FAKULTÄT FÜR MATHEMATIK

TECHNISCHE UNIVERSITÄT MÜNCHEN

LEHRSTUHL FÜR GEOMETRIE UND VISUALISIERUNG

Discrete Spin Geometry for Surfaces

Zi Ye

Acknowledgments

Abstract

This thesis proposes a discrete framework for spin geometry, which can be roughly understood as a variant of Riemannian geometry with the structure group $SO(n)$ replaced by the spin group $Spin(n)$, for surfaces.

Specifically, we discretize the basic notions in spin geometry, such as the spin structure, spin connection and Dirac operator, over a discrete surface, called the face edge-constraint net, i.e., the polygonal surfaces with normals defined on faces. Within this framework, two types of Dirac operators, namely the intrinsic Dirac operator and the extrinsic Dirac operator, are closely related as in the smooth case. Moreover, they both induce the discrete conformal immersion with prescribed mean curvature half-density.

In addition, we consider the corresponding numerical problems and present various applications in computer graphics, shape analysis and biological imaging. For example, we build a generative model for 3D surfaces based on our framework. Since the curvature is explicitly encoded in our model, the local structure of shapes will be more precisely captured than in existing 3D machine learning models. Furthermore, our model substantially reduces the influence from translation and rotation.

Zusammenfassung

In dieser Arbeit wird ein diskreter Zugang zur Spin-Geometrie vorgestellt. Dieser Zugang kann quasi als Variante der Riemannschen Geometrie verstanden werden, wobei die Strukturgruppe $SO(n)$ durch die Spin-Gruppe $Spin(n)$ ersetzt wird. Der Fokus dieser Arbeit liegt in der Geometrie der Flächen. Insbesondere diskretisieren wir die grundlegenden Begriffe der Spin-Geometrie, wie zum Beispiel die Spin-Struktur, den Spin-Zusammenhang und den Dirac Operator über einer diskreten Fläche. Das von uns verwendete Modell für eine diskrete Fläche wird als Face-Edge-constraint Netz bezeichnet, das ist eine polygonale Fläche bei welcher jedem Polygon eine Normale zugeordnet wird. Auch in diesem Rahmen gibt es zwei Varianten für den Dirac Operator, nämlich den intrinsischen und den extrinsischen Dirac Operator. Wie auch in der glatten Theorie sind beide Operatoren eng verwandt. Darüber hinaus induzieren beide die diskret-konforme Immersion mit vorgeschriebener Halbdichte der mittleren Krümmung.

Anschließend betrachten wir die entsprechenden numerischen Probleme und stellen verschiedene Anwendungen in den Bereichen Computergrafik, Formenanalyse und biologischer Bildgebung vor. Insbesondere bauen wir auf Basis unseres Frameworks ein generatives Modell für 3D-Flächen auf. Anders als bei bisherigen 3D Modellen für maschinelles Lernen, wird die mittlere Krümmung in unserem Modell explizit kodiert. Dadurch wird die lokale Struktur der Formen präziser erfasst. Des Weiteren wird der Einfluss von Translationen und Rotationen durch unser Modell deutlich reduziert.

Contents

1 Introduction

The Dirac operator was originally invented by Dirac as the square root of the Laplace operator. It then became a fundamental equation in quantum field theory to describe the behavior of spin $\frac{1}{2}$-particles. In the 1950s, Atiyah and Singer generalized the formalism from 4-dimensional Minkowski space to the semi-Riemmnian manifolds. It turned out to be the key tool to study the symbol of elliptic operators on manifolds.

Generally speaking, in the realm of spin geometry people consider the $\mathrm{Pin}(n)$ groups ($\mathrm{Spin}(n)$ for oriented manifolds) as the underlying structure groups in place of the conventional orthogonal groups $\mathrm{O}(n)$ ($\mathrm{SO}(n)$ for oriented manifolds). Though it looks like a small change ($\mathrm{Spin}(n)$ is the two-fold covering of $\mathrm{SO}(n)$), the tools in spin geometry turned out to be very powerful in solving a large variety of problems, e.g., determining the metrics with positive scalar curvature [GL83] and proving the positive mass conjecture [Wit81].

Besides these famous results, people realized that spin geometry is a convenient tool to study surface immersion problems, e.g., in $\mathbb{R}^3$ and S^3. This can be actually traced back to the celebrated Weierstrass representation of minimal surfaces. In fact, every minimal surface can be represented by one holomorphic and one meromorphic function. The formulation can be generalized to conformal immersions with any prescribed mean curvature and in more complicated spaces [Ken79; KS96; Fri98; RR13; Mor04]. Kamberov and Pinkall [KPP98] tackled the almost equivalent problem from the extrinsic aspect. Instead of the complex Clifford representation used by others, they used the real representation, which is isomorphic to the quaternionic one, and constructed an quaternionic extrinsic Dirac operator based on the immersed surfaces. Due to the use of a quaternionic representation, the formulation of the conformal immersion became simply a scale rotation in $\mathbb{R}^3$, thereby giving a more intuitive geometric picture. By observing the formulas and properties of both operators it is not so surprising to see that the intrinsic and extrinsic Dirac operators are closely related.

In addition to the aforementioned smooth theories, spin geometry has been developed in the discrete setting, from both the theoretical and applied viewpoints. In theory, the idea has been used for the discrete integrable surfaces [Bob; BP99; Hof99; SB08; HSFW16; Lam16]. Even though the concept of spin geometry was rarely mentioned in these papers, the 2×2 complex matrices, which were frequently used as the orthonormal frame in these papers, are actually spinor fields. For applications especially in the field of computer graphics, Crane et al. [CPS11; CPS13] discretized the extrinsic Dirac operator and obtained the applications such as curvature painting and conformal surface fairing.

In this thesis, we establish a new framework for discrete spin geometry for surface immersion problems in the spirit of the discrete differential geometry, namely we not

only discretize the Dirac equation, as was done in [CPS11], but discretize the whole theory.

In Chapter 2 we present some of the fundamentals of the smooth theory, in which we reformulate the results in previous work in a systematic way.

In Chapter 3 we present the framework of our discrete theory. The underlying surface is called the face edge-constraint net, which involves a polygonal surface with normals defined on the faces. The integrated mean curvature, which arises naturally in this setting by means of Steiner's formula, can be manipulated by the discrete Dirac equation. One can use this idea to construct discrete minimal surfaces and their associated families. In the Section 3.3 we consider a more abstract intrinsic net, i.e., a cell complex with a length assigned to each edge. A discrete spinor bundle, together with a spinor connection, can then be constructed over this net. Furthermore, several results coming from the smooth theory can be shown to still hold in our setting: an even Euler characteristic implies the existence of a spin structure and the first Betti number determines the number of spin structures. The discrete intrinsic Dirac operator follows naturally and one can build a realization of the intrinsic net with prescribed integrated mean curvature in $\mathbb{R}^3$, which is a face edge-constraint net, by solving the Dirac equation. In the end we will see, just as in the smooth case, that there is a nice connection between the extrinsic and intrinsic Dirac operators.

In Chapter 4 we analyze and develop the numerical methods based on the theory. Since the face-based operator fails to be numerically stable in some circumstances, due to there being too many degrees of freedom for the normals, we propose several variant formulations that behave well in practice, enabling us to calculate the immersions corresponding to the Dirac eigenfunctions. This further led us to some applications in computer graphics and shape analysis, such as conformal parameterization, shape matching, and conformal shape filtering. Finally, we build an intrinsic representation of discrete shapes in $\mathbb{R}^3$ which can be consistently applied to a collection of shapes with fixed topology. We have shown that this representation can be used as a data structure for machine learning of 3D shapes. Based on this finding we can create a 3D shape generative model that learns and generates the curvature and the conformal structure.

2 Smooth theory

Section 2.5 is taken from a preprint submitted by the author [HY18].

2.1 From Spin Transformation to Spinor Weierstrass

Suppose given an immersion of surface into $\mathbb{R}^3$. Its differential is actually a $\mathbb{R}^3$-valued one-form $\mathrm{d}f$. We always consider $\mathbb{R}^3$ as embedded in the imaginary part of $\mathbb{H}$. Hence df is actually a $\mathbb{H}$-valued one-form, which has the wedge product defined in [KPP98]

$$\omega \wedge \eta(X, Y) := \frac{1}{2} \left(\omega(X)\eta(Y) - \omega(Y)\eta(X) \right).$$

Suppose given a quaternion-valued function $\phi : M \to \mathbb{H}$. It can be thought of as a continuously varying rotation. We apply it to every tangent plane of f:

$$\tilde{d}f = \overline{\phi} \cdot \mathrm{d}f \cdot \phi,$$

the resulting $\mathbb{R}^3$-valued one-form is closed, i.e., $\mathrm{d}\tilde{d}f = 0$, if and only if

$$D_f \phi = \rho \phi,$$

where ρ is a real-valued function (see [CPS11]).
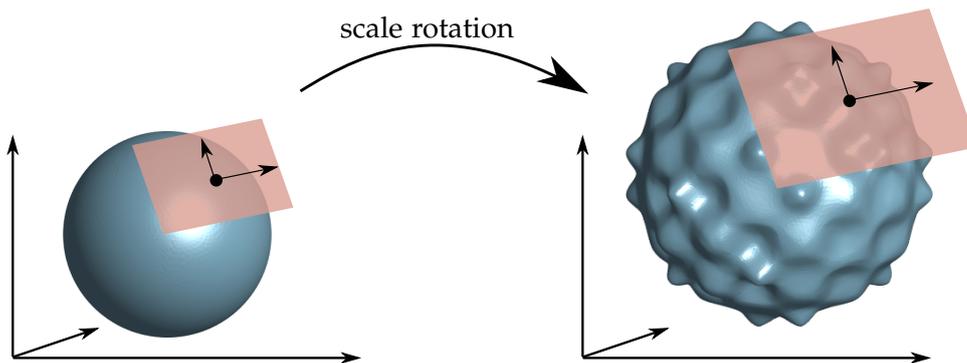
Figure 2.1: Spin transformation.

Now, suppose we only have a Riemannian surface M without the immersion f. Is it possible to do something similar? Recall that at each point $p \in M$ there is a chart

$f : U \to \mathbb{R}^2$. Let us consider $\mathbb{R}^2$ as a plane embedded in the $i - j$-plane of $\mathbb{H}$. Therefore, in this local neighbourhood U_p we can define a quaternion-valued function $\phi_p \to \mathbb{H}$ and apply the corresponding spin transformation locally:

$$d\tilde{f}_U = \overline{\phi_U} \cdot df_U \cdot \phi_U.$$

In order to extend globally this transformation, we need some compatibility condition so that it is well-defined on the overlap between any two charts U and V, i.e.,

$$d\tilde{f}_U = d\tilde{f}_V \Rightarrow \overline{\phi_U} \cdot df_U \cdot \phi_U = \overline{\phi_V} \cdot df_V \cdot \phi_V.$$

If $df_U = \overline{g_{UV}} \cdot df_V \cdot g_{UV}$, then

$$g_{UV} \cdot \phi_U = \phi_V. \tag{2.1}$$

Therefore, to have a globally well-defined 'spin transformation' for an intrinsic Riemannian surface, we need

1. Some quaternion-valued functions ϕ_U for every charts.

2. They should satisfy the compatibility condition (2.1) on the overlaps.
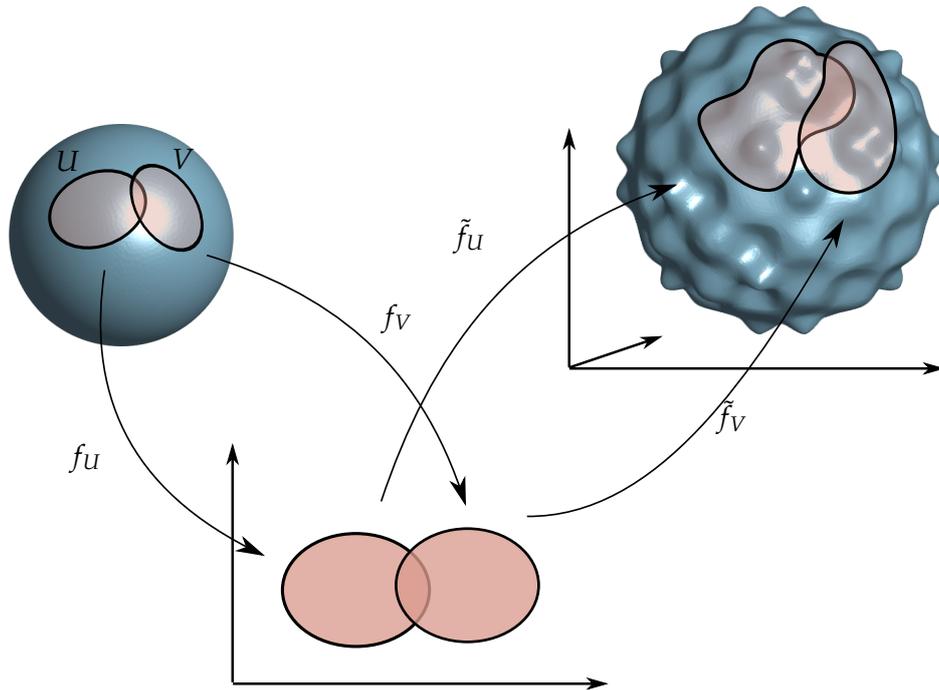
Figure 2.2: Intrinsic analogue of the spin transformation.

In fact, a collection of such (ϕ_U, g_{UV}) is called a spinor field, which can be rigorously formulated as a section of the spinor bundle by setting g_{UV} to be the transition functions

of the 2-dimensional real spinor bundle. We call the intrinsic version of the spin trans-formation the spinorial Weierstrass representation, because it generalizes the classical Weierstrass representation of the minimal surface.

In the following sections, we will establish the these theorems rigorously.

2.2 The spinorial Weierstrass representation

This section is in fact a reformulation of [Fri98]. However, instead of using the complex Clifford representation, we use the real representation, which gives a clearer geomet-ric picture, because the transformation can be understood as a rotation in $\mathbb{R}^3$ using quaternions, see Section 2.1 above.

We consider a simply connected two-dimensional manifold X with Riemmnian metric. In this case, the spinor bundle $\mathcal{S}$ is endowed with a quaternion structure. One can actually construct a quaternion hermitian form $\langle \cdot, \cdot \rangle : \Gamma(\mathcal{S}) \times \Gamma(\mathcal{S}) \to \mathbb{H}$ in the following way: In any local neighborhood $U \subset X$ we choose a section $p_U \in P_{\mathrm{Spin}}U$ and take any trivialization

$$f_U : \mathcal{S}_U \cong U \times \mathbb{H}$$
$$\phi = ((p_U)_x, v) \mapsto (x, q),$$

the quaternion hermitian form on U can be simply defined by

$$\langle q_1, q_2 \rangle := \overline{q_1} \cdot q_2$$

under the identification of f_U. To extend it from U to a nearby neighborhood, let V be any neighborhood with a non-empty overlap with U. Assume that p_U and p_V are related by $p_U \cdot g = p_V$ over $U \cap V$, where $g : U \cap V \to \mathrm{Spin}(n)$ is the transition function. Then choose the identification $f_V : \mathcal{S}_V \cong V \times \mathbb{H}$ such that

$$\phi = ((p_V)_x, v) \mapsto (x, g^{-1} \cdot \pi_2(f_U(\phi)))$$

is satisfied on $U \cap V$, where π_2 is the projection onto the second component. This guarantees that f_U and f_V coincide on the overlap and hence the quaternion hermitian form can be extended to V. We can extend $\langle \cdot, \cdot \rangle$ to the whole manifold in the same way and it is clear that it is unique up to a global conjugation

$$\langle \cdot, \cdot \rangle \mapsto \overline{q} \cdot \langle \cdot, \cdot \rangle \cdot q,$$

where q is a unit quaternion. It will be clear later that this ambiguity actually corresponds to the Euclidean motion of the immersion of the surface in $\mathbb{R}^3$.

Any spinor section $\phi \in \Gamma(S)$ induces an $\mathbb{H}$-valued one-form $\Theta \in \mathbb{H} \otimes \Gamma(T^*X)$ by $\Theta(Y) := \langle \phi, Y \cdot \phi \rangle$, $Y \in \Gamma(TX)$. Indeed, Θ is $\mathbb{R}^3$-valued by a simple calculation in local coordinates. Now we are going to prove that the closing condition of Θ is equivalent to the Dirac equation.

Theorem 2.1. $d\Theta = 0$ *if and only if* $D\phi = \rho\phi$, *where* $\rho : X \to \mathbb{R}$ *is any real-valued scalar function.*

Proof. Choose any oriented orthonomal frame $\{e_1, e_2\}$ and let $\{\omega_1, \omega_2\}$ be its dual frame. In this frame, Θ has the form $\Theta = \langle \phi, e_1 \cdot \phi \rangle \omega_1 + \langle \phi, e_2 \cdot \phi \rangle \omega_2$. It follows that

$$
\begin{aligned}
d\Theta &= d\langle \phi, e_1 \cdot \phi \rangle \omega_1 + \langle \phi, e_1 \cdot \phi \rangle d\omega_1 + d\langle \phi, e_2 \cdot \phi \rangle \omega_2 + \langle \phi, e_1 \cdot \phi \rangle d\omega_2 \\
&= \langle \phi, \nabla e_1 \cdot \phi \rangle \wedge \omega_1 + \langle \phi, e_1 \cdot \nabla \phi \rangle \wedge \omega_1 + \langle \nabla \phi, e_1 \cdot \phi \rangle \wedge \omega_1 + \langle \phi, e_1 \cdot \phi \rangle d\omega_1 \\
&\quad + \langle \phi, \nabla e_2 \cdot \phi \rangle \wedge \omega_2 + \langle \phi, e_2 \cdot \nabla \phi \rangle \wedge \omega_2 + \langle \nabla \phi, e_2 \cdot \phi \rangle \wedge \omega_2 + \langle \phi, e_2 \cdot \phi \rangle d\omega_2 \quad (2.2) \\
&= \left(\langle \phi, \nabla_{e_2} e_1 \cdot \phi \rangle + \langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle + \langle \nabla_{e_2} \phi, e_1 \cdot \phi \rangle \right) \omega_2 \wedge \omega_1 + \langle \phi, e_1 \cdot \phi \rangle d\omega_1 \\
&\quad + \left(\langle \phi, \nabla_{e_1} e_2 \cdot \phi \rangle + \langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle + \langle \nabla_{e_1} \phi, e_2 \cdot \phi \rangle \right) \omega_1 \wedge \omega_2 + \langle \phi, e_2 \cdot \phi \rangle d\omega_2
\end{aligned}
$$

Due to the orthogonality of the frame $\{e_1, e_2\}$, we have $\nabla_{e_1} e_2 = \Gamma^1_{12} e_1$ and $\nabla_{e_2} e_1 = \Gamma^2_{21} e_2$. For any one-form ω and two vector fields Y, Z we have the identity [Lee01, Prop. 12.17]

$$
d\omega(Y, Z) = Y(\omega(Z)) - Z(\omega(Y)) - \omega([Y, Z])
$$

Then it follows that

$$
\begin{aligned}
d\omega_1(e_1, e_2) &= -\omega_1(\nabla_{e_1} e_2) = -\Gamma^1_{12} \\
d\omega_2(e_2, e_1) &= -\omega_2(\nabla_{e_2} e_1) = -\Gamma^2_{21}
\end{aligned}
$$

Hence we get $d\omega_1 = -\Gamma^1_{12}\omega_1 \wedge \omega_2$ and $d\omega_2 = -\Gamma^2_{21}\omega_2 \wedge \omega_1$. This leads to some cancellations in (2.2) as follows:

$$
\begin{aligned}
\langle \phi, \nabla_{e_2} e_1 \cdot \phi \rangle \omega_2 \wedge \omega_1 + \langle \phi, e_2 \cdot \phi \rangle d\omega_2 &= \left(\langle \phi, \Gamma^2_{21} e_2 \cdot \phi \rangle - \langle \phi, e_2 \cdot \phi \rangle \Gamma^2_{21} \right) \omega_2 \wedge \omega_1 \\
&= 0,
\end{aligned}
$$

$$
\begin{aligned}
\langle \phi, \nabla_{e_1} e_2 \cdot \phi \rangle \omega_1 \wedge \omega_2 + \langle \phi, e_1 \cdot \phi \rangle d\omega_1 &= \left(\langle \phi, \Gamma^1_{12} e_1 \cdot \phi \rangle - \langle \phi, e_1 \cdot \phi \rangle \Gamma^1_{12} \right) \omega_1 \wedge \omega_2 \\
&= 0
\end{aligned}
$$

The remaining terms of (2.2) become

$$
\begin{aligned}
d\Theta &= \left(\langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle + \langle \nabla_{e_2} \phi, e_1 \cdot \phi \rangle \right) \omega_2 \wedge \omega_1 \\
&\quad + \left(\langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle + \langle \nabla_{e_1} \phi, e_2 \cdot \phi \rangle \right) \omega_1 \wedge \omega_2 \\
&= \left(-\langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle + \langle e_1 \cdot \nabla_{e_2} \phi, \phi \rangle + \langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle - \langle e_2 \cdot \nabla_{e_1} \phi, \phi \rangle \right) \omega_1 \wedge \omega_2 \\
&= \left(-\langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle + \overline{\langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle} + \langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle - \overline{\langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle} \right) \omega_1 \wedge \omega_2 \\
&= 2 \left(-\text{Im}\langle \phi, e_1 \cdot \nabla_{e_2} \phi \rangle + \text{Im}\langle \phi, e_2 \cdot \nabla_{e_1} \phi \rangle \right) \omega_1 \wedge \omega_2 \\
&= 2 \left(\text{Im}\langle \phi, -e_1 \cdot \nabla_{e_2} \phi + e_2 \cdot \nabla_{e_1} \phi \rangle \right) \omega_1 \wedge \omega_2 \\
&= -2 \left(\text{Im}\langle \phi, D\phi \rangle \right) \omega_1 \wedge \omega_2
\end{aligned}
$$

Therefore, $d\Theta = 0$ is equivalent to $\text{Im}\langle \phi, D\phi \rangle = 0$, which means $D\phi = \rho\phi$. $\qquad\square$

Suppose X is simply connected. The condition $d\Theta = 0$ holds if and only if there exists a quaternion-valued function $f : X \to \mathbb{R}^3$ with $df = \Theta$. Hence we think of the equation $d\Theta = 0$ as the local closing condition of an immersed surface in $\mathbb{R}^3$, namely the Gauss–Codazzi equation. Therefore we get the following corollary.

Corollary 2.2. *Let X be an oriented surface with Riemannian metric. Every solution ϕ of the Dirac equation*

$$D\phi = \rho\phi \tag{2.3}$$

gives a conformal immersion $f : X \to \mathbb{R}^3$ if M is simply connected. Otherwise, ϕ gives a conformal immersion of its universal covering $f : \tilde{X} \to \mathbb{R}^3$.

Given that Θ indeed comes from the differential of the immersion of a surface, we are going to compute its mean curvature. It turns out that it is determined by the function ρ.

It is easy to see that the first fundamental form with respect to $\{e_1, e_2\}$ is

$$\text{I} = \begin{pmatrix} |\phi|^4 & 0 \\ 0 & |\phi|^4 \end{pmatrix}$$

and the Gauss map of the immersion $f : X \to \mathbb{R}^3$ is given by $n = \langle \phi, e_1 e_2 \cdot \phi \rangle / |\phi|^2$. The second fundamental form is then given by

$$\text{II}(Y, Z) = \langle Y(\langle \phi, Z \cdot \phi \rangle), n \rangle_{\mathbb{R}^3}.$$

Theorem 2.3. *The immersed surface $f : M \to \mathbb{R}^3$ obtained by the (2.3) has the mean curvature*

$$H = \frac{\rho}{|\phi|^2}. \tag{2.4}$$

Proof. The mean curvature H is given by

$$H = \frac{\text{TrII}}{2|\phi|^4} = \frac{1}{2|\phi|^4} \left(\langle e_1(\langle \phi, e_1 \cdot \phi \rangle), n \rangle_{\mathbb{R}^3} + \langle e_2(\langle \phi, e_2 \cdot \phi \rangle), n \rangle_{\mathbb{R}^3} \right). \tag{2.5}$$

The left component in the bracket is

$$e_1(\langle \phi, e_1 \cdot \phi \rangle) + e_2(\langle \phi, e_2 \cdot \phi \rangle) = \langle \nabla_{e_1}\phi, e_1 \cdot \phi \rangle + \langle \phi, \nabla_{e_1} e_1 \cdot \phi \rangle + \langle \phi, e_1 \cdot \nabla_{e_1}\phi \rangle$$
$$+ \langle \nabla_{e_2}\phi, e_2 \cdot \phi \rangle + \langle \phi, \nabla_{e_2} e_2 \cdot \phi \rangle + \langle \phi, e_2 \cdot \nabla_{e_2}\phi \rangle \tag{2.6}$$

Observe that $\langle \phi, \nabla_{e_i} e_j \cdot \phi \rangle$ lies in the tangent plane, hence $\langle \langle \phi, \nabla_{e_i} e_j \cdot \phi \rangle, N \rangle_{\mathbb{R}^3} = 0$. The remaining terms in (2.6) are

$$\langle \nabla_{e_1}\phi, e_1 \cdot \phi \rangle + \langle \phi, e_1 \cdot \nabla_{e_1}\phi \rangle + \langle \nabla_{e_2}\phi, e_2 \cdot \phi \rangle + \langle \phi, e_2 \cdot \nabla_{e_2}\phi \rangle$$
$$= -\langle e_1 \cdot \nabla_{e_1}\phi + e_2 \cdot \nabla_{e_2}\phi, \phi \rangle + \langle \phi, e_1 \cdot \nabla_{e_1}\phi + e_2 \cdot \nabla_{e_2}\phi \rangle$$
$$= -\langle e_1 e_2 \cdot D\phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot D\phi \rangle$$
$$= \rho \left(-\langle e_1 e_2 \cdot \phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot \phi \rangle \right) \tag{2.7}$$

Since $-\langle e_1 e_2 \cdot \phi, \phi \rangle = \langle \phi, e_1 e_2 \cdot \phi \rangle = n|\phi|^2$, substituting (2.7) into (2.5) gives

$$H = \frac{1}{2|\phi|^4} \langle 2\rho n |\phi|^2, n \rangle_{\mathbb{R}^3} = \frac{\rho \langle n, n \rangle_{\mathbb{R}^3}}{|\phi|^2} = \frac{\rho}{|\phi|^2}$$

$$\square$$

2.3 The Lawson Correspondence

It is well known that minimal surfaces in S^3 are in one-to-one correspondence with constant mean curvature surfaces in $\mathbb{R}^3$ [Law+70]. We will show that the formulation in last section can be adapted to describe this correspondence.

Since S^3 is homeomorphic as a topological space to the Lie group $SU(2) \cong \mathrm{Spin}(1)$, any immersion $f : X \to S^3$ of a surface induces a Maurer–Cartan form

$$\Theta : T_p M \to \mathfrak{su}(2), \quad \Theta(Y) = f^{-1}\, \mathrm{d}f(Y)$$

for any $X, Y \in T_p M$. It is well known that the Maurer–Cartan form satisfies the Maurer–Cartan equation

$$\mathrm{d}\Theta(Y, Z) + [\Theta(Y), \Theta(Z)] = 0 \tag{2.8}$$

Note that, in the following context we will occasionally use the identification $\mathbb{R}^3 \cong \mathfrak{su}(2) \cong \mathfrak{spin}(1)$.

Theorem 2.4. *Let $\Theta \in TM \otimes \mathfrak{su}(2) \cong TX \times \mathbb{R}^3$ be a $\mathbb{R}^3$-valued one-form, defined by $\Theta(Y) = \langle \phi, Y \cdot \phi \rangle$. Then Θ satisfies the Maurer–Cartan equation if and only if*

$$D\phi + e_1 e_2 \cdot \phi = \rho \phi \tag{2.9}$$

Proof. We will again express Θ in the oriented orthonormal frame $\{e_1, e_2\}$, as in the proof of theorem 2.1, as $\Theta = \langle \phi, e_1 \cdot \phi \rangle \omega_1 + \langle \phi, e_2 \cdot \phi \rangle \omega_2$. In theorem 2.1, the differential part $\mathrm{d}\Theta$ has been derived as

$$\mathrm{d}\Theta = \left(-\langle \phi, D\phi \rangle + \langle D\phi, \phi \rangle \right) \omega_1 \wedge \omega_2$$

The Lie bracket in (2.8) can be written as $[\langle \phi, e_1 \cdot \phi \rangle, \langle \phi, e_2 \cdot \phi \rangle] \omega_1 \wedge \omega_2$. Let $\{E_1, E_2, E_3\}$ be an orthonormal frame in $\mathbb{R}^3 \cong \mathfrak{su}(2)$. We know that the Lie bracket of $\mathfrak{su}(2)$ is $[E_1, E_2] = 2E_3$. Now, since $\langle \phi, e_1 \cdot \phi \rangle$, $\langle \phi, e_2 \cdot \phi \rangle$ and $\langle \phi, e_1 e_2 \cdot \phi \rangle$ form a orthogonal frame with the same length, we have

$$[\langle \phi, e_1 \cdot \phi \rangle, \langle \phi, e_2 \cdot \phi \rangle] = 2\langle \phi, e_1 e_2 \cdot \phi \rangle = -\langle e_1 e_2 \cdot \phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot \phi \rangle$$

Overall, we obtain that

$$-\langle \phi, D\phi - e_1 e_2 \cdot \phi \rangle + \langle D\phi - e_1 e_2 \cdot \phi, \phi \rangle = 2\mathrm{Im}\langle D\phi - e_1 e_2 \cdot \phi, \phi \rangle = 0$$

which implies that $D\phi - e_1 e_2 \phi = \rho \phi$, where $\rho : X \to \mathbb{R}$ is real valued. □

The formula for the mean curvature is actually the same as in the case of $\mathbb{R}^3$.

Theorem 2.5. *Any immersion $f : X \to S^3$ obtained by eq. (2.9) has the mean curvature $H = \frac{\rho}{|\phi|^2}$.*

Proof. Since f is conformal with the scaling factor $|\phi|^4$, the first fundamental form is $\mathrm{I} = \begin{pmatrix} |\phi|^4 & 0 \\ 0 & |\phi|^4 \end{pmatrix}$. The Gauss map is given by $n_{f(p)} = (L_{f(p)})_* (\langle \phi, e_1 e_2 \cdot \phi \rangle) \frac{1}{|\phi|^2}$. This formula for H is similar as that of (2.5) but we have to replace the $\mathbb{R}^3$-vector differential with the Levi–Civita connection for S^3, denoted by $\tilde{\nabla}$, i.e.,

$$H = \frac{1}{2|\phi|^4} \left(\langle \tilde{\nabla}_{e_1} ((L_{f(p)})_* \langle \phi, e_1 \cdot \phi \rangle), n_p \rangle_{S^3} + \langle \tilde{\nabla}_{e_2} (L_{f(p)})_* (\langle \phi, e_2 \cdot \phi \rangle), n_p \rangle_{S^3} \right) \quad (2.10)$$

Observe that

$$\tilde{\nabla}_{e_i} ((L_{f(p)})_* \langle \phi, e_i \cdot \phi \rangle) = \tilde{\nabla}_{e_i} ((L_{f(p)})_* \langle \phi, e_i \cdot \phi \rangle_p) + (L_{f(p)})_* e_i (\langle \phi, e_i \cdot \phi \rangle)$$

where in first term of the RHS we fix an element in the Lie algebra, meaning that $(L_{f(p)})_* \langle \phi, e_1 \cdot \phi \rangle_p$ is a left-invariant vector field. We know that, $\tilde{\nabla}_Y Y = 0$ for any left-invariant vector field Y, since $\tilde{\nabla}$ is the Levi–Civita connection of the bi-invariant metric for the Lie group. Therefore, the first term vanishes. It remains to examine the scalar product of the second term and the Gauss map, which turns out to be the same as in (2.6):

$$\langle (L_{f(p)})_* e_i (\langle \phi, e_i \cdot \phi \rangle), (L_{f(p)})_* (\langle \phi, e_1 e_2 \cdot \phi \rangle) \frac{1}{|\phi|^2} \rangle_{S^3}$$

$$= \langle e_i (\langle \phi, e_i \cdot \phi \rangle), (\langle \phi, e_1 e_2 \cdot \phi \rangle) \frac{1}{|\phi|^2} \rangle_{\mathfrak{su}(2)}$$

By some calculations similar to those for (2.6) and (2.7) we can obtain

$$\begin{aligned}
e_1 (\langle \phi, e_1 \cdot \phi \rangle) + e_2 (\langle \phi, e_2 \cdot \phi \rangle) &= -\langle e_1 e_2 \cdot D\phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot D\phi \rangle \\
&= -\langle e_1 e_2 \cdot (\rho - e_1 e_2 \cdot) \phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot (\rho - e_1 e_2 \cdot) \phi \rangle \\
&= -\langle \rho e_1 e_2 \cdot \phi + \phi, \phi \rangle + \langle \phi, \rho e_1 e_2 \cdot \phi + \phi \rangle \\
&= \rho \left(-\langle e_1 e_2 \cdot \phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot \phi \rangle \right) - |\phi|^2 + |\phi|^2 \\
&= \rho \left(-\langle e_1 e_2 \cdot \phi, \phi \rangle + \langle \phi, e_1 e_2 \cdot \phi \rangle \right)
\end{aligned}$$

which has the same result as (2.7), hence the mean curvature has the same formula as well

$$H = \frac{\rho}{|\phi|^2}.$$

$\square$

Theorem 2.6 (Lawson Correspondence). *Any simply connected minimal surface in S^3 gives a constant mean curvature surface in $\mathbb{R}^3$.*

Proof. For any simply connected minimal surface $f : X \to S^3$, we take the metric from the ambient space S^3. The Dirac equation (2.9) gives a spinor field with unit length $|\phi| = 1$ such that

$$D\phi + e_1 e_2 \cdot \phi = 0$$

Let $\varphi = \frac{1+e_1 e_2 \cdot}{2} \phi$. Then

$$
\begin{aligned}
D\varphi &= \frac{1}{2} D\phi - \frac{1}{2} e_1 e_2 \cdot D\phi \\
&= \frac{1}{2} e_1 e_2 \cdot \phi + \frac{1}{2} \phi \\
&= \varphi
\end{aligned}
$$

Therefore, the spinor field φ with the unit length $|\varphi| = 1$ gives a constant mean curvature surface with $H = 1$ in $\mathbb{R}^3$ by corollary 2.2. $\qquad\square$

2.4 Harmonicity of the Gauss Map

Now we are going to prove the fact that, in our spinor formulation, the constant mean curvature surface has the harmonic Gauss map.

Consider the surface $f : X \to \mathbb{R}^3$ with the Gauss map $n = \langle \phi, e_1 e_2 \cdot \phi \rangle$ with the spinor field with unit length. By definition, the Gauss map is harmonic if and only if $\mathrm{Tr} \nabla \, dN = 0$. Write

$$
dn = n_*(e_1)\omega_1 + n_*(e_2)\omega_2 \in \Gamma(T^*X \otimes TS^2).
$$

We need to show that

$$
\nabla_{e_1} n_*(e_1) + \nabla_{e_2} n_*(e_2) - \Gamma_{22}^1 N_*(e_1) - \Gamma_{11}^2 N_*(e_2) = 0 \tag{2.11}
$$

where we used $(\nabla_{e_1} \omega_1)e_1 = 0$, $(\nabla_{e_2} \omega_2)e_2 = 0$, $(\nabla_{e_2} \omega_1)e_2 = -\Gamma_{22}^1$ and $(\nabla_{e_1} \omega_2)e_1 = -\Gamma_{11}^2$. Via the canonical embedding $S^2 \hookrightarrow \mathbb{R}^3$, the tangent bundle of S^2 can be identified with $TS^2 \cong S^2 \times \mathbb{R}^3 \subset S^2 \times \mathbb{H}$ and thus

$$
n_*(e_1) = \langle \nabla_{e_1} \phi, e_1 e_2 \cdot \phi \rangle + \langle \phi, e_1 e_2 \cdot \nabla_{e_1} \phi \rangle
$$

$$
n_*(e_2) = \langle \nabla_{e_2} \phi, e_1 e_2 \cdot \phi \rangle + \langle \phi, e_1 e_2 \cdot \nabla_{e_2} \phi \rangle
$$

since $e_1 e_2$ is always parallel. Therefore the covariant derivative on TS^2 can be obtained by projecting the ordinary derivative in $\mathbb{H}$ onto the tangent plane of S^2:

$$
\nabla_{e_1} n_*(e_1) + \nabla_{e_2} n_*(e_2) = \pi(e_1(n_*(e_1)) + e_2(n_*(e_2)))
$$

where π is the projection onto the plane spanned by $\{\langle \phi, e_1 \cdot \phi \rangle, \langle \phi, e_2 \cdot \phi \rangle\}$. It follows that

$$
\begin{aligned}
&\pi(e_1(n_*(e_1)) + e_2(n_*(e_2))) \\
={}&\pi(\langle \nabla_{e_1} \nabla_{e_1} \phi, e_1 e_2 \cdot \phi \rangle + \langle \nabla_{e_1} \phi, e_1 e_2 \cdot \nabla_{e_1} \phi \rangle \\
&+ \langle \nabla_{e_1} \phi, e_1 e_2 \cdot \nabla_{e_1} \phi \rangle + \langle \phi, e_1 e_2 \cdot \nabla_{e_1} \nabla_{e_1} \phi \rangle \\
&+ \langle \nabla_{e_2} \nabla_{e_2} \phi, e_1 e_2 \cdot \phi \rangle + \langle \nabla_{e_2} \phi, e_1 e_2 \cdot \nabla_{e_2} \phi \rangle \\
&+ \langle \nabla_{e_2} \phi, e_1 e_2 \cdot \nabla_{e_2} \phi \rangle + \langle \phi, e_1 e_2 \cdot \nabla_{e_2} \nabla_{e_2} \phi \rangle)
\end{aligned}
$$

The terms with the form $\langle \nabla_{e_i} \phi, e_1 e_2 \cdot \nabla_{e_i} \phi \rangle$ all vanish under the projection. We define the connection Laplacian Δ by (see (6.8) or [LM90, Chap. 2.8])

$$\Delta := -\left(\nabla_{e_1} \nabla_{e_1} + \nabla_{e_2} \nabla_{e_2} - \nabla_{\nabla_{e_1} e_1} - \nabla_{\nabla_{e_2} e_2} \right)$$

By the Lichnerowicz–Weitzenböck formula we know that

$$D^2 = \Delta + \frac{1}{2} K$$

yields

$$\pi(e_1(n_*(e_1)) + e_2(n_*(e_2)))$$
$$= \pi(\langle (-D^2 + \frac{1}{2}K)\phi, e_1 e_2 \cdot \phi \rangle + \langle \nabla_X \phi, e_1 e_2 \cdot \phi \rangle$$
$$+ \langle \phi, e_1 e_2 \cdot (-D^2 + \frac{1}{2}K)\phi \rangle + \langle \phi, e_1 e_2 \cdot \nabla_X \phi \rangle)$$
$$= \pi(X \langle \phi, e_1 e_2 \cdot \phi \rangle) = X \langle \phi, e_1 e_2 \cdot \phi \rangle = X(n)$$

Then (2.11) becomes

$$X(n) - \Gamma_{22}^1 n_*(e_1) - \Gamma_{11}^2 n_*(e_2) = (\nabla_{e_1} e_1 + \nabla_{e_2} e_2)n - \Gamma_{22}^1 n_*(e_1) - \Gamma_{11}^2 n_*(e_2)$$
$$= \Gamma_{11}^2 e_2(n) + \Gamma_{22}^1 e_1(n) - \Gamma_{22}^1 n_*(e_1) - \Gamma_{11}^2 n_*(e_2)$$
$$= 0$$

2.5 The Extrinsic and Intrinsic Dirac Operators

In this section we will describe the exact connection between the extrinsic and intrinsic Dirac operators (for a more detailed treatment of spin structures and Dirac operators, see [LM90]). Again we start with the smooth setup:

Given a smooth immersion $f : X \to \mathbb{R}^3 \subset \mathbb{H}$ of a surface and a smooth quaternion-valued function $\phi : X \to \mathbb{H}$, a smooth scale rotation of every tangent plane can be constructed by ([KPP98; KNPP02], see Section 3.1 for more details)

$$\widetilde{(\mathrm{d}f)} = \overline{\phi} \cdot \mathrm{d}f \cdot \phi \tag{2.12}$$

If there exists a further smooth surface $\tilde{f}$ such that $\mathrm{d}(\tilde{f}) = \widetilde{(\mathrm{d}f)}$, then it follows that

$$0 = \mathrm{d}\,\mathrm{d}(\tilde{f}) = \mathrm{d}\widetilde{(\mathrm{d}f)} = \mathrm{d}(\overline{\phi} \cdot f \cdot \phi)$$

which gives:

$$\mathrm{D}_f(\phi) = \rho\phi \tag{2.13}$$

where $\rho : X \to \mathbb{R}$ is a real scalar function and $\mathrm{D}_f = -\frac{\mathrm{d}f \wedge \mathrm{d}}{|\mathrm{d}f|^2}$ is called the Dirac operator with respect to the immersion f. Since D_f depends on the immersion f (and in order to

distinguish it from the intrinsic Dirac operator of Atiyah), we call it the extrinsic Dirac operator in the following context.

Suppose X is an oriented surface and $f : X \to \mathbb{R}^3$ is an immersion. Let $Cl_3 \to \mathbb{R}^3$ be the trivial Clifford bundle over $\mathbb{R}^3$ and let $\mathcal{S} \to \mathbb{R}^3$ be the corresponding trivial spinor bundle. Both these bundles can be pulled back to X through the map f: $Cl_3|_X = f^*(Cl_3)$ and $\mathcal{S}|_X = f^*(\mathcal{S})$. Furthermore, since there is a natural identification $Cl_2 \hookrightarrow Cl_3^{even}$ by $v \mapsto n \cdot v$ where n is the normal to X in $\mathbb{R}^3$, we can define the Clifford representation

$$\rho : Cl_2 \to \text{End}(\mathcal{S})$$
$$v \mapsto \rho_3(n \cdot v)$$

where ρ_3 is the Clifford representation of Cl_3.

Note that $\mathcal{S}|_X$ can be identified with the spinor bundle $\mathcal{S}_X$ of X. Suppose $\phi \in \Gamma(\mathcal{S}_X)$ the Dirac operator is

$$D : \Gamma(\mathcal{S}_X) \to \Gamma(\mathcal{S}_X)$$
$$\phi \mapsto \rho(e_1) \cdot \nabla_{e_1}\phi + \rho(e_2) \cdot \nabla_{e_2}\phi$$

where $\{e_1, e_2\}$ is an oriented orthonormal frame of X and ∇ is the Levi–Civita connection of X.

Theorem 2.7. *Let $f : X \hookrightarrow \mathbb{R}^3$ be an isometric immersion of a surface. Let $c \in \Gamma(\mathcal{S})$ be the global parallel section of the spinor bundle and $c|_X = f^*(c) \in \Gamma(S_X)$ be the pull-back of c by f. Note that c induces an isomorphism:*

$$\mathfrak{c} : \Gamma(S_X) \to \Omega(X, \mathbb{H})$$
$$(c, \phi) \mapsto \phi$$

where $\Omega(X, \mathbb{H})$ is the set of quaternion-valued functions. Then we have

$$\mathfrak{c} \circ (D - H) \circ \mathfrak{c}^{-1} = D_f. \tag{2.14}$$

Proof. Note that the covariant derivative of the ambient space and its hypersurface differ by a second fundamental form (see [HMZ01])

$$\begin{aligned}
\nabla_X Y &= \tilde{\nabla}_X Y - \langle \tilde{\nabla}_X Y, n \rangle n \\
&= \tilde{\nabla}_X Y + \langle Y, \tilde{\nabla}_X n \rangle n \\
&= \tilde{\nabla}_X Y - \text{II}(X, Y) n
\end{aligned}$$

and the corresponding spinor connection satisfies

$$\nabla_X \phi = \tilde{\nabla}_X \phi - \frac{1}{2}\text{II}(e_1, X)e_1 \cdot n \cdot \phi - \frac{1}{2}\text{II}(e_2, X)e_2 \cdot n \cdot \phi \,.$$

This yields

$$
\begin{aligned}
D\phi &= \rho(e_1) \cdot \nabla_{e_1}\phi + \rho(e_2) \cdot \nabla_{e_2} \\
&= \rho_3(n) \cdot \rho_3(e_1) \cdot \left(\tilde{\nabla}_{e_1}\phi - \frac{1}{2}\mathrm{II}(e_1,e_1)\rho_3(e_1) \cdot \rho_3(n) \cdot \phi \right. \\
&\quad \left. - \frac{1}{2}\mathrm{II}(e_1,e_2)\rho_3(e_2) \cdot \rho_3(n) \cdot \phi \right) + \rho_3(n) \cdot \rho_3(e_2) \cdot \left(\tilde{\nabla}_{e_2}\phi \right. \\
&\quad \left. - \frac{1}{2}\mathrm{II}(e_2,e_1)\rho_3(e_1) \cdot \rho_3(n) \cdot \phi - \frac{1}{2}\mathrm{II}(e_2,e_2)\rho_3(e_2) \cdot \rho_3(n) \cdot \phi \right) \\
&= \rho_3(N) \cdot \rho_3(e_1) \cdot \tilde{\nabla}_{e_1}\phi + \rho_3(n) \cdot \rho_3(e_2) \cdot \tilde{\nabla}_{e_2}\phi + H\phi
\end{aligned}
\tag{2.15}
$$

where $\tilde{\nabla}$ is the Levi–Civita connection of $\mathbb{R}^3$.

Now let us take the global parallel frame c with the following identifications

$$
e_1 \mapsto \mathrm{d}f(e_1), \quad e_2 \mapsto \mathrm{d}f(e_2), \quad n \mapsto N
$$

where $\mathrm{d}f(e_1)$, $\mathrm{d}f(e_2)$, and N are imaginary quaternions. Since c is parallel, the covariant derivative reduces to the partial derivative ∂. Hence (2.15) becomes:

$$
\begin{aligned}
\mathfrak{c} \circ (D - H) \circ \mathfrak{c}^{-1} &= N \cdot \mathrm{d}f(e_1) \cdot \partial_{e_1} + N \cdot \mathrm{d}f(e_2) \cdot \partial_{e_2} \\
&= \mathrm{d}f(e_2)\partial_{e_1} - \mathrm{d}f(e_1)\partial_{e_2}.
\end{aligned}
\tag{2.16}
$$

On the other hand we have (see [Chu16] for more details)

$$
\begin{aligned}
D_f &= -\frac{\mathrm{d}f \wedge \mathrm{d}}{|\mathrm{d}f|^2} \\
&= -\frac{(\mathrm{d}f(e_1)e_1^* + \mathrm{d}f(e_2)e_2^*) \wedge (e_1^*\partial_{e_1} + e_2^*\partial_{e_2})}{|\mathrm{d}f|^2} \\
&= -\frac{(\mathrm{d}f(e_1)\partial_{e_2} - \mathrm{d}f(e_2)\partial_{e_1})\, e_1^* \wedge e_2^*}{|\mathrm{d}f|^2} \\
&= -\mathrm{d}f(e_1)\partial_{e_2} + \mathrm{d}f(e_2)\partial_{e_1}.
\end{aligned}
\tag{2.17}
$$

Comparing (2.16) with (2.17) we finally find

$$
\mathfrak{c} \circ (D - H) \circ \mathfrak{c}^{-1} = D_f
$$

$\square$

3 Discrete Theory

Sec. 3.1 to Sec. 3.4 are taken from a preprint submitted by the author [HY18].

3.1 Quaterinionic interpretation of 3D rotations

We start by gathering some basic notions about quaternions and how they encode rotations in $\mathbb{R}^3$. Let $\mathbb{H}$ denote the algebra of quaternions: the four dimensional real vector space $\mathbb{H} = \mathrm{span}\{1, \mathbb{i}, \mathbb{j}, \mathbb{k}\}$ together with the product relations $\mathbb{i}^2 = \mathbb{j}^2 = \mathbb{k}^2 = -1, \mathbb{i}\mathbb{j} = \mathbb{k}, \mathbb{j}\mathbb{k} = \mathbb{i}$, and $\mathbb{k}\mathbb{i} = \mathbb{j}$. Then $\mathrm{Im}(\mathbb{H}) := \mathrm{span}\{\mathbb{i}, \mathbb{j}, \mathbb{k}\}$ is a three dimensional subspace canonically isomorphic to $\mathbb{R}^3$ via

$$(x, y, z) \mapsto x\mathbb{i} + y\mathbb{j} + z\mathbb{k}.$$

Given a vector $w \in \mathbb{R}^3$ the rotation of w around a non-vanishing vector $u \in \mathbb{R}^3$ can be described in the following way: First let the vectors w and u be embedded in the imaginary quaternions in the above way. Then the rotation can be computed by:

$$\mathrm{R}_u^\theta(w) = q^{-1} \cdot w \cdot q$$

where R_u^θ denotes the rotation of w around u through the angle θ and

$$q = |q|\left(\cos \frac{\theta}{2} - \sin \frac{\theta}{2} \frac{u}{|u|}\right)$$

Note that the angle θ is measured by the counterclockwise angle as one sees in the opposite direction of u.

Lemma 3.1. *Let w_1, w_2 and u be non-vanishing vectors in $\mathrm{Im}(\mathbb{H})$ such that $|w_1| = |w_2|$ and let $\theta \in (-\pi, \pi)$ denote the bending angle between two the planes $P_1 = \mathrm{span}\{w_1, u\}$ and $P_2 = \mathrm{span}\{w_2, u\}$.*

1. If $w_1 - w_2 \perp u$, then there is an uniquely defined unit quaternion q such that

$$\mathrm{Im}(q) = \begin{cases} \frac{u}{|u|}|\mathrm{Im}(q)| & \theta \neq 0 \\ 0 & \theta = 0 \end{cases} \tag{3.1}$$

and

$$q^{-1} \cdot w_1 \cdot q = w_2 \tag{3.2}$$

---

2. *If $w_1 + w_2 \perp u$, then there is an unique real number H such that*

$$(H + u)^{-1} \cdot w_1 \cdot (H + u) = -w_2$$

and we have

$$H = |u| \tan \frac{\theta}{2}.$$

Proof. Let w_1, w_2, u and θ be as above.

1. Since $w_1 - w_2 \perp u$, w_2 can be obtained by rotating w_1 around u by the angle θ. There are two quaternions $q = \pm(\cos \frac{\theta}{2} - \sin \frac{\theta}{2} \frac{u}{|u|})$ satisfying eq. (3.2), but only one of them

$$q = \begin{cases} -\cos \frac{\theta}{2} + \sin \frac{\theta}{2} \frac{u}{|u|} & \sin \frac{\theta}{2} \geq 0 \\ \cos \frac{\theta}{2} - \sin \frac{\theta}{2} \frac{u}{|u|} & \sin \frac{\theta}{2} < 0 \end{cases}$$

satisfies eq. (3.1).

2. Since $w_1 + w_2 \perp u$, $-w_2$ can be obtained by rotating w_1 around u by the angle $\theta + \pi$.

$$\begin{aligned} H + u &= |u| \tan \frac{\theta}{2} + u \\ &= |u| \left(\tan \frac{\theta}{2} + \frac{u}{|u|} \right) \\ &= \frac{|u|}{\cos \frac{\theta}{2}} \left(\sin \frac{\theta}{2} + \cos \frac{\theta}{2} \frac{u}{|u|} \right) \\ &= \frac{|u|}{\cos \frac{\theta}{2}} \left(-\cos(\frac{\pi}{2} + \frac{\theta}{2}) + \sin(\frac{\pi}{2} + \frac{\theta}{2}) \frac{u}{|u|} \right) \\ &= -\frac{|u|}{\cos \frac{\theta}{2}} \left(\cos(\frac{\pi + \theta}{2}) - \sin(\frac{\pi + \theta}{2}) \frac{u}{|u|} \right) \end{aligned}$$

It follows that $(H + u)^{-1} \cdot w_1 \cdot (H + u) = -w_2$ and it is also the unique quaternion with the imaginary part being exactly u.

$\square$

3.2 The Extrinsic Dirac Operator

We are now interested in a discretization of D_f. Note that, the inner product $\langle \cdot, \cdot \rangle$ on the differential forms induced by the metric can be defined by

$$\omega \wedge *\eta = \langle \omega, \eta \rangle \, dvol \, .$$

Then D_f can be formally reformulated as

$$D_f(\phi) = -\frac{df \wedge d\phi}{|df|^2} = \langle df, * d\phi \rangle \, . \tag{3.3}$$

Hence, in the discrete setting it is more natural to think of ϕ as the function of the dual vertices.

A net is a cell complex $X = (V, E, F)$ such that

1. The faces are all polygons, but not necessary planar.

2. The intersection of two adjacent faces contains always only one edge.

By oriented nets we mean in every face we choose a preferred direction for every edge such that the common edge in two adjacent faces has the reversed direction. An immersed net is a net with each vertex assigned with a position in $\mathbb{R}^3$. The notation e_{ij} indicates the immersed edge incident to the faces Δ_i and Δ_j and with the orientation in face Δ_i. It is clear that

$$e_{ij} = -e_{ji}$$

Our basic object is the face edge-constraint net, which is similar as the one in [HSFW16].
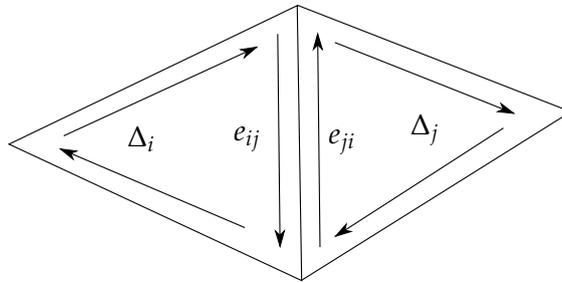
Figure 3.1: Orientation

Instead of considering normals at the vertices, in this setting the normals are defined on the faces.

Definition 3.2. A face edge-constraint net $\mathfrak{X} = (X, f, n)$ is an oriented net $X = (V, E, F)$ with an immersion $f : V \to \mathbb{R}^3$ and unit normals $n : F \to \mathbb{S}^2$ assigned to each face, such that

$$n_i + n_j \perp e_{ij} \tag{3.4}$$

holds for every pair of adjacent faces Δ_i and Δ_j.

Remark 3.3. An immersed oriented net with all faces being planar and n_i being the normal of the face Δ_i is always a face edge-constraint net. We call such nets classical nets.

An advantage of the face edge-constraint nets is that they come with a natural notion of mean curvature that arises from a face offset Steiner's formula, as we will see below. We are then able to introduce a discrete spin transformation and Dirac operator such that the Dirac equation guarantees the closing condition of the spin transformation. Moreover, one can control the mean curvature with the Dirac equation exactly as in the smooth case.

Definition 3.4. Given a face edge-constraint net the bending angle θ_{ij} from the face Δ_i to Δ_j is defined to be the angle from the plane P_i to P_j, where $P_i = \text{span}\{n_i, e_{ij}\}$ and $P_j = \text{span}\{n_j, e_{ij}\}$.
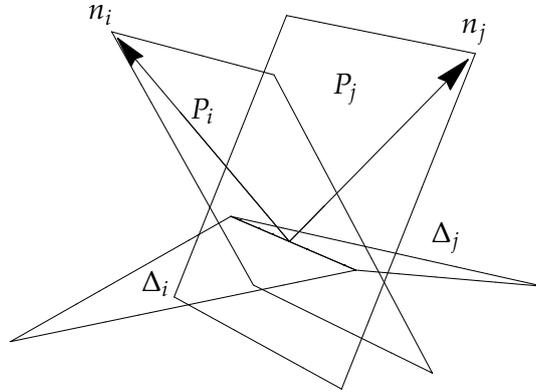
Figure 3.2: Bending angle

Definition 3.5. For a face edge-constraint net the integrated mean curvature of the edge e_{ij} is defined by

$$\mathbf{H}_{ij} = \frac{1}{2}|e_{ij}| \tan \frac{\theta_{ij}}{2} .$$

The mean curvature of a face is defined to be the sum of the mean curvatures of all the edges around the face:

$$\mathbf{H}_i = \sum_j \mathbf{H}_{ij}$$

where j runs through all the adjacent faces of Δ_i.

Remark 3.6. Suppose X is a smooth immersed surface and X_t is the surface offset obtained by shifting every point of X along the normals with distance t (fig. 3.3a). Then, Steiner's formula for the infinitesimal area dA of X_t gives

$$dA(X_t) = (1 + 2Ht + Kt^2) \, dA(X) \tag{3.5}$$

where H and K stand for the mean curvature and Gauss curvature of X respectively. In order to be consistent with the terminology in [CPS11; KPP98], we choose the sign of H which is different from the one in [KW14a].

Now let us consider a classical face edge-constraint net $\mathfrak{X}$. If we move the plane of the face Δ_i along n_i, as well as all the faces Δ_j adjacent to Δ_i along n_j, with the distance t, then we obtain the face offset Δ_i^t (fig. 3.3b). The area of Δ_i^t is

$$\text{Area}(\Delta_i^t) = \left(1 + \frac{\mathbf{H}_i}{\text{Area}(\Delta_i)} t + o(t^2)\right) \text{Area}(\Delta_i) \tag{3.6}$$

hence our mean curvature can be thought of as the the mean curvature integrated over the face Δ_i.

(a) Smooth

(b) Face offset

Figure 3.3: The Steiner formula

Proof. See [KW14a, Thm 2.4]. □

The next definition ties together all the edge-located information in one quaternionic object:

Definition 3.7. The hyperedge $E_{ij} \in \mathbb{H}$ is a quaternion whose real part is the mean curvature of the edge e_{ij} and whose imaginary part is the natural embedding of the edge into $\mathbb{H}$, i.e.,

$$E_{ij} := 2\mathbf{H}_{ij} + e_{ij}$$

It is easy to see the following two properties of hyperedges:

Proposition 3.8. *For any hyperedge one finds:*

1. $E_{ij} = \overline{E_{ji}}$

2. *If the bending angle $\theta_{ij} = 0$, then $E_{ij} = e_{ij}$ is purely imaginary.*

One can read hyperedges as rotation quaternions. This way we obtain

Proposition 3.9.
$$E_{ij}^{-1} \cdot n_i \cdot E_{ij} = -n_j$$

Proof. Direct computation yields

$$
\begin{aligned}
E_{ij} &= \tan \frac{\theta_{ij}}{2} |e_{ij}| + e_{ij} \\
&= |e_{ij}| \cos \frac{\theta_{ij}}{2} \left(\sin \frac{\theta_{ij}}{2} + \cos \frac{\theta_{ij}}{2} \frac{e_{ij}}{|e_{ij}|} \right) \\
&= |e_{ij}| \cos \frac{\theta_{ij}}{2} \left(-\cos \frac{\theta_{ij}+\pi}{2} + \sin \frac{\theta_{ij}+\pi}{2} \frac{e_{ij}}{|e_{ij}|} \right) .
\end{aligned}
$$

Apparently n_i gets mapped to $-n_j$ by the rotation around the axis $\frac{e_{ij}}{|e_{ij}|}$ with the angle $\theta_{ij} + \pi$. □

Definition 3.10. Let $\mathcal{H}$ be the space of functions from the set of faces F to $\mathbb{H}$. We also refer to the elements in $\mathcal{H}$ as the spinors. The discrete extrinsic Dirac operator, also denoted by D_f, is defined as follows:

$$D_f : \mathcal{H} \to \mathcal{H}$$

$$D_f(\phi)|_i = \frac{1}{2} \sum_j E_{ij} \cdot (\phi_j - \phi_i) .$$

D_f has a similar form as its smooth counterpart eq. (3.3). Since the sum of the imaginary parts of hyperedges around a face vanishes, the Dirac operator can be rewritten as

$$
\begin{aligned}
D_f(\phi)|_i &= \frac{1}{2} \sum_j E_{ij} \cdot \phi_j - \frac{1}{2} \Big(\sum_j E_{ij} \Big) \cdot \phi_i \\
&= \frac{1}{2} \sum_j E_{ij} \cdot \phi_j - \Big(\sum_j \mathbf{H}_{ij} \Big) \phi_i \\
&= \frac{1}{2} \sum_j E_{ij} \cdot \phi_j - \mathbf{H}_i \phi_i
\end{aligned}
$$

Proposition 3.11. *Let $\langle \cdot, \cdot \rangle$ be the scalar product defined on $\mathcal{H}$*

$$\langle \phi, \varphi \rangle = \sum_i \overline{\phi_i} \varphi_i$$

where i runs through all the faces of X and suppose X is a closed net. Then, the discrete extrinsic Dirac operator D_f is self-adjoint.

Proof. Let j_i be the indices of the faces neighbouring to i.

$$
\begin{aligned}
\langle D_f \phi, \varphi \rangle &= \sum_i \overline{D_f \phi_i} \varphi_i \\
&= \sum_i \sum_{j_i} \overline{\frac{1}{2} E_{ij} \cdot \phi_{j_i} - \mathbf{H}_i \phi_i} \varphi_i \\
&= \sum_i \sum_{j_i} \Big(\frac{1}{2} \overline{\phi_{j_i}} \overline{E_{ij_i}} \varphi_i - \mathbf{H}_i \overline{\phi_i} \varphi_i \Big)
\end{aligned}
$$

If X is closed then we can switch the indices in the first term and it yields

$$
\begin{aligned}
\langle D_f \phi, \varphi \rangle &= \sum_i \sum_{j_i} \left(\frac{1}{2} \overline{\phi_i \widehat{E_{j_i i}}} \varphi_{j_i} - \mathbf{H}_i \overline{\phi_i} \varphi_i \right) \\
&= \sum_i \sum_{j_i} \left(\frac{1}{2} \overline{\phi_i} E_{ij_i} \varphi_{j_i} - \mathbf{H}_i \overline{\phi_i} \varphi_i \right) \\
&= \sum_i \overline{\phi_i} \sum_{j_i} \left(\frac{1}{2} E_{ij_i} \varphi_{j_i} - \mathbf{H}_i \varphi_i \right) \\
&= \sum_i \overline{\phi_i} D_f \varphi_i \\
&= \langle \phi, D_f \varphi \rangle
\end{aligned}
$$

$\square$

We will now define a scale-rotation type of transformation for face edge-constraint nets in the spirit of equation (2.12) together with a condition for the result to be integrable into a new face edge-constraint net:

Definition 3.12. Let $\mathfrak{X}$ be a face edge-constraint net. The discrete spin transformation s_ϕ with respect to ϕ is given by (fig. 3.4):

$$
\begin{aligned}
s_\phi(E_{ij}) &= \overline{\phi_i} \cdot E_{ij} \cdot \phi_j \\
s_\phi(n_i) &= \phi_i^{-1} \cdot n_i \cdot \phi_i
\end{aligned}
$$
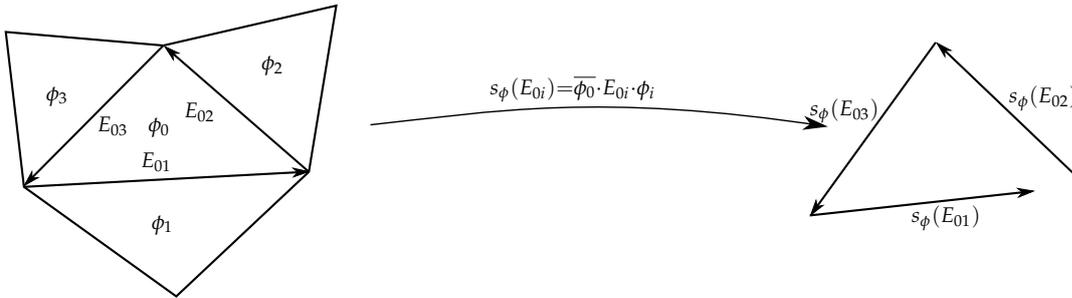
Figure 3.4: Discrete spin transformation

Theorem 3.13. *For a simply-connected net $\mathfrak{X}$, if*

$$
D_f \phi = \rho \phi \tag{3.7}
$$

where $\rho : F \to \mathbb{R}$ is a real function, then the imaginary parts of the hyperedges obtained by spin transformation are closed around every face.

Proof. The spin transformation of the face Δ_i is

$$\sum_j s_\phi(E_{ij}) = \sum_j \overline{\phi_i} \cdot E_{ij} \cdot \phi_j$$
$$= \overline{\phi_i} \cdot \left(\sum_j E_{ij} \cdot \phi_j \right)$$
$$= 2\overline{\phi_j}(\boldsymbol{\rho}_i + \mathbf{H}_i)\phi_i$$
$$= 2(\boldsymbol{\rho}_i + \mathbf{H}_i)|\phi_i|^2$$

which is a real number. Hence the imaginary parts of the transformed hyperedges add to zero. $\qquad\square$

The following proposition shows that the spin transformation maps a face edge-constraint net again to a face edge-constraint net.

Proposition 3.14. *Let s be a spin-transformation as above. Then*

$$s(E_{ij})^{-1} \cdot s(n_i) \cdot s(E_{ij}) = -s(n_j).$$

Proof. A direct calculation yields:

$$s(E_{ij})^{-1} \cdot s(n_i) \cdot s(E_{ij}) = (\overline{\phi_i} \cdot E_{ij} \cdot \phi_j)^{-1} \cdot \phi_i^{-1} \cdot n_i \cdot \phi_i \cdot (\overline{\phi_i} \cdot E_{ij} \cdot \phi_j)$$
$$= \phi_j^{-1} \cdot E_{ij}^{-1} \cdot n_j \cdot E_{ij}^{-1} \cdot \phi_j$$
$$= -\phi_j^{-1} \cdot n_j \cdot \phi_j$$
$$= -s(n_j).$$

$\qquad\square$

Let $\mathcal{X}$ be the space of all face edge-constraint nets. For every $f \in \mathcal{X}$, every solution ϕ to equation (3.7) gives rise to a new transformed face edge-constraint net $\tilde{f}$. Its mean curvature $\widetilde{\mathbf{H}}$ changes from the original one $\mathbf{H}$ in the following way:

$$\widetilde{\mathbf{H}} = (\boldsymbol{\rho} + \mathbf{H})|\phi|^2 \tag{3.8}$$

Remark 3.15. In smooth case we have the formula (see [KPP98])

$$\tilde{H}|\mathrm{d}\tilde{f}| = H|\mathrm{d}f| + \rho|\mathrm{d}f| \tag{3.9}$$

Let $h = H|\mathrm{d}f|$ be the mean curvature half-density, then (3.9) turns to

$$\tilde{h} = h + \rho|\mathrm{d}f| \tag{3.10}$$

Since the integrated mean curvature $\mathbf{H}$ is approximately $H|\mathrm{d}f|^2$, we define the discrete mean curvature half-density by

$$\mathbf{h}_i := \frac{\mathbf{H}_i}{|\mathrm{d}f|} = \frac{\mathbf{H}_i}{\sqrt{\mathrm{Area}_i}} \tag{3.11}$$

W=12.59≈4π   W=12.62   W=12.60   W=19.78≈2π²   W=293.53   W=293.54
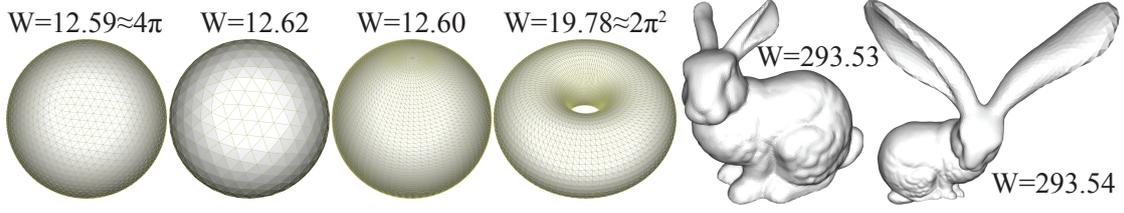
Figure 3.5: On the left, comparing our discrete Willmore energy to its known values for the sphere and torus (left). On the right, comparing the energy values for two surfaces, one of which is a Möbius transformation of the other.

then by $\widetilde{\text{Area}}_i \approx |\phi|^4 \text{Area}_i$ we have

$$\tilde{\mathbf{h}} \approx \mathbf{h} + \frac{\rho}{\sqrt{\text{Area}}}$$

therefore if we think of ρ as the integrated curvature potential, i.e., $\rho \approx \rho|\mathrm{d}f|^2$, it yields

$$\tilde{\mathbf{h}}_i \approx \mathbf{h} + \rho|\mathrm{d}f| \tag{3.12}$$

which concides with the equation in smooth case (3.10).

Based on the mean curvature half-density, the *Willmore energy W* can be defined. In the continuous case, the energy is defined as $W = \int h^2$ [CPS11]. Based on this formula, we define the discrete Willmore energy using our discretized curvatures by

$$W := \sum_i \frac{\mathbf{H}_i^2}{\text{Area}_i} \tag{3.13}$$

In Fig. 3.5 we show Willmore energy calculations for discrete versions of surfaces for which the value of the energy is known in the continuous case, and find that the discrete energy values are close to the predicted ones. The figure also indicates that our discrete energy is numerically invariant to Möbius transformations, as expected.

3.2.1 Minimal Surfaces and their Associated Family

We call a face edge-constraint net a minimal surface, if $\mathbf{H}_i = 0$ for all i. We know that if ϕ is a solution to the Dirac equation

$$D_f \phi = -\mathbf{H}\phi \tag{3.14}$$

then the spin transformation gives a minimal surface by eq. (3.8). Recall that in smooth case a minimal surface doesn't come alone but always with an associated family [Bob94]. In complete analogy, we will now see that there is a corresponding construction for face edge-constraint minimal surfaces. Suppose ϕ_i is a solution to (3.14), then it is easy to verify that the following quaternionic functions parametrized by λ all satisfy (3.14) as well

$$\phi(\lambda)|_i = (\cos \lambda + \sin \lambda n_i) \cdot \phi_i$$
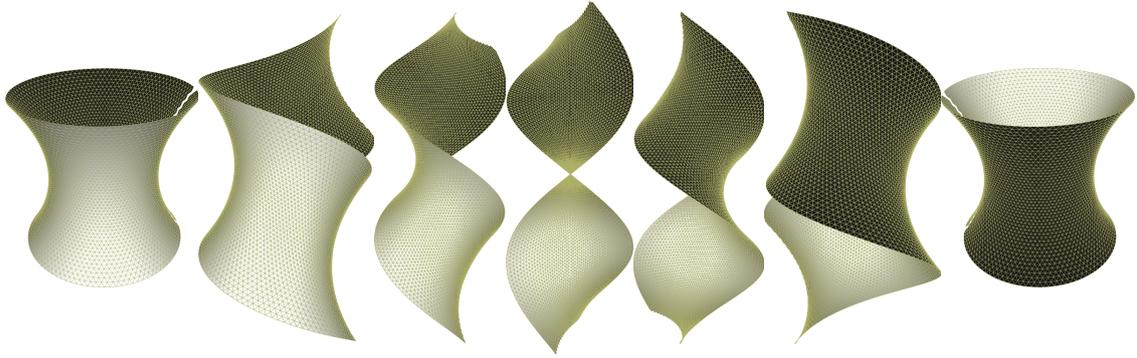
Figure 3.6: A family of discrete surfaces.

The explicit formula tor the associated family then is given by

$$
\begin{aligned}
s(\lambda)(E_{ij}) &= \overline{\phi(\lambda)_i} E_{ij} \phi(\lambda)_j \\
&= \overline{(\cos \lambda + \sin \lambda n_i) \phi_i} E_{ij} \cdot (\cos \lambda + \sin \lambda n_j) \phi_j \\
&= \overline{\phi_i} (\cos \lambda - \sin \lambda n_i) E_{ij} (\cos \lambda + \sin \lambda n_j) \phi_j \\
&= \overline{\phi_i} (\cos \lambda E_{ij} - \sin \lambda n_i E_{ij}) (\cos \lambda + \sin \lambda n_j) \phi_j \\
&= \overline{\phi_i} (\cos^2 \lambda E_{ij} + \cos \lambda \sin \lambda E_{ij} n_j \\
&\quad - \sin \lambda \cos \lambda n_i E_{ij} - \sin^2 \lambda n_i E_{ij} n_j) \phi_j \\
&= \overline{\phi_i} (\cos 2\lambda E_{ij} - \sin 2\lambda n_i E_{ij}) \phi_j
\end{aligned}
$$

Remark 3.16. In [Lam16] Lam shows that there exists an associated family which contains two types of well-known minimal surfaces, *A*-minimal surfaces coming from the discrete integrable system and *C*-minimal surfaces coming from an area variational approach. It's easy to see that these two types of minimal surfaces are also minimal surfaces in our framework. In fact a face edge-constraint minimal surface is *A*-minimal surface if and only if $\mathbf{H}_{ij} = 0$ for every edge *ij*. And the *C*-minimal surfaces are just the minimal classical nets.

3.2.2 A Weierstrass Representation

Recall that in [LP16] Lam and Pinkall define a discrete holomorphic quadratic differential $q : E \to \mathrm{Im}(\mathbb{C})$ on a planar triangulated mesh $z : V \to \mathbb{C}$, $X = (V, E, F)$ in the complex plane by

$$
\sum_j q_{ij} = 0
$$

$$
\sum_j q_{ij} / \, \mathrm{d}z(e_{ij}) = 0
$$

and they show that this gives rise to the two types minimal surfaces mentioned in remark 3.16 by means of a discrete analogue of the Weierstrass representation. By

remark 3.16 we know that our face edge-constraint minimal surfaces can be considered as a generalization of these two types minimal surfaces and indeed we can generalize the discrete holomorphic quadratic differential, removing the restriction on q of being purely imaginary, and obtain the following generalized discrete holomorphic quadratic differential:

Definition 3.17. Given a planar net on the complex plane $z : V \to \mathbb{C}$. A holomorphic quadratic differential is a funtion $q : E \to \mathbb{C}$ such that

$$\sum_j q_{ij} = 0$$

$$\sum_j q_{ij} / \, \mathrm{d}z(e_{ij}) = 0$$

Now, we are going to show that this holomorphic quadratic differential always gives a family of minimal surfaces in a similar manner to [LP16].

Theorem 3.18. *Let* $z : V \to \mathbb{C}$ *be a realization of a simply connected triangular mesh and* $q : E \to \mathbb{C}$ *a holomorphic quadratic differential. Then there exists a minimal face edge-constraint net* $\mathfrak{X}_q$:

$$E_{ij} = \mathrm{Re}\left(q_{ij} + \frac{q_{ij}}{i(z_j - z_i)} \left((1 - z_i z_j)\mathbb{i} + i(1 + z_i z_j)\mathbb{j} + (z_i + z_j)\mathbb{k} \right) \right)$$

$$n = \frac{1}{|z|^2 + 1} \begin{pmatrix} 2\,\mathrm{Re}\,z \\ 2\,\mathrm{Im}\,z \\ |z|^2 - 1 \end{pmatrix}$$

where Re *means taking the real part of each component of the quaternion.*

Proof. To see that the imaginary parts of the hyperedges are closed around each face, we refer to the proof of Theorem 6.3 in [LP16]. By direct computation we have

$$E_{ij}^{-1} \cdot n_i \cdot E_{ij} = -n_j$$

indicating that $\mathfrak{X}_q$ is indeed a face edge-constraint net. Note, that the integrated mean curvature for an edge is $H_{ij} = \mathrm{Re}(q_{ij})$, hence

$$H_i = \sum_j \mathrm{Re}(q_{ij}) = 0$$

at any face Δ_i by assumption, showing that $\mathfrak{X}_q$ is minimal. $\qquad \square$

Remark 3.19. We can construct the associated family of a minimal surface by rotating q_{ij} with a constant unit complex number, $q_{ij} \to e^{\lambda i} q_{ij}$, which is basically equivalent to what we have done in the last section.

3.2.3 A Spin Multi-Ratio

In this section we shall investigate an invariant of the spin transformation. It turns out that this invariant – we will call it the spin multi-ratio – actually fully characterizes face edge-constraint nets up to spin equivalence.

Definition 3.20. A path in a net X is a sequence of faces

$$\gamma = (\gamma(1), \gamma(2), \dots, \gamma(n))$$

where $\gamma(i)$ and $\gamma(i+1)$ are neighbouring faces or $\overrightarrow{\gamma(i)\gamma(i+1)} \in E^*$. The length of the path is defined by the number of dual edges in the path, i.e.,

$$|\gamma = (\gamma(1), \gamma(2), \cdots, \gamma(n))| = n - 1$$

Given a face edge-constraint net $\mathfrak{X} = (X, f, n)$ the spin multi-ratio $\mathrm{cr}_{\mathfrak{X}}$ is a map from the set of all the paths to the quaternions

$$\mathrm{cr}(\gamma) = \begin{cases} \overline{E_{\gamma(1),\gamma(2)}}^{-1} \cdot E_{\gamma(2),\gamma(3)} \cdot \dots \cdot E_{\gamma(n-1),\gamma(n)} & |\gamma| \text{ is even} \\ E_{\gamma(1),\gamma(2)}^{-1} \cdot E_{\gamma(2),\gamma(3)} \cdot \dots \cdot \overline{E_{\gamma(n-1),\gamma(n)}}^{-1} & |\gamma| \text{ is odd} \end{cases}$$
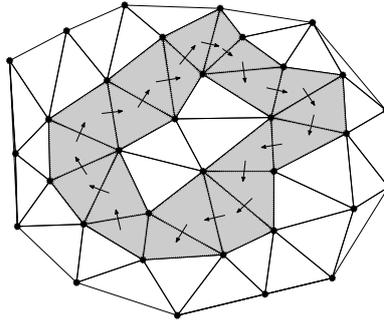
Figure 3.7: A loop

Definition 3.21. A loop at Δ_i is a path starting and ending both at the same face Δ_i. Let's define an equivalence relation on the sets of all loops at *i* by:

$$(\cdots, i, j, i, \cdots) \sim (\cdots, i, \cdots)$$

Then the set of all the loops at Δ_i modulo the equivalence relation is endowed with a group structure by:

$$\gamma_1 \cdot \gamma_2 = (\gamma_1(1), \gamma_1(2), \cdots, \gamma_1(n), \gamma_2(1), \gamma_2(2), \cdots, \gamma_2(m), \gamma_2(1))$$

where $\gamma_1 = (\gamma_1(1), \cdots, \gamma_1(n), \gamma_1(1))$ and $\gamma_2 = (\gamma_2(1), \cdots, \gamma_2(m), \gamma_2(1))$ and

$$\gamma_1^{-1} = (\gamma_1(1), \gamma_1(n), \cdots, \gamma_1(2), \gamma_1(1))$$

We denote this group at i by $\mathcal{O}_i$. Furthermore, $\mathcal{O}_i^{even}$ is the subgroup which consists of all the loops of even length at i, i.e.,

$$\mathcal{O}_i^{even} = \{\gamma \in \mathcal{O}_i \mid |\gamma| \text{ is even}\}$$

Note that the map $cr_{\mathfrak{X}}$ restricted on $\mathcal{O}_i^{even}$ is a group homomorphism to $\mathbb{H}$.

The next proposition shows how the spin multi-ratio changes under a spin transformation.

Proposition 3.22. *Let* s_ϕ *be the spin transformation*

$$s_\phi : \mathfrak{X} \mapsto \mathfrak{X}'$$

with respect to the spinor ϕ. Then

$$cr_{\mathfrak{X}'}(\gamma_i) = \begin{cases} \phi_i^{-1} \cdot cr_{\mathfrak{X}}(\gamma_i) \cdot \phi_i & |\gamma_i| \text{ is even} \\ \phi_i^{-1} \cdot cr_{\mathfrak{X}}(\gamma_i) \cdot \overline{\phi_i}^{-1} & |\gamma_i| \text{ is odd} \end{cases}$$

Therefore the argument and the norm of the spin multi-ratio are preserved if the length of the loop is even.

From now on we simply index the faces in the loop by $\gamma = (1, 2, \cdots, n, 1)$.

Remark 3.23. The norm of the spin multi-ratio contains the information of the edge length as well as the bending angles:

$$|cr(\gamma)| = |E_{12}^{-1}| \cdot |E_{23}| \cdot \cdots \cdot |E_{n1}|^{(-1)^n}$$
$$= |\cos\frac{\theta_{12}}{2}| \cdot \cdots \cdot |\cos\frac{\theta_{n1}}{2}|^{-1^{n+1}} \cdot |e_{12}|^{-1} \cdot \cdots \cdot |e_{n1}|^{(-1)^n}$$

Proposition 3.24. *For a loop γ of even length the axis of the spin multi-ratio $cr(\gamma)$ is always parallel to the normal n_i. For a loop with odd length the spin multi-ratio is always purely imaginary and perpendicular to n_i.*

Proof. Consider the rotation of n_i by $cr(\gamma)$:

$$cr(\gamma)^{-1} \cdot n_i \cdot cr(\gamma)$$

it can be decomposed to successive rotations and each of these rotations takes the normal $n_{\gamma(i)}$ to the $-n_{\gamma(i+1)}$. Hence after an even number of rotations the normal n_i comes back to itself, i.e.,

$$cr(\gamma)^{-1} \cdot n_i \cdot cr(\gamma) = n_i$$

Since n_i is a fix point of rotation represented by $cr(\gamma)$, the axis of $cr(\gamma)$ is exactly n_i. In case of an odd number of rotations one ends up with

$$cr(\gamma)^{-1} \cdot n_i \cdot cr(\gamma) = -n_i$$

so $cr(\gamma)$ must furnish a 180 degree rotation (thus it is purely imaginary) with an axis perpendicular to n_i. $\qquad\square$

With remark 3.23 and proposition 3.24 we have a clear understanding of the geometric meaning of the norm and direction of the spin multi-ratio. Next we are going to show some geometric interpretation of its argument. Since now we only care about the argument, we use a modified version of spin multi-ratio, denoted by $\hat{cr}$, for the purpose of simplicity.

$$\hat{cr}(\gamma) := E_{12} \cdot E_{23} \cdot \cdots \cdot E_{n1}$$

which differs from the true spin multi-ratio only by a scalar factor.

The rough idea is the following: one can rigidly unfold a classical net so that the spin multi-ratio would be factorized into two parts, both of which are easily understood. If the net is not classical one can first project the edges onto the planes perpendicular to the normals and carry out the unfolding.

Lemma 3.25. *Let e_{ij}^i be the pure imaginary quaternion with the same length as E_{ij} and parallel to the projection of e_{ij} onto the plane perpendicular to n_i, i.e.,*

$$e_{ij}^i = \frac{|E_{ij}| \cdot (e_{ij} - \langle e_{ij}, n_i \rangle n_i)}{|e_{ij} - \langle e_{ij}, n_i \rangle n_i|}$$

Then E_{ij} can be factorized into $E_{ij} = e_{ij}^i \cdot h_{ij}$, where h_{ij} is the quaternion satisfying the following properties:

1. *h_{ij} is a unit quaternion with positive real part.*

2. *The axis of h_{ij} is perpendicular both to n_i and n_j.* (3.15)

3. *$h_{ij}^{-1} \cdot n_i \cdot h_{ij} = n_j$.*

Proof. It is easy to show that $|e_{ij}^i| = |E_{ij}|$ and hence $|h_{ij}| = 1$. Then we have

$$h_{ij} = \epsilon \left(-e_{ij} + \langle e_{ij}, n_i \rangle n_i \right) \cdot E_{ij}$$

$$= \epsilon \left(-e_{ij} + \langle e_{ij}, n_i \rangle n_i \right) \cdot \left(\tan \frac{\theta_{ij}}{2} |e_{ij}| + e_{ij} \right)$$

$$= \epsilon \left(|e_{ij}|^2 - \langle e_{ij}, n_i \rangle^2 - \tan \frac{\theta_{ij}}{2} |e_{ij}| e_{ij} + \tan \frac{\theta_{ij}}{2} |e_{ij}| \langle e_{ij}, n_i \rangle n_i \right.$$

$$\left. + \langle e_{ij}, n_i \rangle n_i \times e_{ij} \right)$$

where ϵ is some positive number. It follows that

$$\text{Re}(h_{ij}) = |e_{ij}|^2 - \langle e_{ij}, n_i \rangle^2 = |e_{ij}^i|^2 > 0$$

and

$$\langle \text{Im}(h_{ij}), n_i \rangle = \langle -\tan \frac{\theta_{ij}}{2} |e_{ij}| e_{ij} + \tan \frac{\theta_{ij}}{2} |e_{ij}| \langle e_{ij}, n_i \rangle n_i + \langle e_{ij}, n_i \rangle n_i \times e_{ij}, n_i \rangle$$

$$= -\tan \frac{\theta_{ij}}{2} |e_{ij}| \langle e_{ij}, n_i \rangle + \tan \frac{\theta_{ij}}{2} |e_{ij}| \langle e_{ij}, n_i \rangle$$

$$= 0.$$

Note that $\text{Im}(e^i_{ij}) \perp n_i$ and $e^i_{ij} \cdot n_i \cdot (e^i_{ij})^{-1}$ represents the transformation which rotates n_i around the axis $\text{Im}(e^i_{ij})$ about 180 degree, hence

$$e^i_{ij} \cdot n_i \cdot (e^i_{ij})^{-1} = -n_i$$

and therefore

$$
\begin{aligned}
h^{-1}_{ij} \cdot n_i \cdot h_{ij} &= E^{-1}_{ij} \cdot e^i_{ij} \cdot n_i \cdot (e^i_{ij})^{-1} \cdot E_{ij} \\
&= -E_{ij} \cdot n_i \cdot E_{ij} \\
&= n_j
\end{aligned}
$$

which as well implies that

$$\text{Im}(h_{ij}) \perp n_j .$$

$\square$

Theorem 3.26. *The modified spin multi-ratio can be written as*

$$\hat{cr}_{\mathfrak{X}}(\gamma) = |\hat{cr}_{\mathfrak{X}}(\gamma)|(\mathfrak{e}_{12} \cdot \mathfrak{e}_{23} \cdots \cdots \mathfrak{e}_{n,1}) \cdot (h_{12} \cdots \cdots h_{n,1})$$

where $\mathfrak{e}_{i,i+1} \perp n_1$. If $\mathfrak{X}$ is classical then

$$\angle(\mathfrak{e}_{i-1,i}, \mathfrak{e}_{i,i+1}) = \angle(e_{i-1,i}, e_{i,i+1}).$$

Proof. Factorizing all the hyperedges E_{ij} the spin multi-ratio becomes

$$
\begin{aligned}
\hat{cr}_{\mathfrak{X}}(\gamma) &= E_{12} \cdot E_{23} \cdots \cdots E_{n,1} \\
&= e^1_{12} \cdot h_{12} \cdot e^2_{23} \cdot h_{23} \cdots \cdots e^n_{n,1} \cdot h_{n,1} \\
&= e^1_{12} \cdot (h_{12} \cdot e^2_{23} \cdot h^{-1}_{12}) \cdot (h_{12}h_{23} \cdot e^3_{34} \cdot h^{-1}_{23}h^{-1}_{12}) \cdots \cdot \\
&\quad\quad \cdot (h_{12}h_{23} \cdots \cdots h_{n-1,n} \cdot e^n_{1,n} \cdot h^{-1}_{n-1,n} \cdots \cdots h^{-1}_{23}h^{-1}_{12}) \\
&\quad\quad \cdot (h_{12} \cdot h_{23} \cdots \cdots h_{n-1,n}h_{n,1}).
\end{aligned}
$$

Let $\mathfrak{e}_{i,i+1} = h_{12} \cdots \cdots h_{i-1,i} \cdot e_{i,i+1} \cdot h^{-1}_{i-1,i} \cdots \cdots h^{-1}_{12}$ then, by (3.15) we have $\mathfrak{e}_{i,i+1} \perp n_1$ and $\hat{cr}(\gamma)$ has the form

$$\hat{cr}_{\mathfrak{X}}(\gamma) = |\hat{cr}_{\mathfrak{X}}(\gamma)|(\mathfrak{e}_{12} \cdot \mathfrak{e}_{23} \cdots \cdots \mathfrak{e}_{n,1}) \cdot (h_{12} \cdots \cdots h_{n,1}).$$

If $\mathfrak{X}$ is classical, then $e^i_{ij} = e_{ij}$ and

$$\angle(e_{i-1,i}, e_{i,i+1}) = \angle(e_{i-1,i}, h_{i-1,i} \cdot e_{i,i+1} \cdot h^{-1}_{i-1,i})$$

because the axis of $h_{i-1,i}$ is parallel to $e_{i,i-1}$. Applying the same rotation on $e_{i-1,i}$ and $h_{i-1,i} \cdot e_{i,i+1} \cdot h^{-1}_{i-1,i}$ we get

$$\angle(\mathfrak{e}_{i-1,i}, \mathfrak{e}_{i,i+1}) = \angle(e_{i-1,i}, e_{i,i+1}).$$

$\square$
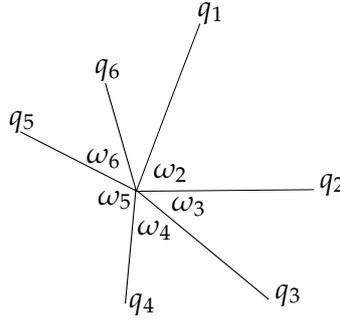
Figure 3.8: The product of quaternions in $\mathbb{i}\mathbb{j}$-plane

Therefore, up to scaling, the spin multi-ratio can be written as the product of two factors: We call $\mathfrak{e}_{12} \cdot \mathfrak{e}_{23} \cdot \ldots \cdot \mathfrak{e}_{n,1}$ the edge part and $h_{12} \cdot h_{23} \cdot \ldots \cdot h_{n,1}$ the curvature part. To understand the edge part we need the following lemma:

Lemma 3.27. *Suppose n is an even number. Let $q_1 = \cos \omega_1 \mathbb{i} + \sin \omega_1 \mathbb{j}$ and*

$$q_i = \cos(\omega_1 - \sum_{i=2}^{n} \omega_i)\mathbb{i} + \sin(\omega_1 - \sum_{i=2}^{n} \omega_i)\mathbb{j}$$

(see fig. 3.8). Then

$$q_1 \cdot q_2 \cdot \ldots q_n = \begin{cases} \cos(\Phi) + \sin(\Phi)\mathbb{k} & n = 0 \mod 4 \\ -\cos(\Phi) - \sin(\Phi)\mathbb{k} & n = 2 \mod 4 \end{cases}$$

where $\Phi = \sum_{i=1}^{n/2} \omega_{2i}$.

We can prove the case that $n = 2, 4$ by direct computation and generalize it by the induction.

Since $\mathfrak{e}_{i,i+1}$ are all coplanar, by lemma 3.27 we have:

$$\mathfrak{e}_{12} \cdot \mathfrak{e}_{23} \cdot \ldots \cdot \mathfrak{e}_{n,1} = \pm(\cos(\Phi) + \sin(\Phi)\mathbb{k})$$

where $\Phi = \sum_{i=1}^{n/2} \omega_{2i}$ and ω_i is the angle between the edges $\mathfrak{e}_{i-1,i}$ and $\mathfrak{e}_{i,i+1}$.

The Argument of the Spin Multi-Ratio and the Angular Defect

The angular defect around a vertex is known to be a polyhedral analog of Gaussian curvature and as such plays an important role in discrete differential geometry and we will show that it is closely related to the argument of the spin multi-ratio.

From now on we consider, for simplicity, a special set of loops which enclose only one vertex without duplicated dual edges. We call these loops fundamental. The even fundamental loops are the fundamental loops enclosing a vertex with even degree. In the
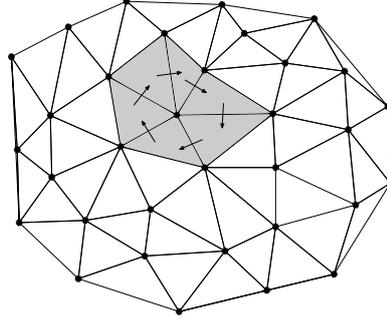
Figure 3.9: A fundamental loop

following cr(v) denotes the spin multi-ratio of the fundamental loop enclosing the vertex v. If no starting point of the fundamental loop is specified then cr(v) is well-defined up to conjugation in $\mathbb{H}$.

A vertex is called regular if and only if

$$\langle e_{i,i+1} \times e_{i-1,i}, n_i \rangle > 0 \tag{3.16}$$

holds for all incident edges. The angular defect of a regular vertex is defined by

$$\kappa(v) = 2\pi - \sum_{i=1}^{n} \omega_i$$

where ω_i is the angle between $\mathfrak{e}_{i-1,i}$ and $\mathfrak{e}_{i,i+1}$.

Let $u \in S^2 \subset \mathbb{R}^3 \cong \text{Im}(\mathbb{H})$ and

$$Q_u = \left\{ q = a + b \cdot u \,\middle|\, a, b \in \mathbb{R}, a^2 + b^2 = 1 \right\}$$

we have a map i_u defined by:

$$\mathbb{R} \to Q_u$$
$$\theta \mapsto \cos\frac{\theta}{2} + \sin\frac{\theta}{2}u$$

It is easy to check that i_u restricted to $(-2\pi, 2\pi) \to Q_u \setminus \{-1\}$ is bijective.

Theorem 3.28. *Let $h_{i,i+1}$ be the quaternions satisfying the conditions* (3.15). *Then*

$$h_{12} \cdot h_{23} \cdot \ldots \cdot h_{n,1} = i_{n_1}(\kappa(v))$$

Proof. There are two unit quaternions, which differ by a sign, satisfying $h_{i,i+1}^{-1} \cdot n_i \cdot h_{i,i+1} = n_{i+1}$, so $h_{i,i+1}$ with positive real part is uniquely defined. Note that

$$h_{n,1}^{-1} \cdot \ldots \cdot h_{12}^{-1} \cdot n_1 \cdot h_{12} \cdot \ldots \cdot h_{n,1} = n_1$$

the axis of $h_{12} \cdot \ldots \cdot h_{n,1}$ is parallel to n_1 hence indeed

$$h_{12} \cdot \ldots \cdot h_{n,1} \in Q_{n_1}.$$

If we cut along the edge $e_{n,1}$, fix the face Δ_1 and unfold the faces along the path, then it gives a planar pattern, where the original edge $e_{n,1}$ incident to face Δ_1 is denoted by $e^1_{n,1}$ and the edge $e_{n,1}$ incident to Δ_n is denoted by $e^n_{n,1}$. It follows that

$$h^{-1}_{n,1} \cdot \cdots \cdot h^{-1}_{12} \cdot e^1_{n,1} \cdot h_{12} \cdot \cdots \cdot h_{n,1} = e^n_{n,1}$$

and hence

$$h_{12} \cdot \cdots \cdot h_{n,1} = i_{n_1}(\pm\kappa(v)).$$

To see that it indeed gives the right sign, observe that any pattern of vertex star can be deformed continuously to a planar pattern. Moreover we can always continously increase the angular defect while it's negative and decrease it while it's positive until $\kappa = 0$. During the deformation the value $h_{12} \cdot \cdots \cdot h_{n,1}$ changes continuously until it becomes 1 and it will never go through the value -1. Therefore we only have to check the sign for the planar pattern and the sign of the other cases will be determined accordingly. In fact, the planar vertex star has $\kappa = 0$, and all h_{ij} would be just 1. Hence we have

$$h_{12} \cdot \cdots \cdot h_{n,1} = 1 = Q_{n_1}(0).$$

$\square$

Remark 3.29. We can take the following example to visualize the map $i_{n_1} : (-2\pi, 2\pi) \to Q_{n_1}$. Assuming that two vertex stars S_1 and S_2 in fig. 3.10 have the same rotation angle
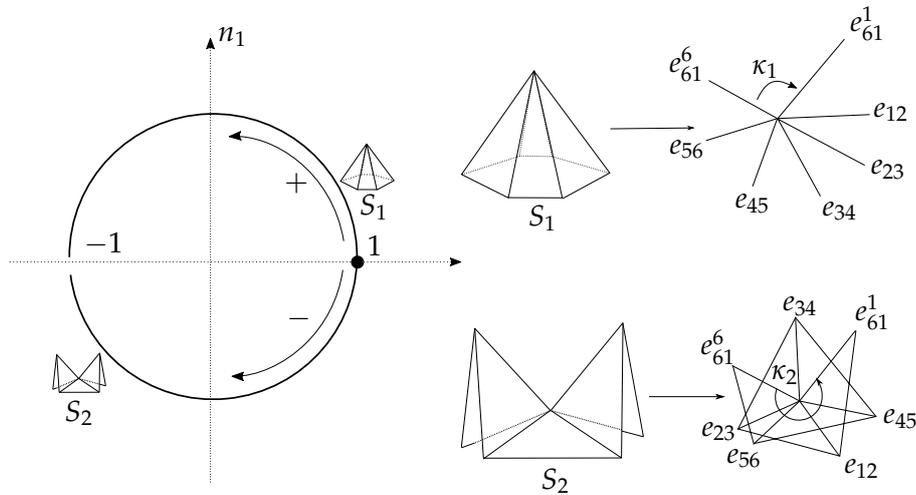
Figure 3.10: A sketch of the map i_{n_1}.

between $e^1_{n,1}$ and $e^n_{n,1}$, we can determine their positions up to the antipodal points on the circle. Observe that S_1 can be deformed to the planar vertex star without going through any pattern with angular defect $\pm\pi$, which are corresponding to the points $\pm n_1$ on the circle. Hence S_1 should sit in the first quadrant. By the analogous argument S_2 should sit in the third quadrant.

As a result the the argument of the spin multi-ratio can be characterized as follows:

Theorem 3.30. *Suppose γ is a loop of even length. The spin multi-ratio can be written as*

$$\frac{\mathrm{cr}_{\mathfrak{X}}(\gamma)}{|\mathrm{cr}_{\mathfrak{X}}(\gamma)|} = \pm\left(\cos\frac{\Phi}{2} + \sin\frac{\Phi}{2}n_1\right)$$

where $\Phi = \kappa(v) + 2 \cdot \sum\limits_{i=1}^{n/2} \omega_{2i}$.

Remark 3.31. The argument of a vertex star with angular defect κ is the sum of the angles for the shaded regions in fig. 3.11.

Since $\kappa = 2\pi - \sum\limits_{i=1}^{n} \omega_i$, we can rewrite the argument as the alternating sum of the angles ω_i:

$$\Phi = 2\pi + \sum_{i=1}^{n}(-1)^i\omega_i .$$
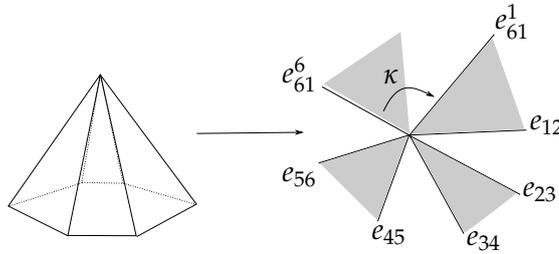
Figure 3.11: The argument of the spin multi-ratio.

3.2.4 Spin Equivalence

We are now able to show, that the spin multi-ratio determines the net up to spin transformations.

Definition 3.32. Given two face edge-constraint nets $\mathfrak{X}$ and $\mathfrak{X}'$ if there exists a spinor ϕ with $|\phi_i| \neq 0$ for all i such that

$$\mathrm{s}_\phi(\mathfrak{X}) = \mathfrak{X}'$$

then we say that $\mathfrak{X}$ and $\mathfrak{X}'$ are spin equivalent.

Theorem 3.33. *Given two face edge-constraint nets $\mathfrak{X}$ and $\mathfrak{X}'$, if $\mathrm{cr}_{\mathfrak{X}}(\gamma)$ and $\mathrm{cr}_{\mathfrak{X}'}(\gamma)$ have the same argument and norm for all $\gamma \in \mathcal{O}_i^{even}$ then they are spin equivalent. Moreover, if all the vertices in X have even degree then there are a family of the spinor ϕ_λ, parametrized by S^1, giving the spin transformation between $\mathfrak{X}$ and $\mathfrak{X}'$. If there is exists at least one vertex with odd degree then the spinor is unique.*

Proof. First consider the case with only even degree vertices, then all the loops have even length. Choose the ϕ_i such that $n_i' = \phi_i^{-1} \cdot n_i \cdot \phi_i$. Note that all the possible choices

form a S^1-parametrized set. Now we want to determine the value at the face j. First take a path from i to j

$$\gamma = (i = 1, 2, \cdots, n = j)$$

and by induction let

$$\phi_{m+1} = E_{m,m+1}^{-1} \cdot \overline{\phi_m^{-1}} \cdot E'_{m,m+1} \tag{3.17}$$

for $m = 1, \cdots, n-1$. Now, we just need to check that the value of ϕ is independent on the choice of path. Suppose γ_1 and γ_2, with $|\gamma_1| = m_1$ and $|\gamma_2| = m_2$ are two paths connecting i and j. Label the in-between vertices by:

$$\gamma_1 = (i = \gamma_1(1), \gamma_1(2), \cdots, \gamma_1(m_1) = j)$$

and

$$\gamma_2 = (i = \gamma_2(1), \gamma_2(2), \cdots,, \cdots, \gamma_2(m_2) = j).$$

Since $|\gamma_1| + |\gamma_2|$ is even, $|\gamma_1|$ and $|\gamma_2|$ are either both even or both odd. Suppose that they are both even, then computing the value of ϕ_j along the path γ_1 we obtain that

$$\phi'_j = E_{\gamma_1(m_1-1),\gamma_1(m_1)}^{-1} \cdot \cdots \cdot \overline{E_{\gamma_1(1),\gamma_2(2)}} \cdot \phi_i \cdot \overline{E'_{\gamma_1(1),\gamma_1(2)}}^{-1} \cdots \cdot E'_{\gamma_1(m_1-1),\gamma_1(m_1)}.$$

Then, computing the value of ϕ_j along the path γ_2 we find

$$\phi''_j = E_{\gamma_2(m_2-1),\gamma_2(m_2)}^{-1} \cdot \cdots \cdot \overline{E_{\gamma_2(1),\gamma_2(2)}} \cdot \phi_i \cdot \overline{E'_{\gamma_2(1),\gamma_2(2)}}^{-1} \cdots \cdot E'_{\gamma_2(m_2-1),\gamma_2(m_2)}.$$

Note that $\gamma_1 \cdot \gamma_2^{-1}$ forms an even loop, so $\mathrm{cr}_{\mathfrak{x}}(\gamma_1 \cdot \gamma_2^{-1})$ and $\mathrm{cr}_{\mathfrak{x}'}(\gamma_1 \cdot \gamma_2^{-1})$ have the same argument and norm. Besides, the axis of $\mathfrak{X}(\gamma_1 \cdot \gamma_2^{-1})$ is parallel to n_i and the axis of $\mathrm{cr}_{\mathfrak{x}'}(\gamma_1 \cdot \gamma_2^{-1})$ is parallel to n'_i. Therefore we have

$$\phi_i^{-1} \cdot \mathrm{cr}_{\mathfrak{x}}(\gamma_1 \cdot \gamma_2^{-1}) \cdot \phi_i = \mathrm{cr}_{\mathfrak{x}'}(\gamma_1 \cdot \gamma_2^{-1})$$

where by definition

$$\mathrm{cr}_{\mathfrak{x}}(\gamma_1 \cdot \gamma_2^{-1}) = \overline{E_{\gamma_1(1),\gamma_1(2)}}^{-1} \cdot E_{\gamma_1(2),\gamma_1(3)} \cdot \cdots \cdot E_{\gamma_2(2),\gamma_2(1)}$$

and

$$\mathrm{cr}_{\mathfrak{x}'}(\gamma_1 \cdot \gamma_2^{-1}) = \overline{E'_{\gamma_1(1),\gamma_1(2)}}^{-1} \cdot E'_{\gamma_1(2),\gamma_1(3)} \cdot \cdots \cdot E'_{\gamma_2(2),\gamma_2(1)}.$$

It then follows that

$$\begin{aligned}
\phi''_i &= E_{\gamma_2(m_2-1),\gamma_2(m_2)}^{-1} \cdot \cdots \cdot \overline{E_{\gamma_2(1),\gamma_2(2)}} \cdot \mathrm{cr}_{\mathfrak{x}}^{-1}(\gamma_1 \cdot \gamma_2^{-1}) \cdot \phi_i \\
&\quad \cdot \mathrm{cr}_{\mathfrak{x}'}(\gamma_1 \cdot \gamma_2^{-1}) \cdot \overline{E'}_{\gamma_2(1),\gamma_2(2)}^{-1} \cdots \cdot E'_{\gamma_2(m_2-1),\gamma_2(m_2)} \\
&= \phi'_i.
\end{aligned}$$

The argument is analogous for the case of $|\gamma_1|$ and $|\gamma_2|$ both being odd.

If there exists an odd loop $\gamma_o \in \mathcal{O}_i$, then we can first determine all the values of ϕ

lying on the loop γ_o by (3.17). Since $\text{cr}_{\mathfrak{x}}(\gamma_o)$ and $\text{cr}_{\mathfrak{x}'}(\gamma_o)$ are both pure imaginary and perpendicular to n_i and n_i' respectively, there is a unique ϕ_i satisfying the following conditions:

$$\phi_i^{-1} \cdot n_i \cdot \phi_i = n_i',$$
$$\overline{\phi_i} \cdot \text{cr}_{\mathfrak{x}}(\gamma_o) \cdot \phi_i = \text{cr}_{\mathfrak{x}'}(\gamma_o).$$

Fixing this ϕ_i, the values of the other ϕ on the loop ϕ_o are then all compatibly determined. To determine the values of ϕ on the other vertices j away from γ_o we just need to again take some path between i and j, if the path has even length, we are done. Otherwise we can precompose the path with γ_o and obtain a even path. It remains to determine the values of ϕ on this path by (3.17). $\qquad \square$

3.3 A Discretization of the Intrinsic Dirac Operator

Next, we aim to find a discrete version of the above relation. We start with

3.3.1 A Discrete Principal Bundle

Following the ideas from [KP16] we construct the discrete principal bundle by the connection between neighbouring faces.

Definition 3.34. Let X be an oriented net. We call (P, X, G, η) a discrete principal bundle with connection if

1. each face Δ_i is assigned with a manifold P_i with a right action, free and transitive, by a Lie group G.

2. $P = \{P_i\}$ is a collection of the manifolds P_i.

3. each oriented dual edge $\vec{ij}$ is endowed with a connection $\eta_{ij} : P_i \to P_j$ such that $\eta_{ij}(p \cdot g) = \eta_{ij}(p) \cdot g$ and $\eta_{ji} \circ \eta_{ij} = \text{Id}$.

Integrating the connections along the fundamental loop around a vertex v we obtain the holonomy $\Omega_p^v \in G$:

$$p \cdot \Omega_p^v := \eta_{n,1} \circ \ldots \circ \eta_{23} \circ \eta_{12}(p)$$

It is easy to see that $\Omega_{pg}^v = \text{Ad}_{g^{-1}} \Omega_p^v$, hence the holonomy of the same fibre all lie in the same conjugate class.

We know that the spin group $\text{Spin}(n)$ is a two-fold covering of $\text{SO}(n)$, namely the following short exact sequence holds:

$$0 \to \mathbb{Z}_2 \to \text{Spin}(n) \xrightarrow{\xi_0} \text{SO}(n) \to 0$$

where ξ_0 is the adjoint representation. Given a $\text{SO}(n)$-principal bundle

$$(P_{SO}, X, \text{SO}(n), \eta),$$

a lifting is a $\text{Spin}(n)$-principal bundle $(P_{Spin}, X, \text{Spin}(n), \tilde{\eta})$ together with a set of maps $\xi_i : P^i_{Spin} \to P^i_{SO}$ which are compatible with the connections, i.e. the following diagram commutes at each dual edge $\vec{ij}$:

$$
\begin{array}{ccc}
P^i_{Spin} & \xrightarrow{\eta^{ij}} & P^j_{Spin} \\
\downarrow{\scriptstyle \xi} & & \downarrow{\scriptstyle \xi} \\
P^i_{SO} & \xrightarrow{\tilde{\eta}^{ij}} & P^j_{SO}
\end{array}
$$

If $n = 2$, then since $\text{SO}(2)$ and $\text{Spin}(2)$ are both abelian groups, the holonomy of the loop is well-defined without specifying an point p in the fibre.

3.3.2 Discrete Associated Bundle

Definition 3.35. We consider a principal G-bundle P_G and a vector space W with the left action by G. Take a product space $P_G \times W$ modulo the relation $\sim$:

$$(p, v) \sim (pg^{-1}, gv)$$

We call $P_G \times_\sim W$ the associated bundle to P_G. The connection on the associated bundle is

$$(p, v)_i \mapsto (\eta_{ij}(p), v)_j.$$

Since

$$
\begin{array}{ccc}
(p, v)_i & \longmapsto & (\eta_{ij}(p), v)_j \\
\downarrow{\scriptstyle \sim} & & \downarrow{\scriptstyle \sim} \\
(p \cdot g^{-1}, gv) & \longmapsto & (\eta_{ij}(p) \cdot g^{-1}, gv)
\end{array}
$$

commutes, the connection is well-defined on the associated bundle. In order to define the Clifford multiplication on bundle level we need to check the covariance. Let S denote the irreducible Clifford module. Since there is a bundle isomorphism $P_{SO} \times W \cong P_{Spin} \times_{Ad} W$, the Clifford multiplication can be defined as follows

$$
\begin{aligned}
\left(P_{Spin} \times_{Ad} W\right) \times \left(P_{Spin} \times S\right) &\to P_{Spin} \times S \\
(p, v) \times (p, x) &\mapsto (p, v \cdot x)
\end{aligned}
$$

If we change p to pg^{-1}, it yields

$$(pg^{-1}, gvg^{-1}) \cdot (pg^{-1}, gx) = (pg^{-1}, gv \cdot x) = (p, vx).$$

Hence, the multiplication is independent of the choice of p. It is also easy to see that the Clifford multiplication is compatible with the connection, i.e. $\eta_{ij}(v) \cdot \tilde{\eta}_{ij}(x) = \tilde{\eta}_{ij}(v \cdot x)$,

or

$$(p,v)_i \times (p,x)_i \xrightarrow{\tilde{\eta}_{ij}} (\tilde{\eta}_{ij}(p),v)_j \times (\tilde{\eta}_{ij}(p),x)_j$$

$$(p, v \cdot x)_i \xrightarrow{\tilde{\eta}_{ij}} (\tilde{\eta}_{ij}(p), v \cdot x)_j$$

commutes.

3.3.3 The Discrete Dirac Operator

In order to introduce a discrete version of the spinor connection, which is necessary for the intrinsic Dirac operator, we propose the following setting of discrete intrinsic nets, which mimics smooth surfaces with Riemannian metric. In the end we will show that the discrete intrinsic Dirac operator arising from this setting couples with discrete extrinsic Dirac operator introduced in section 3.2 very well. Therefore they form a consistent framework together with the face edge-contraint net setting in section 3.2. The notion of discrete spinor connection is compatible with the one in the recent work [CKPS18], which is used for shape embedding problems.

Definition 3.36. An intrinsic net is an oriented net such that each face Δ_i is endowed with an Euclidean affine plane Affine(Δ_i) and every oriented edge e_{ij} in Δ_i is identified with a vector in Affine(Δ_i), denoted by e_{ij}^i such that the common edge is identified with the same length in the neighbouring faces, i.e., $|e_{ij}^i| = |e_{ij}^j|$.
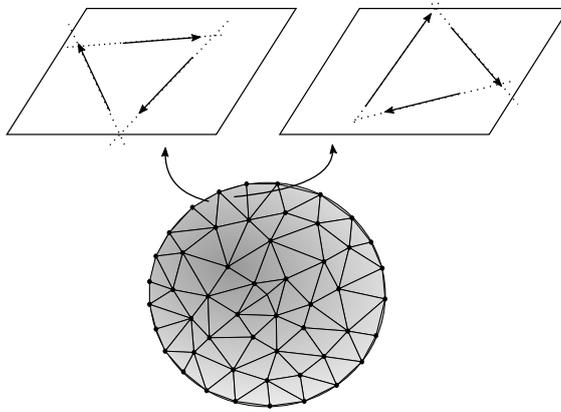
Figure 3.12: The intrinsic net

Remark 3.37. The the edges in a face do *not* need to form a closed polygon. However, it makes sense to define the angle between any pair of edges in a face by taking the angle between their extension lines (see fig. 3.12).

Definition 3.38. An oriented orthonormal frame of a face Δ_i is an oriented affine isometric map

$$p_i : \mathbb{R}^2 \to \text{Affine}(\Delta_i).$$

Let $p_1^i := p(\begin{pmatrix} 1 \\ 0 \end{pmatrix})$ and $p_2^i := p(\begin{pmatrix} 0 \\ 1 \end{pmatrix})$. Given a frame at Δ_i, the vector e_{ij}^i can be represented by a linear combination of that frame, denoted by $p^i(e_{ij}^i)$ or $\mathfrak{e}_{ij}^i$.

Definition 3.39. Suppose X is an intrinsic net. An orthonormal frame bundle with Levi-Civita-connection $P_{\mathrm{SO}}^{LC} \to X$ is a SO(2)-bundle consisting all the orthonormal frames at each face Δ_i satisfying $\big(\eta_{ij}(p^i)\big)\big(e_{ij}^j\big) = p^i(e_{ij}^i)$.

Now one can take any lift of the principal bundle with Levi-Civita-connection $P_{\mathrm{Spin}}^{LC} \to P_{\mathrm{SO}}^{LC}$. Then the tangent bundle can be constructed by

$$\mathrm{T}X := P_{\mathrm{Spin}}^{LC} \times_{\mathrm{Ad}} \mathbb{R}^2$$

and the spinor bundle can be constructed by

$$\mathcal{S} = P_{\mathrm{Spin}}^{LC} \times_{\mathrm{L}} S$$

where $S \cong \mathbb{H}$ is the irreducible Clifford module of Spin(2) and L denotes the left action of Spin(2) on S. Note that there is an isomorphism

$$\mathrm{T}X_i \xrightarrow{\cong} \mathrm{Affine}(\Delta_i),$$
$$(e, v) \mapsto e(v).$$

Therefore the Clifford multiplication is defined by

$$\mathrm{Affine}(\Delta_i) \times \mathcal{S}_i \to \mathcal{S}_i$$
$$(e, v) \times (e, x) \mapsto (e, v \cdot x)$$

and with this we are finally able to formulate a discrete intrinsic Dirac operator as follows:

Definition 3.40 (Discrete Dirac operator). Given an intrinsic net X and the principal bundle $P_{\mathrm{Spin}} \to P_{\mathrm{SO}}$ over X. The Dirac operator D is a map $\Gamma(\mathcal{S}) \to \Gamma(\mathcal{S})$ defined as follows:

$$D(\phi)_i = \frac{1}{2} \sum_j e_{ij} \cdot \tilde{\eta}_{ji}(\phi_j).$$

Note, that there is a well-defined Hermitian product

$$\Gamma(\mathcal{S}) \times \Gamma(\mathcal{S}) \to \mathcal{H},$$
$$\langle (p, x_1), (p, x_2) \rangle = \overline{x_1} \cdot x_2 .$$

Theorem 3.41. *Any ϕ satisfying the Dirac equation*

$$D\phi = \rho\phi$$

where $\rho : F \to \mathbb{R}$ is a real-valued function, gives rise to a face edge-constraint net by:

$$E_{ij} = \langle \phi_i, e_{ij} \cdot \tilde{\eta}_{ji}(\phi_j) \rangle,$$

$$n_i = \frac{1}{|\phi_i|^2} \langle \phi_i, \mathbb{k} \cdot \phi_i \rangle.$$

Proof. Compute

$$\sum_j E_{ij} = \sum_j \langle \phi_i, e_{ij} \cdot \tilde{\eta}_{ji}(\phi_j) \rangle$$
$$= \langle \phi_i, 2(D\phi)_i \rangle$$
$$= 2 \langle \phi_i, \rho \phi_i \rangle$$
$$= 2\rho |\phi|^2$$

which is a real number. $\qquad\square$

We will call these a face edge-constraint realization of the underlying intrinsic net with respect to the spinor ϕ.

3.3.4 Explicit Construction

Now let us derive an explicit formula for the Dirac equation as well as the face edge-constraint realizations. We begin by choosing an orthonormal frame $p_i = (p_1^i, p_2^i)$ at each face.

Let $g_{ij} \in \text{Spin}(2)$ be defined by $p_i \cdot g_{ij} = \tilde{\eta}_{ji}(p_j)$. Since $\tilde{\eta}_{ij} \circ \tilde{\eta}_{ji} = \text{Id}$, we have $g_{ij} = g_{ji}^{-1}$. Then we take an isometric embedding of the the affine plane $\text{Affine}(\Delta_i)$ and $\text{Affine}(\Delta_j)$ into $\mathbb{i}$-$\mathbb{j}$-plane such that

1. the common edge e_{ij}^i and e_{ij}^j coincide in this embedding.

2. p_1^i is mapped to $\mathbb{i}$ and p_2^i is mapped to $\mathbb{j}$.

Now, every vector in these two affine planes can be identified with a quaternion in the $\mathbb{i}$-$\mathbb{j}$-plane by:

$$v = x p_1^i + y p_2^j \mapsto x\mathbb{i} + y\mathbb{j}.$$

In particular

$$p_1^j \mapsto c_{11}\mathbb{i} + c_{12}\mathbb{j},$$
$$p_2^j \mapsto c_{21}\mathbb{i} + c_{22}\mathbb{j}.$$

We can find a quaternion g_{ji} such that

$$c_{11}\mathbb{i} + c_{12}\mathbb{j} = g_{ij}\mathbb{i}g_{ij}^{-1},$$
$$c_{21}\mathbb{i} + c_{22}\mathbb{j} = g_{ij}\mathbb{j}g_{ij}^{-1}.$$

In fact g_{ij} is uniquely defined up to a sign, which represent different liftings of the connection. We will see in the next section that the choice of the lifting actually determines the spin multi-ratio.

The parallel transport from a neighbouring face Δ_j is:

$$\tilde{\eta}_{ji}((p_j, \phi_j)) = (p_i \cdot g_{ij}, \phi_j)$$
$$= (p_i, g_{ij} \cdot \phi_j).$$

In Affine(Δ_i) we can write

$$e_{ij}^i = x p_1^i + y p_2^i \mapsto \mathbb{e}_{ij}^i = x\mathbb{i} + y\mathbb{j}.$$

Therefore, the Dirac operator becomes

$$D(\phi)_i = \frac{1}{2} \sum_j \mathbb{e}_{ij}^i \cdot g_{ij} \cdot \phi_j$$

and in the local frame p_i the Dirac equation has the form

$$\frac{1}{2} \sum_j \mathbb{e}_{ij}^i \cdot g_{ij} \cdot \phi_j = \rho_i \phi_i.$$

Moreover, a face edge-constraint realization is given by the explicit formula

$$E_{ij} = \overline{\phi_i} \cdot \mathbb{e}_{ij}^i \cdot g_{ij} \cdot \phi_j, \tag{3.18}$$

$$n_i = \phi_i^{-1} \cdot \mathbb{k} \cdot \phi_i. \tag{3.19}$$

To see that this realization is well-defined, we first compute

$$E_{ji} = \overline{\phi_j} \cdot \mathbb{e}_{ji}^j \cdot g_{ji} \cdot \phi_i.$$

Note that $\mathbb{e}_{ji}^j = -\mathbb{e}_{ij}^j = -g_{ij}^{-1} \cdot \mathbb{e}_{ij}^i \cdot g_{ij}$ and $g_{ij} = g_{ji}^{-1}$. This implies

$$E_{ji} = -\overline{\phi_j} \cdot g_{ij}^{-1} \mathbb{e}_{ij}^i \cdot g_{ij} \cdot g_{ji} \cdot \phi_i \tag{3.20}$$

$$= -\overline{\phi_j} \cdot g_{ij}^{-1} \cdot \mathbb{e}_{ij}^i \cdot \phi_i \tag{3.21}$$

and by $\overline{g_{ij}^{-1}} = g_{ij}$ we obtain $E_{ij} = \overline{E_{ji}}$. Finally we need to show that

$$E_{ij}^{-1} \cdot n_i \cdot E_{ij} = -n_j.$$

By direct computation we see

$$E_{ij}^{-1} \cdot n_i \cdot E_{ij} = \left(\overline{\phi_i} \cdot \mathbb{e}_{ij}^i \cdot g_{ij} \cdot \phi_j \right)^{-1} \cdot \phi_i^{-1} \cdot \mathbb{k} \cdot \phi_i \cdot \overline{\phi_i} \cdot \mathbb{e}_{ij}^i \cdot g_{ij} \phi_j \tag{3.22}$$

$$= \phi_j^{-1} g_{ij}^{-1} \cdot (-\mathbb{e}_{ij}^i) \cdot \mathbb{k} \cdot \mathbb{e}_{ij}^i \cdot g_{ij} \cdot \phi_j. \tag{3.23}$$

Since $\mathbb{e}_{ij}^i$ lies in the $\mathbb{i}$-$\mathbb{j}$-plane,

$$-\mathbb{e}_{ij}^i \cdot \mathbb{k} \cdot \mathbb{e}_{ij}^i = -\mathbb{k}$$

and g_{ij} has the axis parallel to $\mathbb{k}$, so $g_{ij}^{-1} \cdot \mathbb{k} \cdot g_{ij} = \mathbb{k}$, it follows that

$$E_{ij}^{-1} \cdot n_i \cdot E_{ij} = -\phi_j^{-1} \cdot \mathbb{k} \cdot \phi_j = -n_j.$$

3.3.5 The Preferred Choice for the Lifting

We know that in an intrinsic net each edge admits two liftings with opposite sign, hence an intrinsic net with n edges have 2^n different spinor connections. Now we are going to show that among all these spinor connections there are some more reasonable ones, called the preferred liftings, which correspond to the spinor structures in the smooth case.

Similar to eq. (3.16) we call a vertex in an intrinsic net regular if and only if

$$\langle \mathfrak{e}^i_{i-1,i} \times \mathfrak{e}^i_{i,i+1}, \Bbbk \rangle > 0.$$

Let v be a regular vertex with even degree and $\mathfrak{X}$ be the face edge-constraint realization of $(X, \mathcal{A})$ with respect to the spinor ϕ, then

$$
\begin{aligned}
\mathrm{cr}_{\mathfrak{X}}(v) &= \overline{E_{12}}^{-1} \cdot E_{23} \cdots \overline{E_{n-1,n}}^{-1} \cdot E_{n,1} \\
&= \phi_1^{-1} \mathfrak{e}^1_{12} \cdot g_{12} \frac{\phi_2}{|\phi_2|^2} \cdot \overline{\phi_2} \cdot \mathfrak{e}^2_{23} \cdot g_{23} \cdot \phi_3 \cdots \\
&\qquad \cdots \phi_{n-1}^{-1} \mathfrak{e}^{n-1}_{n-1,n} \cdot g_{n-1,n} \frac{\phi_n}{|\phi_n|^2} \cdot \overline{\phi_n} \cdot \mathfrak{e}^n_{n,1} \cdot g_{n,1} \cdot \phi_1 \\
&= \phi_1^{-1} \cdot \mathfrak{e}^1_{12} \cdot g_{12} \cdot \mathfrak{e}^2_{23} \cdot g_{23} \cdots \mathfrak{e}^{n-1}_{n-1,n} \cdot g_{n-1,n} \cdot \mathfrak{e}^n_{n,1} \cdot g_{n,1} \cdot \phi_1 \\
&= \phi_1^{-1} \cdot \mathfrak{e}^1_{12} \cdot (g_{12} \mathfrak{e}^2_{23} g_{12}^{-1}) \cdot g_{12} \cdot g_{23} \cdot \mathfrak{e}^3_{34} \cdots \mathfrak{e}^{n-1}_{n-1,n} \cdot g_{n-1,n} \cdot \mathfrak{e}^n_{n,1} \cdot g_{n,1} \phi_1 \\
&= \phi_1^{-1} \cdot \mathfrak{e}^1_{12} \cdot (g_{12} \mathfrak{e}^2_{23} g_{12}^{-1}) \cdot (g_{12} g_{23} \mathfrak{e}^3_{34} g_{23}^{-1} g_{12}^{-1}) \cdots \\
&\qquad \cdots (g_{12} \cdots g_{n-1,n} \mathfrak{e}^n_{n,1} g_{n-1,n}^{-1} \cdots g_{12}^{-1}) \cdot (g_{12} \cdots \cdots g_{n,1}) \cdot \phi_1 \\
&= \phi_1^{-1} \cdot \mathfrak{e}^1_{12} \cdot \mathfrak{e}^1_{23} \cdot \cdots \mathfrak{e}^1_{n-1,n} \cdot \mathfrak{e}^1_{n,1} \cdot (g_{12} \cdots \cdots g_{n,1}) \cdot \phi_1 \, .
\end{aligned}
$$

We call $\mathfrak{X} = (X, f, n)$ a classical realization of $(X, \mathcal{A})$ if and only if $\mathfrak{X}$ is classical and all the internal angles are preserved:

$$\angle(e_{i-1,i}, e_{i,i+1}) = \angle(\mathfrak{e}^i_{i-1,i}, \mathfrak{e}^i_{i,i+1}).$$

For a classical realization, observe that $e^1_{12} \cdot e^1_{23} \cdot \cdots \cdot e^1_{n-1,n} \cdot e^1_{n,1}$ actually coincides with the edge part of the spin multi-ratio of the classical realization. Hence $g_{12} \cdots \cdots g_{n,1}$ should coincide with the curvature part of the spin multi-ratio.

Definition 3.42. Let $(X, \mathcal{A})$ be an intrinsic net with only regular vertices. A choice of lifting is called a preferred lifting if

$$g_{12} \cdots \cdots g_{n,1} = \mathrm{i}_{\Bbbk}(\kappa(v))$$

holds for all vertices.

Lemma 3.43 (A Gauss-Bonet theorem for intrinsic nets). *Let $(X, \mathcal{A})$ be an intrinsic net such that for each face Δ_i the extension lines of the vectors e^i_{ij} form an oriented convex polygon.*

Suppose the total angular defect $\sum_i \kappa(v)$ is the sum of the angular defects of all the vertices. Then we have

$$\sum_{vertices} \kappa(v) = 2\pi\chi$$

where χ is the Euler characteristic.

Proof. We have

$$\sum_{vertices} \kappa(v) = \sum_{vertices} (2\pi - \Sigma(v))$$
$$= 2\pi|V| - \Sigma$$
$$= 2\pi|V| - \sum_{faces} \Sigma(\Delta_i)$$

where $\Sigma(v)$ is the sum of the interior angles at the vertex v, $\Sigma(\Delta_i)$ is the sum of the interior angles in Δ_i and Σ is the sum of all the interior angles. Assuming that in the face Δ_i the extension lines of the vectors form an oriented convex s_i-sided polygon, then the sum of the interior angles is $(s_i - 2)\pi$ and

$$\Sigma(\Delta_i) = (s_i - 2)\pi.$$

Further note, that $\sum_{faces} s_i = 2|E|$, hence

$$\sum_{vertices} \kappa(v) = 2\pi|V| - \sum_{faces} (s_i - 2)\pi$$
$$= 2\pi|V| - 2\pi|E| + 2\pi|F|$$
$$= 2\pi\chi.$$

$\square$

Theorem 3.44. *Every intrinsic net $(X, \mathcal{A})$ satisfying the condition in Lemma 3.43 has a preferred lifting.*

Proof. Any choice of the lifting g_{ij} gives a 2-cochain σ in the following way. Let μ be a map from the vertices to $\mathrm{Spin}(2)$ defined by

$$\mu[v] = g_{12} \cdot \cdots \cdot g_{1,n}$$

and let ν be the map defined by

$$\nu[v] = \mathrm{i}_{\Bbbk}(\kappa(v)).$$

Since $g_{i,i+1}$ all lie in the i-j-plane, μ and ν both indeed have the codomain $\mathrm{Spin}(2)$. Since $\mathrm{Spin}(2)$ is abelian, μ and ν can be linearly extended to the 2-cochains of X^*, i.e.,

$$\mu, \nu \in \mathrm{C}^2(X^*, \mathrm{Spin}(2)).$$

The 2-cochain σ is defined by

$$\sigma[v] := \mu[v] \cdot \nu[v]^{-1}.$$

Since $g_{12} \cdot \cdots \cdot g_{n,1} = \pm(i_{\Bbbk}(v))$, σ is actually a 2-cochain with coefficient $\mathbb{Z}_2$, i.e.,

$$\sigma \in C^2(X^*, \mathbb{Z}_2).$$

Clearly σ takes the value:

$$\sigma[v] = \begin{cases} 1 & i_{\Bbbk}(g_{21} \cdot \cdots \cdot g_{1,n}) = \kappa(v) \\ -1 & i_{\Bbbk}(g_{21} \cdot \cdots \cdot g_{1,n}) = -\kappa(v) \end{cases}.$$

If g_{ji} is a preferred lifting then $\sigma = 0$. If we change the lifting at some edges, then it leads to a 2-cochain σ' which only differs from σ by a differential of a 1-cochain:

$$\sigma' = \sigma + d\delta$$

where $\delta \in C^1(X^*, \mathbb{Z}_2)$. It implies that even though σ as a cochain depends on the lifting g_{ji},

$$\bar{\sigma} \in H^2(X^*, \mathbb{Z}_2)$$

as a cohomology class doesn't depend on the choice of the lifting but only depends on the SO-connection. Moreover $\bar{\sigma} = 0$ if and only if there exists a preferred lifting. Observe that

$$\sigma[X^*] = \mu[X^*] \cdot \nu[X^*]^{-1}$$

and we have $\mu[X^*] = \mathrm{Id}$ because every g_{ji} and g_{ij} always appear in pair in X^*. Furthermore $\sum_{v \in V} \kappa(v) = \chi \cdot 2\pi$ by lemma 3.43, which is always an even number for a oriented surface. Hence $\nu[X^*] = i_{\Bbbk}(\chi) = \mathrm{Id}$ and then

$$\sigma[X^*] = 1.$$

We know that there is only one nontrivial class $\omega \in H^2(X^*, \mathbb{Z}_2)$ but $\omega[X^*] = -1$, thus $\omega \neq \bar{\sigma}$ and $\bar{\sigma} = 0$. $\qquad\square$

Definition 3.45. Given an intrinsic net $(X, \mathcal{A})$ satisfying the condition in lemma 3.43, the spin equivalence class is the set of the pairs $(X, \mathcal{A}, \tilde{\eta})$ where $\tilde{\eta}$ is a preferred lifting of $(X, \mathcal{A})$ modulo the spin equivalence relation.

Theorem 3.46. *The spin equivalence class of an intrinsic net with Betti number b has 2^b elements.*

Proof. Let $(X, \mathcal{A}, \eta)$ and $(X, \mathcal{A}, \eta')$ be two preferred liftings of the same underlying intrinsic net. Since the spin multi-ratio at each vertex v should be the same for two liftings, at each vertex there should be even numbers of incident edges e_{ij} such that the η_{ij} and η'_{ij} have reversed signs. Hence all these edges form some closed boundaries.

For a simply-connected net these boundaries would create some separated disk-like areas. It's easy to see that any loops always cross these boundaries with an even number of times. Therefore the spin multi-ratio for all the even loops are the same for the liftings η and η', meaning that they are spin equivalent.

Suppose the X has the Betti number b, we can always find $2b$ closed curves which represent different non-trivial homology classes. Pick any of such a closed curve, flip the signs of the spinor connections all along this curve and we obtain a new spin equivalence class. □

Remark 3.47. Recall that in the smooth theories, an oriented manifold has the spin structure if and only if the second Stiefel-Whitney class is zero. Hence a oriented surface is spin if and only if the Euler characteristic is even (which is true for all oriented surfaces). Furthermore a spin manifold has 2^{2b} number of spin structures. Clearly theorem 3.44 and theorem 3.46 show that our discretization preserves all these results.

3.4 The Discrete Extrinsic and Intrinsic Dirac Operators

In the last section we started with an intrinsic net and constructed face edge-constraint realizations by solving the Dirac equation. Now we are going to discuss the question: How can we construct the intrinsic net from a given face edge-constraint net? In fact we will see that each face edge-constraint net is associated with an intrinsic net and a constant spinor field ϕ_c with unit length in the ambient space $\mathbb{R}^3$ induces a spinor field on the intrinsic net. With this induced spinor field one can reconstruct the original egde-constraint net from the associated Riemannain net. Moreover the relation between the extrinsic and intrinsic operators still holds in the discrete case. Precisely, the ideas can be depicted as follows: For each face Δ_i the hyperplane perpendicular to n_i gives a

$$\mathfrak{X} = (X, f, n) \xrightarrow[\text{solving } D_f \phi = H'\phi]{\mathrm{d}f' = \overline{\phi} \cdot \mathrm{d}f \cdot \phi} \mathfrak{X}' = (X, f', n')$$

with vertical arrow labelled "realization w.r.t ϕ_c" and "forget the immersion" down to $(X, \mathcal{A})$, and diagonal arrow labelled "realization w.r.t $\phi_c \cdot \phi$".
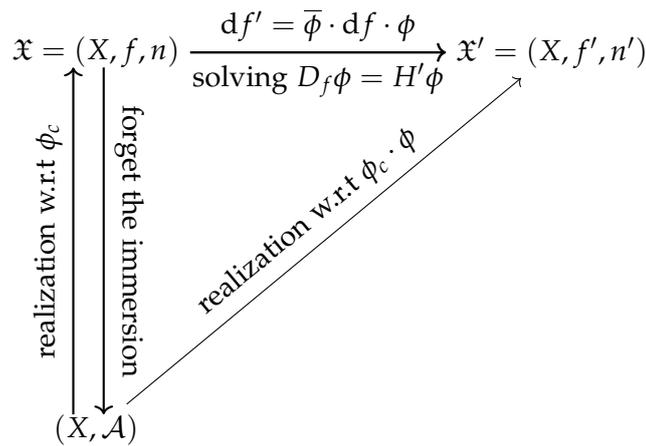
Figure 3.13: The relation between intrinsic and extrinsic Dirac operators

affine structure $\text{Affine}(\Delta_i)$, we then can identify the edge e_{ij} by

$$e_{ij}^i = |E_{ij}| \frac{e_{ij} - \langle e_{ij}, n_i \rangle n_i}{|e_{ij} - \langle e_{ij}, n_i \rangle n_i|}. \tag{3.24}$$

Fix a reference frame p_i for $\text{Affine}(\Delta_i)$ and then e_{ij}^i can be represented with p_i, denoted by $\mathfrak{e}_{ij}^i$.

Recall that in the smooth case there is a section of the spinor bundle $\mathcal{S} \to \mathbb{R}^3$ given by $\phi_c = (c, 1)$ where c is the globally parallel section of the spin bundle. An immersion of the surface $X \hookrightarrow \mathbb{R}^3$ induces a section of $\mathcal{S} \to X$ by restricting ϕ_c on X.

Now choose a unit quaternion $g_i \in \text{Spin}(3)$ such that

$$e_{ij}^i = g_i^{-1} \cdot \mathfrak{e}_{ij}^i \cdot g_i$$

The constant section of the spin bundle can be formally defined by

$$c = p_i \cdot g_i$$

Then we can rewrite the spinor field $(c, 1)$ as

$$\begin{aligned}
(c, 1) &= (p_i \cdot g_i, 1) \\
&= (p_i, g_i).
\end{aligned}$$

The spinor connection is then given by

$$g_{ij} = g_i \cdot h_{ij} \cdot g_j^{-1}$$

where h_{ij} is defined in lemma 3.25 with $E_{ij} = e_{ij}^i \cdot h_{ij}$. The Dirac equation yields:

$$\begin{aligned}
2D(\phi_c) &= \sum_j (p_i, \mathfrak{e}_{ij}^i) \cdot \tilde{\eta}_{ji}(c, 1) = \sum_j (p_i, \mathfrak{e}_{ij}^i) \cdot \tilde{\eta}_{ji}(p_j \cdot g_j, 1) \\
&= \sum_j (p_i, \mathfrak{e}_{ij}^i) \cdot \tilde{\eta}_{ji}(p_j, g_j) = \sum_j (p_i, \mathfrak{e}_{ij}^i) \cdot (p_i, g_{ij} \cdot g_j) \\
&= \sum_j (p_i, \mathfrak{e}_{ij}^i \cdot g_{ij} \cdot g_j) = \sum_j (p_i, g_i \cdot e_{ij}^i \cdot g_i^{-1} \cdot g_{ij} \cdot g_j) \\
&= \sum_j (p_i, g_i \cdot e_{ij}^i \cdot h_{ij}) = \sum_j (p_i, g_i \cdot E_{ij}) \\
&= (p_i, g_i \cdot (\sum_j E_{ij})) = 2\mathbf{H}_i \cdot (p_i, g_i) = 2\mathbf{H}_i \cdot (c, 1) \\
&= 2\mathbf{H}_i \cdot \phi_c.
\end{aligned}$$

It shows that the section ϕ_c satisfies the Dirac equation and the induced face edge-constraint realization exactly recovers the original face edge-constraint net.

Let $\mathcal{H}$ be functions from the faces to $\mathbb{H}$ and $\Gamma(\mathcal{S})$ be the spaces of the sections of the spinor bundle. The map $\mathfrak{c}$ is constructed by:

$$\mathfrak{c} : \Gamma(\mathcal{S}) \to \mathcal{H}$$
$$(c, \phi_i) \mapsto \phi_i.$$

The arguments above also imply that

$$\mathfrak{c} \circ (D - \mathbf{H}) \circ \mathfrak{c}^{-1} = D_f.$$

Compared with theorem 2.7 this shows that the discretization of the both operators preserves the relation of their smooth correspondence. Note that, with the affine structure (3.24) the intrinsic Dirac operator is different from the one in [YDT+18] by a cosine factor, which was introduced for the purpose of numerics, because in that case the Dirac operator would be covariant under edge-length preserving deformations. In our case, the intrinsic Dirac operator is covariant under hyperedge-length preserving deformations and hence it is more consistent with the extrinsic one (fig. 3.13).

3.5 Alternative discretizations

Although the discretization above inherit some nich properties, e.g., the closing condition and the formula for the mean curvature, from the smooth counterpart, we still do not have the discrete version of the Lichnernowicz forumla, which relates the Dirac operator with the more famous Laplace operator. It has been shown that some Lichnernowicz-type formula can be derived from the discrete extrinsic Dirac operator defined in [CPS11], which is defined for triangular meshes and maps vertex-based quaternion-valued functions to face-based quaternion-valued functions. In the following we first review this vertex-to-face extrinsic Dirac operator and then propose a similar but intrinsic one, from which we derive some discrete intrinsic Lichnernowicz forumla. However, we do not see that the closing condition and the mean curvature formula will emerge in this alternative setting.

3.5.1 The vertex-to-face extrinsic Dirac operator

We first derive the formulas for the vertex-to-face Dirac operator in [CPS11], denoted by $\mathfrak{D}_f$, using the finite element scheme. In the triangle k with vertices (v_1, v_2, v_3) we use the coordinate system (x, y) (see inset). Then the immersion f can be written as

$$f = v_1 + (v_2 - v_1)x + (v_3 - v_1)y$$

Let $a = v_3 - v_2$, $b = v_1 - v_3$ and $c = v_2 - v_1$. We have

$$\mathrm{d}f = c\,\mathrm{d}x - b\,\mathrm{d}y,$$
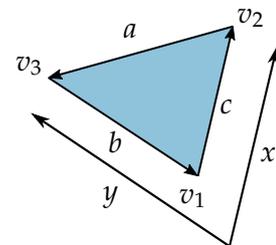
Similarly, it yields that ϕ can be written as the linear combination

$$\phi = \phi_1 + x(\phi_2 - \phi_1) + y(\phi_3 - \phi_1),$$

where ϕ_i is the linear functions which is supported in the star of v_i and vanish at the boundary of the star. It follows that

$$d\phi = (\phi_2 - \phi_1)dx + (\phi_3 - \phi_1)dy,$$

and hence the contributions of v_1, v_2 and v_3 on the triangle k is

$$
\begin{aligned}
(\mathfrak{D}_f\phi)_k &= -\frac{df \wedge d\phi}{|df|^2} = -\frac{1}{2A_k}\left(a(\phi_3 - \phi_1) + b(\phi_2 - \phi_1)\right) \\
&= -\frac{1}{2A_k}\left(a\phi_1 + b\phi_2 + c\phi_3\right),
\end{aligned}
\tag{3.25}
$$

where A_k is the area of the triangle k.

The spin transformation of the edge c is obtained by

$$
\begin{aligned}
\int_0^1 \overline{\phi} \cdot c\,dx \cdot \phi &= \int_0^1 ((1-x)\overline{\phi_1} + x\overline{\phi_2})c\,dx((1-x)\phi_1 + x\phi_2) \\
&= \frac{1}{3}\overline{\phi_1}c\phi_1 + \frac{1}{6}\overline{\phi_1}c\phi_2 + \frac{1}{6}\overline{\phi_2}c\phi_1 + \frac{1}{3}\overline{\phi_2}c\phi_2
\end{aligned}
$$

In smooth case, the extrinsic Dirac operator has the Lichnernowicz-type forumla [Cra13, Section 3.3]

$$D_f^2 = \boldsymbol{\Delta} + *(dn \wedge d\phi),$$

where the Laplacian here is Hodge-Laplacian defined on the trivial quaternion bundle, i.e. $\blacksquare = *d*d$. We found the discrete analogy in an unpublished manuscript by Ulrich Pinkall:

Since the operator $\mathfrak{D}_f$ goes from vertices to faces, its adjoint $\mathfrak{D}_f^*$ is given by the relation

$$(D_f\phi, \varphi)_F = (\phi, D_f^*\varphi)_V,$$

where in discrete case these inner products should be defined by

$$(\phi, \varphi)_F := \sum_k \overline{\phi}_k \varphi_k A_k,$$

and

$$(\phi, \varphi)_V := \sum_i \overline{\phi}_i \varphi_i A_i,$$

where A_i is the vertex area. Hence it is clear that $(\mathfrak{D}_f^*)_i = A_i^{-1}(\tilde{\mathfrak{D}}_f^T)_i$, where $\tilde{\mathfrak{D}}_f$ is just $\mathfrak{D}_f$ without the face area factor, namely

$$(\mathfrak{D}_f^*)_{1k} = \frac{1}{2A_1}a.\tag{3.26}$$

Actually the square of the Dirac operator in this case should be defined by

$$\mathfrak{D}_f^2 := \mathfrak{D}_f^* \circ \mathfrak{D}_f.$$

Apparently $\mathfrak{D}_f^2$ only sends the function values from one vertex to its neighboring vertices. Combining (3.25) and (3.26) we find any function value will go from vertex i to j through two faces k and l as follows
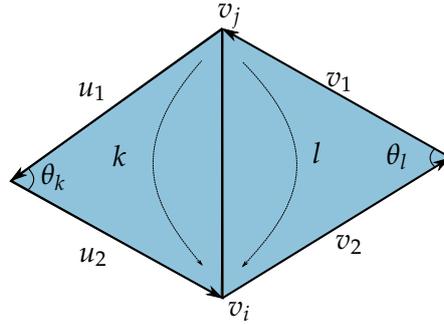

Figure 3.14: The square of the Dirac operator.

and hence

$$(\mathfrak{D}_f)_{ij} = -\frac{1}{4A_i}\left(\frac{u_1 \cdot u_2}{A_l} + \frac{v_1 \cdot v_2}{A_k}\right)$$

$$= \frac{1}{2A_i}(\underbrace{\cot\theta_k + \cot\theta_k}_{\text{real}}\underbrace{-n_k + n_l}_{\text{imaginary}})$$

Observe that the real part above does coincides with the cotangent Laplacian. Recall that the Laplace operator is associated with the Dirichlet energy.

In smooth case, we know that $(\Delta f, f) = (\nabla f, \nabla f)$, where the inner product is defined by $(f, g) = \int_X fg \, dA$. Hence the eigenfunction $(\Delta f, f) = \lambda|f|^2 = |\nabla f|^2$, which means the smoothness of f, measured by $|\nabla f|^2$ is controlled by the magnitude of the eigenvalue λ.

In discrete case let us take the cotangent Laplacian for example. We know that the cotangent Laplacian for a triangular graph with metric is a $|V| \times |V|$ matrix L such that

$$L_{ij} = \frac{\omega_{ij}}{A_i},$$

and

$$L_{ii} = -\frac{\sum_j \omega_{ij}}{A_i},$$

where the coefficients are given by $\omega_{ij} = \frac{1}{2}(\cot\theta_k + \cot\theta_l)$. Then we have

$$E(f) := (\Delta f, f)_V = \sum_{ij} \omega_{ij}|f_i - f_j|^2$$

Suppose f is an eigenfunction with small eigenvalue λ, namely $Lf = \lambda f$. Its Dirichlet energy is then satisfies

$$\sum_{ij} \omega_{ij} |f_i - f_j|^2 = (\Delta f, f)_V = \lambda |f|^2$$

3.5.2 Search for the Lichnernowicz formula

In smooth case, due to the famous Lichnerowicz–Weitzenböck formula (6.8), the Dirac eigenfunctions with $D\phi = \lambda\phi$ satisfy

$$\lambda^2 |\phi|^2 = \left((\Delta + \frac{1}{2}K)\phi, \phi \right),$$

it follows that

$$|\nabla \phi| = (\lambda^2 - K)|\phi|^2$$

which implies that the manifold with positive scalar curvature does not admit spinor harmonics.

Inspired by the formula in the last section, we think that it is easier to get the Lichnerowicz-formula with the vertex-to-face type operators. Therefore, we try to modify the extrinsic vertex-to-face Dirac operator by adding spin connections. We will see this modification brings the intrinsicness and the its square is related to a spin connection Laplacian.

We first define the vertex normal $n_i : V \to S^2$ at every normals. The intrinsic vertex-to-face Dirac operator is defined to be a quaternion-valued $|F| \times |V|$ matrix such that

$$\mathfrak{D}_{ki} = -\frac{1}{2A_k} e_k^i \cdot g_{ik}, \tag{3.27}$$

where e_k^i is the edge in face j opposite to the vertex i and g_{ik} is the unit quaternion with positive real part such that

$$g_{ik}^{-1} \cdot n_i \cdot g_{ik} = n_k.$$

Note that (3.27) differs from (3.25) only by the spin connection g_{ik}. Similar as (3.26) we have

$$(\mathfrak{D}^*)_{ik} = \frac{1}{2A_i} g_{ki} \cdot e_k^i, \tag{3.28}$$

where $g_{ki} = g_{ik}^{-1}$. To simplify the notation, we use the indices in the following patch:
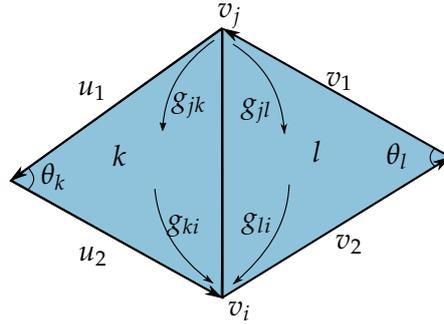
Figure 3.15: Spin connection from vertices to faces.

and it yields

$$
\begin{aligned}
(\mathfrak{D}^* \circ \mathfrak{D})_{ij} &= -\frac{1}{4A_i}\left(\frac{1}{A_k}g_{ki} \cdot u_1 \cdot u_2 \cdot g_{jk} + \frac{1}{A_l}g_{li} \cdot v_1 \cdot v_2 \cdot g_{jl}\right) \\
&= -\frac{1}{2A_i}\Big(\frac{1}{\sin\theta_k|u_1||u_2|}g_{ki}(-\langle u_1, u_2\rangle + u_1 \times u_2)g_{jk} \\
&\quad + \frac{1}{\sin\theta_l|v_1||v_2|}g_{li}(-\langle v_1, v_2\rangle + v_1 \times v_2)g_{jl})\Big) \\
&= \frac{1}{2A_i}\left(\cot\theta_k \cdot g_{ki}g_{jk} - g_{ki}n_kg_{jk} + \cot\theta_l \cdot g_{li}g_{jl} + g_{li}n_lg_{jl}\right) \\
&= \frac{1}{2A_i}\left(\cot\theta_k \cdot g_{ki}g_{jk} + \cot\theta_l \cdot g_{li}g_{jl} + n_i(-g_{ki}g_{jk} + g_{li}g_{jl})\right) \\
&= \frac{1}{2A_i}\Big(\underbrace{\cot\theta_k \cdot g_{ji}^k + \cot\theta_l \cdot g_{ji}^l}_{\text{spinor Laplacian}} + \underbrace{n_i(-g_{ji}^k + g_{ji}^l)}_{\text{Gauss curvature}}\Big),
\end{aligned}
\tag{3.29}
$$

where for the second last equation we used the fact that $g_{ki}n_kg_{ik} = g_{li}n_lg_{il} = n_i$, and for the last equation we define the spin connection between vertices through the left face by

$$
g_{ji}^k := g_{ki}g_{jk},
$$

and the spin connection through the right face by

$$
g_{ji}^l := g_{li}g_{jl}.
$$

Observe that the first two terms of (3.29) can be interpreted as the spin connection Laplacian, since by taking $(\mathfrak{D}^* \circ \mathfrak{D}\phi, \phi)$ they contribute to the Dirichlet energy:

$$
E(\phi) := \sum_{ij}\cot\theta_k|\phi - g_{ji}^k\phi|^2 + \cot\theta_l|\phi - g_{ji}^l\phi|^2,
$$

which measures the difference between the spinor at one vertex and the spinor transported from its neighbourings. The second part of (3.29) stands for the Gauss curvature,

because it measures the difference between the spin parallel transports through two different paths.

Unfortunately, the Gauss curvature terms appear in the off-diagonal entries, which is not the case in smooth case.

4 Numerics and Applications

Sec. 4.1 and Sec. 4.2 are taken from a paper published by the author [YDT+18]. Sec. 4.3 is taken from a manuscript submitted by the author [YUIH].

Many applications in geometric analysis and processing have benefited from *spectral* methods, namely ones that utilize eigenvalues and eigenfunctions of linear operators defined on the underlying geometric object – typically a polygonal mesh. Applications include mesh segmentation, feature extraction, mappings and correspondence, smoothing, deformation, remeshing, and various others [LZ10]. Various properties of spectral methods have made them popular in geometry processing: the frequency domain, namely the space spanned by the operator eigenfunctions, typically enables multi-resolution shape analysis via independent processing of the various "frequencies", and allows for a level of mesh independence. It also provides a canonical space on which to map shapes so as to jointly perform operations on multiple shapes simultaneously.

Among the operators used for spectral analysis, the Laplace-Beltrami operator is the most ubiquitous. The Laplace-Beltrami operator acts on the space of scalar functions defined on the shape, is elliptic and self-adjoint, and thus its eigenfunctions provide a basis on which it is possible to express any real-valued signal defined on the mesh. One of the appealing properties of the Laplacian operator is that it is *intrinsic*, namely it is invariant to isometric deformation of the underlying geometry, including rigid transformations. Hence, any analysis, filtering, or processing done using its eigenfunctions will similarly be independent of such transformations.

However, recent works in geometry processing [LJC17; CSBC+17] have pointed out that often, capturing extrinsic information can be useful for some applications; for example, in order to distinguish between dissimilar shapes that are however intrinsically similar, e.g. a cylinder and a plane, or different poses of the same shape. To address this issue, Liu et al.[LJC17] propose to use operators inspired by the quaternionic *Dirac operator*, which acts on the space of *quaternions* defined on a mesh. The extrinsic version of the Dirac operator was introduced in the geometry processing literature in the context of conformal deformations by Crane et al.[CPS11]. Liu et al. introduced a parameterizable family of operators obtained by interpolating between a fully intrinsic and a fully extrinsic operator via a user-defined parameter. These two operators correspond to the fully intrinsic and the fully extrinsic part extracted from the square of the extrinsic Dirac operator of [CPS11]; the fully intrinsic operator is in fact a quaternionic Laplacian operator on the mesh. By interpolating, Liu et al. enable a trade-off between the intrinsic vs. extrinsic nature of these two operators.

The classical Dirac operator has a well-known relation to pointwise conformal transformations that are *integrable*, namely map closed surfaces to closed surfaces. This property
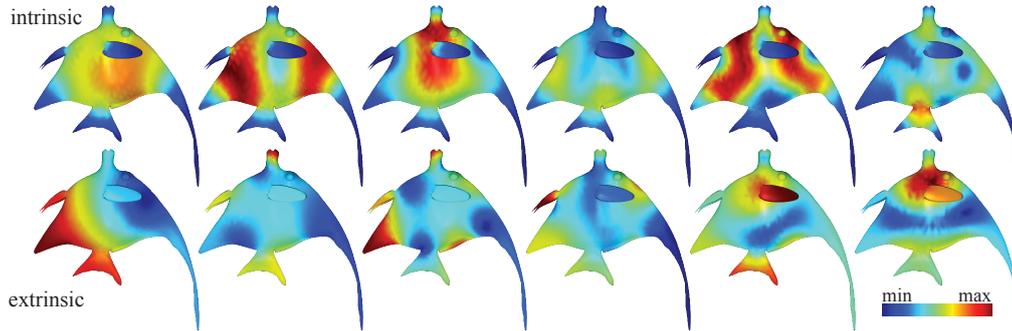
Figure 4.1: The eigenfunction magnitudes of the intrinsic and extrinsic Dirac operators, discretized in a unified way using our framework. The two operators capture different geometric information on meshes, highlighting different types of distortion and bending / stretching. Intrinsic Dirac eigenfunctions attain low values in regions of high Gaussian curvature – extrinsic eigenfunctions manifest maxima in curvy regions.

has also been exploited in the discrete case [CPS13] to design conformal deformations on meshes. This property no longer holds for the square of the Dirac operator in [LJC17].

We analyze various properties of our operator to gain insight into its accuracy, and showcase frequently improved numerics over previous operators. By studying its eigenfunctions and eigenvalues, we provide initial theoretical results on properties pertaining to the eigenfunctions that remain invariant under rigid or general isometric transformations. These considerations are useful when using the operator for applications. In this paper, we apply our operator for curvature painting (Fig. 4.2), canonical shape construction (Fig. 4.10 and 4.8), fast surface fairing (Fig. 4.9), shape correspondence (Fig. 4.13) and close-to-conformal shape filtering (Fig. 4.14). As future work, further applications can be considered, among which feature extraction, object classification, shape correspondence, deformation transfer and distance calculations. We hope that the availability of an intrinsic and extrinsic quaternionic operator discretized within the same framework will inspire further such applications using spectral methods.

Spectral Mesh Processing Frequency-based ideas have been widely used for mesh analysis and processing in a broad range of applications - see [LZ10] and papers therein for a recent survey. The majority of these works utilize the spectrum (eigenfunctions / eigenvalues) of the Laplace-Beltrami operator on discrete manifolds [DRW10] – mostly triangular meshes, although extensions to polygonal cases exist [HKA15]. Applications include, among others, mesh smoothing, compression, shape segmentation, matching, and parameterization. More recently, the spectral approach was also involved in on-surface distance approximation [LRF10], feature extraction [SOG09; Rus10; ASC11], mesh editing [HSTP11], remeshing [LHJ+14], inter-surface functional-based mappings [OBCS+12] and on-surface tangent field design and processing [ABCCO13; ACBCO17; AOCBC15]. The Laplace-Beltrami operator acts on functions defined on a manifold.
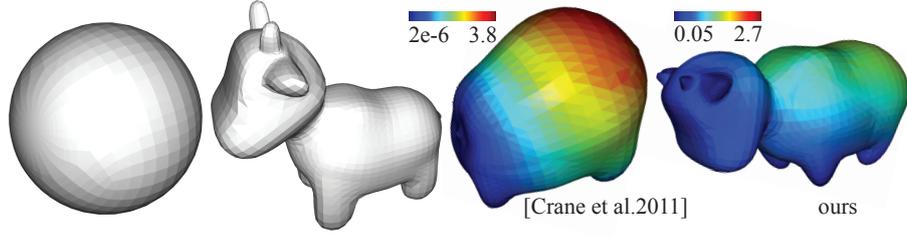
Figure 4.2: Recovering a cow from a sphere using the mean curvature half-density to design a spin transformation.

Instead, our work uses a discrete version of the *Dirac operator*, acting on quaternionic functions on manifolds. In the continuous case, the square of the Dirac operator closely relates to the Laplace-Beltrami.

4.1 Numerical Method

4.1.1 Modified Discrete intrinsic Dirac operator

In practice, we use the modified discrete intrinsic Dirac operator by

$$(D\phi)_i = \frac{1}{2} \sum_j E_{ij} \cos \frac{\theta_{ij}}{2} \phi_j \tag{4.1}$$

Note the presence of an additional cosine factor. Without it, the operator would be *covariant* under transformations of the form

$$E_{ij} \mapsto \bar{g}_i \cdot E_{ij} \cdot g_j$$
$$n_i \mapsto g_i^{-1} \cdot n_i \cdot g_i$$

with $|g_i| = 1$. Such a transformation preserves the norm of the hyperedges $|E_{ij}|$; it does not, however, preserve edge lengths. Note also that the norm of the quaternion $E_{ij} \cos \frac{\phi_{ij}}{2}$ is the same as the edge length $|e_{ij}|$. Hence we introduce the cosine factor in this formula as a correction factor in the discrete case, to ensure the covariance under *edge-length preserving* deformations; we found that this produces much better results in practice. Similarly to the extrinsic case, we prove the equivalent of Eq.(2.3), regarding the relation between the intrinsic Dirac operator and conformal immersions, which now holds in the discrete case.

Theorem 4.1. *Any $\phi \in \Omega$ satisfying the intrinsic Dirac equation*

$$D\phi = \rho\phi \tag{4.2}$$
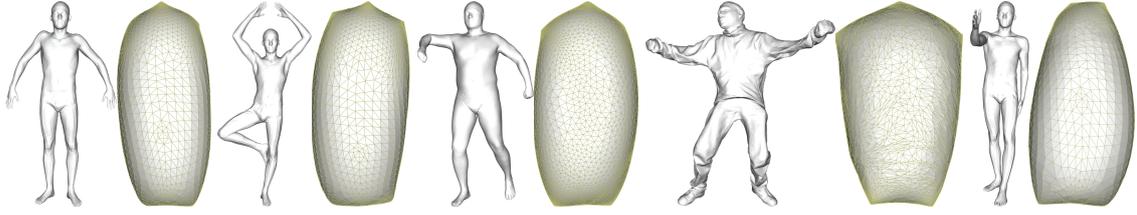
Figure 4.3: The first Dirac humans. We compute the eigenfunction of our intrinsic Dirac operator corresponding to its smallest eigenvalue, and apply it as a spin transformation to deform the input human mesh. Note the similar appearance of the resulting surfaces.

where $\rho : F \to \mathbb{R}$ is a face-based real valued function, gives another immersed edge-constraint net by the spin transformation:

$$E_{ij} \mapsto \qquad \cos \frac{\theta_{ij}}{2} \overline{\phi}_i \cdot E_{ij} \cdot \phi_j := \tilde{E}_{ij} \qquad (4.3)$$

$$n_i \mapsto \qquad \phi_i^{-1} \cdot n_i \cdot \phi_i := \tilde{n}_i \qquad (4.4)$$

The integrated mean curvature of the new surface is given by the discrete analogue of Eq. (2.4), namely

$$\tilde{\mathbf{H}}_i = \rho_i |\phi_i|^2 \qquad (4.5)$$

The following theorem (proven in the appendix) shows that this discrete intrinsic Dirac operator is indeed intrinsic.

Theorem 4.2. *The intrinsic Dirac operator in Eq. (4.1) is covariant under an isometric deformation.*

In fact it is possible to construct the intrinsic Dirac operator using only the metric (edge lengths) and some extra structure called the spinor connection, i.e. without knowing the vertex positions. Since all of the applications in this paper are based on immersed surfaces, we will not present the details on this process but refer the readers to [HY18]. Remarkably, in parallel to our work, a closely related spinor connection structure has been developed by Chern et al. [CKPS18] for the purposes of isometric shape embeddings.

4.1.2 Numerical methods

In this section we outline common numerical problems involving our Dirac operators that frequently appear in applications and experiments in the remainder of the paper.

Spin transformations. Frequently, we need to prescribe a curvature potential $\rho : F \to \mathbb{R}$ on a given mesh and compute the spin transformation (per-face quaternion-valued function ϕ) that will deform the mesh so it obtains the prescribed mean curvature.

For this we need to solve the Dirac equation Eq. (4.2) of the form $D\phi = \rho\phi$ with the prescribed ρ. Note, however, that the existence of the solution ϕ is not always guaranteed for all ρ. In order to solve for the transformation ϕ, we follow the same process as in [CPS11, Section 4] , namely we solve the following eigenvalue problem for an eigenvalue λ with small magnitude.

$$(D - \rho)\phi = \lambda\phi \tag{4.6}$$

Spectrum Calculations. In another common scenario, we wish to compute the eigenvalues and the eigenfunctions of the (intrinsic or extrinsic) Dirac operator

$$D\phi_i = \lambda_i\phi_i \tag{4.7}$$

where $\cdots \leq \lambda_{-1} \leq 0 \leq \lambda_1 \leq \cdots$. The recovered ϕ's may then be used for analysis, or as a spin transformation to deform a surface for the relevant application.

Numerical experiments show that the computed ϕ_i's are smooth if the target ρ in Eq. (4.6) has a small magnitude or only few eigenfunctions are required in Eq. (4.7). As the magnitude of ρ increases or more eigenfunctions are needed, numerical instabilities with the face-based Dirac operator may create non-smooth solutions. In order to improve the numerical stability, we propose the following approaches in practice.

Averaging the function on vertices In these cases, we use an additional face-to-vertex averaging matrix A (the $4|F| \times 4|V|$ incidence matrix) and solve the generalized eigenvalue problem

$$A^T \cdot \hat{D} \cdot M^{-1} \cdot \hat{D} \cdot A\phi = \lambda A^T \cdot \hat{D} \cdot A\phi \tag{4.8}$$

Here, M is the diagonal matrix with the face areas as entries, and $\hat{D} = D - P$, where P is a $4|F| \times 4|F|$ diagonal matrix containing the curvature information ρ as in (4.5) . If only eigenvalues of small magnitude are needed, we solve the simpler singular problem

$$A^T \cdot \hat{D} \cdot \hat{D} \cdot A\phi = \lambda A^T M \cdot A\phi. \tag{4.9}$$

Note that both these equations still only contain symmetric matrices on both sides, implying that the eigenvalues and eigenfunctions are always real-valued. After the ϕ's are computed, we use the transpose of the averaging matrix to compute a per-face quaternion function, which we can use for spin transformations if needed.

Regularization We observe that, even though this face-based Dirac operator gives the exact solution, it is not numerically stable, because its solution space is often too large (technically, some solutions that give the edge-constraint normals far from the actual face normal will result in unwanted transformations). On the other hand, while the vertex-based operators in [CPS11; YDT+18] works well in many cases, they are not able to faithfully recover the high curvature regions on the surface, because their solution spaces are too limited. To have a balance between these two approaches we propose the following regularized energy based on the face-based operator:

$$E_D(t) = \hat{D}^T(t) \cdot \hat{D}(t) + cR,$$

where c is a positive coefficient and R is the $4|F| \times 4|F|$ regularization matrix such that

$$R = \sum_{ij} |e_{ij}^*|(\phi_i - \phi_j)^2,$$

where the sum runs over all adjacent faces i and j. Note that the weights with the dual edge length are used in [CKPS18]. To have a finer control of the regularizer, one can decompose R into four components and set different weights as in [CKPS18], but we did not see that this will make any obvious difference in our setting. Empirically the coefficient c is set to be $0.001 \max_{ij}|e_{ij}|$.

By the min-max principal, solving the generalized eigenvalue problem

$$E_D(t)\phi = \lambda M\phi,$$

where M is the mass matrix, is actually equivalent to minimizing the energy

$$\min E_D, \text{ s.t. } |\phi| = 1,$$

with the metric defined by $|\phi|^2 := \phi^T \cdot M \cdot \phi$.

Finally, the edges are constructed by the spin transformation

$$e_{ij} \mapsto \text{Im}(\overline{\phi_i} \cdot E_{ij} \cdot \phi_j),$$

the position of vertices v_i are recovered by solving the Poisson equation (see Sec. 3 of [SA07] or Sec. 5.6 of [CPS11]). In the attached videos, we prescribe the mean curvature half-density of two shapes (red) on their conformal parameterization (blue) and it shows deformation from the sphere to the original shapes.

Area calibration   Even though the Dirac operator with regularization term improves the accuracy of reconstruction, we observe that some area distortion is still visible, especially at the region with really high curvature. To overcome this problem, we would like to make the reconstruction algorithm be aware of the area scaling factor. In the work by Chern et al. [CPS15] one prescribes a volumetric scaling factor e^u and obtains the close-to-conformal volumetric deformation by minimizing an energy E_u depending on u. While the energy E_u in [CPS15] is specifically designed for 3D volumetric meshes, an analogy for 2D surfaces still holds in smooth case:

Theorem 4.3. *Let $f : M \to \mathbb{R}^3 \subset \mathbb{H}$ be an isometric immersion and $h : M \to \mathbb{R}$ be any function. The quaternion gradient is defined by*

$$\text{grad}_f h = df(\text{grad}\, h).$$

The spin transformation $d\tilde{f} := \overline{\phi} \cdot df \cdot \phi$ with $D_f\phi = 0$ is closing if

$$d\phi\phi^{-1} = -\frac{1}{2}Gdf, \tag{4.10}$$

where $G := \text{grad}_f u$ is the gradient of the logarithmic factor $e^u := |\phi|^2$.

Proof. See App. 7. ◻

Therefore, given a spin transformation induced from ϕ with the area factor $u = \log|\phi|$, the quaternion-valued 1-form

$$\omega := d\phi + \frac{1}{2} G df \phi$$

vanishes. Practically we minimize the energy $|\omega|^2$, where the metric for quaternion-valued 1-form is defined by

$$\langle \omega, \eta \rangle := \int_M \overline{\omega} \wedge (*\eta). \tag{4.11}$$

In discrete case, minimizing the energy E_u again amounts to solving a generalized eigenvalue problem for a $4|V| \times 4|V|$ matrix (see Sec. 8.3). To avoid introducing the scaling factor as one more function in our representation and subsequently increasing the data size, we first apply the isotropic remeshing with approximate equalized face area ([FAKG10]) for all shapes in the dataset. In this case the logarithmic factor u should be set to $u_i = \log(1/\sqrt{|\tilde{A}_i|})$, where $\tilde{A}_i$ is the face area of the conformal parameterization $\Phi(M)$.

Deforming a surface with a spin transformation Given a spin transformation, we can compute the new surface by first applying the transformation to the hyperedges. We then recover the coordinates of the vertices by solving a linear system, similarly to [CPS11, Section 5.6]. Specifically, we collect the imaginary parts of the transformed hyperedges as the coordinates of the edge vectors, and then solve the underdetermined sparse linear system

$$v_j - v_i = e_{ij}$$

for all the edges e_{ij}, where v_i, v_j are the vertices incident to e_{ij}. To solve this system we need to fix the coordinates of any one vertex. In practise, since numerical issues occurring during the Dirac eigenvalue problem might prevent this system from having an exact solution, we solve it in a least squares fashion, which we found yields satisfying results.

4.2 Evaluation and Analysis

In this section we validate our discrete operators against known properties of the continuous operators and evaluate their behaviour.

4.2.1 Spectrum verification

Eigenvalues on spheres. For the round unit sphere, the intrinsic Dirac operator eigenvalues are integers with multiplicity equal to their absolute value plus one, namely

$$\cdots, -3, -3, -3, -2, -2, -1, 1, 2, 2, 3, 3, 3, \cdots$$
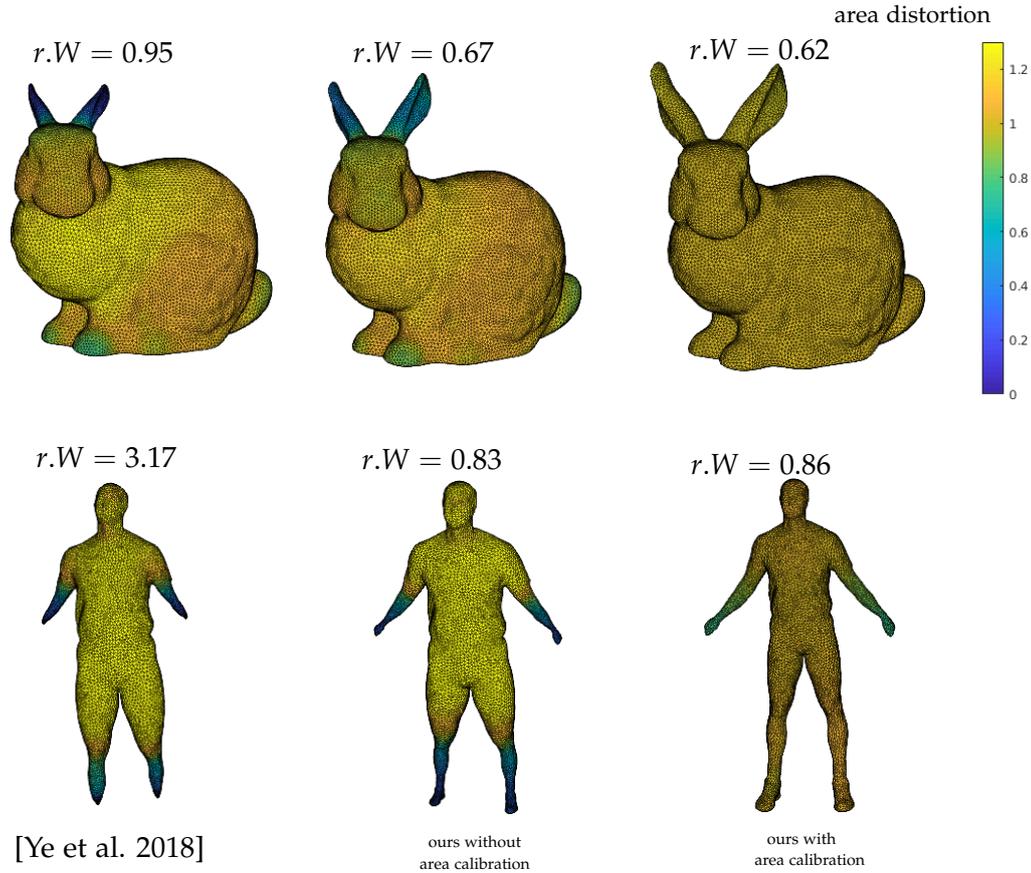
Figure 4.4: Reconstruction of shapes from their conformal parameterization. While the Willmore energy is defined by $W = \sum_i h_i^2$, we define the relative Willmore energy between two meshes with identical connectivity by $r.W := \sum_i ((h_1)_i - (h_2)_i)^2$, which measures how close the mean curvature half-density of two meshes are. This experiment shows that our method substantially improves the accuracy of curvature reconstruction. Furthermore, the area distortion, which usually appears at the regions with high curvature, gets much reduced by the area calibration. Note that, in contrast to [CPS15], we only encode the expected scaling factor in the energy $|\omega|^2$ and the factual scaling factor $|\phi|^4$ is determined by the optimizer.

Note that the intrinsic Dirac operator cannot have a zero eigenvalue on a sphere, since the corresponding spin transformation from Eq.(2.3) would result in a minimal sphere-topology surface with zero minimal curvature, which is impossible. In contrast, the extrinsic Dirac does have a zero eigenvalue, corresponding to eigenfunctions that are rigid transformations. We verify our discretization by performing eigenanalysis, using Eq. (4.8), on our intrinsic Dirac operator on various meshes of the unit sphere. We use various triangulations of different resolution, element quality, regularity and meshing types. Fig. 4.5 shows that our discrete operator gives exactly the same spectra regardless
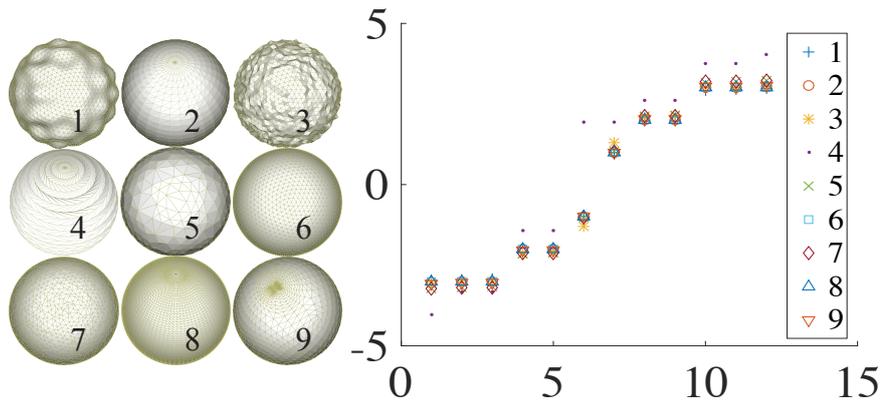
Figure 4.5: The eigenvalues of our discrete intrinsic Dirac operator, computed on various meshings of spheres, agree with their known values from the continuous case. The only exception is the fourth mesh with really bad normal approximation. Additionally, they are relatively robust against noise in the meshing or the geometry.

of meshing.

Eigenvalues on tori. The Dirac operator can be used to distinguish between surfaces that are isometric but have different *spin structure*. The spin structure of Riemannian manifolds is an abstract notion. However, in the case where an immersion $\mathbb{R}^3$ is available, it is more intuitive to understand: two surfaces have the same spin structure iff they are related by a regular homotopy [Pin85]. All simply-connected surfaces have a single spin structure; in the case of tori, 4 different spin structures exist. In Fig. 4.6 we construct two tori that are approximately isometric but differ in spin structure. To this end, we create the tori with the same radii in such a way that the big radius is much larger than the small one. In this way, both tori look locally like a flat cylinder, and are therefore approximately isometric. As shown in the figure, the two tori are indistinguishable in terms of the eigenvalues of their Laplace-Beltrami operators, due to the approximate isometry. On the other hand, they can be clearly distinguished based on the Dirac operator eigenvalues.

Invariance to isometries. In order to verify that our discrete intrinsic Dirac operator is indeed invariant under isometric deformations, we construct a simple experiment based on a planar mesh (Fig. 4.7). By displacing the vertices of the mesh while keeping the edge length unchanged, we obtain a non-planar mesh that is an isometric deformation of the original plane. As can be seen from the figure, both the eigenvalues and the patterns of the norm of eigenfunctions are unaffected by the deformation.
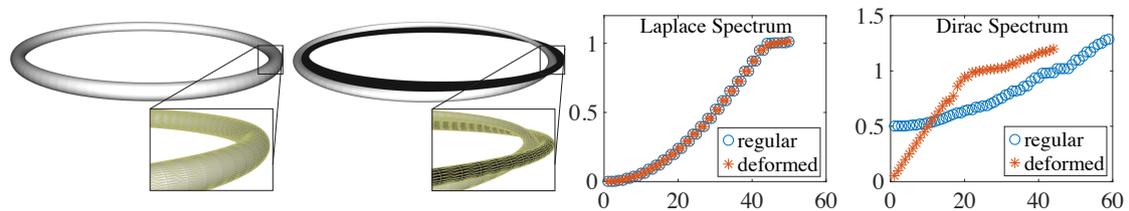
Figure 4.6: Laplace-Beltrami and Dirac spectra of two tori, constructed to be approximately isometric but different in terms of their spin structure. The Laplace-Beltrami eigenvalues for the two tori are almost identical, given the approximate isometry. However, the Dirac eigenvalues help distinguish between the two.
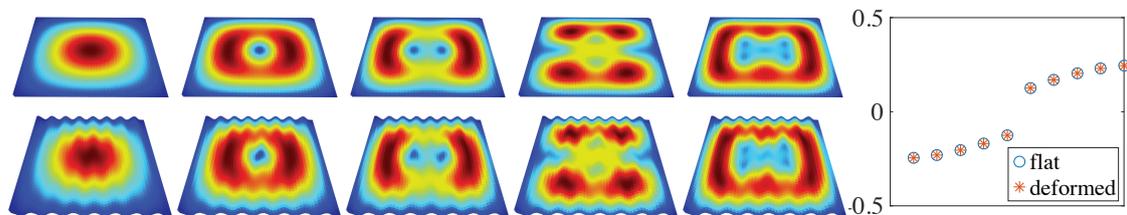
Figure 4.7: Spectra of the discrete intrinsic Dirac operator for two isometric surfaces. The bottom mesh is the result of translating the vertices of the top planar mesh while keeping the edge lengths unchanged. The colors show the magnitude of the eigenfunctions corresponding to the first five eigenvalues for the two meshes. On the right, the plot shows the eigenvalues of the operator for the two meshes, which are identical.

4.2.2 Recovering a conformal immersion

As noted in [KPP98] and [CPS13], apart from some extreme cases, the mean curvature half-density actually determines the conformal immersion that describes the embedding of a surface in $\mathbb{R}^3$. We validate this statement for the case of our discrete operators in Fig. 4.2, by "extruding" a target mesh from a mesh of a unit sphere. In order to achieve this, we first compute the mean curvature half-density of the target mesh using Eq. (3.11), namely $\rho_i^{\text{target}} = \mathbf{H}_i^{\text{target}} / \sqrt{A_i^{\text{target}}}$, where A_i^{target} is the face area on the target mesh. Since this is an integrated quantity, we adapt it for the sphere mesh by $\rho_i = \rho_i^{\text{target}} \sqrt{A_i^{\text{sphere}}}$, and prescribe it as ρ in Eq. (4.2). We then compute the resulting spin transformation using equation Eq. (4.9) and apply it onto the spherical mesh. For comparison, in Fig. 4.2 we also show the result using the discrete extrinsic Dirac operator of Crane et al. [CPS11]. Note that since the operator in [CPS11] is extrinsic, we need to use the *difference* between the target mean curvature half-density and the mean curvature half-density of the sphere as the ρ in Eq. (2.13). Our discretization seems to be recovering the target surface more faithfully.

4.2.3 Dirac immersions
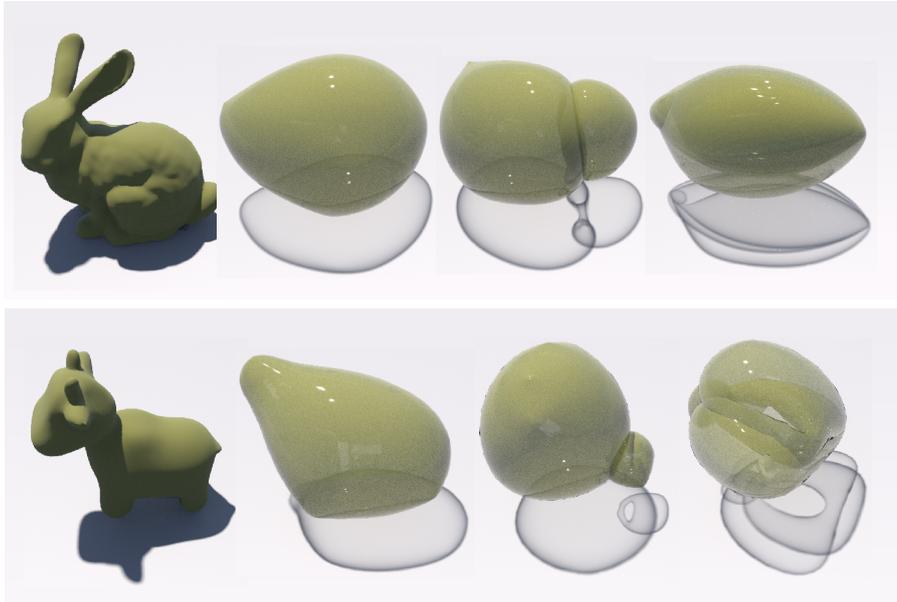
Figure 4.8: Dirac animals. We compute the eigenfunctions of our intrinsic Dirac operator corresponding to its smallest eigenvalues, and apply them as a spin transformation to deform an input mesh.

By *Dirac immersions* we refer to the conformal immersions that correspond to the intrinsic Dirac eigenfunctions. In [CPS11] such immersions (the *Dirac spheres*) were
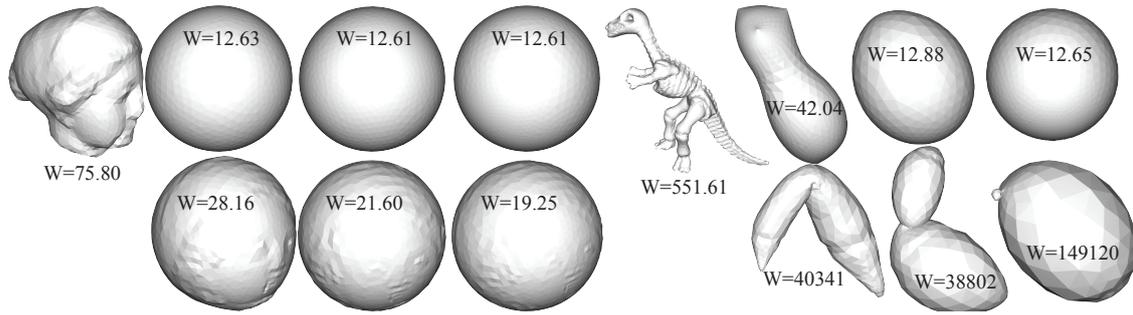
Figure 4.9: For each mesh, the top row shows the results of iteratively applying the first Dirac eigenfunction as a spin transformation. This produces a progressively smoother shape, as reflected by the reduced Willmore energy values. The bottom row shows, for the same meshes, the application of conformal Willmore flow with a big time step $\frac{1}{2}$. Note that while both techniques work for the first shape, which deforms fast towards the round sphere, the conformal Willmore flow with big time step is quite unstable for the second shape.

computed and visualized for the unit round sphere, for which closed-form expressions are known [Ric97] and find applications in physics. Note that the operator in [CPS11] is extrinsic and cannot be used to compute Dirac immersion for general surfaces – however, since the round sphere is of constant mean curvature $H = 1$, and using Eq. (2.14), it can be seen that the extrinsic and intrinsic Dirac operator only differ by a constant 1 in this case. Therefore, for the special case of a unit round sphere, one can extract the intrinsic Dirac eigenfunctions from the extrinsic Dirac eigenfunctions. Our purely intrinsic Dirac operator can be computed for any mesh – solving the eigenvalue problem (4.8) then allows us to directly visualize Dirac immersions of various degrees for arbitrary simply-connected surfaces (Fig. 4.8 and 4.3). Notably, the first Dirac immersions of the various human meshes are of similar shape, which might be of interest in applications.

Fast surface fairing. In the continuous case, the following theorem can be proven (see appendix):

Theorem 4.4. *Let $f : M \to \mathbb{R}^3$ be the simply-connected immersed surface with Willmore energy W, and f_1 the immersion corresponding the Dirac eigenfunction with smallest eigenvalue. Then we have $W_1 \leq W$, where W_1 is the Willmore energy of f_1.*

Namely, the first Dirac immersion is always of lower Willmore energy than the initial surface, and thus smoother. This is also reflected in our discrete results. We can use this fact to rapidly transition from the surface to the round sphere in only a few steps, as shown in Fig. 4.9. We do this by repeatedly applying the spin transformation (Eq. (4.8)) corresponding to the first eigenfunction and recomputing the operator at each iteration.

Polygonal meshes. While our work mainly targets triangular meshes, we show some preliminary support for polygonal meshes – Fig. 4.10 shows simple curvature painting and a Dirac immersion for discrete spheres discretized as quadrilateral meshes.
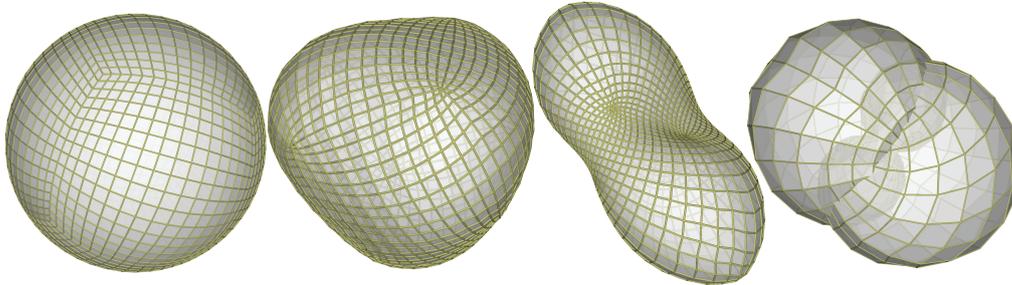

Figure 4.10: Our intrinsic operator applied on a non-triangular mesh of a sphere (left). The middle two images show the spin transformations induces by setting ρ to equal the y euclidean coordinate at each point, and the product xyz respectively. The rightmost image shows the first Dirac immersion.

Dirac immersions and isometries. The Dirac immersions of two isometric surfaces only differ by an Euclidean motion, as expected by the invariance of the eigenfunctions, shown in the proof of Theorem 4.2. We verify this statement for the two isometric surfaces of Fig. 4.7 : Fig. 4.11 shows the first few immersions for the plane (top) and its isometric deformation (bottom). The colors show the magnitude of the first five eigenfunctions for the two meshes, indicating the invariance of the immersions to isometries.
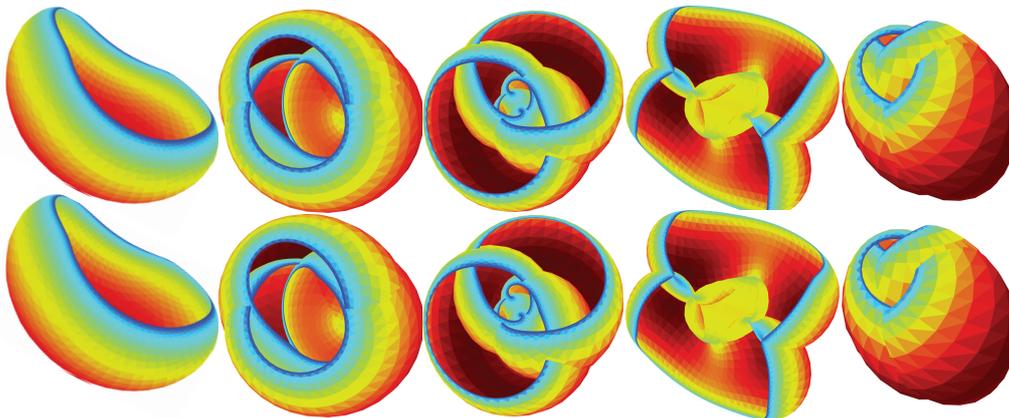

Figure 4.11: The Dirac eigenfunctions of the two isometric surfaces of Fig. 4.7 applied as spin transformations onto their corresponding surfaces. Note that they are identical up to a rigid transformation. In pseudocolor, the magnitude of the corresponding eigenfunctions.

Comparison to Crane et al. [CPS13], [CPS11]. One can apply the formula $D = D_f + H$ on the discretization in [CPS11] and obtain another discrete intrinsic Dirac operator. Indeed, in some circumstances this operator gives similar results for regular meshes. However, due to the lack of invariant properties (see Theorem 4.2), it is more sensitive to the meshing than ours, see Fig. 4.16.

Our fast surface fairing looks very similar to Crane et al.'s conformal Willmore flow [CPS13] with time step $\frac{1}{2}$. However, for Willmore flow the potential needs to be projected onto the feasible space in order to ensure the existence of a solution [CPS13, Section 6.2]. With a relatively big time step this projection might not be effective and produce unexpected shapes, especially for complicated meshes. In contrast, it is guaranteed by our Theorem 4.4 that the first Dirac immersion will always reduce the Willmore energy. A drawback of our fast fairing approach is that it only works for simply-connected surfaces. The fine-grained, per-iteration control of the conformal Willmore flow in [CPS13], necessary to ensure global integrability, allows instead for practical applicability of this flow also to surfaces of higher genus.

4.2.4 Invariant properties.

$$i\text{-comp.} \qquad j\text{-comp.} \qquad k\text{-comp.} \qquad \sum(\text{Imag}(\cdot)) \quad \Big| \quad \text{norm}(\cdot) \qquad \cos\frac{\theta}{2} \qquad \langle u, n \rangle$$
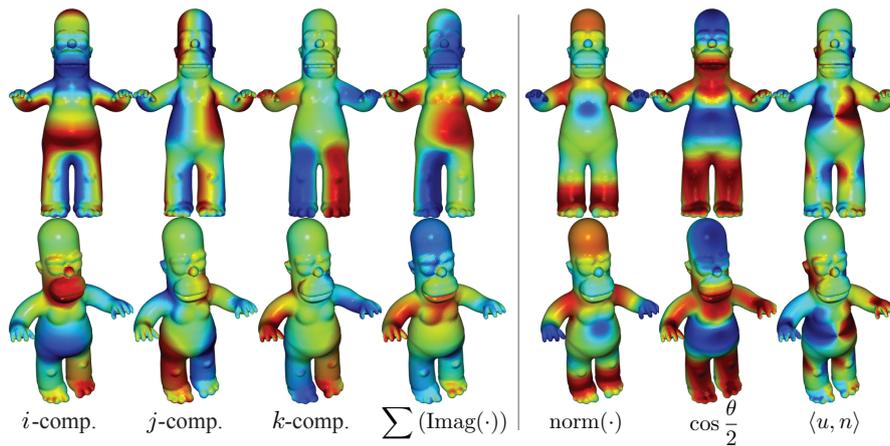
Figure 4.12: Various quantities derived from the intrinsic Dirac eigenfunctions that vary (left) or remain invariant (right) under rigid transformations of an input mesh.

Inspired by the widespread usage of the eigenfunctions of the Laplacian operator in shape analysis, a natural question might be whether it is possible to similarly extract geometric information from the Dirac eigenfunctions as well. For example, one might consider using the intrinsic Dirac operator for spectral feature extraction, similarly to HKS/WKS [SOG09; ASC11]. Additionally, as argued in [LJC17], some applications might benefit from intrinsicness of the respective operator, while others might require extrinsicness, e.g. when different kinds of symmetries need to be extracted. So a natural question that arises is, which geometric quantities extracted from the intrinsic Dirac

eigenfunctions are *intrinsic*, namely invariant under general isometric deformations, and which are *extrinsic* – which in this context means invariant under only rigid transformations. Such considerations affect the quality of the extracted descriptors and determine their potential for applications.

For example, the individual components of the quaternions corresponding to the intrinsic Dirac eigenfunctions (when viewed as $\mathbb{R}^4$) do not produce intrinsic quantities. Fig. 4.12 shows that, even under the canonicalization process of [LJC17], the individual quaternion components are still not invariant even under rigid transformation. In the appendix, we show that the per-face norm $|\phi_i|$ of the intrinsic Dirac eigenfunctions is an intrinsic quantity. Furthermore, the Dirac eigenfunctions contain two more extrinsic quantities, which are not available from the Laplacian eigenfunctions, as per the following theorem (proven in the appendix). That shows that the Dirac eigenfunctions reflect the extrinsic information, even though one uses the intrinsic Dirac operator.

Theorem 4.5. *Let ϕ be the the eigenfunctions of extrinsic or intrinsic Dirac operator. At each face, we write $\phi_i = |\phi_i| \left(\cos \frac{\theta_i}{2} + \frac{\sin \theta_i}{2} u_i \right)$. The following two quantities are invariant under the rigid transformation:*

- *The rotation angle, or the normalized real part, $\cos \frac{\theta_i}{2}$.*

- *The angle between the rotation axis and the face normal $\langle u_i, n_i \rangle$.*

Following the idea in [LJC17], we construct the shape descriptors, in the framework as the wave kernel signature [ASC11], based on the norm of Dirac eigenfunctions and two quantities above. The norm contributes to detecting the intrinsic geometric features, while the other two quantities are responsible for disambiguating the features, which are intrinsically similar but extrinsically different. The Dirac kernel signature in [LJC17] succeeds in detecting the extrinsically different features, however, it suffers from not being invariant under rigid transformations, because it contains the quantity, the sum over four quaternion components of the eigenfunctions, which varies under Euclidean motions. Since all of the quantities we used are invariant under rigid transformations, Fig. 4.13 shows that our shape descriptor fixes this problem.

Although our intrinsic Dirac operator is invariant under edge-length-preserving deformations, it is, in general, not invariant to

different tessellations of the same surface. In the top inset figure, we consider two meshes inscribed in the unit sphere: one is a subdivided icosahedron (first from left), and the other is obtained by flipping the edges of the first (second from right). Note that the Dirac eigenvalues and Willmore energy of the first mesh are close to their predicted values; this is less so for the second mesh. Additionally, performing Willmore flow / fast fairing does not affect the first mesh (second from left); but doing the same on the second mesh decreases its Willmore energy (first from right). The change on the mesh is hardly noticeable though, implying that our operator remains stable for this second mesh.
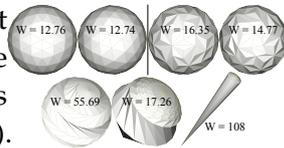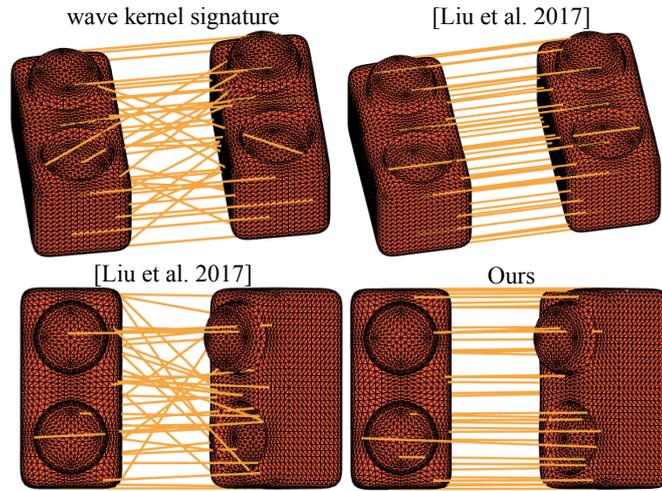
Figure 4.13: Point-to-point correspondence between isometrically identical surfaces with different meshes. The Laplace-Beltrami-based wave kernel signature fails to differentiate between the inward and outward bumps. The Dirac kernel signature based on [LJC17] captures these extrinsically different features, but yields poor matches when one of the objects is slightly rotated. The same descriptor based on our operator overcomes both problems.

However, if the input is a mesh, still inscribed in the unit sphere but with a significantly worse normal approximation (bottom inset), fast fairing can become unstable. In this case, it first reduces the Willmore energy (bottom-middle); later the result becomes unstable and its Willmore energy grows uncontrollably. Improving our intrinsic operator to deal with such meshes with extremely bad normal approximation remains an avenue for future work.

Comparison to Liu et al. In [CPS11], the relative Dirac operator is constructed by extracting the extrinsic part of the square of the extrinsic Dirac operator of [CPS11]. This modified operator, while being purely extrinsic, does not relate to spin transformations via Eq. (2.3) – neither does the intrinsic operator of [LJC17], which is essentially a quaternionic Laplacian. In that sense, our operators are more faithful to the continuous Dirac operator; Liu et al.'s operator cannot be used to compute spin transformations or immersions, for example. On the other hand, since the square of the [CPS11] operator also involves the Laplacian component, the relative Dirac operator can be thought of as the extrinsic Dirac operator without the impact of the Laplacian. Hence it is possibly "more extrinsic" than the operator of [CPS11] or ours, which might be interesting to consider in applications.

Our operators could possibly be applied to feature extraction using heat/wave kernel signatures (HKS/WKS), by using the norm of the Dirac eigenfunctions as opposed to Laplace-Beltrami. Our preliminary experiments in shape matching on state-of-the-art
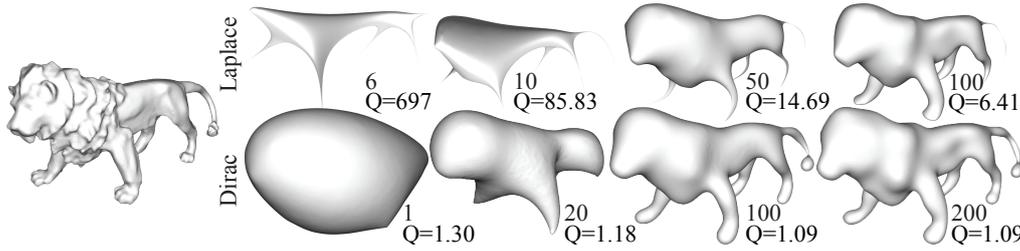
Figure 4.14: Top: the classical spectral decomposition of a mesh, using Laplace-Beltrami eigenfunctions. Bottom: close to conformal decomposition using intrinsic Dirac eigenfunctions; the first shape here is the first Dirac immersion. We plot the quasi-conformal distortion [SSGH01] induced by the low-pass filtering, which is much smaller using our method. Note that we generally use twice as many Dirac eigenfunctions (counted by quaternionic multiplicity) as Laplace eigenfunctions, since the intrinsic Dirac operator always has symmetric eigenvalues $\{\lambda_1, -\lambda_1, \lambda_2, -\lambda_2, \ldots\}$.

databases, however, do not show an obvious advantage in using our operator. We conjecture that the information contained in the *norm* of the Dirac eigenfunctions is equivalent to that of Laplace-Beltrami, and it is sufficient for these purely intrinsic shape matching experiments. Extract additional intrinsic scalar-valued information from the Dirac operator remains an interesting research topic.

4.2.5 Close-to-conformal spectral shape decomposition

In [VL08], the eigenfunctions of the Laplace-Beltrami operator were used to decompose a shape. This was done by projecting the vertex coordinate functions onto a truncated set of Dirac eigenfunctions of increasing eigenvalue magnitude; owing to the relation between the Laplacian and Fourier frequency analysis, using less eigenfunctions generally produces smoother versions of the input surface. In a similar fashion, we can now use the eigenfunctions of the Dirac operator, which will produce a close-to-conformal decomposition. To this end, we can represent the input shape as the identity quaternionic transformation with the constant value $\phi = (1, 1, \ldots, 1)$ on all faces and project this function onto a subset of the Dirac eigenfunctions (Eq. (4.8)). We then apply the reconstructed sum as a spin transformation on the original surface. The results are shown in Fig. 4.14. Note that the weighted sum of a truncated Dirac eigenfunction basis does not produce an integrable spin transformation in general – while we did not find this to be a problem in practise, an additional projection step could be applied to ensure integrability if necessary.

4.2.6 Convergence under different meshings.

Following [CPS11], we use the quasi-conformal error [SSGH01] to measure the quality of our conformal mappings. At each face, we compute the ratio Q_i between the largest
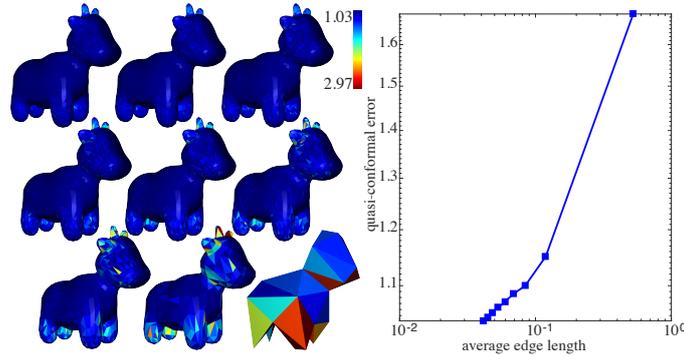
Figure 4.15: On the left, the spatial distribution of the quasi-conformal errors between the original shape and its first Dirac immersion for different mesh resolutions. On the right, the average quasi-conformal error for different mesh resolutions, plotted against the average edge length.

and smallest singular value of the differential df. The ideal value is $Q_i = 1$, implying that the map is uniform scaling and rotation. We compute the total error Q for a surface by averaging the quasi-conformal errors per face using area weighting. Fig. 4.15 shows that the quasi-conformal error between a shape and its first Dirac immersion goes to 1 as the average edge-length goes to zero. Additionally, Fig. 4.14 and Fig. 4.16 show that our method generally induces low quasi-conformal error.

4.3 Deep Curvature Learning

While the convolutional neural network has achieved significant success in 2D image processing, more and more attention has recently been drawn to applying the technique to the domain of 3D shapes. Unlike 2D images, which are typically represented by a multidimensional tensor, the representation of 3D shapes is usually unstructured, hence the convolutional neural network is not directly applicable. Thus the main challenge is how to create a suitable representation for 3D shapes which can take advantage of the state-of-art machine learning frameworks. Several such representations based on point clouds [FSG17; ADMG18; GFK+18], volumetric data [TDB17; WSLT18; WLG+17] and meshes [BHMK+18] have been proposed with different applications. However, all these representations are built on extrinsic data, e.g., position of points, vertices or voxels.

In this paper, we propose a 3D deep generative model based on mean curvature and metric, which in discrete case are expressed by two functions that are invariant under Euclidean motion. It has the following advantages against the models based on extrinsic representation:

Firstly, our model would preserve more detailed structure of shapes in case that the curvature plays a critical role, especially when the surface is highly folded and convoluted like the cortical surfaces in Fig. 4.17. The convolutional
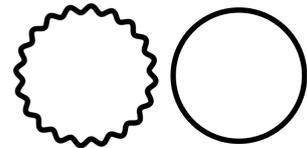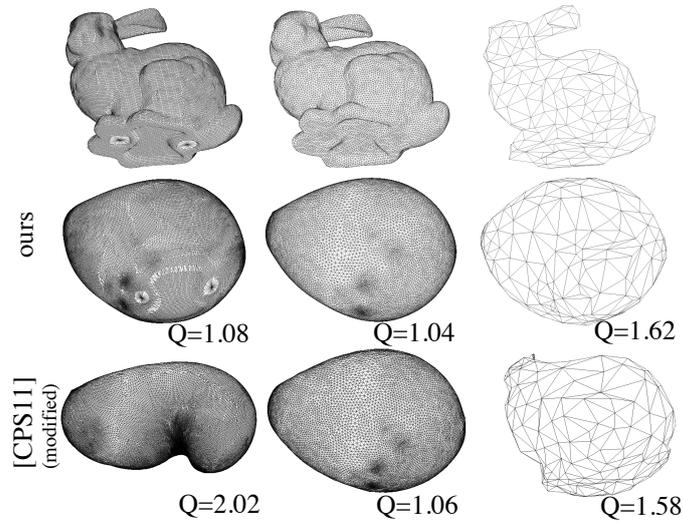
Figure 4.16: The Dirac bunnies with different meshings. Top row, left to right: the original bunny mesh, an isotropic remeshing and a highly simplified mesh. Middle row: the Dirac immersions for these meshes using our operator. The output shapes are of very similar appearance despite the very different input tesselations. Bottom row: the immersions produced via the modified version of the operator in [CPS11], i.e., $D = D_f + H$. In contrast to our result, these differ significantly in appearance for the different inputs.

neural network (CNN) is known to be good at capturing
not only the global features but also the local fine structure of data. Its effectiveness, however, relies on a proper distance function defined on the space of features. For example, the Euclidean distance between two points is usually used in extrinsic models. As the result, the bumpy circle (inset) will tend to be deformed through the neural network to the round circle, which is more regular and is close to the bumpy one under the measurement by Euclidean distance. In contrast, we adopt curvature representation and subsequently the distance between curvatures, by which two circles are clearly distinguishable, hence the small hills will be safely preserved. Secondly, our model is less affected by rigid transformation and uniform scaling. Thanks to the invariant quantities that constitute our representation and the CNN on sphere (see Sec. 4.3.3 for detailed discussion), we provides a simple and efficient way to deal with the dataset without alignment.

The input shapes for our model are required to be surfaces with consistent simply-connected topology, e.g., the disk-like surface or the spherical surface. We first map the input surface to a canonical domain such as a sphere, where mean curvature and vertex density are extracted and recorded as the input data for the neural network. For generative models like VAE, the output is a variant of the input so it has the same form as the input. To reconstruct the shape, we first create a triangulation of the canonical domain by randomly sampling the points with respect to the generated density function
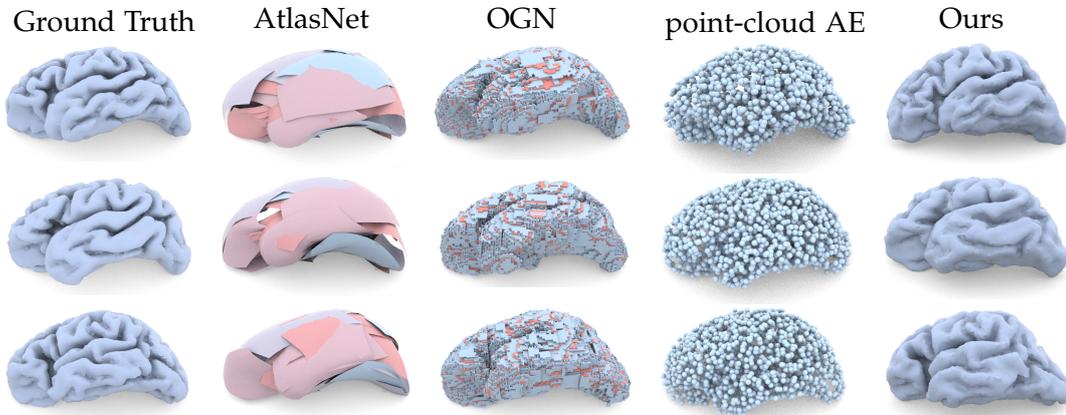
Figure 4.17: Brain autoencoder. We build a curvature-to-curvature autoencoder and compare to the models based on point clouds, the AtlasNet [GFK+18] (point clouds to surface) and the point-cloud AE [ADMG18] (point clouds to point clouds), and the voxel-based model OGN [TDB17] (IDs to voxels). All the neural networks, except for OGN, are trained on 1400 cortical surfaces and validated on 200 surfaces. Three of the predicted surfaces from the validation data are shown above. Although all the models are able to restore the brain structure in large scale, only our model preserves the local fine structure. For more details see Sec.4.3.3 and Fig. 8.1.
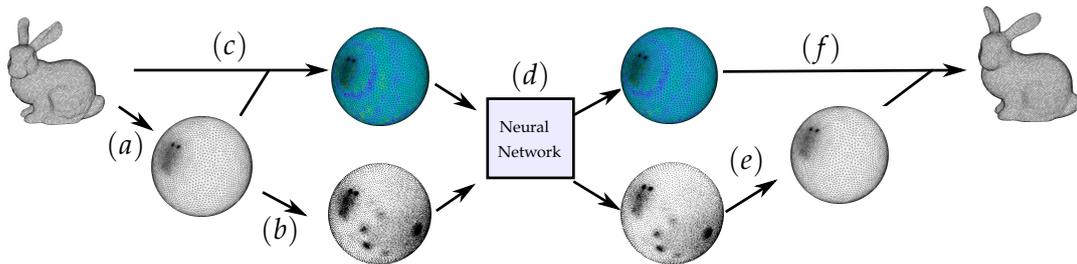
Figure 4.18: The pipeline of our model for generating variant shapes: (a) the conformal parameterization, (b) the density function extraction (Sec. 4.3.2), (c) the mean curvature half-density extraction (Sec. 4.3.2), (d) learning and generating (Sec. 4.3.2), (e) the isotropic remeshing (Sec. 4.3.1), (f) solving the Dirac equation and applying the spin transformation.
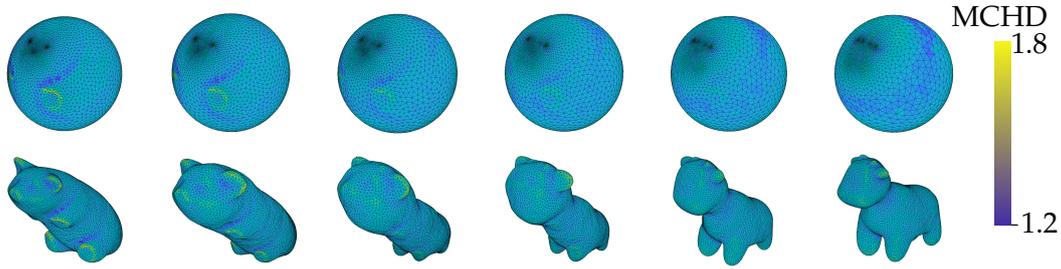
Figure 4.19: The spherical conformal paratmeterizations of two animals are aligned by a Möbius transformation with three landmark points. Then, they are packed into tensors with dimension $320 \times 32 \times 32 \times 2$. This figure shows a linear interpolation between the curvature representation of two shapes and the resulting shape reconstruction from the curvature representation.

and applying the isotropic remeshing. Then, we deform the triangulation gradually towards the target shape with the prescribed mean curvature. The deformation (see the attached videos) is obtained by solving the Dirac equation based on an approach improved from the works by Crane et al. [CPS11] and Ye et al. [YDT+18].

A reconstruction algorithm with high accuracy is critical for generating plausible shapes. Recall that in [CPS13] and [YDT+18] the deformation between the domain and target shapes is given by the solution of the Dirac equation. We propose a modified equation with larger solution space and it results in the reconstruction comparatively closer to the target shape. Furthermore, previous methods fail to locally scale the shape correctly at regions with large curvature. In fact, it is hard to directly manipulate the local scaling with the Dirac equation. Therefore we design a new algorithm inspired by Chern et al. [CPS15] to calibrate the area scaling factor. This compensates for the shortcoming of the Dirac equation and significantly stabilize the reconstruction.

We evaluate our reconstruction algorithm on several shapes, showing that our method outperforms previous methods visually and quantitatively. In addition to some preliminary applications such as shape remeshing, interpolation and clustering, we demonstrate randomly generated shapes from various datasets and compare to other 3D generative models.

In summary, the contribution of this paper is 1) an improved algorithm for shape reconstruction from curvature with area calibration and 2) a 3D shape deep learning framework based on curvature.

4.3.1 Related Work

Which invariant quantities determine an immersed surface in $\mathbb{R}^3$?   It is well-known that an immersed surface in $\mathbb{R}^3$ is determined up to a Euclidean motion by its first and second fundamental forms. However, they are tensorial data whose representation

depends on a choice of coordinate. Hence, in order to consistently represent 3D shapes based on the two fundamental forms, an identical triangulation for all shapes, which is not always possible, is required.

Other options are point-wise shape descriptors such as the heat kernel signature [SOG09] and the wave kernel signature [ASC11]. Indeed, they have been employed in discriminative models for 3D shape classification and segmentation [BMM+15]. But they can hardly be used for generative models, because it is unclear whether these shape descriptors completely determine the shapes or how to reconstruct shapes from them.

The idea of this paper comes originally from Bonnet [Bon67]. In fact, except for some very special cases, an immersed surface is completely determined by conformal structure, regular homotopy class and mean curvature half-density, which is a scale-independent variant of the mean curvature [Kam98]. The exceptions, called the Bonnet immersions, includes minimal surfaces, constant mean curvature surfaces and Bonnet pairs. In our case the regular homotopy class is unnecessary, since we only consider the simply-connected surfaces which have only one unique regular homotopy class [Pin85]. In summery, generic simply-connected immersed surfaces are uniquely determined by the conformal structure and the mean curvature half-density.

Generative modeling and variational autoencoder The generative modeling is a specific area in machine learning which learns distribution of data and generate new instances from the distribution. Among various generative frameworks the generative adversarial network (GAN) [GPAM+14] and the variational autoencoder (VAE) [KW14b] have drawn the most attention. In this paper, we only implement our 3D shape representation with the latter one, though it is possible to build our model with GANs with slight modification.

The VAE has a "bottleneck" structure constructed by successively composing three networks: the encoder, the latent space and the decoder. Input data will be compressed in the encoder network and then be reparametrized in the latent space, where it is assumed to satisfy the unit Gauss distribution. Then the data will go through the decoder network and be reconstructed as the output. The loss function is the sum of the reconstruction loss, which measures how much the output deviates from the input data, and the KL-divergence which measures how much the distribution in the latent space is different from the unit Gauss distribution. After the model is well trained, we can feed random Gauss noises into the latent space and obtain randomly generated new instances as the output. For details see the original paper [KW14b] and the tutorial [Doe16].

3D shape generation Various representations of surfaces have been proposed for 3D shape generation, e.g., models based on volumetric representation [WZX+16; TDB17; SM17; WLG+17; WSLT18], or point clouds representation [FSG17; NW17; ADMG18]. These methods are particularly applicable for the dataset with inconsistent topology. However, without knowing the mesh structure it is hard to capture the fine structure of

certain highly complicated surfaces (see Fig. 4.17).

Our model is closer to the following works, which take the mesh structure into account. Ben-Hamu et al. [BHMK+18] proposed a representation based on multiple charts, which conformally map different parts of shapes to a domain. Since features over each charts are normalized separately, the fine structure will be better preserved than with a single chart. However, while the creation of such charts requires a sparse correspondence, reconstruction of shapes from the charts needs a template shape, which amounts to a dense correspondence. In order to find such correspondence, one has to introduce a time-consuming workflow beforehand. Groueix et al. [GFK+18] learns a parameterization of shapes with multiple embedded charts. Hence one does have to manually create the charts. However, the generated charts do not always perfectly fit with each other, nor do they preserve as much details as the ones in [BHMK+18]. Umetani [Ume17] developed a depth map representation with cube as the domain. This representation works well for close-to-convex shapes like cars, but would be difficult to be applied on highly curved and non-convex shapes. Kostrikov et al. [KJP+18] used the same Dirac operator as ours. But they merely replaced the Laplace-Beltrami operator in the neural network with the Dirac operator, thus the real power of the Dirac operator, namely its connection to conformal transformation, is not exploited.
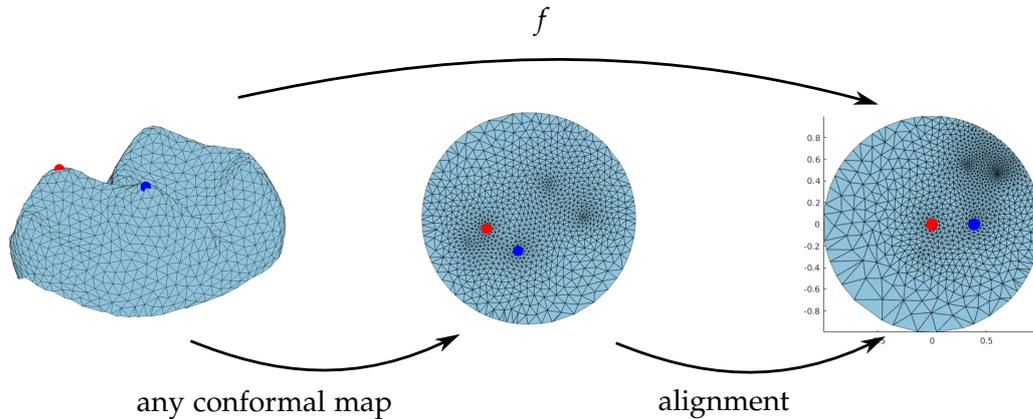
Figure 4.20: For disk-like surfaces, given two landmark points there is a unique conformal map which maps the first point (red) to zero and maps the second one (blue) to the *x*-axis.

Conformal parameterization Our method relies on a conformal parameterization, i.e., a conformal mapping of surfaces to the canonical domain, e.g., the unit disk for disk-like surfaces and the unit sphere for spherical surfaces. However, these maps are not unique but differ by a conformal automorphism of the domain. To deal with the ambiguity one may choose from the following approaches depending on the situation:

Landmark alignment   We know that the conformal automophism of the 2-sphere, i.e., the Möbius transformation, is fully determined by three distinguished points and the conformal automorphism of the disk is determined by one point and one rotation. Hence we choose two landmark points for disk-like surfaces and three landmark points for closed surfaces and align these landmarks via conformal mappings. One example is shown in Fig.4.20.

Landmark-free alignment   For example, [BCK18] proposed a canonical Möbius transformation such that the mass center is aligned with the sphere center. Then we register two spherical meshes of centered Möbius transformations by searching for a optimal rotation.

Without any alignment at all   This will result in a larger shape latent space and consequently poses higher demands on the capacity of neural network, because, for example, a rotation of shapes might also yield a rotation of the curvature function. However, our model is particularly good at capturing this uncertainty (see the discussion in Sec. 4.3.3).

Specifically, there are many available algorithms for conformal parameterization of open and closed surfaces, e.g., [GWC+04; CPS13; CL15; CLL15; SC17; YDT+18]. In fact, we did not observe obvious difference of these algorithms in our experiments.

Centroidal Voronoi tessellation   The isotropic meshing is usually constructed by centroidal Voronoi tessellation [DFG99]. Given a set of points $\{v_i\}$ in $\mathbb{R}^2$. The Voronoi region V_i corresponding to v_i is defined by

$$V_i = \{x \mid |x - v_i| \leq |x - v_j|, j \neq i\},\tag{4.12}$$

which are polygons. Given a density function $\mathfrak{d}$, the centroid v_i^* of the polygon V_i is given by

$$v^* = \frac{\int_V y\mathfrak{d}(y)dy}{\int_V \mathfrak{d}(y)dy}.\tag{4.13}$$

We call a point set $\{v_i\}$ the weighted centroidal Voronoi tesselation if $v_i = v_i^*$ holds true for all i.

In this paper we use Llyod relaxation to compute the CVT. Given a point set $\{v_i\}$ we iteratively update the point v_i with the corresponding centroid v_i^* until it converges (see App. 8.1 for the formula of computing weighted centroid of polygons).

Analogously, for the point set distributed on the sphere with the density function $\mathfrak{d} : S^2 \to \mathbb{R}$, the spherical CVT is obtained by changing the domain and metric in (4.12) and (4.13) accordingly.

| Sampling | Voronoi D. | 1-st iter. | 5-th iter. | Delaunnay Tri. |
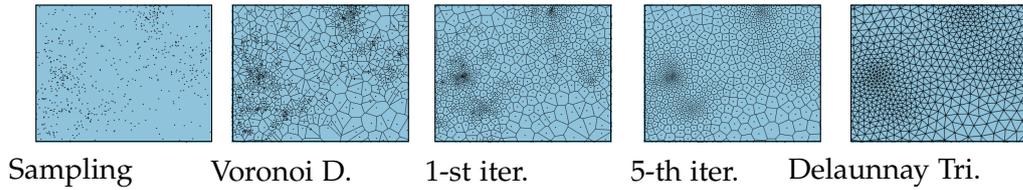
Figure 4.21: Centroidal Voronoi Tessellation. In order to obtain an isotropic meshing with respect to a given density, we first sample a point set according to the density and repeatedly apply the Lloyd's relaxation. Observe that the point set becomes more and more isotropic as the iteration goes.

4.3.2 Method

Encoding the conformal structure   In discrete case, how to encode shapes in the scheme of the Bonnet problem (Sec. 4.3.1)? While the mean curvature half-density can be represented by a vertex-based or face-based function, it is not straightforward to pack the conformal structure in a form that is suitable for machine learning pipeline. For example, we can recover the shape of a cow from its spherical conformal parameterization ((b) in Fig. 4.22) by prescribing the function of mean curvature half-density ((c) in Fig. 4.22). But it is not clear how to construct a spherical mesh, which is conformal equivalent to a given shape, purely from scalar functions. One might consider the notion of discrete conformal equivalence for triangular meshes by length cross-ratio on edges ([SSP08]). But it is unclear how to transfer the length cross-ratio across different meshes.

   Recall that the conformal structure is the set of metrics modulo the equivalence relation $g \sim e^{2u}g$, i.e., two metrics are identified if they only differ by a scaling at each point. Therefore, instead of encoding the conformal structure, we encode the metric of shapes. In general, the space of all metrics is extremely large, thus we focus on a smaller subset, i.e., the isotropic meshing. Since the conformal map is locally isotropic, i.e., it takes an isotropic mesh to a mesh close to isotropic (see the zoom-in in Fig. 4.22), and we know that the isotropic meshing is usually generated by the centroidal Voronoid tessellation (CVT) with respect to a density function [ADVDI03], this density function can be actually utilized as an approximation of a metric. At the beginning of our pipeline all the input shapes are isotropically remeshed (like (a) in Fig. 4.22). The vertex density functions are then extracted from their conformal parameterizations ((b) in Fig. 4.22). To recover the spherical mesh (b) in Fig. 4.22 one constructs the CVT with respect to the density function. Although the CVT is not unique in general for a given density function, this representation induces satisfied reconstructed shapes in our experiments.

Disk-like surfaces   Since the disk-like surface is homeomorphic to a standard square, it is easy to adapt the curvature-type data to the standard machine learning framework. The surfaces are first conformally mapped to a standard square and then the functions
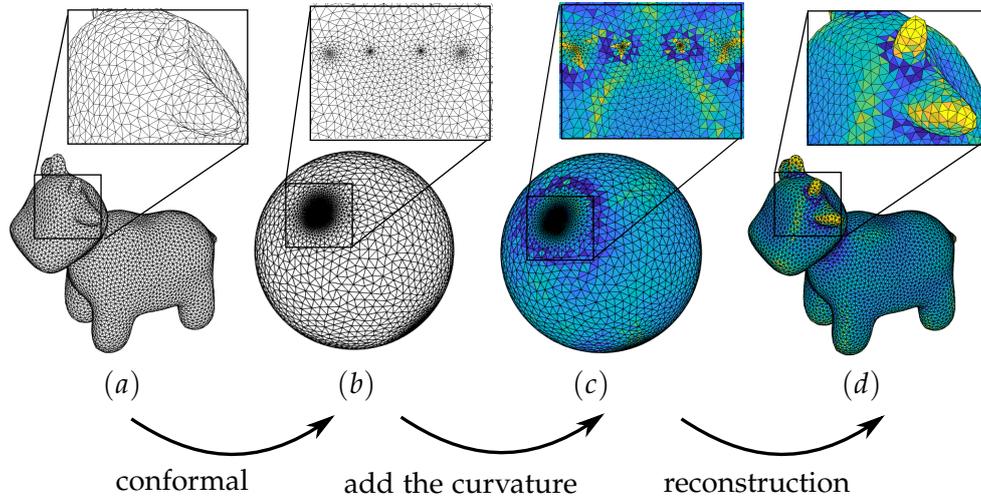
Figure 4.22: [YDT+18] shows that a simply-connected surface in $\mathbb{R}^3$ can be faithfully reconstructed from its conformal parameterization by prescribing the mean curvature half-density.

can be interpolated on some fixed grids. Note that each surface is now represented just like a 2D image with two channels.

To reconstruct the metric from a density function one iteratively applies the following steps (see Fig. 4.23):

1. Randomly sample the points with respect to the density $\mathfrak{d}$ (defined in Sec. 4.3.2).

2. Create the Voronoi diagram. We have to be a bit careful that the Voronoi cells close to the boundary are mostly unbounded. Hence we reflect the points close to the boundary, so that all the Voronoi cells inside or close to the unit disk are bounded.

3. Compute the (weighted) centroids of the bounded Voronoi cells and remove the points lying outside the disk.

Then, a Delaunay triangulation is constructed by taking the dual of the Voronoi diagram. Generally, this triangulation does not perfectly fit the disk at the boundary, but it does not significantly affect the global appearance of shapes. In the end, we solve the Dirac equation with free boundary condition.

From now on we will focus on the case of spherical surfaces.

Construction of local patches Some previous works aimed at building CNNs on arbitrary graphs or surfaces, see [BZSL14; KW17; MBM+17; BMM+15; MBBV15; MGA+17] and the survey [BBL+17]. In our case the underlying domain is fixed, namely a regular mesh of the round sphere. Hence we opt for a simpler solution. The idea is that we construct several small tangent patches which cover the whole sphere and project the

| Voronoi D. | Flipping | Update | Remove |
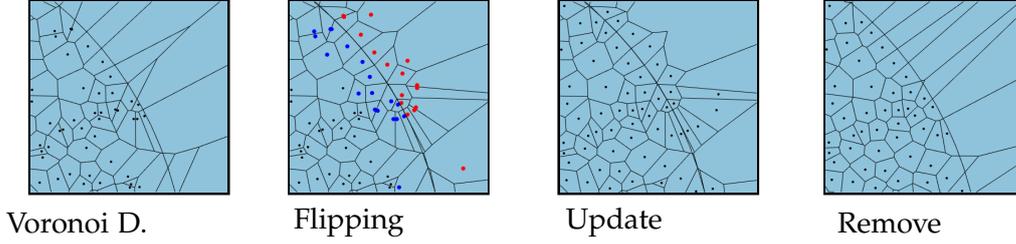
Figure 4.23: Constraint CVT. To avoid dealing with unbounded Voronoi cells, we flip the points, which are close to the boundary, such that the cells close to the boundary are all bounded.

data on these patches. The distortion caused by the projection is neglectable when the size of patches is small. The corresponding convolution is easy to construct since it just consists of several ordinary 2D convolutions. Similar ideas are used in a more general setting [TPKZ18].

We choose the mesh $S_0 = (V_0, E_0, F_0)$, obtained by iteratively taking the 1-to-4 subdivision of the unit icosahedron, as the domain. Each face $T_i \in F_0$ is assigned with a tangent plane, identified with $\mathbb{R}^2$, at the barycentric center of f_i. Then we take a square $[-l, l] \times [-l, l]$ in the tangent plane such that the projection π_i of face T_i would entirely lie in the square. We call the map π_i a patch associated with the face T_i and it gives a local coordinate system of points in the pre-image $\pi_i^{-1}([-l, l] \times [-l, l])$. For the whole mesh S_0 we fix the length l such that every faces T_i are projected inside the square.
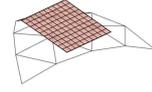
Representation of functions Now, any function $f : S^2 \to \mathbb{R}$ can be locally represented by $f_i := f \circ \pi_i^{-1} : [-l, l] \times [-l, l] \to \mathbb{R}$. We either interpolate the function f_i on some fixed grid of the domain $[-l, l] \times [-l, l]$, or consider an orthonormal basis of $[-l, l] \times [-l, l]$, e.g., the eigenfunctions of the Laplacian, and represent f_i by some coefficients of the orthogonal decomposition $c_{ij} := \langle f_i, e_j \rangle$. In this paper we adopt the direct interpolation method for the purpose of simplicity.

With a conformal map $\Phi : M \to S^2$ any points on the shape are registered with a point on the sphere. In this context, we are particularly interested in two functions over the sphere, i.e, the mean curvature half-density h and the vertex density function $\mathfrak{d}$ of the conformal parameterization $\Phi(M)$.

By [YDT+18] the mean curvature half-density h is a face-based function given by

$$h_i = \frac{\sum_j |e_{ij}| \tan \theta_{ij}/2}{2\sqrt{A_i}}, \tag{4.14}$$

where the sum runs over all the edges e_{ij} of the face T_i, θ_{ij} are bending angles at the edge e_{ij} and A_i are the face area. We also notice that Rusinkiewicz proposed a face-based

mean curvature estimation $\mathfrak{H} : F \to \mathbb{R}$ by minimizing a least-square error [Rus04]. It turns out that the mean curvature half-density given by

$$\mathfrak{h}_i := \mathfrak{H}_i \sqrt{A_i} \tag{4.15}$$

is usually smoother than (4.14) and hence numerically more robust. Therefore, in the context of machine learning, we advocate for $\mathfrak{h}_i$ rather than (4.14) and compose it with Φ so that $\mathfrak{h} \circ \Phi^{-1} : S^2 \to \mathbb{R}$ is stored in our representation.

There are several ways to approximate the vertex density, e.g., the density kernel estimation on each local patch. For simplicity, we estimate the density function $\mathfrak{d}$ by the reciprocal of face area, $\mathfrak{d}_i := 1/\tilde{A}_i$, where $\tilde{A}_i$ is the face area of the conformal parameterization $\Phi(M)$. We do not normalize the density $\mathfrak{d}$, since the integral of the piecewise constant function $\int d = \sum_i \mathfrak{d}_i A_i$, which is equal to the number of points inside the patch, gives us the information how many points should be initially sampled at the beginning of the metric reconstruction. During our experiments it occurs to us that the logarithmic density $\tilde{\mathfrak{d}} := \log \mathfrak{d}$ is usually more evenly distributed. Therefore, the logarithmic density of shapes is in fact recorded as the second function in our representation.
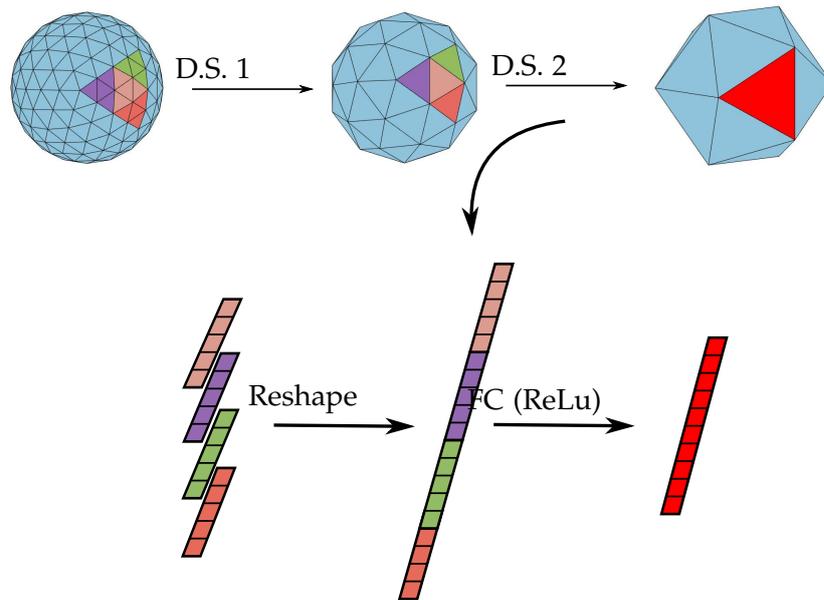

Figure 4.24: Downsampling layers based on the subdivision structure of the spherical meshes. The tensors in the previous layer, which are corresponding to a common triangle in the next layer, are merged and fully connected to the tensor associated with the father triangle. These downsampling layers respect the spatial relations among the triangles.

The architecture of the neural network   Assume that the spherical domain is obtained by subdividing the isocahedron k times. So we have $N_p = 20 \times 4^k$ patches and each patch is assigned with a $n \times n$ grid. Then every surface is associated with the data of size $N_p \times n \times n \times 2$. The encoder reduces the spatial dimension by the following 2 types of layers:

Convolution layer   Each patch is associated with the data of size $n \times n \times 2$. Hence we just apply the convolution layers with kernel size 4×4 and strides 2 several times. Note that the filter weights are shared across different patches.

Downsampling layer   The dimension reduction can not be proceeded any more as the spatial dimension reaches 1. To decrease the dimension further, we design a downsampling layer which takes the connectivity relation among the patches into account. Note that, since our spherical domain is constructed by subdividing a isocahedron, it is naturally endowed with a hierarchical structure (Fig. 4.24). We denote these subdivided spherical meshes by M_0, M_1 and so on. Suppose four triangles F_{i1}, F_{i2}, F_{i3}, F_{i4} in M_i are given by subdividing the triangle F_{i-1} in M_{i-1}. We merge the tensors with dimension n_i corresponding to these four small triangles to a single vector and fully connect them to the tensor with dimension n_{i-1}, which is corresponding to the triangle F_{i-1} (Fig. 4.24).

For all layers we set the activation functions to be leaky ReLU with rate 0.2 except that the last layer is endowed with the sigmoid activation. For each experiment the detailed architecture is shown in App. 8.4.

Reconstruction of parameterization   Suppose a density function is given in each local patch. We first randomly sampling the points with respect to the density in each patch and then project these points from the tangent plane back to the unit sphere. For each patch, we remove the points that are projected outside the corresponding triangle.

From this spherical point clouds with respect to the density, the reconstruction of the spherical CVT is similar as the one described in Sec. 4.3.2. We just replace the planar Voronoi diagram with the spherical one and no boundary issue has to be considered.

4.3.3 Results

We use the Matlab package gptoolbox [Jac+18] for data pre-processing and Tensorflow [AAB+15] to build and train the neural networks. All the neural networks are trained and evaluated with the GPU GeForce GTX 1080 with 8GB memory.

Preliminary applications   We first present some simple applications that are unrelated to machine learning.

In smooth case, the mean curvature half-density changes covariantly $h \mapsto m \cdot h$ under the parameterization scaling $x \mapsto m \cdot x$, $m \in \mathbb{R}$. Analogously, in discrete case, one can adjust the parameterization by scaling the vertex density, i.e., multiplies the density $\eth$
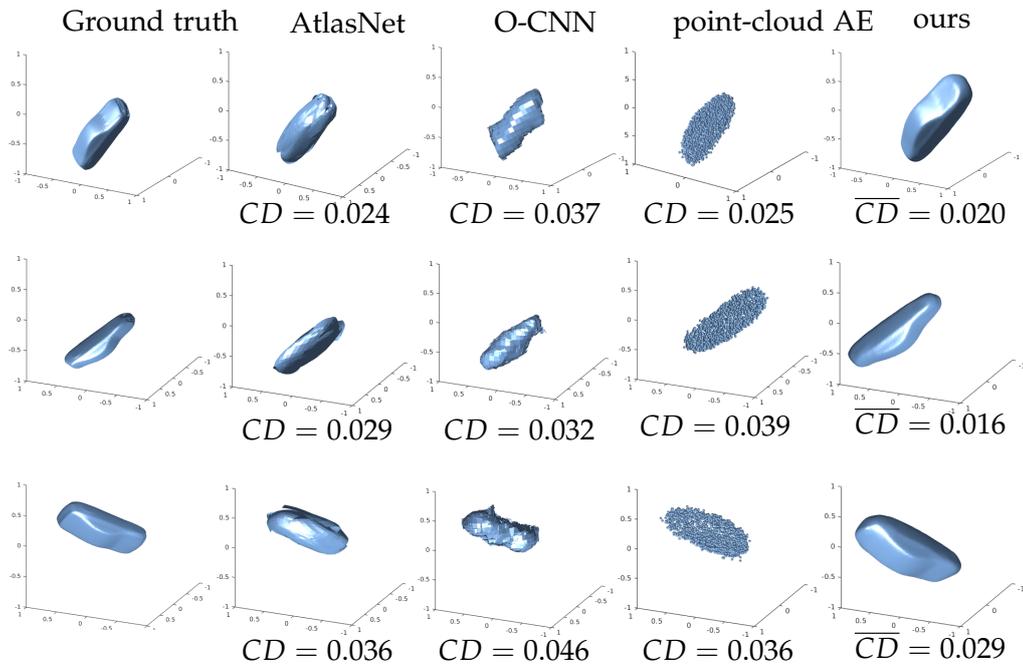
Figure 4.25: Autoencoder for transformed cars. We transform a shape of car by applying random translation, scaling and rotation. We demonstrate our results with other three models based on the point clouds, namely the point-cloud AE [ADMG18] and the AtlasNet [GFK+18] and based on voxels, namely the O-CNN [WSLT18]. Other methods, though were shown to achieve satisfied results on aligned dataset, do not correctly capture the symmetry of various transformations. In contrast, our model succeeds in producing convincing transformed shapes. We evaluate the results by measuring the Champfer distance $\overline{CD}$. However, since our our model lose the information of translation and scaling, we have to first normalize the volume of the results with centered position (unnormalized shapes are shown above). In the end we compute the Chamfer distance of the normalized outputs $\overline{CP}$.
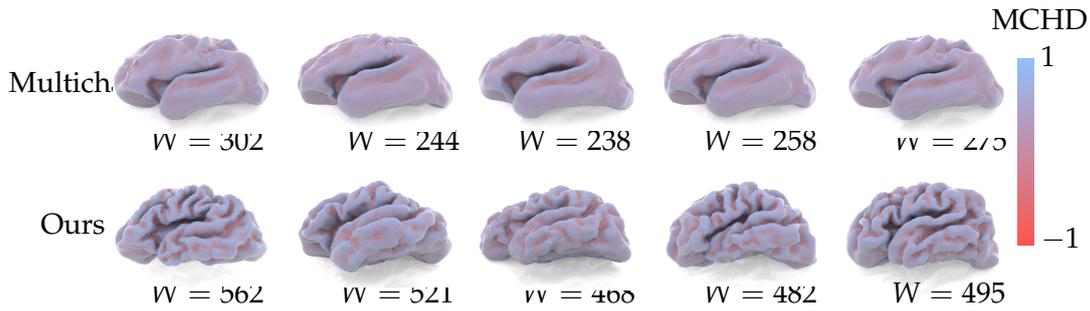
Figure 4.26: The randomly generated cortical surfaces by Multi-chart GAN [BHMK+18] and the VAE based on our representation. Our representation has dimension $320 \times 32 \times 32 \times 2 = 655360$, which is around 3 times that of Multi-chart, i.e., $16 \times 64 \times 64 \times 3 = 196608$. However, we only require 3 landmark points for alignment, while the Multi-chart needs a dense correspondence for surface reconstruction. The surfaces are labeled by the mean curvature half-density. Note that, the training data mostly have the Willmore energy from 900 to 1000. Although the generated surfaces from our model have been smoothed to a certain extent (partly due to a well-known limitation of VAE), our model apparently preserves more fine structure than the position-based model.

with a constant number, $\mathfrak{d} \mapsto m\mathfrak{d}$. In order to preserve the shape, one has to adjust the mean curvature half-density by $h \mapsto \frac{h}{\sqrt{m}}$. The reconstructed shapes from the modified representation are actually remeshings with approximately $m|V|$ vertices, where $|V|$ is the number of vertices of the original mesh. Fig. 4.27 shows that our method will preserve the smooth features on the shape. However, the regions of high curvature tend to be smoothed with declining vertex number.
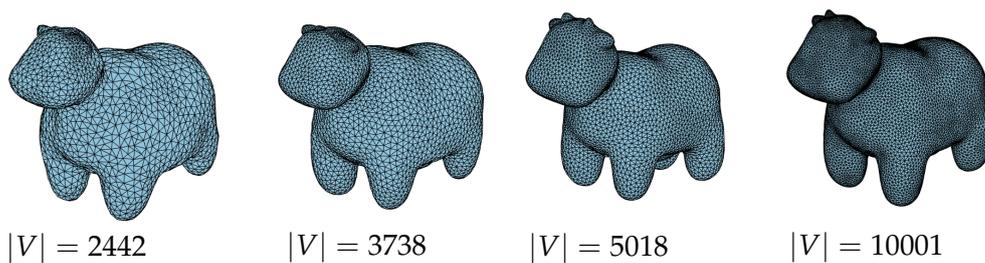

Figure 4.27: Given an original shape of cow with $|V| = 5000$, the density is modified by multiplying with 0.25, 0.75, 1 and 2. The mean curvature half-density changes accordingly such that the mean curvature is preserved.

Shape interpolation We visualize the interpolation of our curvature-based representation. Fig. 4.19 shows the reconstructed shapes from a linear interpolation of two

animals, whose conformal parameterizations are matched by a Möbius transformation that aligns 3 chosen landmark points. In addition, one can interpolate the latent space representation of a trained autoencoder (see Sec. 4.3.3). Fig. 4.28 shows two latent space bi-linear interpolations of a dataset of cars.
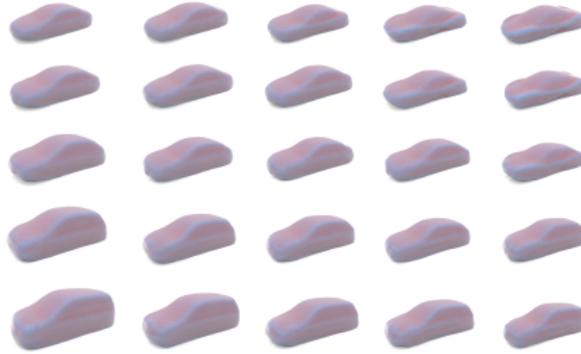

Figure 4.28: Latent space interpolation. We choose four examples in the car dataset and interpolate their mean values in the latent space of VAE. The left lower triangle is a bilinear interpolation of a van, a car and a SUV. The right upper triangle is a bilinear interpolation of a van, a car and a race car.

Random generation of disk-like and spherical shapes We test our model for disk-like surfaces on a dataset of anatomical shapes provided by [BLC+11]. In particular, we choose the shapes of teeth, which is one of three types of bone in this dataset. To create the representation, we first take an intermediate conformal map, which maps the teeth to the unit disk by the algorithm from [CL15].

Several landmark points are available in [BLC+11], hence we choose two landmark points u_i, v_i for every shape M_i. We know that the conformal automorphisms of the unit disk have the form

$$f(z) = e^{i\theta} \frac{z - a}{1 - \bar{a}z},$$

where $\theta \in \mathbb{R}$ and $a \in \mathbb{C}$. Set $a = u_i$ and θ such that $f(v_i) \in \mathbb{R}$. Clearly, this uniquely determined map $f_{a,\theta}$ satisfies $f(u_i) = 0$ and $f(v_i) \in \mathbb{R}$. Fixing a reference shape M_0, for any shape M_i we apply the alignment map $f_0^{-1} \circ f_i$ for every shapes.

All the aligned disk meshes are then mapped to the square via the Schwarzt-Christoffel mapping. The functions are interpolated on the 64×64 grid using the `scatteredInterpolant` function in Matlab.

For spherical surfaces we take the dataset of 1240 cars from ShapeNet [CFG+15]. All the shapes are converted into genus-0 surfaces by [Ume17]. Then we create the aligned conformal parameterization by the canonical Möbius transformation [BCK18] and pack them with $320 \times 32 \times 32 \times 2$ dimensional tensors.

The randomly generated teeth and cars are shown in the appendix as well as their curvature representation.

Generation of unaligned data

Discussion of local invariance We call two functions f_1 and f_2 local invariant if they have the same function value but only differ by a transformation g of domain, i.e., $f_1 = f_2 \circ g$. Traditional CNNs are able to capture the translational features such as (a) of inset. Hence one would expect the CNNs for 3D shapes with the similar properties like local invariance under translation, rotation or even scaling. However, 3D generative models based on position, such as point cloud and mesh, will not have such properties due to the varied function value of coordinates (see (b)). This makes it more difficult for CNNs to extract meaningful information. The voxel-based models are local invariant, but they are not applicable for data with high resolution due to the high cost of memory and computation. Some multi-resolution representation, e.g., octree [TDB17; WSLT18], are designed to overcome this problem, but the local invariant

(a) local invariant

(b) not local invariant

(c) local invariant

property does not hold any more. In contrast, our model (sketched by (c)), together with the CNN on the sphere, provides an efficient way to learn the 3D data without a certain alignment. We verify our argument with the following two examples.
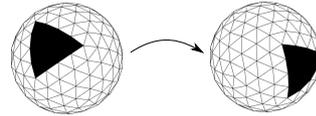
Learning unaligned anatomical data We merge three different anatomical models in [BLC+11] and create the representations without any alignment methods. Insect shows the randomly generated bones of different types. Compared with Fig. 8.1 the bones get smoothed due to the expanded shape space. However, we show that our model is still capable to extract the meaningful information from the am-

biguity by visualizing the latent space distribution (Fig. 4.29). We compare the result with a baseline model that has the same network architecture but operates on the coordinate functions.
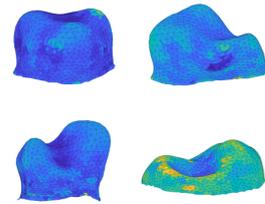
Generation of transformed cars In this experiment we would like to see whether the 3D generative models are able to correctly predict shapes with various transformation. The dataset is created by randomly translating, rotating and scaling a single shape of car in the cube of size $[-1,1] \times [-1,1] \times [-1,1]$. We train autoencoders based on different models on 900 training data and test them on 100 validation data. The comparison shows that our method produces more accurate prediction than others (see Fig. 4.25).
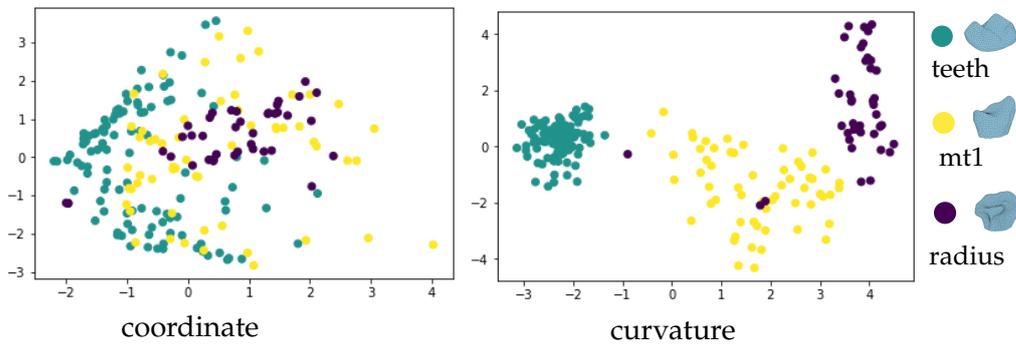
Figure 4.29: Latent space visualization. The dataset is composed of three different types of anatomical surfaces. We project the latent space representation on a 2-dimensional space by PCA. Though all the shapes are packed without alignment, the three types of bones are clearly separated in the latent space. In contrast, the model based on the coordinate failed to learn the structure of the bones, so their distribution in the latent space is not well separated.

Since only our model considers the mesh structure of shapes, to make a fair comparison, we evaluate the results with Chamfer distance which only depends on the underlying point clouds. Note that, as a trade-off, our representation loses the information of translation and scaling. Thus we first normalize the shapes reconstructed from our model and then calculate the Chamfer distance to the ground truth.

**Cortical surface generation**   To show that our model is particularly good at preserving the fine structure, we perform the experiment on human cortical surfaces, which are highly folded with a lot of "hills" and "valleys". A dataset of cortical surfaces are available on the Open Access Series of Imaging Studies (OASIS) [MFC+10]. The MRI images are converted to genus-0 surfaces via the open source reconstruction software FreeSurfer (http://surfer.nmr.mgh.harvard.edu/).

We first compare our model to three other state-of-art autoencoders for 3D shapes. Fig. 4.17 shows that, although all models succeed in characterizing the shapes in large scale, our model preserves much more small features, e.g., the curvature, than the others.

Training details   Our model is trained with 200 epochs for around 5 hours. The point-cloud AE [ADMG18] with 2048 points for each data and AtlasNet [GFK+18] with 2500 points for each data are both trained with 500 epochs for approximately 4 hours. Although the point-cloud based models above have smaller data size than ours, the training of their neural networks already exhausted our GPU memory. The OGN, with the octree representation of $128 \times 128 \times 128$ dimensional voxels, is trained with 4000 epochs with 5 hours. While other models produce the shapes instantly after training, it takes 2 minutes with our method to reconstruct a mesh with 10000 vertices from
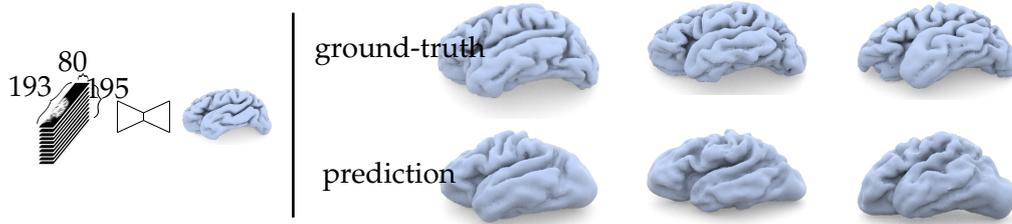
Figure 4.30: Volume-Curvature autoencoder. The input is the MRI volumetric data from [MFC+10]. Since only the left hemisphere is generated, we align the volume using FreeSurfer [RRF10] and chop the volume properly such that the dimension is $193 \times 80 \times 195$. The encoder is shown in Fig. 8.3 and the decoder is the same as Fig. 8.1.

curvature.

Next, we compare the cortical surfaces randomly generated by our VAE to the ones by Multi-chart GAN [BHMK+18] (Fig. 4.26). While both mesh-based models generate significantly more faithful results than other types of representation in Fig. 4.17, the "hills" and "valleys" are much more visible with our model. Moreover, we only choose 3 landmark points on each shape to align the conformal parameterization, while it requires 21 landmark points to create 16 charts as in [BHMK+18], and even a template shape, which amounts to a dense correspondence, to reconstruct the final shapes.

At last, we try to create an autoencoder that converts the 3D MRI images of brain to cortical surfaces. In this case, the encoder consists of several 3D convolutional layers (see Fig. 8.3) and the decoder is the same as the ones in previous experiments. Fig. 4.30 shows that our model is able to predict the cortical surface from the MRI volume to a certain extent, but the accuracy is not yet optimal, because the neural network failed to capture the spatial correspondence between the volumetric data and the spherical data. We leave the construction of a finer 3D-to-2D autoencoder to future work.

5 Conclusion

We presented a new discretization framework for the Dirac operator on discrete surface meshes. Our framework allows for the discretization of both an extrinsic and an intrinsic Dirac operator, both of which are in alignment with their continuous counterparts in their basic properties. Most importantly, they both relate to integrable conformal deformations, which is a fundamental property of the continuous operator.

However, the open problems in this area is far from being exhausted. For instance, it is tempting to clarify the relation between our setting and the discrete Dirac operators and spin structures in other areas, e.g., the discrete complex analysis and physics [Cim12; Ken02; CR07; CR08; Mer01]. Besides, Arnold. et al. [AFW06; AFW10] showed that the stability and convergence of some numerical PDE are guaranteed for some certain discretization that captures the key structures of the de Rham cohomology and Hodge theory. In the following work by Leopardi and Stern [LS16] constructed a discrete Hodge-Dirac operator which shows good numerical stability. Hence it is interesting to see if our Dirac operator, which sits in different bundle as Hodge-Dirac, can be adapted to their framework so that it allows for a more systematic analysis of the numerical stability. In the end, we found some similarity between our treatment of the discrete parallel transport and the one for the discrete gauge theory [CH12]. To investigate and discover the possibility for applying our theory in physics is the long-term goal of our future research.

6 Appendix A. Basic Spin Geometry

6.1 Clifford Algebras and Their Representations

We basically follow the general theory in [LM90] and carry out some calculations in 2 and 3 dimensions.

Let V be a vector space over $\mathbb{R}$ or $\mathbb{C}$ with a bilinear form $\langle \cdot, \cdot \rangle$. The Clifford algebra is defined by

$$Cl(V, \langle \cdot, \cdot \rangle) = \sum_{i=0}^{\infty} \otimes^i V / \sim, \tag{6.1}$$

where the equivalence relation $\sim$ is defined by

$$v \otimes w - w \otimes v \sim \langle v, w \rangle.$$

In particular, we always consider the Euclidean vector spaces V or its complexification $V \otimes \mathbb{C}$, then $\langle \cdot, \cdot \rangle$ will be assumed be the Euclidean metric or the natural extension to complex bilinear form.

The Clifford representation (also called Clifford module) $\mathcal{S}$ is a vector space with an action from the Clifford algebra:

$$Cl_n \times \mathcal{S} \xrightarrow{\rho} \mathcal{S}$$

In the following we show the explicit formula of the Clifford algebra and their irreducible representations.

In fact, $Cl_2 := Cl(\mathbb{R}^2) \cong \mathbb{H}$, and we usually identify an orthonormal basis $\{e_1, e_2\}$ by

$$e_1 \mapsto \mathbb{i}, \quad e_2 \mapsto \mathbb{j}$$

In this case the irreducible Clifford representation is isomorphic to $\mathbb{H}$ and the map ρ is also given by the quaternion multiplication.

And $Cl_3 := Cl(\mathbb{R}^3) \cong \mathbb{H} \oplus \mathbb{H}$, and we usually identify an orthonormal basis $\{e_1, e_2, e_3\}$ by

$$e_1 \mapsto (\mathbb{i}, \mathbb{i}), \quad e_2 \mapsto (\mathbb{j}, \mathbb{j}), \quad e_3 \mapsto (\mathbb{k}, -\mathbb{k})$$

In this case, there are two irreducible Clifford representations, both isomorphic to $\mathbb{H}$, given by first projecting the Clifford algebra to one component of $\mathbb{H} \oplus \mathbb{H}$ and taking the quaternion multiplication. Observe that the key difference between these two representations are $\rho_{e_1 e_2} = \rho_{e_3}$ v.s. $\rho_{e_1 e_2} = -\rho_{e_3}$. We will always use the first

representation, since it coincides with our conventional orientation of surfaces. Therefore for the representation we frequently use the identification:

$$e_1 \mapsto \mathbb{i}, \quad e_2 \mapsto \mathbb{j}, \quad e_3 \mapsto \mathbb{k}$$

Let $i : V_2 \hookrightarrow V_3$ be a subspace of V_3. Then i induces an inclusion of Clifford algebra $i_* : Cl_2 \hookrightarrow Cl_3$ by naturally extending

$$v \mapsto n \cdot v,$$

where $n \in V_3$ is the unit vector which is perpendicular to V_2. Hence any Cl_3 representation $\rho_3 : Cl_3 \times \mathcal{S} \to \mathcal{S}$ can be pulled back to the representation of Cl_2:

$$v \cdot \phi := i_*(v) \cdot \phi,$$

where $\phi \in \mathcal{S}$. If ρ_3 is an irreducible representation, then by counting the dimension we know that $i^*(\rho_3)$ is actually equivalent to the irreducible representation of Cl_2.

6.2 The G-bundle

Definition 6.1. Let X be a manifold. A G-bundle (E_G, X, π) is a triple where E_G is a manifold, G is a Lie group (called the structure group) and $\pi : E_G \to X$ is a surjective map, with the following conditions

- E_G is endowed with a right action of G such that $\pi(p) = \pi(p \cdot g)$ for any $p \in E_G$ and $g \in G$.

- For any $x \in X$ there is a neighbourhood U such that

$$\pi^{-1}U \cong U \times V,$$

which is called the local trivialization and V is called the fibre.

In fact, one can construct the G-bundle ina geometric way. The idea is that we first build the trivial bundle $U_i \times V$ over each $U_i \in \mathcal{U}$, where $\mathcal{U}$ is a set of open sets that cover the manifold. Then we glue all these trivial bundles at the overlaps by the group action (called the transition functions):

$$g_{ij} : U_i \cap U_j \to G,$$

defined on every twofold overlaps $U_i \cap U_j$, such that g satisfies

$$g_{ij}g_{jk}g_{ki} \equiv 1, \tag{6.2}$$

for every three-fold overlaps.

In the end we construct the bundle by

$$E_G := \bigcup U_i \times V / \sim,$$

where $(x, v_j) \sim (x, g_{ij}v_i)$.

We will frequently use the following two types of G-bundles:

Definition 6.2. If the group action is transitive and free and consequently the fibre is isomorphic to G, then we call it a principal bundle P_G.

Example 6.1. *The following two principal bundles will be frequently used:*

- *In particular, let X be a oriented Riemmanian manifold, the orthonormal frame of its tangent bundle forms a principal $\mathrm{SO}(n)$-bundle, denoted by $P_{\mathrm{SO}}X$.*

- *We know that the spin group $\mathrm{Spin}(n)$ is a two-fold covering $\xi_0 : \mathrm{Spin}(n) \to \mathrm{SO}(n)$. The spin bundle $P_{\mathrm{Spin}} \xrightarrow{\xi} P_{\mathrm{SO}} \to X$ is a principal bundle such that ξ commutes with the group actions.*

Now let $P_{\mathrm{SO}} \to X$ be the SO-principal bundle. Take an open covering $\mathcal{U} = \{U_i\}$ such that P_{SO} is trivial on each open set U_i. The corresponding transition functions are $g_{ij} : U_i \cap U_j \to \mathrm{SO}$ satisfying (6.2). For each g_{ij} we pick any lifting $\tilde{g}_{ij} : U_i \cap U_j \to \mathrm{Spin}(n)$ such that $g_{ij} = \varepsilon \circ \tilde{g}_{ij}$. The new transition functions induce a Spin-principal bundle if and only if (6.2) holds for $\tilde{g}_{ij}$, i.e.,

$$w_{ijk} := \tilde{g}_{ij}\tilde{g}_{jk}\tilde{g}_{ki} \equiv 1,$$

Since $\varepsilon(w_{ijk}) \equiv 1 \in \mathrm{SO}(n)$, $w_{ijk} \equiv \pm 1 \in \mathrm{Spin}(2)$, meaning that w is actually a $\mathbb{Z}_2$-valued cocycle. One can show that the cohomology class represented by w is exactly the 2nd Stiefel-Whitney class.

Theorem 6.3. *The spin structure exists if and only if the second Stiefel-Whitney class vanishes.*

Proof. See [LM90]. $\square$

Definition 6.4. Given a principal bundle P_G and a vector space V with a left action of G. The associated vector bundle is defined by $P_G \times V \backslash \sim$, where

$$(p, v) \sim (p \cdot g^{-1}, gv).$$

Example 6.2. *We will frequently use the following associated bundles:*

- *Let X be a n-dimensional Riemannain manifold and $\rho_n : \mathrm{SO}(n) \times \mathbb{R}^n \to \mathbb{R}^n$ be the canonical representation, then*

$$P_{\mathrm{SO}}X \times_\rho \mathbb{R}^n \cong TX.$$

- *One can actually extends ρ_n to the Clifford algebra, giving*

$$cl(\rho_n) : \mathrm{SO}(n) \times Cl_n \to Cl_n.$$

The Clifford bundle is defined by

$$\mathbf{Cl}(X) := P_{\mathrm{SO}} \times_{cl(\rho_n)} Cl_n$$

- *Let S be the irreducible Clifford module with $\rho : Cl_n \times S \to S$, the spinor bundle is defined by:*

$$\mathcal{S} = P_{\mathrm{Spin}}(X) \times_\rho S$$

6.3 The Connections and the Dirac Operator

Definition 6.5. A connection on the principal G-bundle is a $\mathfrak{g}$-valued one-form $\omega :$ $\mathfrak{g} \otimes \Omega(P_G)$ such that

$$\omega((L_p)_*(v)) = v,$$

where $v \in \mathfrak{g}$ and

$$g^*(\omega) = \mathrm{Ad}_{g^{-1}}\omega$$

, where $\mathfrak{g}$ is the Lie algebra of G.

The connection on the principal bundle induces the covariant derivative on the associated bundle $\nabla : \Gamma(TM) \times \Gamma(P_G \times_\sim V) \to \Gamma(P_g \times_\sim V)$, given by

$$\nabla_X(p, v) = (p, d_X v) + \omega((p_*)(X)) \cdot v$$

where $\cdot$ denotes the Lie albegra representation induced from the Lie group action.

Example 6.3. *In particular, the we have the following two covariant derivatives:*

- *The conventional covariant derivative of the Riemmanian manifold can be obtained from a connection on P_{SO}. In a local neighborhood U we choose an oriented orthonormal frame $\{e_1, \dots, e_n\}$ which is actually a local section $p \in \Gamma_U(P_{\mathrm{SO}})$. Then the covariant derivative is given by $\nabla : \Gamma(TM) \times \Gamma(TM) \to \Gamma(TM)$ such that*

$$\nabla e_i = \sum e_j \otimes \omega_{ji},$$

  where $\omega_{ji} = \langle \omega(\partial e_j), e_i \rangle$.

- *The connection ω on P_{SO} induces the spinor connection $\tilde{\omega}$ on $P_{\mathrm{Spin}}M$ by $\tilde{\omega} = \xi^*\omega$. Furthermore, it induces the spinor covariant derivative on the spinor bundle $\mathcal{S}$. More precisely, choose an orthonormal frame $\{\tilde{e}_1, \dots, \tilde{e}_n\}$, which is actually a local section of $\tilde{p} \in \Gamma_U(P_{\mathrm{Spin}})$. Then the spinor covariant derivative $\tilde{\nabla} : \Gamma(TM) \times \Gamma(\mathcal{S}) \to \Gamma(\mathcal{S})$ satisfies*

$$\tilde{\nabla}\tilde{e}_k = \frac{1}{2}\sum_{i<j} \omega_{ji} \otimes e_i e_j \cdot \tilde{e}_k.$$

In the end, the Dirac operator is a first-order differential operator defined by:

Definition 6.6. The Dirac operator $D : \Gamma(S) \to \Gamma(S)$ is defined by

$$D\phi = \sum_{i=1}^{n} e_i \cdot \tilde{\nabla}_{e_i}\phi$$

Let the manifold X be compact. A global inner product on $\Gamma(S)$ is induced from the pointwise inner product

$$(\phi, \varphi) := \int_X \langle \phi, \varphi \rangle.$$

In the sense of this inner product the Dirac operator is self-adjoint:

Proposition 6.7. *Let X be a closed manifold. We have*

$$(D\phi, \varphi) = (\phi, D\varphi).$$

Proof. See Prop 5.3 in [LM90]. □

The connection Laplacian is defined by

$$\boldsymbol{\Delta} := -\text{tr}(\nabla\nabla),$$

where $\nabla\nabla$ is a tensor given by

$$\nabla\nabla(X, Y) := \nabla_X\nabla_Y - \nabla_{\nabla_X Y}.$$

The connection Laplacian relates with the square of the Dirac operator by the Lichnerowicz formula, which has the following form in dimension 2:

Theorem 6.8. *Let X be an oriented two dimensional manifold. Then one has*

$$D^2 = \boldsymbol{\Delta} + \frac{1}{2}K$$

Proof. For simplicity we take the normal coordinate at any point $p \in X$ such that all the Christoffel symbols vanish at p. Let $\{e_1, e_2\}$ be the corresponding orthonormal frame. It yields

$$
\begin{aligned}
D^2 &= (e_1 \cdot \nabla_{e_1} + e_2 \cdot \nabla_{e_2})(e_1 \cdot \nabla_{e_1} + e_2 \cdot \nabla_{e_2}) \\
&= e_1 e_1 \nabla_{e_1}\nabla_{e_1} + e_2 e_2 \nabla_{e_2}\nabla_{e_2} + e_1 e_2 \nabla_{e_1}\nabla_{e_2} + e_2 e_1 \nabla_{e_2}\nabla_{e_1} \\
&= -\nabla_{e_1}\nabla_{e_1} - \nabla_{e_2}\nabla_{e_2} + e_1 e_2 (\nabla_{e_1}\nabla_{e_2} - \nabla_{e_2}\nabla_{e_1}) \\
&= \boldsymbol{\Delta} + e_1 e_2 \frac{1}{2}(\partial_{e_1}\Gamma_{21}^2 - \partial_{e_2}\Gamma_{11}^2)e_1 e_2 \\
&= \boldsymbol{\Delta} - \frac{1}{2}(\partial_{e_1}\Gamma_{21}^2 - \partial_{e_2}\Gamma_{11}^2) \\
&= \boldsymbol{\Delta} + \frac{1}{2}K,
\end{aligned}
$$

where the last equation follows from the intrinsic formula of the Gauss curvature. □

7 Appendix B. Proofs

Proof of Theorem 4.2. Let $h_{ij} \in \mathbb{H}$ be the unit quaternion which rotates n_i to n_j. Then we can define the modified hyperedges $\mathcal{E}_{ij} := \cos \frac{\theta}{2} E_{ij} = e_{ij} \cdot h_{ij}$, which are used in the intrinsic Dirac operator. Now, assume that two classical nets $\mathcal{X}$ and $\tilde{\mathcal{X}}$ are related by an isometric deformation. This means that we will have $e_{ij} = g_i^{-1} \tilde{e}_{ij} g_i$ and $h_{ij} = g_i^{-1} \tilde{h}_{ij} g_j$, where g_i are unit quaternions. Hence $\tilde{\mathcal{E}}_{ij} = g_i^{-1} \mathcal{E}_{ij} g_j$. If we now look at an eigenfunction ϕ of the intrinsic Dirac operator on $\mathcal{X}$, it will satisfy

$$(D\phi)_i = \frac{1}{2} \sum_j \mathcal{E}_{ij} \phi_j = \lambda \phi_i.$$

and thus

$$\frac{1}{2} \sum_j g_i^{-1} \mathcal{E}_{ij} g_j g_j^{-1} \phi_j = \lambda g_i^{-1} \phi_i.$$

Plugging in the previous equation we get

$$\frac{1}{2} \sum_j \tilde{\mathcal{E}}_{ij} \tilde{\phi}_j = \lambda \tilde{\phi}_i$$

for $\tilde{\phi}_i = g_i^{-1} \phi_i$. Therefore λ and $\tilde{\phi}$ are the eigenvalue and eigenfunction of the intrinsic Dirac operator on $\tilde{\mathcal{X}}$.

Note that one can always right multiply ϕ with a constant quaternion q such that ϕq is again an eigenfunction with the same eigenvalue. This means that the eigenvalues of the operators on $\mathcal{X}$ and $\tilde{\mathcal{X}}$ correspond, and their eigenfunctions are of the form ϕ_i and $g_i^{-1} \phi_i q$ respectively. As a result, the norm of the intrinsic Dirac eigenfunctions is invariant to isometries, i.e. $|\phi_i| = |g_i^{-1} \phi_i q|$, since g_i and q have unit length. Moreover, the spin transformation of these two isometric surfaces according to those eigenfunctions (the Dirac immersions) are:

$$\mathcal{E}_{ij} \mapsto \overline{\phi}_i \cdot \mathcal{E}_{ij} \cdot \phi_j$$
$$\tilde{\mathcal{E}}_{ij} \mapsto \overline{g_i^{-1} \phi_i q} \cdot g_i^{-1} \mathcal{E}_{ij} g_j \cdot g_j^{-1} \phi_j q = \overline{q} \overline{\phi}_i \cdot \mathcal{E}_{ij} \cdot \phi_j q$$

which only differ by a rotation represented by the unit norm q, meaning that they are related by a rigid transformation.

We can similarly prove that the extrinsic Dirac operator is covariant under rigid transformations, by replacing all per-face g_i above with a constant unit quaternion $g \in \mathbb{H}$, the same for all faces.

Discussion on spinor covariance. A reasonable question might be why the eigenfunction is not invariant under isometric deformations but differs by the factor g_i^{-1}, given that the Dirac operator is intrinsic. In fact, the spinor eigenfunction, which is more like a vector field rather than a scalar field, will only be covariant rather than invariant under the coordinate transformation. For example, assume a vector field V with the expression $V = c_1 e_1 + c_2 e_2$ in a frame $\{e_1, e_2\}$. Then, in another frame (e_1', e_2') such that $(e_1', e_2') = (e_1, e_2) \cdot G$, the expression should change accordingly by $(c_1', c_2')^T = G \cdot (c_1, c_2)^T$. Analogously, the quaternion-valued function ϕ is actually the expression of the spinor with respect to a frame which is the pullback of a parallel frame by the immersion $f : M \to \mathbb{R}^3$. If one uses another isometric immersion $df' = g^{-1} df g$, one should expect the corresponding spinor transforms like $\phi \mapsto g^{-1}\phi$.

Proof of Theorem 4.4. The first eigenvalue for the spherical surface has the estimated bounds[Bär98]:

$$\frac{4\pi}{\text{Area}} \leq \lambda_1^2 \leq \frac{1}{\text{Area}} \int_M H^2 \tag{7.1}$$

Hence the immersion induced by the $D\phi = \lambda_1 \phi$ has the mean curvature half-density

$$h = \lambda_1 |df| \leq \sqrt{\frac{1}{\text{Area}} \int_M H^2} |df|$$

By $W = \int h^2$ we have

$$W_1 \leq \left(\frac{1}{\text{Area}} \int_M H^2 \right) \int_M |df|^2 = \int_M H^2 = W$$

Proof of Theorem 4.5. Two nets $\mathcal{X}$, $\tilde{\mathcal{X}}$, which only differ by an Euclidean motion, are related by

$$\tilde{E}_{ij} = g^{-1} \cdot E_{ij} \cdot g$$
$$\tilde{n}_i = g^{-1} \cdot n_i \cdot g$$

where g is constant and has unit length. By the same argument as in the proof of Theorem 4.2 above, any eigenfunction ϕ_i of the intrinsic or extrinsic Dirac operator on $\mathcal{X}$ is corresponding to the eigenfunction $g\phi$ on $\tilde{\mathcal{X}}$. Besides, any two eigenfunctions corresponding to the same eigenvalue might differ by a right multiplication of a unit quaternion $q \in \mathbb{H}$. Taking this into account, corresponding eigenfunctions of $\mathcal{X}$ and $\tilde{\mathcal{X}}$ will have the form ϕ and $g^{-1}\phi q$. Now let's take the canonical representation proposed in [LJC17, Section 4.3]:

$$\phi \mapsto \phi \cdot \frac{(\sum_i c_i \phi_i)^{-1}}{|(\sum_i c_i \phi_i)^{-1}|}$$
$$g\phi q \mapsto g\phi \cdot \frac{(\sum_i c_i \phi_i)^{-1}}{|(\sum_i c_i \phi_i)^{-1}|} g^{-1}$$

where c_i are some fixed real numbers, particularly in [LJC17] they are set to be the areas. It is easy to see that these two eigenfunctions belong to the same conjugate class. Therefore we have the following two rigid-transform invariants:

- The rotation angle, or the normalized real part, $\cos\frac{\theta_i}{2}$, which is invariant under the conjugate class.

- The angle between the rotation axis and the face normal $\langle u_i, n_i \rangle$, because

$$\langle u_i, n_i \rangle = \langle gu_ig^{-1}, gn_ig^{-1} \rangle$$

Proof of Thm. 4.3 (Closing condition for prescribing the area factor)

Proof. Let (x, y) be a conformal coordinate of the immersion $f : M \to \mathbb{R}^3$. The left hand side of (4.10) is actually

$$\phi_x \cdot \phi^{-1}dx + \phi_y \cdot \phi^{-1}dy,$$

while the right hand side reads

$$-\frac{1}{2}(-u_xf_x^{-1} - u_yf_y^{-1}) \cdot (f_xdx + f_ydy)$$
$$= \frac{1}{2}((u_x + u_yf_y^{-1}f_x)dx + (u_y + u_xf_x^{-1}f_y)dy)$$
$$= \frac{1}{2}((u_x + u_yn)dx + (u_y - u_xn)dy). \tag{7.2}$$

The equation (4.10) implies that

$$\phi_x \cdot \phi^{-1} = \frac{1}{2}(u_x + u_yn),$$
$$\phi_y \cdot \phi^{-1} = \frac{1}{2}(u_y - u_xn).$$

Substituting the equations above into the Dirac operator in local form, we obtain

$$D_f\phi = f_x\phi_y - f_y\phi_x$$
$$= \frac{1}{2}f_x(u_y - u_xn) - \frac{1}{2}f_y(u_x + u_yn)$$
$$= 0,$$

by $f_x \cdot n = -f_y$ and $f_y \cdot n = -f_x$. $\qquad\square$

8 Appendix C. Implementation details

8.1 Compute the weighted centroid of polygons

The weighted centroid of a polygon is given by

$$v^* = \frac{\int_V y \mathfrak{d}(y) dy}{\int_V \mathfrak{d}(y) dy}.$$

A Voronoi cell is naturally decomposed in several triangles, of which we first compute the weighted centroid.

Denote the density on the vertex v_i by $\mathfrak{d}_i$ and we assume that the density is linearly interpolated on every triangles. The denominator of v^* is called the weighted area, which is given by $\mathfrak{A}_i = \frac{\mathfrak{d}(v_1) + \mathfrak{d}(v_2) + \mathfrak{d}(v_3)}{3} A_i$, where A_i is the triangle area.

Integrating the linear function on the triangle i, we obtain

$$v_i^* = \frac{(2\mathfrak{d}_1 + \mathfrak{d}_2 + \mathfrak{d}_3)v_1 + (\mathfrak{d}_1 + 2\mathfrak{d}_2 + \mathfrak{d}_3)v_2 + (\mathfrak{d}_1 + \mathfrak{d}_2 + 2\mathfrak{d}_3)v_3}{4(\mathfrak{d}_1 + \mathfrak{d}_2 + \mathfrak{d}_3)}.$$
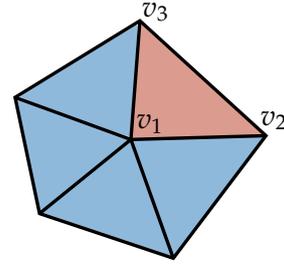
Then, the centroid of the polygon is the weighted sum

$$v^* = \frac{\sum_i v_i^* \cdot \mathfrak{A}_i}{\sum_i \mathfrak{A}_i}.$$

8.2 Finite element method for quaternion gradient

To obtain the discrete formula of the energy $|\omega^2|$, we first derive the formula of the quaternion gradient in discrete case.

Let $h : M \to \mathbb{R}$ be any function. We know that the gradient is defined by $\operatorname{grad} u := (du)^\sharp$, where $\sharp : T^*M \to TM$ is called raising indices defined by

$$\langle \omega^\sharp, v \rangle = \omega(v), \quad \text{for any } v \in TM$$

In a triangle i in quaternion space with the oriented edges a, b, $c \in \mathbb{H}$, we choose a coordinate (x, y) system (inset). Assuming that h is a linear function with the value h_1, h_2, h_3 at the vertices, write dh in local form as:

$$dh = (h_2 - h_1)dx + (h_3 - h_1)dy$$
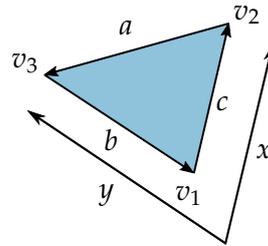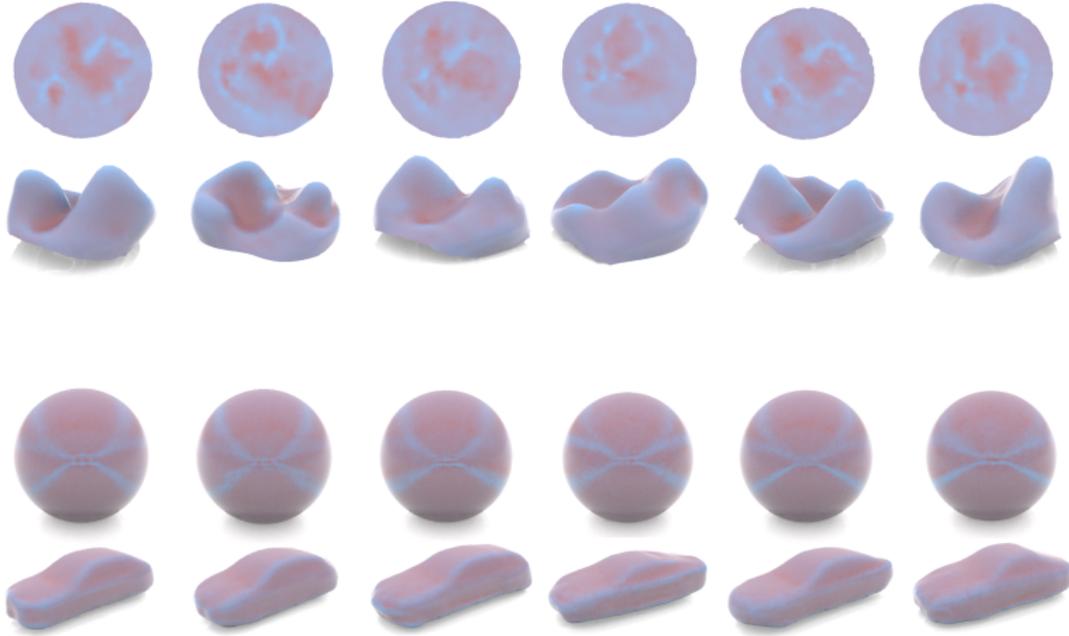
Figure 8.1: Randomly generated teeth and cars via the variational autoencoder. The first and third rows show the isotropic meshings, which are induced from the generated density function, with the generated mean curvature half-density. The second and fourth rows show the resulting reconstruction. The architectures of neural networks are modified from the traditional autoencoders in Fig. 8.2 and Fig. 8.1 to variantional autoencoder.

Since $\langle (dx)^\sharp, \partial x \rangle = 1$ and $\langle (dx)^\sharp, \partial y \rangle = 0$, $df((dx)^\sharp)$ is perpendicular to b and has the length $\frac{1}{|c|\sin\theta} = \frac{|b|}{2A}$, where A is the area of the triangle. Thus $df(dx^\sharp) = \frac{n \cdot b}{2A}$ and, by the same argument, we have $df(dy^\sharp) = \frac{n \cdot c}{2A}$.

Therefore,

$$\text{grad}_f h = \frac{n}{2A}(ah_1 + bh_2 + ch_3)$$

8.3 The energy of quaternion 1-form

We discretize the energy

$$E_u = |\omega|^2 = |d\phi + \frac{1}{2}Gdf\phi|^2$$

in the scheme of finite element method. In the local coordinate system above, the metric and its inverse read:

$$g = \begin{pmatrix} |c|^2 & -\langle c, b \rangle \\ -\langle c, b \rangle & b^2 \end{pmatrix}, \quad g^{-1} = \frac{1}{2A} \begin{pmatrix} b^2 & \langle c, b \rangle \\ \langle c, b \rangle & |c|^2 \end{pmatrix}.$$

With $\omega = \omega_x dx + \omega_y dy$, (4.11) becomes

$$\int (|\omega_x|^2 |b|^2 + \langle c, b \rangle (\overline{\omega_x} \omega_y + \overline{\omega_y} \omega_x) + |\omega_y|^2 |c|^2) dx \wedge dy.$$

Now, we work out the formula $\omega = d\phi + \frac{1}{2} G df \phi$ in one triangle:

$$\omega = \left((\phi_2 - \phi_1) + \frac{1}{2} G \cdot c((1 - x - y)\phi_1 + x\phi_2 + y\phi_3) \right) dx$$
$$+ \left((\phi_3 - \phi_1) - \frac{1}{2} G \cdot b((1 - x - y)\phi_1 + x\phi_2 + y\phi_3) \right) dy$$

where

$$G \cdot c = u_1 - u_2 + \frac{n}{2A} (-\langle a, c \rangle u_1 - \langle b, c \rangle u_2 - |c|^2 u_3)$$
$$G \cdot b = -u_1 + u_3 + \frac{n}{2A} (-\langle a, b \rangle u_1 - |b|^2 u_2 - \langle c, b \rangle u_3)$$

The energy E_u is a $|V| \times |V|$ quaterion-valued matrix. With a tedious calculation the entries related to the triangle are given by

$$|\omega|_{11}^2 = \frac{1}{2} |a|^2 - \frac{1}{6} (|a|^2 u_1 + \langle b, a \rangle u_2 + \langle c, a \rangle u_3) + \frac{1}{6} |G|^2 A^2,$$
$$|\omega|_{23}^2 = \frac{1}{2} \langle b, c \rangle + \frac{|G|^2 A^2}{12}$$
$$+ \frac{1}{12} ((4An + |a|^2) u_1 - (a \cdot b) u_2 - (c \cdot a) u_3)$$

where

$$|G|^2 = \frac{1}{4A^2} (a^2 u_3^2 + b^2 u_2^2 + c^2 u_1^2$$
$$+ 2 \langle a, b \rangle u_1 u_2 + 2 \langle b, c \rangle u_2 u_3 + 2 \langle c, a \rangle u_3 u_1).$$

8.4 Architectures

Encoder

layers	input	output
Conv2D (4 × 4)	320×32×32×2	320×32×32×4
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	320×32×32×4	320×16×16×8
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	320×16×16×8	320×8×8×16
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	320×8×8×16	320×4×4×32
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	320×4×4×32	320×2×2×64
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	320×2×2×64	320×1×1×128
BatchNormalization		
LeakyReLu		
Reshape	320×1×1×128	80×512
FC	80×512	80×256
BatchNormalization		
LeakyReLu		
Reshape	80×256	20×1024
FC	20×1024	20×512
BatchNormalization		
LeakyReLu		
FC	20×512	200

Decoder

layers	input	output
FC	200	20480
BatchNormalization		
LeakyReLu		
Reshape	20480	20×1024
FC	20×1024	20×2048
BatchNormalization		
LeakyReLu		
Reshape	20×2048	80×512
FC	80×512	80×1024
BatchNormalization		
LeakyReLu		
Reshape	80×1024	320×2×2×64
Deconv2D (4 × 4)	320×2×2×64	320×4×4×32
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	320×4×4×32	320×8×8×16
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	320×8×8×16	320×16×16×8
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	320×16×16×8	320×32×32×4
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	320×32×32×4	320×32×32×2

Table 8.1: The architecture for spherical surfaces.

Encoder

layers	input	output
Conv2D (4 × 4)	256×256×2	128×128×4
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	128×128×4	64×64×8
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	64×64×8	32×32×16
BatchNormalization		
LeakyReLu		
Conv2D (4 × 4)	32×32×16	16×16×32
BatchNormalization		
LeakyReLu		
FC	16×16×32	100

Decoder

layers	input	output
FC	100	8192
BatchNormalization		
LeakyReLu		
Reshape	8192	16×16×32
Deconv2D (4 × 4)	16×16×32	32×132×16
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	32×32×16	64×64×8
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	64×64×8	128×128×4
BatchNormalization		
LeakyReLu		
Deconv2D (4 × 4)	128×128×4	256×256×2

Table 8.2: The architecture for disk-like surfaces.

Encoder		
layers	input	output
Conv3D ($4 \times 4 \times 4$)	193×80×195×1	97×40×98×4
BatchNormalization		
LeakyReLu		
Conv3D ($4 \times 4 \times 4$)	97×40×98×4	49×20×49×8
BatchNormalization		
LeakyReLu		
Conv3D ($4 \times 4 \times 4$)	49×20×49×8	25×10×25×16
BatchNormalization		
LeakyReLu		
Conv3D ($4 \times 4 \times 4$)	25×10×25×16	13×5×13×32
BatchNormalization		
LeakyReLu		
Conv3D ($4 \times 4 \times 4$)	213×5×13×32	7×3×7×64
BatchNormalization		
LeakyReLu		
FC	7×3×7×64	200

Table 8.3: The architecture for volumetric data.

Bibliography

[AAB+15]   M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015.

[ABCCO13]   O. Azencot, M. Ben-Chen, F. Chazal, and M. Ovsjanikov. An Operator Approach to Tangent Vector Field Processing. *Computer Graphics Forum* 32.5 (2013), pp. 73–82.

[ACBCO17]   O. Azencot, E. Corman, M. Ben-Chen, and M. Ovsjanikov. Consistent Functional Cross Field Design for Mesh Quadrangulation. *ACM Trans. Graph.* 36.4 (2017), 92:1–92:13.

[ADMG18]   P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. J. Guibas. Learning Representations and Generative Models for 3D Point Clouds. *ICML*. 2018.

[ADVDI03]   P. Alliez, E. C. De Verdire, O. Devillers, and M. Isenburg. Isotropic surface remeshing. *2003 Shape Modeling International.* IEEE. 2003, pp. 49–58.

[AFW06]   D. N. Arnold, R. S. Falk, and R. Winther. Finite element exterior calculus, homological techniques, and applications. *Acta numerica* 15 (2006), pp. 1–155.

[AFW10]   D. Arnold, R. Falk, and R. Winther. Finite element exterior calculus: from Hodge theory to numerical stability. *Bulletin of the American mathematical society* 47.2 (2010), pp. 281–354.

[AOCBC15]   O. Azencot, M. Ovsjanikov, F. Chazal, and M. Ben-Chen. Discrete Derivatives of Vector Fields on Surfaces – An Operator Approach. *ACM Trans. Graph.* 34.3 (2015), 29:1–29:13.

[ASC11]   M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE. 2011, pp. 1626–1633.

[BBL+17]   M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine* 34.4 (2017), pp. 18–42.

[BCK18]   A. Baden, K. Crane, and M. Kazhdan. Möbius Registration. *Computer Graphics Forum* 37.5 (2018), pp. 211–220.

[BHMK+18]   H. Ben-Hamu, H. Maron, I. Kezurer, G. Avineri, and Y. Lipman. Multi-Chart Generative Surface Modeling. *ACM Trans. Graph.* 37.6 (2018).

[BLC+11]   D. M. Boyer, Y. Lipman, E. S. Clair, J. Puente, B. A. Patel, T. Funkhouser, J. Jernvall, and I. Daubechies. Algorithms to automatically quantify the geometric similarity of anatomical surfaces. *Proceedings of the National Academy of Sciences* 108.45 (2011), pp. 18221–18226.

[BMM+15]   D. Boscaini, J. Masci, S. Melzi, M. M. Bronstein, U. Castellani, and P. Vandergheynst. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. *Computer Graphics Forum* 34.5 (2015), pp. 13–23.

[Bob]   Discrete isothermic surfaces. *Journal für die reine und angewandte Mathematik (Crelles Journal)* 1996.475 (1996).

[Bob94]   A. I. Bobenko. Surfaces in terms of 2 by 2 matrices. Old and new integrable cases. *Harmonic Maps and Integrable Systems*. Vieweg+Teubner Verlag, 1994, pp. 83–127.

[Bon67]   O Bonnet. Memoire sur la theorie des surfaces applicables. *JEC Polyt* 42 (1967), pp. 27–29.

[BP99]   A. I. Bobenko and U. Pinkall. Discretization of surfaces and integrable systems. *Oxford lecture series in mathematics and its applications* 16 (1999), pp. 3–58.

[BZSL14]   J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. Spectral Networks and Deep Locally Connected Networks on Graphs. *Proc. ICLR, 2014.* 2014.

[Bär98]   C. Bär. Extrinsic Bounds for Eigenvalues of the Dirac Operator. *Annals of Global Analysis and Geometry* 16.6 (1998), pp. 573–596.

[CFG+15]   A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu. *ShapeNet: An Information-Rich 3D Model Repository*. Tech. rep. arXiv:1512.03012 [cs.GR]. Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[CH12]   S. H. Christiansen and T. G. Halvorsen. A simplicial gauge theory. *Journal of Mathematical Physics* 53.3 (2012), p. 033501.

[Chu16]   D. Chubelaschwili. Variational formulas for immersions into 3 manifolds. PhD thesis. Technische Universität Berlin, 2016.

[Cim12]   D. Cimasoni. Discrete Dirac operators on Riemann surfaces and Kasteleyn matrices. eng. *Journal of the European Mathematical Society* 014.4 (2012), pp. 1209–1244.

[CKPS18]   A. Chern, F. Knöppel, U. Pinkall, and P. Schröder. Shape from Metric. *ACM Trans. Graph.* 37.4 (2018), 63:1–17.

[CL15]   P. T. Choi and L. M. Lui. Fast Disk Conformal Parameterization of Simply-Connected Open Surfaces. *Journal of Scientific Computing* 65.3 (2015), pp. 1065–1090.

[CLL15]   P. T. Choi, K. C. Lam, and L. M. Lui. FLASH: Fast Landmark Aligned Spherical Harmonic Parameterization for Genus-0 Closed Brain Surfaces. *SIAM Journal on Imaging Sciences* 8.1 (2015), pp. 67–94.

[CPS11]   K. Crane, U. Pinkall, and P. Schröder. Spin Transformations of Discrete Surfaces. *ACM Trans. Graph.* 30.4 (2011), 104:1–10.

[CPS13]   K. Crane, U. Pinkall, and P. Schröder. Robust Fairing via Conformal Curvature Flow. *ACM Trans. Graph.* 32.4 (2013), 61:1–10.

[CPS15]   A. Chern, U. Pinkall, and P. Schröder. Close-to-conformal Deformations of Volumes. *ACM Trans. Graph.* 34.4 (2015), 56:1–56:13.

[CR07]   D. Cimasoni and N. Reshetikhin. Dimers on surface graphs and spin structures. I. *Communications in Mathematical Physics* 275.1 (2007), pp. 187–208.

[CR08]   D. Cimasoni and N. Reshetikhin. Dimers on surface graphs and spin structures. II. *Communications in Mathematical Physics* 281.2 (2008), p. 445.

[Cra13]   K. Crane. Conformal Geometry Processing. PhD thesis. Caltech, 2013.

[CSBC+17]   E. Corman, J. Solomon, M. Ben-Chen, L. Guibas, and M. Ovsjanikov. Functional Characterization of Intrinsic and Extrinsic Geometry. *ACM Trans. Graph.* 36.2 (2017), 14:1–14:17.

[DFG99]   Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM review* 41.4 (1999), pp. 637–676.

[Doe16]   C. Doersch. Tutorial on variational autoencoders. *arXiv:1606.05908* (2016).

[DRW10]   T. K. Dey, P. Ranjan, and Y. Wang. Convergence, Stability, and Discrete Approximation of Laplace Spectra. *Proceedings of the Twenty-first Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '10. Society for Industrial and Applied Mathematics, 2010, pp. 650–663.

[FAKG10]   S. Fuhrmann, J. Ackermann, T. Kalbe, and M. Goesele. Direct Resampling for Isotropic Surface Remeshing. *VMV*. Citeseer. 2010, pp. 9–16.

[Fri98]   T. Friedrich. On the spinor representation of surfaces in Euclidean 3-space. *Journal of Geometry and Physics* 28.1 (1998), pp. 143–157.

[FSG17]   H. Fan, H. Su, and L. J. Guibas. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 2463–2471.

[GFK+18]   T. Groueix, M. Fisher, V. G. Kim, B. Russell, and M. Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[GL83]   M. Gromov and H. B. Lawson. Positive scalar curvature and the Dirac operator on complete Riemannian manifolds. *Publications Mathématiques de l'IHÉS* 58 (1983), pp. 83–196.

[GPAM+14]   I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[GWC+04]   X. Gu, Y. Wang, T. Chan, P. Thompson, and S.-T. Yau. Genus Zero Surface Conformal Mapping and Its Application to Brain Surface Mapping. *IEEE Transactions on Medical Imaging* 23.8 (2004), pp. 949–958.

[HKA15]   P. Herholz, J. E. Kyprianidis, and M. Alexa. Perfect Laplacians for Polygon Meshes. *Proceedings of the Eurographics Symposium on Geometry Processing*. SGP '15. Graz, Austria: Eurographics Association, 2015, 211–218.

[HMZ01]   O. Hijazi, S. Montiel, and X. Zhang. Dirac Operator on Embedded Hypersurfaces. *Mathematical Research Letters* 8.2 (2001), pp. 195–208.

[Hof99]   T. Hoffmann. Discrete cmc surfaces and discrete holomorphic maps. *Oxford lecture series in mathematics and its applications* 16 (1999), p. 97.

[HSFW16]   T. Hoffmann, A. O. Sageman-Furnas, and M. Wardetzky. A Discrete Parametrized Surface Theory in $\mathbb{R}^3$. *International Mathematics Research Notices* (2016).

[HSTP11]   K. Hildebrandt, C. Schulz, C. V. Tycowicz, and K. Polthier. Interactive Surface Modeling Using Modal Analysis. *ACM Trans. Graph.* 30.5 (2011), 119:1–119:11.

[HY18]   T. Hoffmann and Z. Ye. A discrete extrinsic and intrinsic Dirac operator. *ArXiv e-prints* (2018). arXiv: 1802.06278 [math.DG].

[Jac+18]   A. Jacobson et al. *gptoolbox: Geometry Processing Toolbox*. 2018.

[Kam98]   G Kamberov. Prescribing mean curvature: existence and uniqueness problems. *Electronic Research Announcements of the American Mathematical Society* 4.2 (1998), pp. 4–11.

[Ken02]   R. Kenyon. The Laplacian and Dirac operators on critical planar graphs. *Inventiones mathematicae* 150.2 (2002), pp. 409–439.

[Ken79]   K. Kenmotsu. Weierstrass formula for surfaces of prescribed mean curvature. *Mathematische Annalen* 245.2 (1979), pp. 89–99.

[KJP+18]   I. Kostrikov, Z. Jiang, D. Panozzo, D. Zorin, and J. Bruna. Surface Networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[KNPP02]   G. Kamberov, P. Norman, F. Pedit, and U. Pinkall. Quaternions, Spinors, and Surfaces. Vol. 299. Contemporary Mathematics. American Mathematical Society, 2002.

[KP16]   F. Knöppel and U. Pinkall. Complex Line Bundles Over Simplicial Complexes and Their Applications. *Advances in Discrete Differential Geometry*. Ed. by A. I. Bobenko. Springer Berlin Heidelberg, 2016, pp. 197–239.

[KPP98]   G. Kamberov, F. Pedit, and U. Pinkall. Bonnet pairs and isothermic surfaces. *Duke Mathematical Journal* 92.3 (1998), pp. 637–644.

[KS96]   R. Kusner and N. Schmitt. The spinor representation of surfaces in space. *arXiv preprint dg-ga/9610005* (1996).

[KW14a]   O. Karpenkov and J. Wallner. On offsets and curvatures for discrete and semidiscrete surfaces. *Beiträge zur Algebra und Geometrie / Contributions to Algebra and Geometry* 55.1 (2014), pp. 207–228.

[KW14b]   D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *Proc. ICLR, 2014*. 2014.

[KW17]   T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *Proc. ICLR, 2017*. 2017.

[Lam16]   W. Y. Lam. Discrete Minimal Surfaces: Critical Points of the Area Functional from Integrable Systems. *International Mathematics Research Notices* (2016).

[Law+70]   H. B. Lawson et al. Complete minimal surfaces in S3. *Ann. of Math* 92.2 (1970), pp. 335–374.

[Lee01]   J. M. Lee. Introduction to smooth manifolds. Springer, 2001.

[LHJ+14]   R. Ling, J. Huang, B. Jüttler, F. Sun, H. Bao, and W. Wang. Spectral Quadrangulation with Feature Curve Alignment and Element Size Control. *ACM Trans. Graph.* 34.1 (2014), 11:1–11:11.

[LJC17]   D. Liu, A. Jacobson, and K. Crane. A Dirac Operator for Extrinsic Shape Analysis. *Computer Graphics Forum (SGP)* 36.5 (2017).

[LM90]   H. B. Lawson and M.-L. Michelsohn. Spin Geometry (PMS-38). Princeton University Press, 1990.

[LP16]   W. Y. Lam and U. Pinkall. Holomorphic Vector Fields and Quadratic Differentials on Planar Triangular Meshes. *Advances in Discrete Differential Geometry*. Ed. by A. I. Bobenko. Springer Berlin Heidelberg, 2016, pp. 241–265.

[LRF10]   Y. Lipman, R. M. Rustamov, and T. A. Funkhouser. Biharmonic Distance. *ACM Trans. Graph.* 29.3 (2010), 27:1–27:11.

[LS16]   P. Leopardi and A. Stern. The Abstract Hodge–Dirac Operator and Its Stable Discretization. *SIAM Journal on Numerical Analysis* 54.6 (2016), pp. 3258–3279.

[LZ10]   B. Lévy and H. R. Zhang. Spectral Mesh Processing. *ACM SIGGRAPH 2010 Courses*. SIGGRAPH '10. ACM, 2010, 8:1–8:312.

[MBBV15]   J. Masci, D. Boscaini, M. M. Bronstein, and P. Vandergheynst. Geodesic convolutional neural networks on Riemannian manifolds. *Proc. of the IEEE International Conference on Computer Vision (ICCV) Workshops*. 2015, pp. 37–45.

[MBM+17]   F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein. Geometric Deep Learning on Graphs and Manifolds Using Mixture Model CNNs. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[Mer01]   C. Mercat. Discrete Riemann surfaces and the Ising model. *Communications in Mathematical Physics* 218.1 (2001), pp. 177–216.

[MFC+10]   D. S. Marcus, A. F. Fotenos, J. G. Csernansky, J. C. Morris, and R. L. Buckner. Open access series of imaging studies: longitudinal MRI data in nondemented and demented older adults. *Journal of cognitive neuroscience* 22.12 (2010), pp. 2677–2684.

[MGA+17]   H. Maron, M. Galun, N. Aigerman, M. Trope, N. Dym, E. Yumer, V. G. Kim, and Y. Lipman. Convolutional Neural Networks on Surfaces via Seamless Toric Covers. *SIGGRAPH* (2017).

[Mor04]   B. Morel. Surfaces in S^3 and $\mathbb{H}^3$ via spinors. *Séminaire de théorie spectrale et géométrie* 23 (2004), pp. 131–144.

[NW17]   C. Nash and C. K. I. Williams. The shape variational autoencoder: A deep generative model of part-segmented 3D objects. *Computer Graphics Forum* 36.5 (2017), pp. 1–12.

[OBCS+12]   M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas. Functional Maps: A Flexible Representation of Maps Between Shapes. *ACM Trans. Graph.* 31.4 (2012), 30:1–30:11.

[Pin85]   U. Pinkall. Regular homotopy classes of immersed surfaces. *Topology* 24.4 (1985), pp. 421–434.

[RR13]   P. Romon and J. Roth. The spinor representation formula in 3 and 4 dimensions. *arXiv preprint arXiv:1309.0457* (2013).

[RRF10]   M. Reuter, H. D. Rosas, and B. Fischl. Highly accurate inverse consistent registration: a robust approach. *Neuroimage* 53.4 (2010), pp. 1181–1196.

[Rus04]   S. Rusinkiewicz. Estimating curvatures and their derivatives on triangle meshes. *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.* IEEE. 2004, pp. 486–493.

[Rus10]   R. M. Rustamov. Robust Volumetric Shape Descriptor. *Proceedings of the 3rd Eurographics Conference on 3D Object Retrieval*. 3DOR '10. Eurographics Association, 2010, pp. 1–5.

[SA07]   O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. *Symposium on Geometry processing*. Vol. 4. 2007, pp. 109–116.

[SB08]   Y. B. Suris and A. I. Bobenko. Discrete differential geometry: Integrable structure. American Mathematical Society, 2008.

[SC17]   R. Sawhney and K. Crane. Boundary First Flattening. *ACM Trans. Graph.* 37.1 (2017), 5:1–5:14.

[SM17]   E. J. Smith and D. Meger. Improved Adversarial Systems for 3D Object Generation and Reconstruction. *CoRL*. 2017.

[SOG09]   J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Computer graphics forum*. Vol. 28. 5. Wiley Online Library. 2009, pp. 1383–1392.

[SSGH01]   P. V. Sander, J. Snyder, S. J. Gortler, and H. Hoppe. Texture Mapping Progressive Meshes. *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '01. Association for Computing Machinery, 2001, 409–416.

[SSP08]   B. Springborn, P. Schröder, and U. Pinkall. Conformal Equivalence of Triangle Meshes. *ACM Trans. Graph.* 27.3 (2008), 1–11.

[TDB17]   M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs. *CoRR* (2017).

[TPKZ18]   M. Tatarchenko, J. Park, V. Koltun, and Q.-Y. Zhou. Tangent Convolutions for Dense Prediction in 3D. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[Ume17]   N. Umetani. Exploring Generative 3D Shapes Using Autoencoder Networks. *SIGGRAPH Asia 2017 Technical Briefs*. SA '17. New York, NY, USA: Association for Computing Machinery, 2017.

[VL08]   B. Vallet and B. Lévy. Spectral Geometry Processing with Manifold Harmonics. *Computer Graphics Forum* 27.2 (2008), pp. 251–260.

[Wit81]   E. Witten. A new proof of the positive energy theorem. *Communications in Mathematical Physics* 80.3 (1981), pp. 381–402.

[WLG+17]   P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong. O-CNN: octree-based convolutional neural networks for 3D shape analysis. *ACM Trans. Graph.* 36 (2017), 72:1–72:11.

[WSLT18]   P.-S. Wang, C.-Y. Sun, Y. Liu, and X. Tong. Adaptive O-CNN: A Patch-Based Deep Representation of 3D Shapes. *ACM Trans. Graph.* 37.6 (2018).

[WZX+16]   J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in Neural Information Processing Systems*. 2016, pp. 82–90.

[YDT+18]   Z. Ye, O. Diamanti, C. Tang, L. Guibas, and T. Hoffmann. A unified discrete framework for intrinsic and extrinsic Dirac operators for geometry processing. *Computer Graphics Forum* 37.5 (2018), pp. 93–106.

[YUIH]   Z. Ye, N. Umetani, T. Igarashi, and T. Hoffmann. Deep curvature learning. *In submission.* ().

[Ric97]   J. Richter. Conformal maps of a Riemannian surface into the space of quaternions. TU Berlin, FB Mathematik, 1997.