distortion (middle). Our framework enables the easy enforcement of curve-to-curve correspondences, allowing points to slide along the specified curves (right). This approach substantially reduces overall distortion by accurately matching the concave corners of the boundary.

### 5.4 Comparisons with Prior Work

We evaluate our method against state-of-the-art approaches, checking robustness to initialization, and quality of the resulting correspondences. In particular, we compare to the constant curvature metric based inter-surface mapping (ISM) approach from [Schmidt et al. 2020], the adaptive triangulations (AT) method from [Schmidt et al. 2023], along with the reversible harmonic maps (RHM) approach of [Ezuz et al. 2019b] and hyperbolic orbifold Tutte embeddings (HOTE) [Aigerman and Lipman 2016]. We additionally compare and highlight important differences of our approach with functional maps based correspondences.

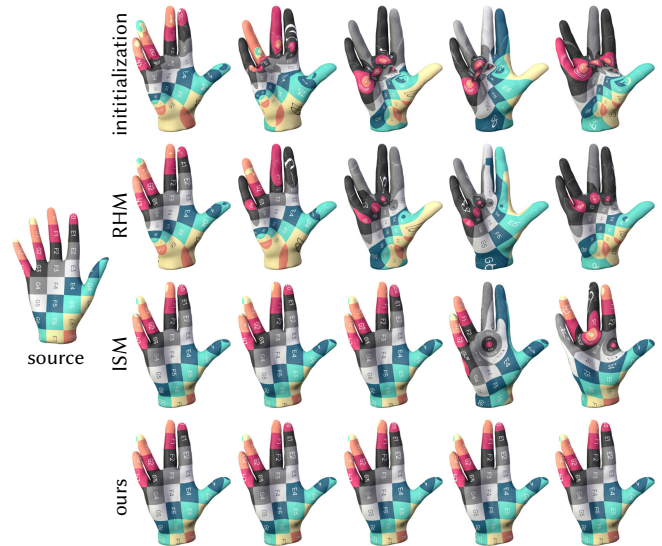


Fig. 23. We replicate the experiment from Schmidt et al. [2020] to evaluate robustness against poor initializations (top row). The RHM method [Ezuz et al. 2019b] becomes trapped in local minima of distortion for each initialization. While Schmidt et al. [2020] (ISM) performs better and achieves consistent results for the first three initializations, our method produces nearly identical low-distortion mappings across all initializations.

*Robustness to Initialization.* A key advantage of our approach lies in its ability to recover high-quality maps even from poor initial correspondences (Sections 5.1 and 5.2). In Figure 23, we replicate the initialization experiment from [Schmidt et al. 2020, Fig. 13], computing correspondence from increasingly distorted initial states. RHM, which directly minimizes the Dirichlet energy, consistently becomes trapped in local minima and fails to realign thin features or correct large distortions. While ISM performs significantly better, it achieves consistent results only for the first three initializations before diverging. In contrast, our method converges to a nearly identical, low-distortion map across all initializations, verifying its untangling capability discussed in Section 5.2.

*Map Quality and Distortion.* We further compare the quality of the final maps against ISM, which also encodes the overlay mesh induced by the correspondences. As shown in Figure 24, while both algorithms converge to geometrically similar correspondences, our implicit Ginzburg-Landau minimization results in consistently lower distortion energies (measured by both the area of the correspondence graph and symmetric Dirichlet energy). The difference is particularly noticeable in the smoothness of the distortion and the regularity of the texture transfer. Our method avoids the localized distortion spikes often seen in explicit remeshing-based approaches.

We also compare the distortion to AT on a challenging pair of inputs with significant non-isometric deformation. Figure 25 shows that while AT produces a valid bijective map, our method produces a more aligned correspondence with lower symmetric Dirichlet energy. While not explicitly enforced, our method better respects

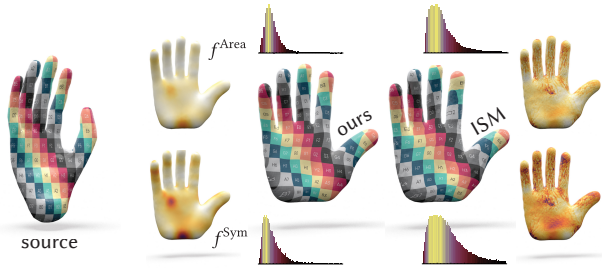


Fig. 24. Both our algorithm and the algorithm of Schmidt et al. [2020] (ISM) represent the common subdivision of the overlay mesh induced by the correspondence. It is needed to evaluate distortion energies of the correspondence intrinsically. While both algorithms find similar correspondences, implicit area minimization produces a less distorted map, even at the finer scales. This is reflected in the smoother distortion distributions, measured both by the area of the correspondence graph given by integrating  $f^{\text{Area}}(\sigma_1, \sigma_2) = ((1 + \sigma_1^2)(1 + \sigma_2^2))^{1/2}$ , and the symmetric Dirichlet energy, integrating  $f^{\text{Sym}}(\sigma_1, \sigma_2) = \sigma_1^2 + \sigma_2^2 + \sigma_1^{-2} + \sigma_2^{-2}$ . The histogram is colored by the height of the bars.

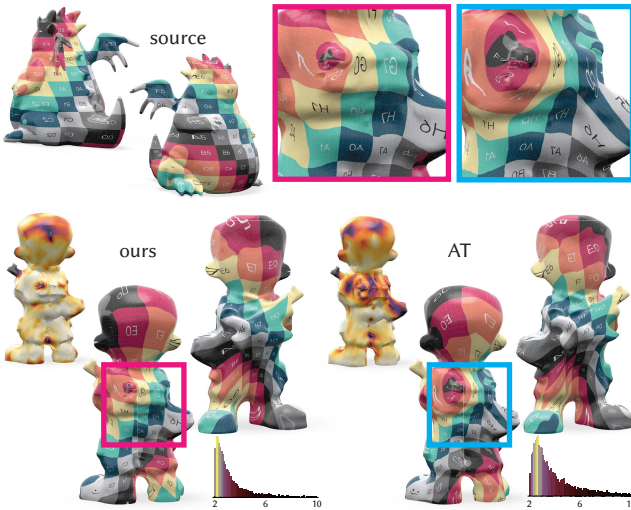


Fig. 25. Distortion is inevitable between non-isometric shapes. Compared with the adaptive triangulations (AT) method [Schmidt et al. 2023], our method produces dramatically less distorted correspondences (as measured by the symmetric Dirichlet energy) that better respect the intrinsic symmetries of the model.

the intrinsic symmetries of the shape (e.g., preserving the left-right symmetry of the human model). In the low-distortion, but non-isometric, regime important in applications like bone registration (Figure 26) our algorithm consistently produces lower distortion correspondences that better respect the intrinsic shape symmetries.

Lastly, while our method and HOTE both produce similar correspondences away from the landmarks, HOTE introduces extremely high distortion near these distinguished points (Figure 27). By contrast, minimizing our global distortion energy produces a correspondence which extends smoothly over landmark points.

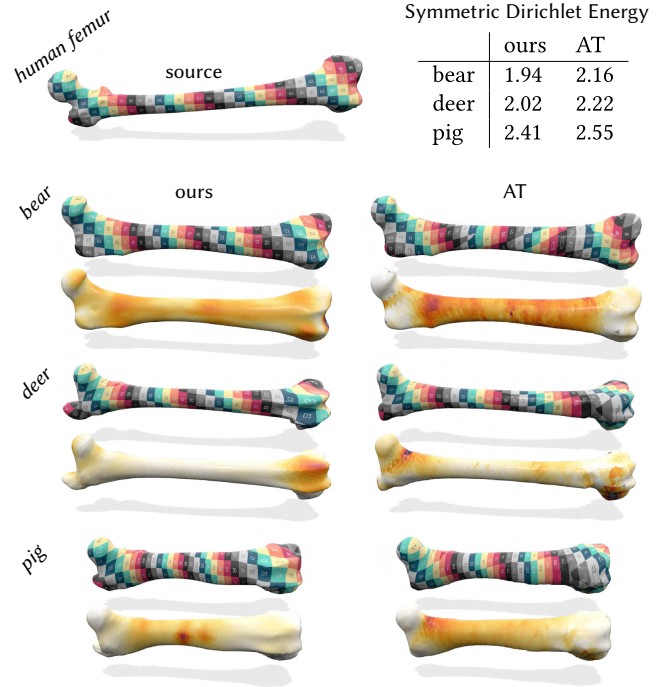


Fig. 26. We use our matching algorithm to compare a human femur with the femurs of a bear, deer, and pig, each computed using four landmarks. Our method (left) consistently achieves lower distortion than the algorithm of Schmidt et al. [2023] (AT) across all three pairs. While both methods realize the same similarity ordering, the correspondence computed using AT exhibits spurious twisting artifacts that are absent in our results.

## 5.5 Implementation and Parameters

We used L-BFGS to minimize the Ginzburg-Landau energy, terminating when the norm of the projected gradient fell below  $10^{-5}$  or after a maximum of 1000 iterations. Comprehensive pseudocode can be found in the supplementary material, and C++<sup>6</sup>, MATLAB<sup>7</sup>, and Python<sup>8</sup> implementations of our method are available.

The main parameter of our method is the Ginzburg-Landau parameter  $\lambda$ , weighting the circular well potential. As discussed in Section 4.2 and Appendix C, setting  $\lambda \approx 100\lambda_{A \times B}$  generally produces high-quality results. However, we observed that examples requiring significant untangling or large deformations from the initialization benefit from a simple annealing scheme, using a smaller  $\lambda$  first to align global features before refining with the default value. When using landmarks, we set equal widths for the Gaussian kernels,  $\sigma_A = \sigma_B$ , setting them depending on the landmarks distribution. We acknowledge that this parameter is sensitive to the specific configuration: the optimal  $\sigma$  depends both on the geometric scale and on the distribution of the landmarks. While the precise value of this parameter does not dramatically change the correspondences, we select this parameter manually to ensure the potential wells

<sup>6</sup><https://github.com/yousufmsoliman/implicit-minimal-surfaces>

<sup>7</sup><https://github.com/etorman/implicit-minimal-surfaces>

<sup>8</sup><https://github.com/RobinMagnet/implicit-minimal-surfaces>

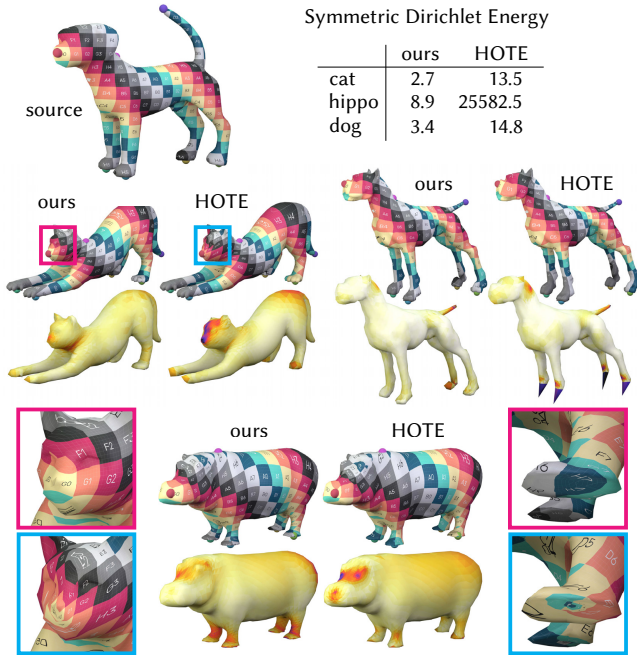


Fig. 27. Our method (*left*) consistently yields lower distortion than maps produced by the hyperbolic orbifold Tutte embedding of Aigerman and Lipman [2016] (HOTE) across all three pairs. In contrast, HOTE does not minimize global mapping distortion and introduces severe artifacts near landmarks. Distortion is measured between the source mesh and the mesh obtained by mapping its vertices onto the target shape.

are sufficiently wide to attract the zero set. We expect that future work on adaptive pinning potentials will allow this parameter to be determined automatically. We refer the reader to the supplementary material for a complete list of parameters used in each figure.

## 6 Limitations and Future Work

While our algorithm offers new perspectives and significant advantages in terms of correspondence quality relative to prior work, it also faces some challenges and suggests avenues for future research.

**Performance.** Without a multiresolution hierarchy, the runtime to compute an implicit minimal surface scales slightly superlinearly with the product of the vertex counts (Figure 28). On a typical pair of models with  $|V_A| \approx |V_B| \approx 500$ , our algorithm takes approximately 20 seconds, while for a pair of models with  $|V_A| \approx |V_B| \approx 5000$  our algorithm takes approximately 1 hour (implemented in C++, measured on an Intel i7-14700K CPU with 64GB of RAM). While the direct application of our method is significantly slower than remeshing based approaches such as [Schmidt et al. 2023], it is competitive with ISM approach of [Schmidt et al. 2020], which reports a runtime of about 3 hours when  $|V_A| \approx |V_B| \approx 4000$ .

The runtime of our algorithm is dominated by the eigenvalue initialization and the minimization of the Ginzburg-Landau functional; the remaining operations, including the construction of surface connections and overlay mesh extraction, are negligible. For the former,

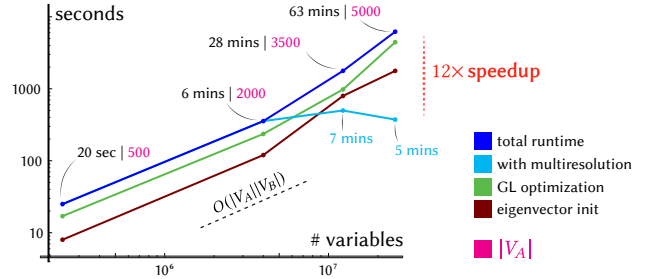


Fig. 28. Computing implicit minimal surfaces scales roughly linearly in the number of variables, with the eigenvalue initialization taking a shade less than half the total runtime. By adopting a multiresolution acceleration scheme, we decrease the effective runtime significantly.

we employ a naive LOBPCG solver, which could be significantly accelerated by incorporating algebraic or geometric multigrid preconditioners [Xu and Zikatanov 2017] adapted to the tensor-product structure of the operator. Similarly, for the energy minimization, we rely on a generic off-the-shelf algorithm (L-BFGS). Using a custom solver that uses the structure of the Ginzburg-Landau energy to exploit parallelization would likely provide significant speedups. These costs are partially mitigated by our multiresolution hierarchy (Section 4.5), which shifts most of the computation to coarser meshes. For the example in Figure 14 ( $|V_A| \approx |V_B| \approx 5000$ ) we obtained a  $12\times$  speedup and computed a distortion minimizing bijective correspondence in only 5 minutes.

Memory limitations place a hard constraint on the input mesh size: although the mass and stiffness matrices are never assembled in the product space, the section  $z$  requires  $|V_A| \times |V_B|$  storage, exceeding capacity for pairs with greater than  $\approx 70k$  vertices each. By using a decimated proxy mesh to compute the correspondence there are further opportunities to decouple the runtime and memory requirements from the input resolution.

**Bijection.** As mentioned in Section 4.2, the choice of Ginzburg-Landau parameter can have a big effect on the quality of the output map. Figure 29 shows maps computed between the same pair of surfaces for a variety of parameters  $\lambda = t\lambda_0$ . In general, we observe that when  $\lambda$  is not sufficiently large, the map degrades in regions of high curvature, yielding a non-bijective correspondence between the two surfaces. In practice, we find that increasing the Ginzburg-Landau parameter allows us to compute low-distortion bijective maps, but further theoretical work would be required to provide guarantees on the quality or bijectivity of the output maps.

First of all non-convexity of the Ginzburg-Landau functional means that the optimization may converge to local minima. While theoretical work suggests that critical points—beyond just global minima—can describe minimal surfaces, it remains unclear whether *all* critical points correspond to such surfaces. Moreover, more analysis is needed to characterize (and constrain) when these implicit minimal surfaces are graphs over both factors (*i.e.* bijections).

**Generalization.** Extending the algorithm to general surfaces, not limited to genus zero surfaces with boundaries, is a compelling direction for future work. The topology of the implicit surfaces,

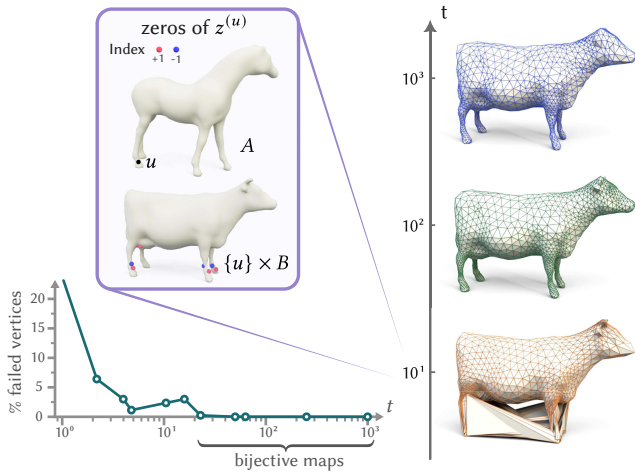


Fig. 29. Coordinate transfer from the horse  $A$  into the cow  $B$ , computed for three values of  $\lambda = t \lambda_0$  (right). A large value of  $\lambda$  produces clean correspondences, but the map degrades near regions of high curvature when  $\lambda$  decreases. The graph (bottom left) plots the average percentage of vertices whose slices have more than one zero, thus breaking bijectivity. We obtain bijective maps once  $t \gtrsim 50$ . Inspecting the failure at  $t = 10^1$ , we find that for a vertex  $u \in A$  mapped incorrectly, the slice  $z^{(u)}$  on  $B$  has several zeros rather than one. Their indices (red +1, blue -1) sum to 1, as required by our construction, and one of the +1 zeros lie on the correct location. This suggests that a specific zero-selection heuristic could recover valid correspondences even in these degenerate cases.

encoded in the curvature of the connection, needs to be modified so that it represents surfaces that are the graphs of bijective correspondences. In addition to total curvature  $2\pi$  on each slice, additional curvature must be concentrated, according to the map homotopy type, on the closed two-dimensional surfaces corresponding to the product of homology generators. Beyond changing the topology of the implicit surfaces, additional complications arise when minimizing the Ginzburg-Landau functional that may obstruct extracting a correspondence. For instance, the energy density of critical points of the Ginzburg-Landau functional may not concentrate around a singularity as  $\varepsilon \rightarrow 0$ , instead spreading out over the entire space according to a harmonic one-form.

In principle, our implicit representation can also be generalized to codimension-3 manifolds, replacing complex line bundles with rank-3 vector bundles. Looking for implicit codimension-3 minimal surfaces in the product of two volumetric domains would be the natural generalization of our method to volumetric correspondences.

Finally, we mention that beyond bijective correspondences, our method can compute covering maps instead of bijections. For instance, by picking a connection on  $A$  which has total curvature  $4\pi$  instead of  $2\pi$ , we can compute a double covering map where every point of  $B$  is covered by two points of  $A$ .



## Acknowledgments

Thanks to Aria Halavati for teaching us about the complex line bundle approach to codimension-two minimal surfaces.

## References

S. Mazdak Abulnaga, Oded Stein, Polina Golland, and Justin Solomon. 2023. Symmetric volume maps: Order-invariant volumetric mesh correspondence with free boundary. *ACM Transactions on Graphics (TOG)* 42, 3 (2023), 1–20. <https://doi.org/10.1145/3572897>

Amandine Aftalion, Etienne Sandier, and Sylvia Serfaty. 2001. Pinning phenomena in the Ginzburg-Landau model of superconductivity. *Journal de mathématiques pures et appliquées* 80, 3 (2001), 339–372. [https://doi.org/10.1016/S0021-7824\(00\)01180-6](https://doi.org/10.1016/S0021-7824(00)01180-6)

Noam Aigerman, Shahar Z Kovalsky, and Yaron Lipman. 2017. Spherical Orbifold Tutte Embeddings. *ACM Trans. Graph.* 36, 4 (2017), 90. <https://doi.org/10.1145/3072959.3073615>

Noam Aigerman and Yaron Lipman. 2015. Orbifold Tutte Embeddings. *ACM Trans. Graph.* 34, 6 (2015), 190–1. <https://doi.org/10.1145/2816795.2818099>

Noam Aigerman and Yaron Lipman. 2016. Hyperbolic Orbifold Tutte Embeddings. *ACM Trans. Graph.* 35, 6 (2016), 217–1. <https://doi.org/10.1145/2980179.2982412>

Noam Aigerman, Roi Poranne, and Yaron Lipman. 2014. Lifted bijections for low distortion surface mappings. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–12. <https://doi.org/10.1145/2601097.2601158>

Noam Aigerman, Roi Poranne, and Yaron Lipman. 2015. Seamless surface mappings. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–13. <https://doi.org/10.1145/2766921>

Giovanni Alberti, Sisto Baldo, and Giandomenico Orlandi. 2005. Variational Convergence for Functionals of Ginzburg-Landau Type. *Indiana University Mathematics Journal* 54, 5 (2005), 1411–1472. <https://www.jstor.org/stable/24902326>

N. El Amrani, P. Roetzer, and F. Bernard. 2025. High-Resolution 3D Shape Matching with Global Optimality and Geometric Consistency. *Computer Graphics Forum* 44, 5 (2025), e70208. <https://doi.org/10.1111/cgf.70208>

Alex Baden, Keenan Crane, and Misha Kazhdan. 2018. Möbius Registration. *Computer Graphics Forum (SGP)* 37, 5 (2018), 211–220. <https://doi.org/10.1111/cgf.13503>

M Faisal Beg, Michael I Miller, Alain Trounev, and Laurent Younes. 2005. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International journal of computer vision* 61, 2 (2005), 139–157. <https://doi.org/10.1023/B:VISI.0000043755.93987.a>

A. Bobenko and B. Springborn. 2007. A discrete Laplace-Beltrami operator for simplicial surfaces. *Discrete & Computational Geometry* 38, 4 (2007), 740–756. <https://doi.org/10.1007/s00454-007-9006-1>

Lucas Brifault, David Cohen-Steiner, and Mathieu Desbrun. 2025. Efficient and Scalable Spatial Regularization of Optimal Transport. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25)*. ACM, Article 120, 10 pages. <https://doi.org/10.1145/3757377.3763976>

Giacomo Canevari, Federico Luigi Dipasquale, and Giandomenico Orlandi. 2023. The Yang-Mills-Higgs functional on complex line bundles:  $\Gamma$ -convergence and the London equation. *Archive for Rational Mechanics and Analysis* 247, 6 (2023), 104. <https://doi.org/10.1007/s00205-023-01933-1>

Albert Chern, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. 2018. Shape from metric. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 63. <https://doi.org/10.1145/3197517.3201276>

Keenan Crane, Mathieu Desbrun, and Peter Schröder. 2010. Trivial Connections on Discrete Surfaces. *Computer Graphics Forum (SGP)* 29, 5 (2010), 1525–1533. <https://doi.org/10.1111/j.1467-8659.2010.01761.x>

Camillo De Lellis. 2014. The regularity of minimal surfaces in higher codimension. *Current Developments in Mathematics* (2014), 153–229.

Guido De Philippis and Alessandro Pigati. 2024. Non-degenerate minimal submanifolds as energy concentration sets: A variational approach. *Communications on Pure and Applied Mathematics* 77, 8 (2024), 3581–3627. <https://doi.org/10.1002/cpa.22193>

Mathieu Desbrun, Anil N Hirani, Melvin Leok, and Jerrold E Marsden. 2005. Discrete exterior calculus. arXiv:math/0508341 [math.DG]

Nicolas Donati, Etienne Corman, Simone Melzi, and Maks Ovsjanikov. 2022. Complex Functional Maps: A Conformal Link Between Tangent Bundles. *Computer Graphics Forum* 41, 1 (2022), 317–334. <https://doi.org/10.1111/cgf.14437>

M. Eisenberger, Z. Löhner, and D. Cremers. 2019. Divergence-Free Shape Correspondence by Deformation. *Computer Graphics Forum (SGP)* 38, 5 (2019), 1–12. <https://doi.org/10.1111/cgf.13785>

Marvin Eisenberger, Zorah Löhner, and Daniel Cremers. 2020. Smooth Shells: Multi-Scale Shape Registration With Functional Maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 12262–12271. <https://doi.org/10.1109/CVPR42600.2020.01228>

D. Ezuz, B. Heeren, O. Azencot, M. Rumpf, and M. Ben-Chen. 2019a. Elastic Correspondence between Triangle Meshes. *Computer Graphics Forum* 38, 2 (2019), 121–134. <https://doi.org/10.1111/cgf.13624>

- Danielle Ezuz, Justin Solomon, and Mirela Ben-Chen. 2019b. Reversible harmonic maps between discrete surfaces. *ACM Transactions on Graphics (TOG)* 38, 2 (2019), 1–12. <https://doi.org/10.1145/3202660>
- Herbert Federer and Wendell H. Fleming. 1960. Normal and Integral Currents. *Annals of Mathematics* 72, 3 (1960), 458–520. <https://doi.org/10.2307/1970227>
- Mark Gillespie, Nicholas Sharp, and Keenan Crane. 2021. Integer coordinates for intrinsic geometry processing. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 13 pages. <https://doi.org/10.1145/3478513.3480522>
- Philipp Gunz and Philipp Mitteroecker. 2013. Semilandmarks: A Method for Quantifying Curves and Surfaces. *Hystrix the Italian Journal of Mammalogy* 24, 1 (May 2013), 103–109. <https://doi.org/10.4404/hystrix-24.1-6292>
- Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas J. Guibas. 2008. Non-rigid registration under isometric deformations. *Computer Graphics Forum (SGP)* 27, 5 (2008), 1449–1457. <https://doi.org/10.1111/j.1467-8659.2008.01285.x>
- Robert L Jerrard and Halil Mete Soner. 2002. The Jacobian and the Ginzburg-Landau energy. *Calculus of Variations and Partial Differential Equations* 14, 2 (2002), 151–191. <https://doi.org/10.1007/s005260100093>
- T. Kanai, H. Suzuki, and F. Kimura. 1997. 3D geometric metamorphosis based on harmonic map. In *Proceedings The Fifth Pacific Conference on Computer Graphics and Applications*. 97–104. <https://doi.org/10.1109/PCCGA.1997.626179>
- Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. 2011. Blended intrinsic maps. *ACM Transactions on Graphics (TOG)* 30, 4 (2011), 1–12. <https://doi.org/10.1145/2010324.1964974>
- Felix Knöppel. 2020. *Riemann Surfaces: Complex Analysis from a Differential Geometric Viewpoint*. (2020). Lecture Notes.
- Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Globally optimal direction fields. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10. <https://doi.org/10.1145/2461912.2462005>
- Felix Knöppel and Ulrich Pinkall. 2016. *Complex line bundles over simplicial complexes and their applications*. Springer Berlin Heidelberg, Berlin, Heidelberg. [https://doi.org/10.1007/978-3-662-50447-5\\_6](https://doi.org/10.1007/978-3-662-50447-5_6)
- A. V. Knyazev, M. E. Argentati, I. Lashuk, and E. E. Ovtchinnikov. 2007. Block Locally Optimal Preconditioned Eigenvalue Solvers (BLOPEX) in Hypre and PETSc. *SIAM Journal on Scientific Computing* 29, 5 (2007), 2224–2239. <https://doi.org/10.1137/060661624>
- Maximilian Kohlbrenner, Singchun Lee, Marc Alexa, and Misha Kazhdan. 2023. Poisson Manifold Reconstruction – Beyond Co-dimension One. *Computer Graphics Forum (SGP)* 42, 5 (2023), e14907. <https://doi.org/10.1111/cgf.14907>
- Vladislav Kraevoy and Alla Sheffer. 2004. Cross-parameterization and compatible remeshing of 3D models. *ACM Transactions on Graphics (TOG)* 23, 3 (2004), 861–869. <https://doi.org/10.1145/1015706.1015811>
- Hao Li, Robert W. Sumner, and Mark Pauly. 2008b. Global Correspondence Optimization for Non-Rigid Registration of Depth Scans. *Computer Graphics Forum (SGP)* 27, 5 (2008), 1421–1430. <https://doi.org/10.1111/j.1467-8659.2008.01282.x>
- Xin Li, Yunfan Bao, Xiaohu Guo, Miao Jin, Xianfeng Gu, and Hong Qin. 2008a. Globally optimal surface mapping for surfaces with arbitrary topology. *IEEE Transactions on Visualization and Computer Graphics* 14, 4 (2008), 805–819. <https://doi.org/10.1109/TVCG.2008.32>
- Fanghua Lin and Tristan Riviere. 1999. Complex Ginzburg-Landau equations in high dimensions and codimension two area minimizing currents. *Journal of the European Mathematical Society* 1, 3 (1999), 237–311. <https://doi.org/10.1007/s100970050008>
- O. Litany, E. Rodolà, A. M. Bronstein, M. M. Bronstein, and D. Cremers. 2016. Non-rigid puzzles. *Computer Graphics Forum (SGP)* 35, 5 (Aug 2016), 135–143. <https://doi.org/10.1111/cgf.12970>
- Nathan Litke, Marc Droske, Martin Rumpf, and Peter Schröder. 2005. An image processing approach to surface matching. In *Proceedings of the Third Eurographics Symposium on Geometry Processing (SGP '05)*. Eurographics Association, 207–es. <https://doi.org/10.2312/SGP/SGP05/207-216>
- Beibei Liu, Yiyi Tong, Fernando De Goes, and Mathieu Desbrun. 2016. Discrete Connection and Covariant Derivative for Vector Field Analysis and Design. *ACM Transactions on Graphics (TOG)* 35, 3, Article 23 (March 2016), 17 pages. <https://doi.org/10.1145/2870629>
- Manish Mandad, David Cohen-Steiner, Leif Kobbelt, Pierre Alliez, and Mathieu Desbrun. 2017. Variance-minimizing transport plans for inter-surface mapping. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–14. <https://doi.org/10.1145/3072959.3073671>
- Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. 2019. ZoomOut: spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–14. <https://doi.org/10.1145/3355089.3356524>
- Takashi Michikawa, Takashi Kanai, Masahiro Fujita, and Hiroaki Chiyokura. 2001. Multiresolution Interpolation Meshes. In *Proceedings of the 9th Pacific Conference on Computer Graphics and Applications (PG '01)*. IEEE Computer Society, 60. <https://doi.org/10.1109/PCCGA.2001.962858>
- John Willard Milnor and James D Stasheff. 1974. *Characteristic Classes*. Number 76 in *Annals of Mathematics Studies*. Princeton University Press.
- Luciano Modica and S Mortola. 1977. Un esempio di  $\Gamma$ -convergenza. *Bollettino della Unione Matematica Italiana B* 14 (1977), 285–299.
- Luca Morreale, Noam Aigerman, Vladimir Kim, and Niloy J. Mitra. 2021. Neural Surface Maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 4637–4646. <https://doi.org/10.1109/CVPR46437.2021.00461>
- Liviu Nicolaescu and Nikhil Savale. 2017. The Gauss-Bonnet-Chern theorem: a probabilistic perspective. *Trans. Amer. Math. Soc.* 369, 4 (2017), 2951–2986. <https://doi.org/10.1090/tran/6895>
- Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. 2012. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–11. <https://doi.org/10.1145/2185520.2185526>
- Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. 2017. Computing and processing correspondences with functional maps. In *ACM SIGGRAPH 2017 Courses*. ACM, Article 5, 62 pages. <https://doi.org/10.1145/3084873.3084877>
- David Palmer, Albert Chern, and Justin Solomon. 2024. Lifting Directional Fields to Minimal Sections. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–20. <https://doi.org/10.1145/3658198>
- David Palmer, Dmitriy Smirnov, Stephanie Wang, Albert Chern, and Justin Solomon. 2022. DeepCurrents: Learning Implicit Representations of Shapes with Boundaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 18644–18654. <https://doi.org/10.1109/CVPR52688.2022.01811>
- Daive Parise, Alessandro Pigati, and Daniel Stern. 2024. Convergence of the self-dual  $U(1)$ -Yang-Mills-Higgs energies to the  $(n-2)$ -area functional. *Communications on Pure and Applied Mathematics* 77, 1 (2024), 670–730. <https://doi.org/10.1002/cpa.22150>
- Chao Peng and Sabin Timala. 2016. Fast mapping and morphing for genus-zero meshes with cross spherical parameterization. *Computers & Graphics* 59 (2016), 107–118. <https://doi.org/10.1016/j.cag.2016.06.001>
- Alessandro Pigati and Daniel Stern. 2021. Minimal submanifolds from the abelian Higgs model. *Inventiones mathematicae* 223 (2021), 1027–1095.
- Roman Poya, Rogelio Ortigosa, and Theodore Kim. 2023. Geometric optimisation via spectral shifting. *ACM Transactions on Graphics (TOG)* 42, 3 (2023), 1–15. <https://doi.org/10.1145/3585003>
- Jing Ren, Simone Melzi, Maks Ovsjanikov, and Peter Wonka. 2020. Maptree: Recovering multiple solutions in the space of maps. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–17. <https://doi.org/10.1145/3414685.3417800>
- Jing Ren, Simone Melzi, Peter Wonka, and Maks Ovsjanikov. 2021. Discrete optimization for shape matching. *Computer Graphics Forum (SGP)* 40, 5 (2021), 81–96. <https://doi.org/10.1111/cgf.14359>
- Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. 2018. Continuous and orientation-preserving correspondences via functional maps. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–16. <https://doi.org/10.1145/3272127.3275040>
- Paul Roetzer and Florian Bernard. 2025. Fast Globally Optimal and Geometrically Consistent 3D Shape Matching. <https://doi.org/10.48550/arXiv.2504.06385> arXiv:2504.06385 [cs]
- Paul Roetzer, Paul Swoboda, Daniel Cremers, and Florian Bernard. 2022. A Scalable Combinatorial Solver for Elastic Geometrically Consistent 3D Shape Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 428–438. <https://doi.org/10.1109/CVPR52688.2022.00052>
- Yusuf Sahillioglu. 2020. Recent advances in shape correspondence. *The Visual Computer* 36, 8 (2020), 1705–1721. <https://doi.org/10.1007/s00371-019-01760-0>
- Patrick Schmidt, Janis Born, Marcel Campen, and Leif Kobbelt. 2019. Distortion-minimizing injective maps between surfaces. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–15. <https://doi.org/10.1145/3355089.3356519>
- Patrick Schmidt, Marcel Campen, Janis Born, and Leif Kobbelt. 2020. Inter-surface maps via constant-curvature metrics. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 119–1. <https://doi.org/10.1145/3386569.3392399>
- P. Schmidt, D. Pieper, and L. Kobbelt. 2023. Surface Maps via Adaptive Triangulations. *Computer Graphics Forum* 42, 2 (2023), 103–117. <https://doi.org/10.1111/cgf.14747>
- John Schreiner, Arul Asirvatham, Emil Praun, and Hugues Hoppe. 2004. Inter-surface mapping. *ACM Transactions on Graphics (TOG)* 23, 3 (2004), 870–877. <https://doi.org/10.1145/1015706.1015812>
- Andrei Sharf, Marina Blumenkrants, Ariel Shamir, and Daniel Cohen-Or. 2006. Snap-paste: an interactive technique for easy mesh composition. *The Visual Computer* 22, 9 (2006), 835–844. <https://doi.org/10.1007/s00371-006-0068-5>
- Nicholas Sharp, Yousuf Soliman, and Keenan Crane. 2019. Navigating Intrinsic Triangulations. *ACM Transactions on Graphics (TOG)* 38, 4, Article 55 (2019), 16 pages. <https://doi.org/10.1145/3306346.3322979>
- Rui Shi, Wei Zeng, Zhengyu Su, Jian Jiang, Hanna Damasio, Zhonglin Lu, Yalin Wang, Shing-Tung Yau, and Xianfeng Gu. 2016. Hyperbolic harmonic mapping for surface registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 5 (2016), 965–980. <https://doi.org/10.1109/TPAMI.2016.2567398>

- Justin Solomon, Andy Nguyen, Adrian Butscher, Mirela Ben-Chen, and Leonidas Guibas. 2012. Soft Maps Between Surfaces. *Computer Graphics Forum (SGP)* 31, 5 (2012), 1617–1626. <https://doi.org/10.1111/j.1467-8659.2012.03167.x>
- Kenshi Takayama. 2022. Compatible intrinsic triangulations. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–12. <https://doi.org/10.1145/3528223.3530175>
- Gary KL Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C Langbein, Yonghui Liu, David Marshall, Ralph R Martin, Xian-Fang Sun, and Paul L Rosin. 2012. Registration of 3D point clouds and meshes: A survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics* 19, 7 (2012), 1199–1217. <https://doi.org/10.1109/TVCG.2012.310>
- Alex Tsui, Devin Fenton, Phong Vuong, Joel Hass, Patrice Koehl, Nina Amenta, David Coeurjolly, Charles DeCarli, and Owen Carmichael. 2013. Globally Optimal Cortical Surface Matching with Exact Landmark Correspondence. In *Information Processing in Medical Imaging*. Springer Berlin Heidelberg, 487–498. [https://doi.org/10.1007/978-3-642-38868-2\\_41](https://doi.org/10.1007/978-3-642-38868-2_41)
- Oliver van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. 2011. A Survey on Shape Correspondence. *Computer Graphics Forum* 30, 6 (2011), 1681–1707. <https://doi.org/10.1111/j.1467-8659.2011.01884.x>
- Matthias Vestner, Roei Litman, Emanuele Rodola, Alex Bronstein, and Daniel Cremers. 2017. Product Manifold Filter: Non-rigid Shape Correspondence via Kernel Density Estimation in the Product Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 6681–6690. <https://doi.org/10.1109/CVPR.2017.707>
- Stephanie Wang and Albert Chern. 2021. Computing minimal surfaces with differential forms. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–14. <https://doi.org/10.1145/3450626.3459781>
- Steffen Weißmann, Ulrich Pinkall, and Peter Schröder. 2014. Smoke rings from smoke. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–8. <https://doi.org/10.1145/2601097.2601171>
- H Whitney. 1957. *Geometric Integration Theory Princeton Univ.* 62 pages.
- Thomas Windheuser, Ulrich Schlickewei, Frank R. Schmidt, and Daniel Cremers. 2011a. Geometrically Consistent Elastic Matching of 3D Shapes: A Linear Programming Solution. In *2011 International Conference on Computer Vision*. IEEE, Barcelona, Spain, 2134–2141. <https://doi.org/10.1109/ICCV.2011.6126489>
- Thomas Windheuser, Ulrich Schlickewei, Frank R. Schmidt, and Daniel Cremers. 2011b. Large-Scale Integer Linear Programming for Orientation Preserving 3 D Shape Matching. *Computer Graphics Forum* 30, 5 (Aug. 2011), 1471–1480. <https://doi.org/10.1111/j.1467-8659.2011.02021.x>
- Jinchao Xu and Ludmil Zikatanov. 2017. Algebraic multigrid methods. *Acta Numerica* 26 (2017), 591–721. <https://doi.org/10.1017/S0962492917000083>
- Yang Yang, Xiao-Ming Fu, Shuangming Chai, Shi-Wei Xiao, and Ligang Liu. 2018. Volume-enhanced compatible remeshing of 3D models. *IEEE Transactions on Visualization and Computer Graphics* 25, 10 (2018), 2999–3010. <https://doi.org/10.1109/TVCG.2018.2861396>
- Yang Yang, Wen-Xiang Zhang, Yuan Liu, Ligang Liu, and Xiao-Ming Fu. 2020. Error-bound compatible remeshing. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 113–1. <https://doi.org/10.1145/3386569.3392434>
- Laurence C Young. 1963. Some extremal questions for simplicial complexes V. The relative area of a Klein bottle. *Rendiconti del Circolo Matematico di Palermo* 12, 3 (1963), 257–274. <https://doi.org/10.1007/BF02851262>

## A Riemannian Geometry in the Product of Surfaces

Consider a pair of oriented smooth surfaces  $A$  and  $B$  with Riemannian metrics  $g_A$  and  $g_B$ , respectively. The product space  $A \times B$  is a Riemannian 4-manifold endowed with the metric  $g_{A \times B}(X_A \oplus X_B, Y_A \oplus Y_B) := g_A(X_A, Y_A) + g_B(X_B, Y_B)$ .

**PROPOSITION A.1.** *Given  $\varphi : A \rightarrow B$  the area of its graph  $\Sigma_\varphi$  can be expressed as the integral of the singular values  $\sigma_1, \sigma_2$  of the deformation  $d\varphi$ :*

$$\text{Area}(\Sigma_\varphi) = \int_A \sqrt{(1 + \sigma_1^2)(1 + \sigma_2^2)} \, \text{vol}_A.$$

**PROOF.** The graph can be parameterized by  $F_\varphi : A \rightarrow \Sigma_\varphi, F_\varphi(p) := (p, \varphi(p))$ , and so  $\text{Area}(\Sigma_\varphi)$  can be computed by integrating the area

form induced by the parameterization:

$$\begin{aligned} \text{Area}(\Sigma_\varphi) &= \int_A \det \left( F_\varphi^* g_{A \times B} \right)^{\frac{1}{2}} \, \text{vol}_A \\ &= \int_A \det \left( I + g_A^{-1} \varphi^* g_B \right)^{\frac{1}{2}} \, \text{vol}_A \\ &= \int_A \left[ 1 + \det \left( g_A^{-1} \varphi^* g_B \right) + \text{tr} \left( g_A^{-1} \varphi^* g_B \right) \right]^{\frac{1}{2}} \, \text{vol}_A \\ &= \int_A \left[ 1 + \sigma_1^2 \sigma_2^2 + \sigma_1^2 + \sigma_2^2 \right]^{\frac{1}{2}} \, \text{vol}_A. \end{aligned}$$

□

As expected the change of variables to  $B$  leads to the same function applied to the inverse singular values:

$$\begin{aligned} \text{Area}(\Sigma_\varphi) &= \int_B \left[ 1 + \sigma_1^2 \sigma_2^2 + \sigma_1^2 + \sigma_2^2 \right]^{\frac{1}{2}} \frac{\det \left( (\varphi^{-1})^* g_B \right)^{\frac{1}{2}}}{\det(g_B)^{\frac{1}{2}}} \, \text{vol}_B \\ &= \int_B \left[ 1 + \sigma_1^2 \sigma_2^2 + \sigma_1^2 + \sigma_2^2 \right]^{\frac{1}{2}} \frac{1}{\sigma_1 \sigma_2} \, \text{vol}_B \\ &= \int_B \left[ \frac{1}{\sigma_1^2 \sigma_2^2} + 1 + \frac{1}{\sigma_2^2} + \frac{1}{\sigma_1^2} \right]^{\frac{1}{2}} \, \text{vol}_B. \end{aligned}$$

## B The Product-Space Dirichlet Energy

Here we discretize the smooth expression of Dirichlet energy in Equation 6 to obtain a discrete Dirichlet energy on the product space. We consider a connection  $\nabla$  that is the tensor product of connections  $\nabla^A$  and  $\nabla^B$  on  $A$  and  $B$ , respectively. For convenience, we reproduce Equation 6 below:

$$\mathcal{E}^D(z) = \int_A \left( \int_B |\nabla z|_B^2 \, \text{vol}_B \right) \text{vol}_A + \int_B \left( \int_A |\nabla z|_A^2 \, \text{vol}_A \right) \text{vol}_B$$

Discretizing the second term yields:

$$\begin{aligned} &\int_{A \times B} |\nabla^A z(p, q)|^2 \, \text{vol}_{A \times B}(p, q) \\ &= \sum_{\substack{ijk \in F_A \\ abc \in F_B}} \underbrace{\int_{abc} \left| \sum_{\substack{u \in ijk \\ v \in abc}} Z_{u,v} \nabla^A \phi_u^A(p) \phi_v^B(q) \right|^2 \, \text{vol}_A(p) \, \text{vol}_B(q)}_{X_{ijk,abc}}. \end{aligned}$$

The integrand evaluates to

$$\begin{aligned} X_{ijk,abc} &= \sum_{\substack{m \in ijk \\ n \in abc}} \bar{Z}_{m,n} \left[ \sum_{\substack{u \in ijk \\ v \in abc}} \left( \int_{ijk} (\nabla^A \phi_m^A)^*(p) \nabla^A \phi_u^A(p) \, \text{vol}_A(p) \right) \right. \\ &\quad \left. \left( Z_{u,v} \int_{abc} \bar{\phi}_n^B(q) \phi_v^B(q) \, \text{vol}_B(q) \right) \right], \end{aligned}$$

where we recognize the integrals as the components of connection Laplacian of  $A$  and the mass matrix of  $B$ . Hence, the connection Laplacian on  $A \times B$  has a tensor-product structure:

$$L_{A \times B}^\nabla = L_A^\nabla \otimes M_B^\nabla + M_A^\nabla \otimes L_B^\nabla. \quad (37)$$

Using a matrix representation  $Z \in \mathbb{C}^{V_A \times V_B}$  of the complex field  $z : A \times B \rightarrow \mathbb{C}$ , the discretization of Equation 6 only depends on

left and right multiplications of the connection Laplacian and mass matrices defined on each manifold:

$$\mathcal{E}^D(Z) = \frac{1}{2} \langle Z, L_A^\nabla Z (M_B^\nabla)^\top \rangle_{\mathbb{C}} + \frac{1}{2} \langle Z, M_A^\nabla Z (L_B^\nabla)^\top \rangle_{\mathbb{C}}.$$

### C The Ginzburg-Landau Parameter

For the following result, we consider a Hermitian positive-definite matrix  $A \in \mathbb{C}^{d \times d}$  along with a diagonal matrix  $M \in \mathbb{R}^{d \times d}$ —for our application, they will be the connection Laplacian and mass matrix on a four-dimensional cell complex, respectively.

**LEMMA C.1.** *Let  $\lambda_0 > 0$  be the smallest eigenvalue of the generalized eigenvalue problem  $Az = \lambda Mz$ . Consider the discrete energy*

$$e(z) = \frac{1}{2} \langle Az, z \rangle + \frac{1}{4} \langle Mu, u \rangle,$$

where  $u_i = 1 - |z_i|^2$  for every vertex  $i = 1, \dots, d$ . If  $0 < \lambda \leq \lambda_0$  then the only critical point of  $e$  is given by  $z \equiv 0$ .

**PROOF.** Taking the scalar product between a stationary point  $\bar{z}$  and the gradient  $\nabla e(\bar{z})$ , we obtain:

$$\langle \bar{z}, \nabla e(\bar{z}) \rangle = \langle \bar{z}, A\bar{z} \rangle - \lambda \langle \bar{z}, M\bar{z} \rangle + \lambda \langle |\bar{z}|^2, M|\bar{z}|^2 \rangle = 0. \quad (38)$$

Using the fact that the smallest eigenvalue of  $A$  is greater than  $\lambda$ , we conclude that  $\bar{z}$  must be equal to zeros:

$$0 \leq \langle \bar{z}, A\bar{z} - \lambda M\bar{z} \rangle = -\lambda \langle |\bar{z}|^2, M|\bar{z}|^2 \rangle = 0 \leq 0 \quad (39)$$

□

In the continuous setting, the connection Laplacian  $\Delta^\nabla = \Delta_A^\nabla \otimes I_B + I_A \otimes \Delta_B^\nabla$  on  $A \times B$  decomposes as a sum of connection Laplacians on each shape. Its minimal eigenvalue is therefore the sum of the minimal eigenvalues on each shape:  $\lambda_0 = \lambda_A^\nabla + \lambda_B^\nabla$ , where  $\lambda_A^\nabla$  and  $\lambda_B^\nabla$  are the minimal eigenvalues of  $\Delta_A^\nabla$  and  $\Delta_B^\nabla$  respectively (defined with respect to the corresponding volume form on each shape).

In practice, however, due to differing discretization choices, we use  $L_{A \times B}^\nabla = L_A^\nabla \otimes M_B^\nabla + M_A^\nabla \otimes L_B^\nabla$  for the Dirichlet energy, but  $M = M_A \otimes M_B$  for the potential energy, using lumped mass matrices  $M_A$  and  $M_B$ . The generalized eigenvalue problem  $Az = \lambda Mz$  then doesn't strictly separate. Motivated by the continuous setting, we approximate the stability threshold using  $\lambda = \lambda_A^\nabla + \lambda_B^\nabla$  where  $\lambda_A^\nabla$  and  $\lambda_B^\nabla$  are the smallest eigenvalues of  $L_A^\nabla$  with respect to  $M_A^\nabla$ , and  $L_B^\nabla$  with respect to  $M_B^\nabla$ , respectively. As both mass matrices approximate the same continuous volume form, we expect this value to provide a close approximation of the true stability threshold.

### D Alternative Surface Connections

We describe two discrete connections with curvature  $\Omega = \frac{1}{2} \Omega^{LC}$ : one based on concentrating and redistributing the curvature of the Levi-Civita connection and another based on spin structures. Below,  $M$  is a triangle mesh describing one of the two input surfaces, with  $r^{LC}$  and  $\Omega^{LC}$  as in Section 3.4.1.

While we obtained similar correspondences irrespective of the choice of connection, the available multiresolution schemes differ (Section 4.5). For instance, to prolongate the sections of the bundles constructed below requires keeping track of parallel transport maps

across the hierarchy of meshes. Further refinements and generalizations of our approach may be possible by changing the surface connections and their curvature.

**A Vector Field Connection.** We can modify the discrete Levi-Civita connection by computing an “offset connection”  $\tilde{r}$  so that  $r = \tilde{r} r^{LC}$  is compatible with  $\Omega$ . Similar to our discretization of the skyscraper bundle, we first select an arbitrary face  $f_0$ , and define

$$\tilde{\Omega} = \Omega^{LC} - 2\pi(\chi(M) - 1)\delta_{f_0},$$

where  $\delta_{f_0}$  is the Kronecker delta on faces. By construction,  $\tilde{\Omega}$  sums to  $2\pi$  on  $M$ . Due to the  $2\pi$  ambiguity in the curvature of a discrete complex line bundle, it is also compatible with  $r^{LC}$ . The offset connection can now be computed by solving a Poisson equation: we set  $\tilde{r}_{ij} = e^{i\tilde{\rho}_{ij}}$  where  $\tilde{\rho}$  solves

$$\arg \min_{\tilde{\rho}} \|\tilde{\rho}\|^2 \quad \text{s.t.} \quad d_1 \tilde{\rho} = \Omega - \tilde{\Omega}. \quad (40)$$

**Spin Connections.** A canonical choice on genus zero surfaces is given by a discrete spin connection [Chern et al. 2018], which is given by computing a square-root of the Levi-Civita connection

$$r_{ij}^{\text{spin}} = \pm (r_{ij}^{LC})^{1/2}.$$

Unlike the other constructions we discussed, the construction does not depend on the choice of an arbitrary face. The signs of the square-root, however, must be chosen appropriately so that  $r^{\text{spin}}$  is compatible with  $\Omega$ . A simple spanning tree based algorithm to choosing the signs is given in [Chern et al. 2018, Algorithm 3].

### E Edge-Edge Intersections

We identify the intersection between an edge of  $A$  and an edge of  $B$  by locating zeros in “edge-edge” faces of  $A \times B$ . On such a face complex section values are interpolated using the tensor product of the basis functions along each edge, so a point  $(s, t)$  in the face is a zero if and only if

$$0 = (1-s)(1-t)e^{t(s\rho_{ij}+t\rho_{jk})}z_i + s(1-t)e^{t(-(1-s)\rho_{ij}+t\rho_{jk})}z_j + ste^{t(-(1-s)\rho_{ij}-(1-t)\rho_{jk})}z_k + (1-s)te^{t(s\rho_{ij}-(1-t)\rho_{jk})}z_l.$$

Dividing through by  $z_i e^{t(s\rho_{ij}+t\rho_{jk})}$ , we obtain

$$0 = (1-s)(1-t) + s(1-t)\frac{z_j}{r_{ij}r_{ji}z_i} + st\frac{z_k}{r_{ij}r_{jk}z_i} + (1-s)t\frac{z_l}{r_{jk}z_i}.$$

Now  $t$  is the solution to the following quadratic equation and  $s$  is the associated quotient:

$$t^2 \text{Im}(a\bar{c}) + t \text{Im}(a\bar{d} - c\bar{b}) + \text{Im}(b\bar{d}) = 0 \quad s = -\frac{\text{Re}(ct+d)}{\text{Re}(at+b)},$$

where

$$b = -1 + r_{jk}^{-1} \frac{z_j}{z_i}, \quad c = -1 + r_{jk}^{-1} \frac{z_l}{z_i}, \quad d = 1$$

$$a = -1 - b - c + r_{jk}^{-1} r_{jk}^{-1} \frac{z_k}{z_i}.$$

# Supplemental Material

To *Implicit Minimal Surfaces for Bijective Correspondences* by Etienne Corman, Yousuf Soliman, Robin Magnet, and Mark Gillespie

## A Pseudocode

This supplement provides detailed pseudocode for computing bijective correspondences via minimal surfaces.

Subroutines and quantities not defined in pseudocode are described in the list below.

- $a_{ijk}$  – area of face  $ijk$ .
- $\theta_i^{jk}$  – corner angle for vertex  $i$  in face  $ijk$ .
- $d_1 \in \mathbb{R}^{F \times E}$  – discrete exterior derivative [Desbrun et al. 2005].
- $*_1 \in \mathbb{R}^{E \times E}$  – Hodge star [Desbrun et al. 2005].
- $M_S \in \mathbb{R}^{V \times V}$  – the (scalar) vertex lumped mass matrix on triangle mesh  $S$ . Note that this is different from the complex connection mass matrix  $M_S^\nabla$  computed in Algorithm 7.
- $d_S(p, q), d_S(p, \gamma)$  – the geodesic distance along  $S$  from point  $p \in S$  to point  $q \in S$ , or to curve  $\gamma \subset S$ . (See e.g. [Crane et al. 2017].)
- $\text{LINEARSOLVE}(A, b)$  – solves a sparse linear system  $Ax = b$ .
- $\text{MINEIGENVALUE}(A, B)$  – computes the smallest eigenvalue of the pair  $A, B$ , i.e. the smallest  $\lambda$  so that exist an  $x$  with  $Ax = \lambda Bx$ .
- $\text{MINEIGENVECTOR}(x \mapsto Ax, x \mapsto Bx)$  – computes the eigenvector of the pair  $A, B$  with smallest eigenvalue. For efficiency, we formulate the problem via the operators  $A, B$  and avoid assembling the whole product space matrices.
- $\text{LBFGS}(f, \nabla f, x_0)$  – minimize  $f$  using LBFGS, starting from initial point  $x_0$ , returning optimized point  $x$
- $\text{NORMALIZETOUNITSURFACEAREA}(M)$  – scale triangle mesh  $M$  so that it has surface area 1.

*Product-space matrices.* Recall from Section 3.3 that the product mesh  $A \times B$  has vertex set  $V_{A \times B} = V_A \times V_B$ , and so it is convenient to represent discrete sections  $z \in \mathbb{C}^{V_{A \times B}}$  by matrices  $Z \in \mathbb{C}^{V_A \times V_B}$ . Similarly, the product mesh has edge set  $E_{A \times B} = E_A \times V_B \cup V_A \times E_B$ , so we can represent a connection  $r \in \mathbb{C}^{E_{A \times B}}$  by a pair of matrices  $r^{E, V} \in \mathbb{C}^{E_A \times V_B}$  and  $r^{V, E} \in \mathbb{C}^{V_A \times E_B}$ , where  $r_{e_A, v_B}^{E, V}$  gives the entry for the product-space edge  $e_A \times v_B$ , and  $r_{v_A, e_B}^{V, E}$  gives the entry for the product-space edge  $v_A \times e_B$ . We often abuse notation and write  $r_{e_A, v_B}$  for the entries of the first matrix in  $\mathbb{C}^{E_A \times V_B}$ , and write  $r_{v_A, e_B}$  for the entries of the second matrix in  $\mathbb{C}^{V_A \times E_B}$ . Finally, the product mesh has face set  $F_{A \times B} = F_A \times V_B \cup E_A \times E_B \cup V_A \times F_B$ , and thus we write vectors  $\Omega \in \mathbb{R}^{F_{A \times B}}$  as triplets of matrices in  $\mathbb{R}^{F_A \times V_B}$ , etc.. We write the  $i$ th row of a matrix  $M$  as  $M_{i, \bullet}$  and the  $j$ th column as  $M_{\bullet, j}$ .

---

### Algorithm 1 SURFACECONNECTION( $S$ )

---

**Input:** A triangle meshes  $S = (V, E, F)$  with edge lengths  $\ell$ .

**Output:** A connection  $r \in \mathbb{C}^E$  and compatible curvature  $\Omega \in \mathbb{R}^F$  such that  $\sum_f \Omega_f = 2\pi$

- 1:  $\Omega_{ijk} \leftarrow \frac{1}{2}(\tilde{\theta}_i^{jk} + \tilde{\theta}_j^{ki} + \tilde{\theta}_k^{ij} - \pi)$  ▷for every face  $ijk \in F$
  - 2: Pick an arbitrary face  $f_0 \in F$
  - 3:  $L \leftarrow d_1 *_1^{-1} d_1^\top$  ▷2-form Laplacian on  $S$
  - 4:  $u \leftarrow \text{LINEARSOLVE}(L, \Omega - 2\pi\delta_{f_0})$
  - 5:  $r_{ij} \leftarrow \exp(t *_1^{-1}(u_{jil} - u_{ijk}))$  ▷for every edge  $ij \in E$
  - 6: **return**  $r, \Omega$
- 

---

### Algorithm 2 FINDTRIANGLEZERO( $ijk, \omega^0, \Omega^0, |z|$ )

---

**Input:** A triangle  $ijk$  with a rotation  $\omega_{ij}^0 \in [-\pi, \pi)$  per edge  $ij$ , the triangle Gaussian curvature  $\Omega_{ijk}^0 \in [-\pi, \pi)$  and field magnitude  $|z_i| \in \mathbb{R}_{>0}$  per vertex  $i$ . The triangle must be singular of index  $\pm 1$ , i.e.  $\omega_{ij}^0 + \omega_{jk}^0 + \omega_{ki}^0 = \pm 2\pi$

**Output:** The barycentric coordinates  $(b_i, b_j, b_k)$  of the zero.

- 1: **for**  $t = 0 \dots 1$  **do** ▷Interpolate from flat to curved triangle
  - 2:  $\omega_{ij} \leftarrow \omega_{ij}^0 + (1-t)\frac{1}{3}(\Omega_{ijk}^0 - 2\omega_{ij}^0 + \omega_{jk}^0 + \omega_{ki}^0)$
  - 3:  $\omega_{jk} \leftarrow \omega_{jk}^0 + (1-t)\frac{1}{3}(\Omega_{ijk}^0 + \omega_{ij}^0 - 2\omega_{jk}^0 + \omega_{ki}^0)$
  - 4:  $\omega_{ki} \leftarrow \omega_{ki}^0 + (1-t)\frac{1}{3}(\Omega_{ijk}^0 + \omega_{ij}^0 + \omega_{jk}^0 - 2\omega_{ki}^0)$
  - 5:  $\Omega_{ijk} \leftarrow t\Omega_{ijk}^0$
  - 6: Find  $b_j, b_k$  solution of Equation 28 using Newton method.
  - 7: **return**  $(1 - b_j - b_k, b_j, b_k)$
- 

---

### Algorithm 3 MAPVERTEX( $A, B, r^B, \Omega^B, Z, v_A$ )

---

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$ , with the connection  $r \in \mathbb{C}^{E_B}$  on  $B$ , curvature  $\Omega \in \mathbb{R}^{F_B}$  on  $B$ , a section  $Z \in \mathbb{R}^{V_A \times V_B}$  encoding a map from  $A \rightarrow B$ , and a vertex  $v_A \in V_A$  which we would like to map.

**Output:** The image of  $v_A$  on  $B$ , given as a face  $ijk \in F_B$  and barycentric coordinates  $(b_i, b_j, b_k)$  recording specific point in face  $ijk$  that  $v_A$  is mapped to.

- 1:  $z^{(v)} \leftarrow Z_{v_A, \bullet}$  ▷Take  $v_A$ 'th row of  $Z$  as a section on  $B$
  - 2:  $\omega_{ij} \leftarrow \arg\left(\frac{z_j^{(v)}}{r_{ij}^B z_i^{(v)}}\right)$  ▷Equation 12 for each  $ij \in E_B$
  - 3:  $\text{ind}_{ijk}^z \leftarrow \frac{1}{2\pi}(d_1\omega + \Omega^B)_{ijk}$  ▷Equation 13 for each  $ijk \in F_B$
  - 4:  $ijk \leftarrow$  first face with  $\text{ind}_{ijk}^z \neq 0$
  - 5:  $(b_i, b_j, b_k) \leftarrow \text{FINDTRIANGLEZERO}(ijk, \omega, \Omega^B, |z^{(v)}|)$
  - 6: **return**  $ijk, (b_i, b_j, b_k)$
-

**Algorithm 4** COMPUTECORRESPONDENCE( $A, B, \varphi, \psi, ((l_k^A, l_k^B)), ((y_k^A, y_k^B))$ )

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$ , and optionally: vertex to face maps  $\varphi : V_A \rightarrow F_B, \psi : V_B \rightarrow F_A$ , pairs of matching landmark points  $\{(l_k^A, l_k^B)\}$  on  $A$  and  $B$  respectively, and/or pairs of matching landmark curves  $\{(y_k^A, y_k^B)\}$  on  $A$  and  $B$  respectively.

**Output:** A discrete section  $z : V_{A \times B} \rightarrow \mathbb{C}$  encoding our optimal bijection between  $A$  and  $B$

▷Normalize inputs

- 1:  $A \leftarrow \text{NORMALIZE\_TO\_UNIT\_SURFACE\_AREA}(A)$
- 2:  $B \leftarrow \text{NORMALIZE\_TO\_UNIT\_SURFACE\_AREA}(B)$
- ▷Build connections on  $A$  and  $B$  (Section 4.1)
- 3:  $r^A, \Omega^A \leftarrow \text{SURFACE\_CONNECTION}(A)$  ▷Algorithm 1
- 4:  $r^B, \Omega^B \leftarrow \text{SURFACE\_CONNECTION}(B)$
- ▷Build FEM Matrices (Section 3.4.3; Algorithm 7)
- 5:  $L_A^\nabla, M_A^\nabla \leftarrow \text{BUILD\_FEM\_CONNECTION\_MATRICES}(A, r^A, \Omega^A)$
- 6:  $L_B^\nabla, M_B^\nabla \leftarrow \text{BUILD\_FEM\_CONNECTION\_MATRICES}(B, r^B, \Omega^B)$
- ▷Build pinning potential (Section 4.6)
- 7:  $V \leftarrow [1] \in \mathbb{R}^{V_A \times V_B}$
- 8:  $\sigma_A \leftarrow 1, \sigma_B \leftarrow 1$  ▷Landmark penalty (Section 4.6)
- 9: **if** landmark points were provided **then**
- 10:   **for**  $i_A \in V_A, i_B \in V_B$  **do**
- 11:      $V_{i_A, i_B} \leftarrow 1 - \max_k \exp\left(-\frac{1}{2\sigma_A^2} d_A(i_A, l_k^A)^2 - \frac{1}{2\sigma_B^2} d_B(i_B, l_k^B)^2\right)$
- 12: **if** landmark curves were provided **then**
- 13:   **for**  $i_A \in V_A, i_B \in V_B$  **do**
- 14:      $V^C \leftarrow 1 - \max_k \exp\left(-\frac{1}{2\sigma_A^2} d_A(i_A, y_k^A)^2 - \frac{1}{2\sigma_B^2} d_B(i_B, y_k^B)^2\right)$
- 15:      $V_{i_A, i_B} \leftarrow \min(V_{i_A, i_B}, V^C)$
- ▷Find initial section  $z_0$
- 16: **if**  $\varphi$  and  $\psi$  were provided **then**
- 17:    $z_0 \leftarrow \text{MAP\_INITIALIZE}(A, B, r^A, r^B, \Omega^A, \Omega^B, \varphi, \psi)$  ▷Algorithm 5
- 18: **else**
- 19:    $z_0 \leftarrow \text{RANDOM\_COMPLEX\_MATRIX}(|V_A|, |V_B|)$
- ▷Set Ginzburg-Landau parameter based on eigenvalues of  $A$  and  $B$  (Section 4.2)
- 20:  $\lambda \leftarrow 100(\text{MIN\_EIGENVALUE}(L_A^\nabla, M_A^\nabla) + \text{MIN\_EIGENVALUE}(L_B^\nabla, M_B^\nabla))$
- ▷Optimize Ginzburg-Landau energy (Algorithm 6)
- 21:  $z \leftarrow \text{LBFGS}(z \mapsto \text{GINZBURG\_LANDAU}(A, L_A^\nabla, M_A^\nabla, B, L_B^\nabla, M_B^\nabla, z, \lambda, V), z_0)$
- 22: **return**  $z$

**Algorithm 5** MAPINITIALIZE( $A, B, r^A, r^B, \Omega^A, \Omega^B, \varphi, \psi$ )

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$  with edge lengths  $\ell_A, \ell_B$ , the surface mesh connections  $r^A, r^B$  with curvatures  $\Omega^A \in \Omega^2(A)$  and  $\Omega^B \in \Omega^2(B)$ , and vertex to face maps  $\varphi : V_A \rightarrow F_B, \psi : V_B \rightarrow F_A$

**Output:** A discrete section  $z : V_{A \times B} \rightarrow \mathbb{C}$  approximating the graphs of  $\varphi$  and  $\psi$

▷First, build a connection  $r^{\varphi, \psi}$  on the product space

- 1:  $L_A \leftarrow d_1^A (*_1^A)^{-1} (d_1^A)^\top$  ▷2-form Laplacian on  $A$
- 2:  $L_B \leftarrow d_1^B (*_1^B)^{-1} (d_1^B)^\top$  ▷2-form Laplacian on  $B$
- 3: **for**  $v_A \in V_A$  **do** ▷Rig connection on  $B$ -slices to map  $v_A$  to  $\varphi(v_A)$
- 4:    $\tilde{\Omega}_{v_A, \bullet} \leftarrow 2\pi\delta_\varphi(v_A)$  ▷target slice curvature
- 5:    $u \leftarrow \text{LINEAR\_SOLVE}(L_B, \Omega^B - \tilde{\Omega}_{v_A, \bullet})$
- 6:    $\rho \leftarrow (*_1^B)^{-1} (d_1^B)^\top u$
- 7:    $r_{(v_A, ij)}^{\varphi, \psi} \leftarrow \exp(i\rho_{ij}) r_{ij}^B$  ▷for every edge  $ij \in E_B$
- 8: **for**  $v_B \in V_B$  **do** ▷Rig connection on  $A$ -slices to map  $v_B$  to  $\psi(v_B)$
- 9:    $\tilde{\Omega}_{\bullet, v_B} \leftarrow 2\pi\delta_\psi(v_B)$  ▷target slice curvature
- 10:    $u \leftarrow \text{LINEAR\_SOLVE}(L_A, \Omega^A - \tilde{\Omega}_{\bullet, v_B})$
- 11:    $\rho \leftarrow (*_1^A)^{-1} (d_1^A)^\top u$
- 12:    $r_{(v_B, ij)}^{\varphi, \psi} \leftarrow \exp(i\rho_{ij}) r_{ij}^A$  ▷for every edge  $ij \in E_A$
- 13: **return**  $\text{MIN\_EIGENVECTOR}($   
 $Z \mapsto \text{SLICEWISE\_CONNECTION\_LAPLACIAN}(A, B, r^{\varphi, \psi}, \tilde{\Omega}, Z),$   
 $Z \mapsto \text{APPLY\_MASS\_MATRIX}(A, B, Z))$

**Algorithm 6** GINZBURGLANDAU( $A, L_A^\nabla, M_A^\nabla, B, L_B^\nabla, M_B^\nabla, z, \lambda, V$ )

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$  and  $B = (V_B, E_B, F_B)$  with connection Laplacians  $L_A^\nabla \in \mathbb{C}^{V_A \times V_A}, L_B^\nabla \in \mathbb{C}^{V_B \times V_B}$ , scalar mass matrices  $M_A \in \mathbb{R}^{V_A \times V_A}, M_B \in \mathbb{R}^{V_B \times V_B}$ , as well as the current section  $z \in \mathbb{C}^{V_A \times V_B}$ , the Ginzburg-Landau parameter  $\lambda \in \mathbb{R}_{>0}$ , and the pinning potential  $V \in \mathbb{R}^{V_A \times V_B}$ .

**Output:** The Ginzburg-Landau energy  $\mathcal{GL}_\lambda$  and its gradient  $\nabla_z \mathcal{GL}_\lambda$

- 1:  $U \leftarrow 0 \in \mathbb{R}^{V_A \times V_B}$
- 2: **for**  $i \in V_A, j \in V_B$  **do**
- 3:    $U_{i,j} \leftarrow |z_{i,j}|^2 - V_{i,j}$
- 4:  $\mathcal{GL}_\lambda \leftarrow \frac{1}{2} \text{Re tr} \left[ z^\dagger \left( L_A^\nabla z \left( M^\nabla B \right)^\top + M_A^\nabla z \left( L_B^\nabla \right)^\top \right) \right]$  ▷Eq.26
- 5:  $\nabla_z \mathcal{GL}_\lambda \leftarrow L_A^\nabla z \left( M_B^\nabla \right)^\top + M_A^\nabla z \left( L_B^\nabla \right)^\top$  ▷Eq.27
- 6:  $\nabla_z \mathcal{GL}_\lambda \leftarrow \nabla_z \mathcal{GL}_\lambda + \lambda \left( M_A U M_B^\top \right) \odot z$  ▷element-wise product
- 6: **return**  $\mathcal{GL}_\lambda, \nabla_z \mathcal{GL}_\lambda$

**Algorithm 7** BUILDFEMCONNECTIONMATRICES( $A, r, \Omega$ )

**Input:** A triangle mesh  $S = (V, E, F)$  with connection  $r \in \mathbb{C}^E$  and curvature  $\Omega \in \mathbb{R}^F$ .

**Output:** The connection Laplacian  $L^\nabla$

```

1:  $L^\nabla, M^\nabla \leftarrow 0 \in \mathbb{C}^{V \times V}, 0 \in \mathbb{R}^{V \times V}$ 
2: for corner  $ijk \in S$  do
     $\triangleright$ Laplace matrix
3:  $\alpha \leftarrow (\ell_{ij}^2 - \ell_{jk}^2 + \ell_{ki}^2)/2$   $\triangleright \alpha = \langle p_j - p_i, p_k - p_i \rangle$ 
4:  $w_L \leftarrow \frac{\bar{r}_{jk}}{a_{ijk}} * \left[ (\ell_{ij}^2 + \ell_{ki}^2) * f_1(\Omega_{ijk}) + \alpha f_2(\Omega_{ijk}) \right]$   $\triangleright$ Algs. 9,10
5:  $L_{jk}^\nabla + = w_L, L_{kj}^\nabla + = \overline{w_L}, L_{ii}^\nabla + = \frac{1}{4a_{ijk}} \left( \ell_{jk}^2 + \Omega_{ijk}^2 \frac{\ell_{ij}^2 + \alpha + \ell_{ki}^2}{90} \right)$ 
     $\triangleright$ Mass matrix
6:  $w_M \leftarrow a_{ijk} \bar{r}_{jk} f_0(\Omega_{ijk})$   $\triangleright$ Alg. 8
7:  $M_{jk}^\nabla + = w_M, M_{kj}^\nabla + = \overline{w_M}, M_{ii}^\nabla + = \frac{1}{6} a_{ijk}$ 
8: return  $L^\nabla, M^\nabla$ 

```

**Algorithm 8**  $f_0(s)$   $\triangleright$ Helper for BUILDFEMCONNECTIONMATRICES

**Input:** A real number  $s \in \mathbb{R}$ .

**Output:** The value  $f_0(s)$  used in Knöppel et al. [2013, Eq.17].

$\triangleright$ The function  $f_0(s)$  has a removable singularity at  $s = 0$ . One can use the Chebyshev expansion provided by Knöppel et al., or the simple Taylor expansion given here

```

1: if  $|s| < \frac{1}{10}$  then
2:   return  $\frac{1}{12} - \frac{s^2}{360} + \frac{s^4}{20160} + l \left( \frac{s}{60} - \frac{s^3}{2520} + \frac{s^5}{181440} \right)$ 
3: else
4:   return  $\frac{1}{3s^4} (-6 - 6ls + 3s^2ls^3 + 6e^{ls})$ 

```

**Algorithm 9**  $f_1(s)$   $\triangleright$ Helper for BUILDFEMCONNECTIONMATRICES

**Input:** A real number  $s \in \mathbb{R}$ .

**Output:** The value  $f_1(s)$  defined by Knöppel et al. [2013, §6.1.1].

```

1: if  $|s| < \frac{1}{10}$  then  $\triangleright$ Taylor expansion to handle singularity
2:   return  $\frac{s^2}{120} - \frac{s^4}{2688} + \frac{s^6}{129600} + l \left( -\frac{s}{24} + \frac{s^3}{504} - \frac{s^5}{17280} \right)$ 
3: else
4:   return  $\frac{1}{s^4} \left( 3 + ls + \frac{s^4}{24} - l \frac{s^5}{60} + (-3 + 2ls + \frac{s^2}{2})e^{ls} \right)$ 

```

**Algorithm 10**  $f_2(s)$   $\triangleright$ Helper for BUILDFEMCONNECTIONMATRICES

**Input:** A real number  $s \in \mathbb{R}$ .

**Output:** The value  $f_2(s)$  defined by Knöppel et al. [2013, §6.1.1].

```

1: if  $|s| < \frac{1}{10}$  then  $\triangleright$ Taylor expansion to handle singularity
2:   return  $-\frac{1}{4} + \frac{s^2}{45} - \frac{s^4}{1120} + \frac{s^6}{56700} + l \left( -\frac{s}{24} + \frac{5s^3}{1008} - \frac{7s^5}{51840} \right)$ 
3: else
4:   return  $\frac{1}{s^4} \left( 4 + ls - l \frac{s^3}{6} - \frac{s^4}{12} + l \frac{s^5}{30} + (-4 + 3ls + s^2)e^{ls} \right)$ 

```

**Algorithm 11** SLICEWISECONNECTIONLAPLACIAN( $A, B, r, \Omega, Z$ )

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$ ,  $B = (V_B, E_B, F_B)$ , a product space connection  $r \in \mathbb{C}^{E_A \times B}$  with curvature  $\Omega \in \mathbb{R}^{F_A \times B}$ , and a section  $Z \in \mathbb{C}^{V_A \times V_B}$

**Output:** Applies the product space connection Laplacian (Equation 34) for connection  $r$  to the section  $Z$  to obtain a new section  $Z'$ .

```

1: for  $v \in V_A, w \in V_B$  do
2:    $L_{B,-}^{\nabla,v} \leftarrow$  BUILDFEMCONNECTIONMATRICES( $B, r_{v,\bullet}, \Omega_{v,\bullet}$ )
3:    $L_{A,-}^{\nabla,w} \leftarrow$  BUILDFEMCONNECTIONMATRICES( $A, r_{\bullet,w}, \Omega_{\bullet,w}$ )
4:    $Z'_{v,w} \leftarrow (M_A)_{v,v} \left( L_{B,-}^{\nabla,v} Z^\top \right)_{w,v} + (M_B)_{w,w} \left( L_{A,-}^{\nabla,w} Z \right)_{v,w}$ 
5: return  $Z'$ 

```

**Algorithm 12** APPLYMASSMATRIX( $A, B, Z$ )

**Input:** Triangle meshes  $A = (V_A, E_A, F_A)$ ,  $B = (V_B, E_B, F_B)$ , and a section  $Z \in \mathbb{C}^{V_A \times V_B}$

**Output:** Applies the product space mass matrix to the section  $Z$ .

```

1: return  $M_A Z' M_B^\top$ 

```

## B Parameters and Meshes Statistics

The parameters used for each figure are reported in Table 1.

Table 1. Parameters for each figure: number of variables, Ginzburg-Landau parameter schedule, pinning parameter (if applicable) and usage of intrinsic triangulation.

	$ V_A  \times  V_B $	$\lambda = t\lambda_0$	$\sigma$	iDT
Fig. 1	4643 × 4818	10, 50, 100	×	✓
Fig. 4	1006 × 998	10, 100	×	✓
Fig. 5 <i>left</i>	3001 × 3001	10, 100	×	×
Fig. 5 <i>right</i>	2999 × 2999	10, 100	×	×
Fig. 12	502 × 2582	100	×	×
Fig. 14	5000 × 4871	100	×	✓
Fig. 15	2485 × 2429	100	1	✓
Fig. 16	3138 × 3146	75	$1/10$	×
Fig. 17	617 × 1338	100	1	✓
Fig. 18 <i>left</i>	512 × 1000	100	×	
Fig. 18 <i>right</i>	4098 × 2500	100	×	
Fig. 19 <i>top</i>	3000 × 3000	100	×	×
Fig. 19 <i>bottom</i>	2000 × 2000	100	×	×
Fig. 13	474 × 512	50	1	×
Fig. 20 <i>left</i>	1502 × 1477	100, 50	×	✓
Fig. 20 <i>center</i>	1502 × 1477	100, 50	×	✓
Fig. 20 <i>right</i>	3000 × 3000	100, 50	×	✓
Fig. 22	954 × 921	100	$1/\sqrt{20}$	×
Fig. 21	2252 × 2277	100	×	✓
Fig. 23	3222 × 6121	10, 100	×	×
Fig. 24	4593 × 4017	100	×	×
Fig. 26 <i>top</i>	1500 × 1500	100	×	×
Fig. 26 <i>middle</i>	1500 × 1500	100	×	×
Fig. 26 <i>bottom</i>	1500 × 1500	100	×	×
Fig. 27 <i>left</i>	2000 × 2019	100	$1/4$	×
Fig. 27 <i>middle</i>	2000 × 3000	100	$1/4$	×
Fig. 27 <i>right</i>	2000 × 3000	100	$1/4$	×
Fig. 25	3863 × 3863	25, 75	×	✓